

# Image Reorderer API



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

## Image Reorderer Overview

The Image Reorderer is a convenience function for applying the Reorderer to images within a collection.. This page provides technical details of the API.

## Creation

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

## 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 Image Reorderer, as described in the [Options](#) section below.

### Status

This component is in [Production status](#)

### On This Page

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

### See Also

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

### Still need help?

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

## Supported 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 <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	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.
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 Image 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.	".flc-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>

### Image Reorderer-specific Selector

The Image Reorderer supports one additional selector:

Name	Description	Default	Examples
imageTitle	Identifies a DOM element that contains a title for the image	".flc-reorderer-imageTitle"	<pre>selectors: {   imageTitle: ".caption" }</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.	<p>The object must be a list of objects containing any subset of the following keys:</p> <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	an object describing how user options should be merged in with defaults For information on options merging, see <a href="#">Options Merging</a>		<pre>mergePolicy: {   keysets: "replace",   "selectors. selectables":     "selectors. movables",   "selectors. dropTargets":     "selectors. movables" }</pre>

### Image Reorderer Options

Name	Description	Values	Default
<b>Deprecated as of 1.1.2:</b> afterMoveCallbackUrl	If an URL is provided with this option, the current state of the component model will be sent to that URL after a move is carried out using a default <code>afterMove</code> event handler. See <a href="#">Deprecated - afterMoveCallback</a> for info.	an URL	none

## Styling the Image Reorderer

The Image Reorderer includes default CSS styles that it applies to the thumbnails. The application of styles is based on known class names. The *default* class names are described below, and can be used by including the Image Reorderer stylesheet:

```
<link href="components/reorderer/css/imageReorderer.css" type="text/css" rel="stylesheet" media="all"/>
```

NOTE that the default class names can be overridden with your own classes using the `styles` option: see the [Options](#) section above.

#### Default classes:

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

## Dependencies

The Image 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/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/FluidDOMUtilities.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="components/reorderer/js/ImageReorderer.js"></script>
```