Enhanced User Interface Contracts

These contracts allow you to add some user feedback to your pages. Many of these contracts implement fairly typical web page behavior in the more recent browsers (at least 4.0 browsers and above).

These contracts can generally be combined to get different and more complex effects.

Limitations

Not all combinations of these contracts have been tested. There are probably combinations that don't work properly. Clearly some combinations make no sense at all.

The border contracts are designed to work together. Contracts that add borders have counterparts that will remove borders. Borders added with Drumbeat style sheets do not interact quite right with the border manipulation contracts. The contract "Add border on page load" is provided to add a border that can be manipulated with these contracts. See implementation and customization notes. Additionally, The element position in the browser is shifted slightly from the position in the Drumbeat layout view when you apply border contracts. The element is shifted by an amount equal to the border thickness to make room for the border.

The "glow" contracts only work on images with transparency. Images that don't have transparency don't have any place to see the glow. Glow works well on text elements.

Special note on contracts that do stuff onMouseOver

There is a minor problem if the user's cursor is directly over an element (an image) when the page is loading into a browser and that element has an interaction that performs some action when the mouse is over it. For example, suppose you apply the contract "Brighten [target] when the mouse is over [source]" to an image. Further, suppose your mouse happens to be directly over that image as the page is loading. You will probably get a JavaScript error. This is due to the fact that the "mouse over" event is fired by the browser prior to the page being fully loaded and until the page is fully loaded some JavaScript may not have been downloaded yet.

The workaround is to also apply the contract "Show [source] when page loads" and mark that element (image) as not visible in the element's attribute sheet. What this does is prevent the MouseOver event from firing until the page loads. This is because as long the element is not visible the mouse is not over it. As soon as the page loads it is marked changed to a visible state and the event is allowed to fire.

Parameters

These contracts share a number of parameters, which are described here.

HighlightColor or Color
> This is the color of the border or glow. The best way to describe the glow is to just try it out and see for yourself.

Style - A drop down list of border styles to choose from. Some border styles require a larger width than others to render properly.
Width - The width or size of the border

Implementation and Customization Notes

The Brighten, Dim, Flip, and Glow contracts are implemented using IE filters. That is why they are only valid in IE4.0 and above browsers. Some of the parameters to these filters are hard wired in the action of these contracts. You can modify these values to suit your needs by clicking the actions tab of the contracts dialog and modifying them. Note that if you modify a contract it affects all elements that use that contract. You can make a copy of the contract and modify the copy to get two versions of the contract with different values for these parameters.

> Example
> The "Brighten" and "Dim" contracts use the IE alpha filter with an opacity setting of 50 (out of 100). The alpha filter is used to dim the object. The "bright" state of the object is its normal unfiltered state. You can modify these contracts to use a different opacity to get a different dimness value. Do this in the "Actions" tab of the contracts dialog. Note that changing this value will change the behavior of all elements that use this contract.

The hide and show contracts utilize the visible style of the element in the browser DOM (Document Object Model). Both 4.0 browsers support this (although slightly differently) so these contracts are valid in all 4.0 browsers.

The border contracts are a little tricky. The trick is to add a margin that is the same width as the border when the page loads. Then the contracts turn on the border by setting its width simultaneously turning off the margin by setting it to zero and vica versa. That way the element doesn't move when the border is added by the JavaScript code. This is also why the element position is offset by an amount equal to the border thickness (to the right and down) in the browser.