



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

Configuration

Contents

- 1 Configuration
 - 1.1 Description
 - 1.2 Example
 - 1.3 Options

Configuration

Description

App uses the configuration property '`_genesys.widgets.main`'. App controls the Genesys Widgets product as a whole, handling themes, languages, and mobile devices.

Example

```
window._genesys.widgets.main = {
  theme: 'dark',
  themes: {
    dark: 'cx-theme-dark',
    light: 'cx-theme-light',
    blue: 'cx-theme-blue',
    red: 'cx-theme-red'
  },
  lang: 'en',
  il8n: '/il8n.json',
  mobileMode: 'auto',
  mobileModeBreakpoint: 600,
  timeFormat: 12
  plugins: [
    'cx-channel-selector',
    'cx-webchat-service',
    'cx-webchat',
    'cx-call-us',
    'cx-cobrowse',
    'cx-send-message',
    'cx-send-message-service',
    'cx-gwe',
    'cx-stats-service'
  ],
  debug: true
}
```

Options

Name	Type	Description	Default	Required
themes	object	An object list containing the CSS classname for each Theme. The	{dark: 'cx-theme-dark', light: 'cx-theme-light'}	

Name	Type	Description	Default	Required
		<p>property names are used to select the theme in the 'theme' property. for example</p> <pre>{ dark:'cx-theme-dark', light:'cx-theme-light', 'red':'cx-theme-red', 'blue':'cx-theme-blue'}</pre> <p>Where 'dark' and 'light' are the built-in themes provided in Genesys Widgets and 'red' and 'blue' are example custom theme names you may create on your own. Note: It is not necessary to define the 'dark' and 'light' theme as shown in this example. It is included to help show how the formatting works. Whatever you put in this object will be merged with the default themes object internally.</p>		
theme	string	Selects the theme to apply to Genesys Widgets from the 'themes' object. Uses the property name of the theme. for example using the example from 'themes' above, possible values for this could be 'dark', 'light', 'red', 'blue'.	dark	
lang	string	Select the language to use from the 'i18n' language pack. Language codes are selected by the customer. Any language code format can be used as long as this property matches one of the language codes in your i18n language pack. For	en	

Name	Type	Description	Default	Required
		more information about localization, see localization .		
i18n	URL string or JSON	Either a path to a remote i18n.json language pack file or an inline JSON language pack definition. For more information about language packs, see localization .	Default English language strings are built into each widget and are displayed by default. Defining this i18n language pack overrides the built-in strings.	
mobileMode	boolean/string	Mobile Mode setting. true = Force Mobile Mode on all devices. false = Disable Mobile Mode completely. 'auto' = Genesys Widgets Automatically switches between Mobile and Desktop Modes using the 'mobileModeBreakpoint' property and UserAgent detection.	auto	
mobileModeBreakpoint	number	The breakpoint width in pixels where Genesys Widgets will switch to Mobile Mode. Breakpoint checked at startup only.	600	
timeFormat	number/string	This sets the time format for the timestamps. It can be 12 or 24	12	
plugins	array	A list of plugin names. This list overrides the default list of plugins that are initialized at startup. You can use this array to dynamically change the available plugins so that different user types have access to different plugins.	['cx-channel-selector', 'cx-webchat-service', 'cx-webchat', 'cx-call-us', 'cx-cobrowse', 'cx-send-message', 'cx-send-message-service', 'cx-gwe', 'cx-stats-service']	

Name	Type	Description	Default	Required
debug	boolean	Enable debug logging from the bus to appear in the browser console.	false	
customStylesheetID	string	The HTML ID of a <style> tag that contains CSS overrides, custom themes, or other custom CSS intended for Genesys Widgets. On startup, Widgets will move this <style> tag to the end of the document so that 1:1 CSS class overrides apply correctly.		