

List Reorderer API

List Reorderer Overview

The List Reorderer allows reordering of elements in a vertical or horizontal list. This page provides technical details of the API. For a simple working example, see [Integrating the Reorderer](#).

Creation

```
var myReorderableList = fluid.reorderList(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 list items to be reorderable. This may be the list itself (e.g. the `` or `` element) or an element that contains the list.

options

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

Status

This component is in [Production status](#)

On This Page

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

See Also

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

Still need help?

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

Supported Events

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

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 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 List 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>

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>
avatarCr eator	a function that returns a valid DOM node to be used as the dragging avatar		The item being dragged will be cloned
afterMove eCallback url	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>
-------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------	--------------

List-specific options

Name	Description	Values	Default
orientation	Indicates whether the list is oriented vertically (the default) or horizontally	fluid.orientation.VERTICAL and fluid.orientation.HORIZONTAL	fluid.orientation.VERTICAL
containerRole	Indicates the role, or general use, for this instance of the Reorderer	fluid.reorderer.roles.LIST fluid.reorderer.roles.GRID fluid.reorderer.roles.REGION	fluid.reorderer.roles.LIST

Dependencies

The List Reorderer dependencies can be met by including the minified `InfusionAll.js` file in the header of the HTML file:

```
<script type="text/javascript" src="InfusionAll.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> <!-- 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>
```