

# Upgrading to Infusion 1.2



This page provides pointers to instructions for upgrading your Fluid Infusion from version 1.1 to version 1.2.

## Overview

This release includes two upgrades that will require changes the import paths in your HTML files:

- New versions of **jQuery** and **jQuery UI**.  
This will require changes to the import paths used to reference jQuery and jQuery UI files.
- Updates to the latest version of **jqUnit**.  
If you are using Infusion's jqUnit, this will require changes to some import paths.

Details on how to update your import paths are provided later on this page.

In addition to these upgrades, the following updates may require changes to your implementations, if you are using them:

- Refactoring of the Reorderer DOM utilities
- Updates to *Infusion Mobile FSS* API
- Added Support for **CKEditor 3.x** to Rich Text Inline Edit (linked through *Fluid CDN*) and deprecated support for FCKEditor 2.x

Details on how to update your files for these changes are also provided on this page.

### On this Page

- [Overview](#)
- [Upgrades to jQuery and jQuery UI](#)
  - [General](#)
- [Component-specific Changes](#)
  - [UI Options](#)
  - [Uploader](#)
  - [Inline Edit](#)
  - [Progress](#)
  - [Reorderer](#)
- [Upgrades to jqUnit](#)
- [Reorderer DOM Utilities](#)
- [Updated Mobile FSS API](#)
- [Updated editor for Rich Text Inline Edit](#)

## Upgrades to jQuery and jQuery UI

### General

Infusion's versions of jQuery and jQuery UI have been upgraded to version 1.4.2 and 1.8 respectively. These upgrades have resulted in some file name changes, as well as some new file requirements for different jQuery UI widgets.

If you are using specific Infusion components, the sections further below provide specific guidance on changes for those components. If you are using Infusion's copy of jQuery and jQuery UI for your own general purposes, you may need to make the following changes:

Old	New
<b>Core</b>	
lib/jquery/ui/js/ui.core.js	lib/jquery/ui/js/jquery.ui.core.js
<b>Dialog</b>	
lib/jquery/ui/js/ui.core.js lib/jquery/ui/js/ui.dialog.js	lib/jquery/ui/js/jquery.ui.core.js lib/jquery/ui/js/jquery.ui.widget.js lib/jquery/ui/js/jquery.ui.position.js lib/jquery/ui/js/jquery.ui.dialog.js
<b>Slider</b>	
lib/jquery/ui/js/ui.core.js lib/jquery/ui/js/ui.slider.js	lib/jquery/ui/js/jquery.ui.core.js lib/jquery/ui/js/jquery.ui.widget.js lib/jquery/ui/js/jquery.ui.slider.js

<b>Accordion</b>	
lib/jquery/ui/js/ui.core.js lib/jquery/ui/js/ui.accordion.js	lib/jquery/ui/js/jquery.ui.core.js lib/jquery/ui/js/jquery.ui.widget.js lib/jquery/ui/js/jquery.ui.accordion.js
<b>Drag and Drop</b>	
lib/jquery/ui/js/ui.core.js lib/jquery/ui/js/ui.draggable.js	lib/jquery/ui/js/jquery.ui.core.js lib/jquery/ui/js/jquery.ui.widget.js lib/jquery/ui/js/jquery.ui.mouse.js lib/jquery/ui/js/jquery.ui.draggable.js

More details about the changes to jQuery UI can be found on [jQuery's Upgrade Guide](#).

## Component-specific Changes

### UI Options

<b>Upgrading From</b>	<b>Upgrading To</b>
lib/jquery/ui/js/ui.core.js lib/jquery/ui/js/ui.accordion.js lib/jquery/ui/js/ui.slider.js	lib/jquery/ui/js/jquery.ui.core.js lib/jquery/ui/js/jquery.ui.widget.js lib/jquery/ui/js/jquery.ui.mouse.js lib/jquery/ui/js/jquery.ui.accordion.js lib/jquery/ui/js/jquery.ui.slider.js

### Uploader

<b>Upgrading From</b>	<b>Upgrading To</b>
lib/jquery/ui/js/ui.core.js	lib/jquery/ui/js/jquery.ui.core.js

### Inline Edit

<b>Upgrading From</b>	<b>Upgrading To</b>
lib/jquery/ui/js/ui.core.js	lib/jquery/ui/js/jquery.ui.core.js

### Progress

<b>Upgrading From</b>	<b>Upgrading To</b>
lib/jquery/ui/js/ui.core.js	lib/jquery/ui/js/jquery.ui.core.js

### Reorderer

<b>Upgrading From</b>	<b>Upgrading To</b>
lib/jquery/ui/js/ui.core.js lib/jquery/ui/js/ui.draggable.js	lib/jquery/ui/js/jquery.ui.core.js lib/jquery/ui/js/jquery.ui.widget.js lib/jquery/ui/js/jquery.ui.mouse.js lib/jquery/ui/js/jquery.ui.draggable.js

## Upgrades to jqUnit

If you are using Infusion's copy of jqUnit for unit tests, you will need to make the following changes:

Changes	Upgrading From	Upgrading To
New qunit dependencies in test templates	<code>lib/qunit/css/testsuite.css</code> <code>lib/qunit/js/testrunner.js</code>	<code>lib/qunit/css/qunit.css</code> <code>lib/qunit/js/qunit.js</code>
New qunit selector's ids	<code>id="header"</code> <code>id="banner"</code> <code>id="userAgent"</code> <code>id="tests"</code>	<code>id="qunit-header"</code> <code>id="qunit-banner"</code> <code>id="qunit-userAgent"</code> <code>id="qunit-tests"</code>

You can find a tutorial on writing JavaScript unit tests at [Writing JavaScript Unit Tests](#). This tutorial also provides an example of how you can now create asynchronous tests with jqUnit.

## Reorderer DOM Utilities

*FluidDOMUtilities.js* is now split into *FluidDOMUtilities.js* and *ReordererDOMUtilities.js* where the latter one consists of those former members of FluidDOMUtilities which are used only by the Reorderer and not by the renderer. Thus the new dependency for the Reorderer on FluidDOMUtilities is now:

Old
<pre>&lt;script type="text/javascript" src="framework/core/js/FluidDOMUtilities.js"&gt;&lt;/script&gt;</pre>
New
<pre>&lt;script type="text/javascript" src="framework/core/js/FluidDOMUtilities.js"&gt;&lt;/script&gt; &lt;script type="text/javascript" src="components/reorderer/js/ReordererDOMUtilities.js"&gt;&lt;/script&gt;</pre>

If you are not using the Reorderer, you will only need *FluidDOMUtilities.js*.

## Updated Mobile FSS API

The [Mobile FSS \(new in v1.1\)](#) is the mobile-friendly version of [Fluid Skinning System \(FSS\)](#). It takes advantage of core FSS philosophies like modular class names and files, it recycles many FSS class names to help take your existing content and make it mobile-friendly, and it's easy to use - just like regular FSS.

You can find more information at [Mobile FSS Cheat Sheet](#).

## Updated editor for Rich Text Inline Edit

The [Rich Text Inline Edit](#) component applies a rich-text editor to an input field. Previously, Infusion included a copy of the FCKEditor for this. This editor has been deprecated and replaced by CKEditor 3.x, linked through *Fluid CDN*.

Old Dependency	New Dependency
<code>tests/manual-tests/lib/fckeditor/fckeditor.js</code>	<code>http://ckeditor-fluid.appspot.com/ckeditor.js</code>

Also, you can now initialize CKEditor similarly to TinyMCE and FCKEditor as follows:

```
fluid.inlineEdit.CKEditor(container, options);
```

