HTML Utilities

HTML utilities allow to add and extend HTML tags and add resources which are common in web applications e.g. a company logo embedded into the UI application as a image resource.

Example File (Builder project Advanced Modeling/UI):

<your example path>\Advanced Modeling\UI\uml\uiHTML.xml
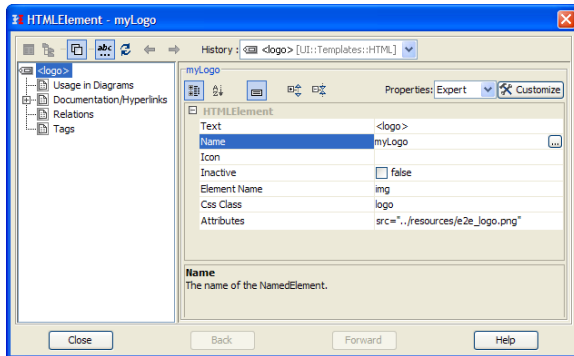

Custom HTML Attributes

The HTML tags generated consist of the basic construction e.g. `<img src="">` . The HTML image can have further attributes to control the image through setting a `width=""` and a `height=""` . This can be achieved using CSS and referencing to the CSS Class parameter of the <<HTMLElement>> or it can be set as additional HTML attributes in the **Attributes** section.
Another example could be a Text Field which needs a limitation on the number of characters that can be written <input name="name" type="text" `maxlength="30">`. This is achieved by adding the `maxlength="30"` in the **Attributes** section.
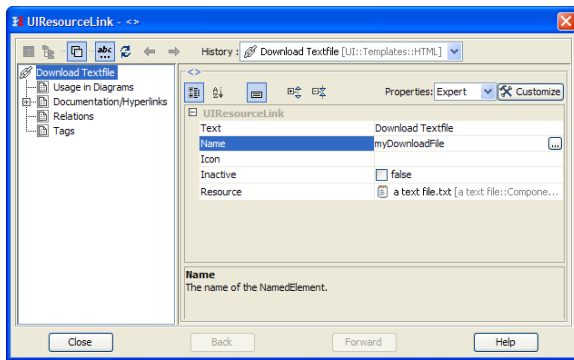
Images

To be able to embed images, a HTML image tag (<img src="path/to/image.png">) needs to be specified. This is done by assigning the <<HTMLElement>> stereotype on a label element. Assign the following settings within the specifications:

The **Text** attribute as well as the **Name** work as a simple placeholder and element names to identify and place the image within a UI structure. The attribute **Element Name** defines the actual HTML tag name <**img** src="path/to/image.png"> and the **Attributes** attribute defines the HTML image tag source to the image <img `src="path/to/image.png">`. A CSS Class can be assigned for more control over the image via CSS.

Download Link to a Resource

The stereotype <<UIResourceLink>> on <<Hyperlink>> element enables to select an imported resource such as a file and place it as a downloadable document within the user interface.
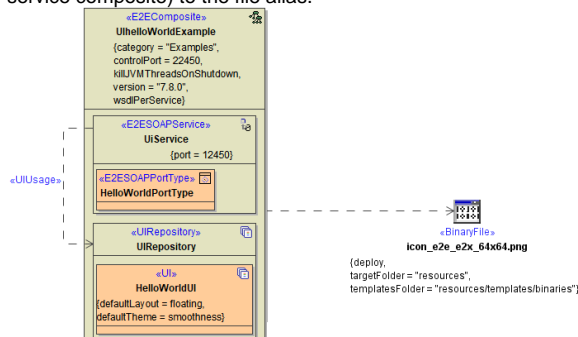
To make the document open in a new window, the stereotype <<HTMLElement>> can be additionally assigned to the <<Hyperlink>> element and within the specifications the attribute target="_blank" can be set.

Providing Needed Resources

Sometimes a resource is needed for the UI to work properly, e.g. a file or an image. To bundle such resources with the service and to directly access them, you can import them as a file resource to the corresponding UI service. That makes them available in folder **repository/htdocs** of the service.
To do this, follow these three steps:

1. Import the resource as a file resource into your Builder project (refer to Importing File Resources for more information on how to do that).
   During import, select the appropriate resource type (e.g. a **BinaryFile** in case of an image). You need to make sure that **deploy** is checked, if you want to deploy the resource together with the service.
2. Drag the imported alias from the folder **Resources** of the imported file resource onto the component diagram. Then, draw a use dependency from the **UIRepository** (residing within the service composite) to the file alias.

3. You can access the resource directly without specifying a path. In case of an image. it would be via **src=<name of the image>**.

HTML Metadata

HTML metadata tags hold various information and instructions which are inserted into the head part of a web page. Other than the well known title tag, information in the head area of your web pages is not visible on the page viewed in browsers. Instead the meta information in the head area is used to communicate information not to be concerned by the web application user. A Meta tag for example can tell a browser which character set is needed to view the served content or a description meta tag can hold an abstract of the pages content.
The meta information is held by the <<UI>> controller having the additional stereotype <<HTMLMeta>>. The meta tags can then be added within the <<UI>> controllers attribute **Meta Attributes**. Multiple meta tags can be defined.

Meta tag	Description
name="author" content="Scheer PAS Bridge"	The author is either a system/software if the page dynamically generated but can also be a human author in case of an article.
name="description" content="An xUML UI application"	The description describes either the content of the applications purpose.

name="keywords" content=" xUML UI, Scheer PAS Bridge"	Keywords hold information for search engines to help them index the content.
name="date" content="2010-04-12"	Some date information
name="robots" content=" noindex"	Instructions for search engine robots to not index/capture the meta information of this page.
http-equiv="content-type" content="text/html; charset=ISO-8859-1"	Instructions for the browser to use ISO-8859-1 character encoding for the content of this page.