

fluid.renderer.selectorsToCutpoints



This functionality is [Sneak Peek](#) status. This means that the **APIs may change**. We welcome your feedback, ideas, and code, but please use caution if you use this new functionality.

fluid.renderer.selectorsToCutpoints(selectors, options)

Produce a list of renderer [cutpoints](#) based on the provided selectors.

```
fluid.renderer.selectorsToCutpoints(selectors, options);
```

File name: `RendererUtilities.js`

Parameters

selectors	(Object) An object containing named selectors.
options	(Object) Options that will configure how the process works. See Options below for more information.

Return Value

Array	An array of cutpoint specifications
--------------	-------------------------------------

See Also

- [fluid.render](#)

Options

The following options configure out the cutpoint creation process occurs:

Name	Description
<code>selectorsToIgnore</code>	A list of selector names to ignore. Cutpoints will not be created for these selectors
<code>repeatingSelectors</code>	A list of selector names that identify elements that will repeat.

Cutpoints

A cutpoint has the form:

```
{
  id: <string>,
  selector: <string>
}
```

Notes

`fluid.renderer.selectorsToCutpoints` will use the selector names as the `{{id}}`s in the cutpoints.

Examples

```

var selectors = {
  cancel: ".csc-confirmationDialogButton-cancel",
  proceed: ".csc-confirmationDialogButton-proceed",
  act: ".csc-confirmationDialogButton-act",
  close: ".csc-confirmationDialog-closeBtn"
};
var cutpoints = fluid.renderer.selectorsToCutpoints(selectors);

```

The code to the left will produce the following cutpoints array:

```

[
  { id: "cancel", selector: ".csc-confirmationDialogButton-cancel" },
  { id: "proceed", selector: ".csc-confirmationDialogButton-proceed" },
  { id: "act", selector: ".csc-confirmationDialogButton-act" },
  { id: "close", selector: ".csc-confirmationDialog-closeBtn" }
]

```

```

var selectors = {
  addToPanel: ".csc-autocomplete-addToPanel",
  authorityItem: ".csc-autocomplete-authorityItem",
  noMatches: ".csc-autocomplete-noMatches",
  matches: ".csc-autocomplete-matches",
  matchItem: ".csc-autocomplete-matchItem",
  longestMatch: ".csc-autocomplete-longestMatch",
  addTermTo: ".csc-autocomplete-addTermTo"
};
var cutpoints = fluid.renderer.selectorsToCutpoints(selectors, {
  selectorsToIgnore: [ "addToPanel" ],
  repeatingSelectors: [ "matchItem", "authorityItem" ],
});

```

The code to the left will produce the following cutpoints array:

```

[
  addToPanel: ".csc-autocomplete-addToPanel",
  { id: "authorityItem:", selector: ".csc-autocomplete-authorityItem" },
  { id: "noMatches", selector: ".csc-autocomplete-noMatches" },
  { id: "matches", selector: ".csc-autocomplete-matches" },
  { id: "matchItem:", selector: ".csc-autocomplete-matchItem" },
  { id: "longestMatch", selector: ".csc-autocomplete-longestMatch" },
  { id: "addTermTo", selector: ".csc-autocomplete-addTermTo" }
]

```

Note that the `addToPanel` selector has not been converted into a cutpoint, and that two cutpoint IDs have a colon (":") added to the name.