

Cutpoints



This page is still being drafted.

A *cutpoint* is a mapping between a component in the renderer component tree and markup in the template. It consists of two fields: The ID of the component in the tree and a selector for the element in the template:

```
{
  id: <string ID of component in tree>,
  selector: <string selector referencing an element in the template>
}
```

Working with Cutpoints

Many renderer-related functions require a list of cutpoints, but it is rare that developers will have to create cutpoints manually. There are three ways to work with cutpoints:

Automatically From Selectors

If you are using [fluid.initRendererComponent](#), you can let the initialization function generate the cutpoints from your selectors. If the default selectors are defined in your component's options, the call to [fluid.initRendererComponent](#) will automatically retrieve the selectors, generate the cutpoints, and provide them to the Renderer wherever necessary. The selector names will be used as the component IDs. This is the simplest option.

```
fluid.default("cspace.autocomplete.popup", {
  selectors: {
    noMatches: ".csc-autocomplete-noMatches",
    matches: ".csc-autocomplete-matches",
    matchItem: ".csc-autocomplete-matchItem",
    longestMatch: ".csc-autocomplete-longestMatch"
  },
  ...
});
cspace.autocomplete.popup = function (container, options) {
  var that = fluid.initRendererComponent("cspace.autocomplete.popup", container, options);
  ...
  return that;
}
```

Generated

If you are using [fluid.initRendererComponent](#), you can provide a function of your own that will be used to generate cutpoints. The function must accept an object containing a set of named selectors, and will also accept options (see [fluid.renderer.selectorsToCutpoints](#) for a description of the options).

```
// Set the cutpointGenerator option of the Renderer component to your cutpoint
// generator function
var rendererFnOptions = $.extend({}, that.options.rendererFnOptions,
  {
    rendererOptions: rendererOptions,
    repeatingSelectors: that.options.repeatingSelectors,
    selectorsToIgnore: that.options.selectorsToIgnore,
    cutpointGenerator: fluid.renderer.personalizeSelectorsToCutpoints
  });

var fluid.renderer.personalizeSelectorsToCutpoints = function (selectors, options) {
  // Override functionality here
};
```

Manually

Finally, cutpoints can be provided directly to the renderer (through [fluid.initRendererComponent](#) or if you are using [fluid.renderer](#)).

```
var cutpoints = [  
  { id: "pattern-row:", selector: ".pattern-table-row" },  
  { id: "name", selector: ".pattern-name" },  
  { id: "sample", selector: ".pattern-sample" }  
];
```