

# Layout Reorderer API



This page has not yet been updated to reflect the latest API changes.

## Layout Reorderer Overview

The Layout Reorderer is a convenience function for applying the Reorderer to portlets, content blocks, or other chunks of layout with minimal effort. This page provides technical details of the API.

## Creation

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

## Parameters

### container

The `container` is a CSS-based [selector](#), single-element jQuery object, or DOM element that identifies the DOM element containing the layout.

### options

The options object is an optional collection of key/value pairs that can be used to further configure the Layout Reorderer, as described below in the [Options](#) section.

### Status

This component is in [Production status](#)

### On This Page

- [Layout Reorderer Overview](#)
- [Creation](#)
  - [Parameters](#)
    - [container](#)
    - [options](#)
- [Supported Events](#)
- [Options](#)
- [Dependencies](#)

### See Also

- [Tutorial - Layout Reorderer](#)
- [List Reorderer API](#)
- [Grid Reorderer API](#)
- [Image Reorderer API](#)
- [Reorderer API](#)
- [Tutorial - Layout Reorderer Selectors](#)

### Still need help?

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

## Supported Events

The Layout Reorderer fires the following events (for more information about events, see [Events](#)):

Event	Type	Description	Parameters	Parameter Description
onShowKeyboardDropWarning	default	This event fires before a drop warning is displayed.	item, kbDropWarning	item: The item being moved. kbDropWarning: The DOM element that contains the drop warning, and that will be displayed.
onSelect	default	This event fires when an item is selected by the user.	item	item: The item being selected.
onBeginMove	"preventable"	This event fires just before a request to move is processed. Because the event is preventable, listeners may prevent the move from happening.	item	item: 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 <a href="#">Talking to the Server Using The afterMove Event</a></p> <p><i>This event replaces the afterMoveCallbackUrl 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 (styles.hover) to/from the item.	item, state	<p>item: The item being moved. state: A boolean indicating whether the cursor is moving to (true) or away from (false) the item.</p>
onRefresh	default	This event fires any time the order of the items changes, or when the refresh() function is called.	none	

## Options

### Selectors

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

Name	Description	Default	Examples
columns	Identifies the DOM elements contained within the Layout Reorderer container that are themselves column containers, that is a container element that corresponds to a single column in the layout.	".flc-reorderer-column"	<pre>selectors: {   columns:     "#column1,     #column2, #column3" }</pre>
modules	Identifies the DOM elements contained within the Layout Reorderer container that are considered to be modules within the columns. Note that modules are assumed to be movable, unless they are identified by the <code>lockedModules</code> selector (see below).	".flc-reorderer-module"	<pre>selectors: {   modules: "div" }</pre>
lockedModules	Identifies the modules that should not be movable.	none	<pre>selectors: {   lockedModules: ".   locked" }</pre>
grabHandle	If present, identifies a single element within a module that the user must click on to drag the module.	"" (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 module 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
selectors	JavaScript object defining CSS-style selectors for important DOM elements. See <a href="#">Selectors</a> for more information.	The object must be a list of objects containing any subset of the following keys:  <pre>dropWarning movables selectables dropTargets grabHandle stylisticOffset</pre>	<pre>selectors: {   dropWarning: ".flc- reorderer-dropWarning",   movables: ".flc- reorderer-movable",   selectables: ".flc- reorderer-movable",   dropTargets: ".flc- reorderer-movable",   grabHandle: "",   stylisticOffset: "" }</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 <a href="#">Supported Events</a>

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>
selectablesTabindex	Normally injected automatically from the layoutHandler	String <a href="#">loC expression</a>	<pre> selectablesTabIndex: "{that}.layoutHandler.options.selectablesTabIndex" </pre>
avatarCreator	a function that returns a valid DOM node to be used as the dragging avatar		The item being dragged will be cloned
disableWrap	This option is used to disable wrapping of elements within the container.	boolean	false

mergePolicy	<p>an object describing how user options should be merged in with defaults          For information on options merging, see <a href="#">Options Merging</a></p>		<pre>mergePolicy: {   keysets: "replace",   "selectors. selectables":   "selectors. movables",   "selectors. dropTargets":   "selectors. movables" }</pre>
-------------	---	--	--

**Layout-specific 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.REGIONS

**Dependencies**

The Layout Reorderer dependencies can be met by including the `MyInfusion.js` file in the header of the HTML file:

```
<script type="text/javascript" src="MyInfusion.js"></script>
```

Alternatively, the individual file requirements are:

```
<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>
<script type="text/javascript" src="framework/core/js/jquery/jquery.keyboard-ally.js"></script>
<script type="text/javascript" src="lib/json/js/json2.js"></script>
<script type="text/javascript" src="framework/core/js/Fluid.js"></script>
<script type="text/javascript" src="framework/core/js/FluidView.js"></script>
<script type="text/javascript" src="framework/core/js/FluidDOMUtilities.js"></script>
<script type="text/javascript" src="framework/core/js/DataBinding.js"></script>
<script type="text/javascript" src="framework/core/js/FluidIoC.js"></script>
<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>
<script type="text/javascript" src="framework/core/js/LayoutReorderer.js"></script>
<script type="text/javascript" src="framework/core/js/ModuleLayout.js"></script>
```