

Reorderer API

Reorderer Overview

The Reorderer will handle almost any arbitrary stream of markup you pass to it, assuming a very simple contract is maintained:

1. The orderable elements are contained within some kind of container element.
2. A selector for the container is passed to the Reorderer upon instantiation.

Optional:

1. Provide a callback function. This will be called by the Reorderer each time the user reorders an element, and is generally used to communicate ordering changes back to the server.
2. Specify configuration options to customize the behaviour of the Reorderer, including the selection of orderable elements within the container. These are documented below.

Creation

To create a generic reorderer:

```
fluid.reorderer(container[, options]);
```

There are four pre-configurations of the Reorderer (List, Image, Grid, and Layout) provided as a convenience for particular contexts and use cases.

List Reorderer

```
fluid.reorderList(container[, options]);
```

Allows reordering of elements in a vertical or horizontal list. More information is available at [List Reorderer API](#).

Grid Reorderer

```
fluid.reorderGrid(container[, options]);
```

Allows reordering of a grid of items. More information is available at [Grid Reorderer API](#).

Layout Reorderer

```
fluid.reorderLayout(container[, options]);
```

Allows reordering of portlets, content blocks, or other chunks of layout arranged in columns. More information is available at [Layout Reorderer API](#).

Image Reorderer

```
fluid.reorderImages(container[, options]);
```

Allows reordering of images, image thumbnails. More information is available at [Image Reorderer API](#).

Parameters

container

The `container` parameter is a CSS-based [selector](#), single-element jQuery object, or DOM element specifying the root node of the Reorderer.

options

The `options` parameter is an optional collection of name-value pairs that configure the Reorderer, as described below in the [fluid:Options](#) section.

Status

This component is in [Production status](#)

On This Page

- [Reorderer Overview](#)
- [Creation](#)
- [Supported Events](#)
- [Options](#)
- [CSS Classes](#)

See Also

- [List Reorderer API](#)
- [Grid Reorderer API](#)
- [Layout Reorderer API](#)
- [Image Reorderer API](#)
- [Talking to the Server Using The afterMove Event](#)
- [Integrating the Reorderer](#)
- [Fluid Component API](#)

Still need help?

Join the [infusion-users mailing list](#) and ask your questions there.

Supported Events

Event	Type	Description	Parameters	Parameter Description
<code>onShowKeyboardDropWarning</code>	default	This event fires before a drop warning is displayed.	<code>item</code> , <code>kbDropWarning</code>	<code>item</code> : The item being moved. <code>kbDropWarning</code> : The DOM element that contains the drop warning, and that will be displayed.
<code>onSelect</code>	default	This event fires when an item is selected by the user.	<code>item</code>	<code>item</code> : The item being selected.
<code>onBeginMove</code>	"preventable"	This event fires just before a request to move is processed. Because the event is preventable, listeners may prevent the move from happening.	<code>item</code>	<code>item</code> : The item being moved.

onMove	default	This event fires just before an item is actually moved.	item, requestedPosition	<p>item: The item being moved. requestedPosition: An object describing the position that the user is trying to move the item into:</p> <pre>requestedPosition = { element, // the drop target position // the position, relative to the drop target, // that a dragged item should be dropped. // One of BEFORE, AFTER, INSIDE, or REPLACE }</pre>
afterMove	default	<p>This event fires after an item has successfully been moved. For more information, see Talking to the Server Using The afterMove Event</p> <p><i>This event replaces the <code>afterMoveCallbackUrl</code> option, which was deprecated at version 1.1.2.</i></p>	item, requestedPosition, movables	<p>item: The item being moved. requestedPosition: An object describing the position that the user is trying to move the item into:</p> <pre>requestedPosition = { element, // the drop target position // the position, relative to the drop target, // that a dragged item should be dropped. // One of BEFORE, AFTER, INSIDE, or REPLACE }</pre> <p>movables: A list of all of the movable elements.</p>
onHover	default	This event fires when the cursor moves over top of an item, and when the cursor moves away from an item. The default listener either adds or removes the hover class (<code>styles.hover</code>) to/from the item.	item, state	<p>item: The item being moved. state: A boolean indicating whether the cursor is moving to (<code>true</code>) or away from (<code>false</code>) the item.</p>
onRefresh	default	This event fires any time the order of the items changes, or when the <code>refresh()</code> function is called.	none	

Options

Selectors

The `selectors` option is an object containing CSS-based selectors for various parts of the Reorderer. Supported selectors are:

Name	Description	Default	Examples
movables	Identifies the DOM elements contained within the Reorderer container that can be moved using the Reorderer.	".f1c-reorderer-movable"	<pre>selectors: { movables: "div.movable" }</pre>

selectables	Identifies the DOM elements contained within the Reorderer container that can be selected using keyboard. Note that selectable elements do not <i>have</i> to be movable.	same as movables	<pre> selectors: { selectables: "div. selectable" } </pre>
dropTargets	Identifies the DOM elements contained within the Reorderer container that can have movable elements dropped relative to them. Note that not all elements within the container need to be drop targets.	same as movables	<pre> selectors: { dropTargets: "div. dropTarget" } </pre>
grabHandle	If present, identifies a single element within a movable item that the user must click on to drag the movable item. (If not specified, the entire movable item can be clicked on.)	"" (empty string)	<pre> selectors: { grabHandle: ". title-bar" } </pre>
dropWarning	Identifies a single element within the DOM that can be shown to display a warning when the user tries to move an item where it can't be moved. It is assumed that this element contains whatever drop warning text and mark-up the implementor desires.	".flc-reorderer-dropWarning"	<pre> selectors: { dropWarning: "#drop-warning" } </pre>

General options

Name	Description	Values	Default
mergePolicy	an object describing how user options should be merged in with defaults For information on options merging, see Options Merging for Infusion Components		<pre> mergePolicy: { keysets: "replace", "selectors. selectables": "selectors. movables", "selectors. dropTargets": "selectors.movables" } </pre>
listeners	JavaScript object containing listeners to be attached to the supported events.	Keys in the object are event names, values are functions or arrays of functions.	See fluid:Supported Events

styles	an object containing CSS class names for styling the Reorderer.	The object may contain any of the keys defined in the default class names (shown to the right). Any class names not provided will revert to the default.	<pre> styles: { defaultStyle: "fl-reorderer- movable-default", selected: "fl- reorderer-movable- selected", dragging: "fl- reorderer-movable- dragging", mouseDrag: "fl- reorderer-movable- dragging", hover: "fl- reorderer-movable- hover", dropMarker: "fl- reorderer- dropMarker", avatar: "fl- reorderer-avatar" } </pre>
keysets	an object containing sets of keycodes to use for directional navigation, and for the modifier key used for moving a movable item.	The object must be a list of objects containing the following keys: modifier : a function that returns true or false, indicating whether or not the required modifier(s) are activated up down right left	<pre> fluid.reorderer. defaultKeysets = [{ modifier : function (evt) { return evt. ctrlKey; }, up : fluid. reorderer.keys.UP, down : fluid. reorderer.keys.DOWN, right : fluid. reorderer.keys. RIGHT, left : fluid. reorderer.keys.LEFT }, { modifier : function (evt) { return evt. ctrlKey; }, up : fluid. reorderer.keys.i, down : fluid. reorderer.keys.m, right : fluid. reorderer.keys.k, left : fluid. reorderer.keys.j }]]; </pre>
avatarCreator	a function that returns a valid DOM node to be used as the dragging avatar		The item being dragged will be cloned
afterMoveCallbackUrl	Deprecated as of 1.1.2: If an afterMoveCallbackUrl is provided, a default listener for this event will post a string version of the model to the URL.	The server URL to POST the new item order to	

<p>New in v1.3 disableWrap</p>	<p>This option is used to disable wrapping of elements within the container.</p> <pre>//example of grid reordering var reordering = fluid.reorderGrid('.myGrid', { selectors: { movables: 'p' }, disableWrap: true }); //example of image reordering var reordering = fluid.reorderImages(".flc-imageReordering", { selectors: { movables: ".flc-imageReordering-item" }, disableWrap: true }); //example of layout reordering var reordering = fluid.reorderLayout("#fluid-LayoutReordering-sample2", { selectors: { columns: ".myColumn", modules: "> div > div", lockedModules: ".locked", dropWarning: ".flc-reordering-dropWarning" }, disableWrap: true }); //example of list reordering var reordering = fluid.reorderList(".todo-list", { selectors: { movables: ".movable" }, styles: { defaultStyle: "demo-reordering-movable-default", selected: "demo-reordering-movable-selected", dragging: "demo-reordering-movable-dragging", mouseDrag: "demo-reordering-movable-mousedrag", hover: "demo-reordering-movable-hover", dropMarker: "demo-reordering-dropMarker", avatar: "demo-reordering-avatar" }, disableWrap: true });</pre>	<p>boolean</p>	<p>false</p>
------------------------------------	--	----------------	--------------

Advanced options

Name	Description	Values	Default
containerRole	indicates the role, or general use, for this instance of the Reorderer	fluid.roles.LIST fluid.roles.GRID fluid.roles.REGIONS	fluid.roles.LIST
layoutHandler		"fluid.listLayoutHandler" "fluid.gridLayoutHandler" "fluid.moduleLayoutHandler"	"fluid.listLayoutHandler"

CSS Classes

The Reorderer applies CSS classes to orderable items, and updates them as they are moved. These classes can be used to apply visual cues to the various states of the orderable items. The class names are configurable. The default class names are:

- fl-reorderer-movable-default - This class is applied to elements in their default state.
- fl-reorderer-movable-selected - This class is applied to the element that has been selected. The selected item can then be moved using keystrokes.
- fl-reorderer-movable-hover - This class is applied to orderable elements when the mouse hovers over them.
- fl-reorderer-movable-dragging - This class is applied to the orderable element that is currently being moved using the keyboard, and to the orderable element that is currently being moved using the mouse.
- fl-reorderer-movable-avatar - This class is applied to the avatar, which defaults to the image of the orderable element as it is being dragged about by the mouse.
- fl-reorderer-movable-dropMarker - This class is applied to the drop target indicator when the mouse is used to drag an item.

```
<script type="text/javascript" src="lib/jquery/core/js/jquery.js"></script>
<script type="text/javascript" src="lib/jquery/ui/js/jquery.ui.core.js"></script>
<script type="text/javascript" src="lib/jquery/ui/js/jquery.ui.widget.js"></script>
<script type="text/javascript" src="lib/jquery/ui/js/jquery.ui.mouse.js"></script>
<script type="text/javascript" src="lib/jquery/ui/js/jquery.ui.draggable.js"></script>
<script type="text/javascript" src="framework/core/js/FluidDocument.js"></script> <!-- New in v1.3 -->
<script type="text/javascript" src="framework/core/js/jquery/jquery.keyboard-ally.js"></script>
<script type="text/javascript" src="framework/core/js//Fluid.js"></script>
<script type="text/javascript" src="framework/core/js/FluidView.js"></script> <!-- New in v1.3 -->
<script type="text/javascript" src="framework/core/js/FluidDOMUtilities.js"></script>
<script type="text/javascript" src="framework/core/js/DataBinding.js"></script> <!-- New in v1.3 -->
<script type="text/javascript" src="framework/core/js/FluidIoC.js"></script> <!-- New in v1.3 -->
<script type="text/javascript" src="framework/core/js/ReordererDOMUtilities.js"></script>
<script type="text/javascript" src="components/reorderer/js/GeometricManager.js"></script>
<script type="text/javascript" src="components/reorderer/js/Reorderer.js"></script>
```