Integrating custom HTML widgets

On this page

- Using the HTML widgets model library
- Integrating HTML widgets in projects
- Widgets in nested Parts supported
- Widgets in UI mockups on a web UI

Simulation supports the integration of custom HTML widgets through the build-in model library named **UI Widgets Library.mdzip**. You can see its usage in **WebUIWidgets** in **Samples** on the Welcome screen under **Simulation**.

For further technical details, please see also the tutorial for integrating widgets for simulation.

> ⚠ **Note**
> HTML widgets work with most web browsers such as Google Chrome, Microsoft Edge (Chromium), Mozilla Firefox, and Opera. However, if you use older versions of Internet Explorer, e.g., IE 11, you must set the document mode to 11 by doing the following:
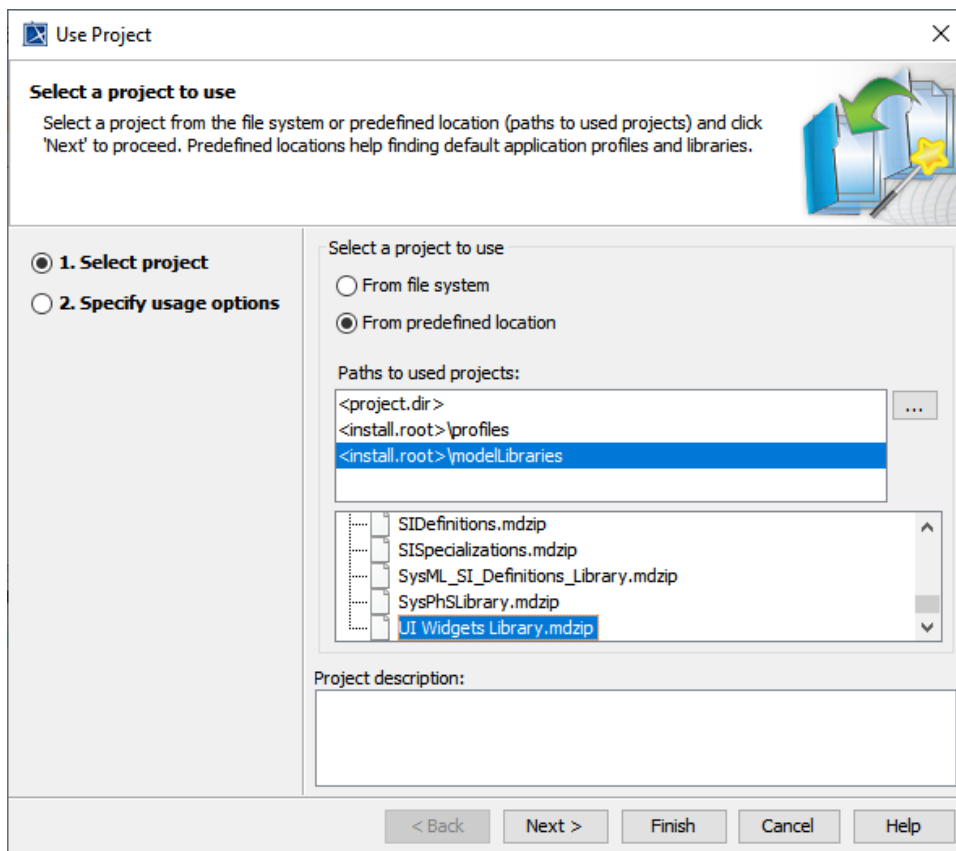
1. Open **Internet Explorer**.
2. Press **F12**. Go to the **Emulation** menu.
3. Under the **Mode** group, select the **Document mode** list and choose **11**.

Using the HTML widgets model library

To use the HTML widgets model library:

1. From the main menu, click **File** > **Use Project** > **Use Local Project**. The **Use Project** dialog opens.
2. In the **Use Project** dialog, do the following:
   - Click the **From predefined location** option.
   - Choose **<install.root>\modelLibraries**.
   - Select **UI Widgets Library.mdzip**.

3. Click **Next**. **UI Widgets Library** will be shown as the shared Package.
4. Click **Finish**. A question dialog about auxiliary resources will appear.

5. Click **Yes**. **UI Widgets Library.mdzip** will be shown in the Containment tree. There are build-in widgets in the Package, e.g., *JQueryKnob*, *LED*, and *KumaGuage*. Those widgets are ready to be customized and used as user interfaces.

> ⚠️ **Note**
>
> - HTML widgets are self-containing, owned properties (not references to any model elements except only themselves).
>
> - To further customize «Widget», open the **UI Widgets Library.mdzip** library and modify, save, and reattach the zip file of «Widget».
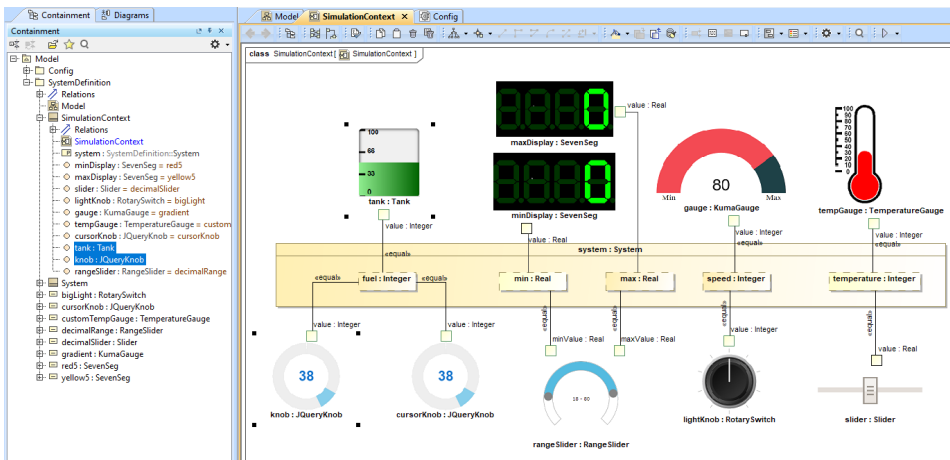
Integrating HTML widgets in projects

You can use widgets in the diagrams based on a Composite Structure diagram as described below.

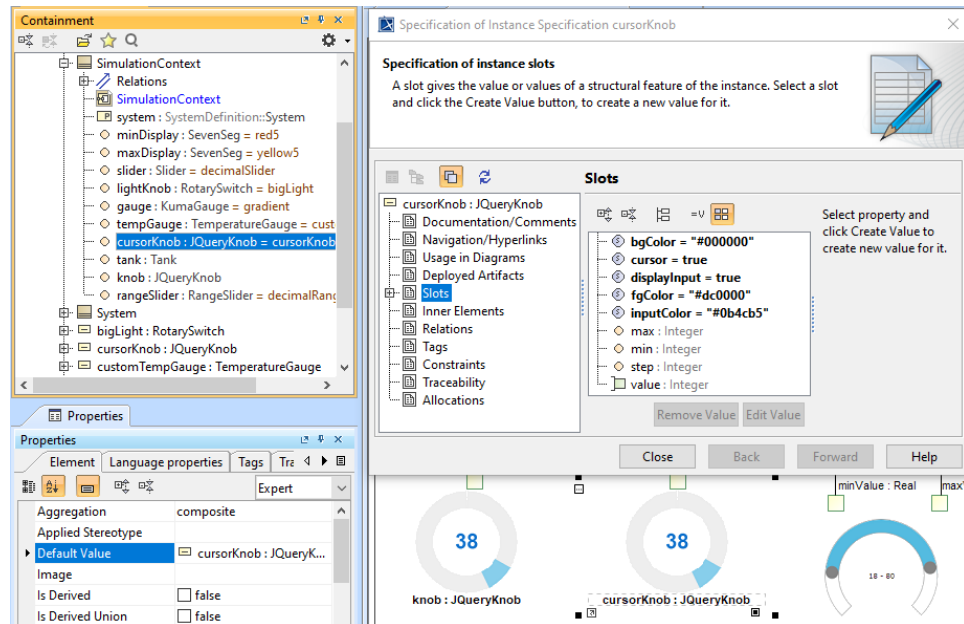> ⚠️ Widgets are not supported in iterative simulations like Monte Carlo.

1. Drag the widget from a **UI Widgets Library** and drop it into the IBD/Composite Structure diagram of the simulation context. Then use a Binding Connector to connect between the value property of the system and Port of the widget with the same Type. The widget can be resized as needed.

2. You can customize properties of any widgets, e.g., changing colors of *cursorKnob*, through the following steps:

   a. Create an Instance Specification of the widget by right-clicking the widget and selecting **Tools** > **Create Instance**. Then select a Part and Package to save.
   b. Open the Specification window of the created Instance Specification. In the left pane of the window, click **Slots** and choose either **Create Value** or **Edit Value**.
   c. Use the modified Instance Specifications as **Default Value** of the Part properties in the simulation context, as shown in the following figure.
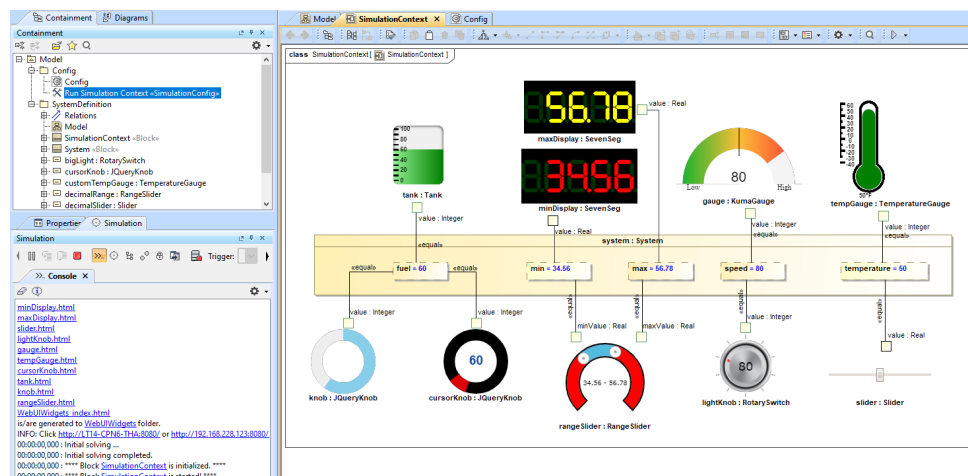

Customizing values through creating instances of the widget, setting values of properties in Slot of widgets, and using them as Part properties.

3. When running from a «SimulationConfig», you must set the executionTarget of the «SimulationConfig» to the simulation context, and the following Events will occur as shown in the next figure below:
   a. «Widget» Parts in the executionTarget, found at any Package levels, will be automatically generated as HTML files. All generated HTML files will be stored in the project's working directory with the same name as the project.
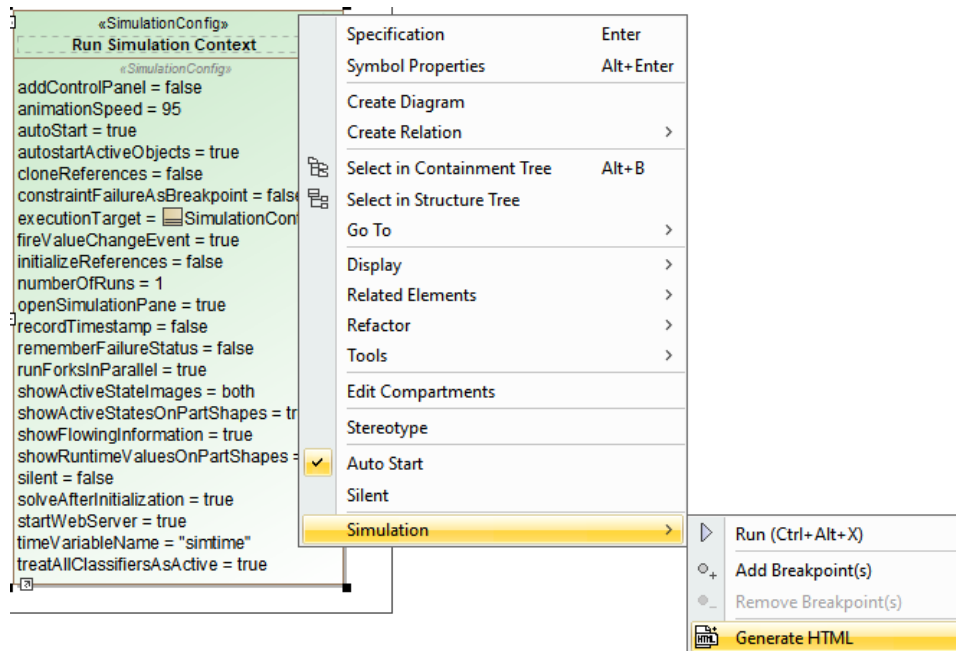   b. The Web Server will automatically start.

   > **Note**
   > If the Web Server plugin is not installed, a warning message with a link to install the plugin will be printed in the **Console** panel.
   > Widgets on the IBD will not work until the plugin has been installed.

   c. The **Console** pane will print all generated HTML files, the generated folder, and URL of the Web UI with links.

Running simulation with HTML widgets from a «SimulationConfig».

4. You can also run the simulation directly from the IBD/Composite Structure diagram that contains widgets either from the right-click menu of the diagram or the Containment tree, or the **Simulation** toolbar of the diagram. Then the simulation will automatically generate HTML files, and the Web Server will be automatically started, similar to Step 3.
5. Depending on the circumstances, you can manually regenerate HTML files for further customization of the web UI. Manually regenerating HTML files is possible by right-clicking a «SimulationConfig» and selecting **Simulation** > **Generate HTML**.

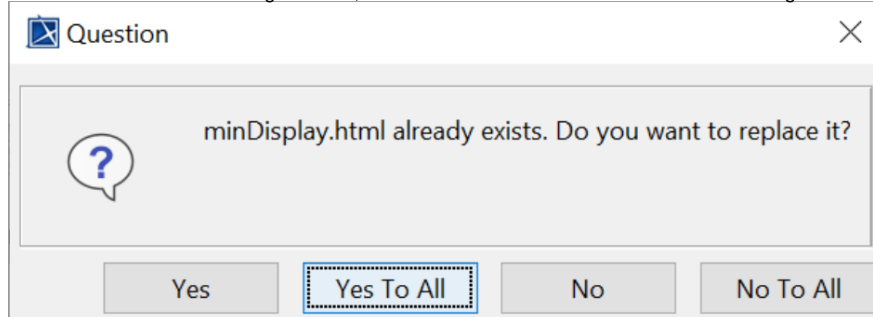Manually regenerating HTML files through the Generate HTML command for HTML widgets.

> ⚠ **Note**
> • The **Generate HTML** command is available only when there is «Widget» as Part (either Class or Block) in the *executionTarget* of a «SimulationConfig».

6. However, if the HTML files already exist, a question dialog will be shown to prevent you from accidentally overwriting them as shown in the figure below.

> • If **HTTP ERROR 404 ..** occurs in the IBD/Composite Structure diagram, you must use the **Generate HTML** command. This is because the error from HTML files is not generated, but the *startWebServer* of the «SimulationConfig» is false.

The question dialog to prevent accidentally overwriting the HTML file.

You can select one of the following buttons for a proper action:

- **Yes**: overwrite the existing *<widget>*.html file.
- **Yes To All**: overwrite all *<widget>*.html files.
- **No**: keep the existing *<widget>*.html file.
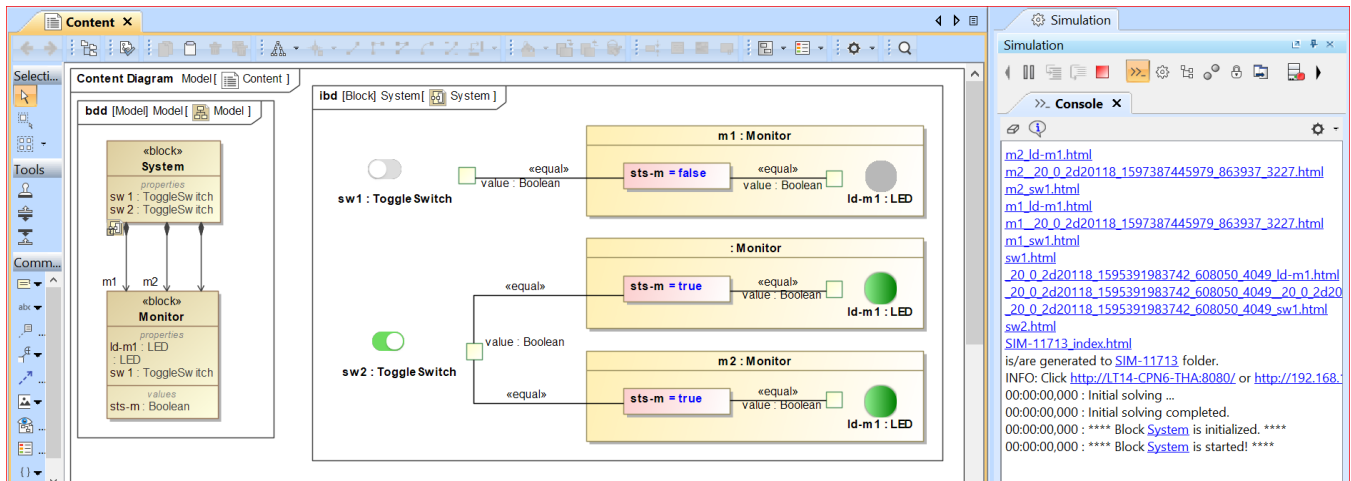- **No To All**: keep all existing *<widget>*.html files.

> ⚠ **Note**
> The *<project>_index*.html file is always regenerated at the end of the process.

Widgets in nested Parts supported

You can use widgets in nested Parts. Simulation uses the name (or the element ID if the name does not exist) of the property to construct the full path name for each generated HTML file name, e.g., a nested widget, *ld-m1:LED*, of *Monitor* is generated **m1_ld-m1.html**, **m2_ld-m1.html**, and **<Element ID>_ ld-m1.html**.

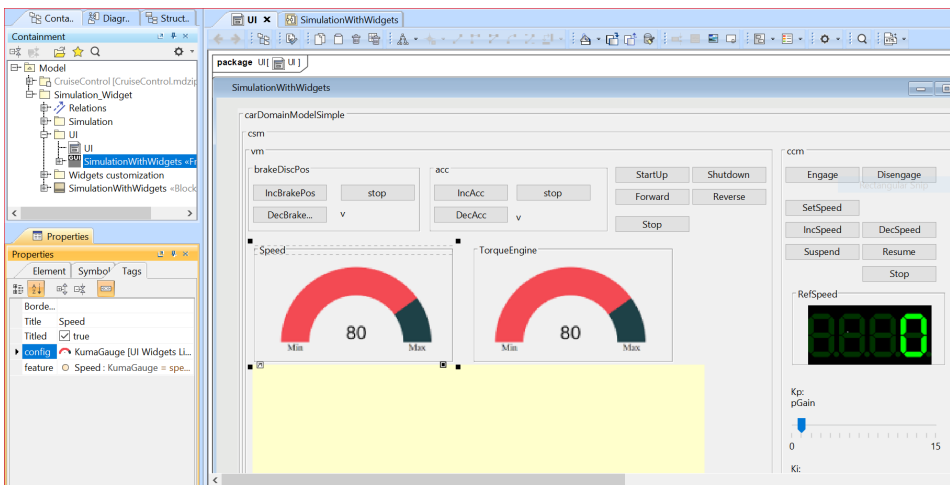Widgets in nested Parts and generated HTML file names.

> **Note**
> When using widgets, you must define your Parts with distinctive names to shorten generated widget file names that cannot normally be longer than 255 characters as limited by operating systems.

Widgets in UI mockups on a web UI

You can reuse web UI widgets in UI mockups on a web UI when running a SimulationConfig with *startWebServer = true* and *UI = «Frame» «UI» hosted widgets*.
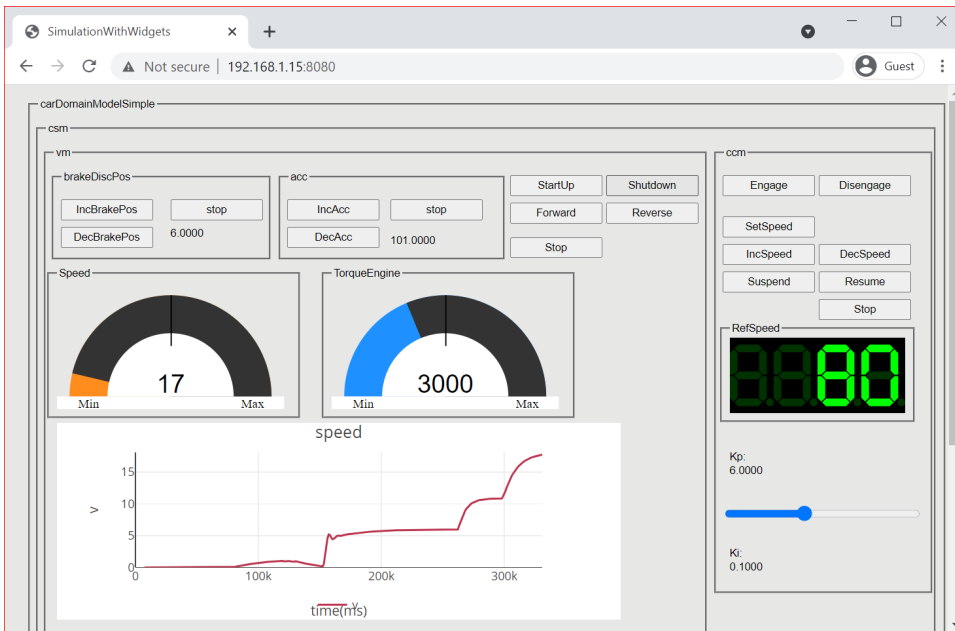
To reuse web UI widgets in UI mockups

1. Create an IBD and connect the Ports of the widgets with value properties to work with runtime values.
2. Drag a property typed widget to a «Frame» «UI». Simulation will automatically set the *config* according to the *feature*. *Image* of the widget will automatically be shown, and «NestedUIConfig» will be automatically set. You can use either a group box (by default) or panel as a place holder shown below.

Web UI widgets in «Frame» «UI» mockups with the config and feature settings.

3. Run a SimulationConfig with *startWebServer = true* and *UI = «Frame» «UI» hosted widgets*. Then, HTML files of the widgets will be generated, and the web server will be automatically started. You must click the URL in the **Console** pane (**INFO: Click** <URL> **to open web UI**) and select either **Embedded Brower** or **Web Browser** to show the web UI as shown below.

Widgets in UI mockups on a web UI through the Chrome browser.