EditLive Drupal Integration Installation Instructions

Files Needed

Before you can get EditLive running in Drupal you need to get a few files…

1. Custom Ephox Drupal module - **`ephox_editlive.zip`**
2. Editor definition files for WYSIWYG module
    a. **`editlive.inc`**
    b. **`editlive.js`**
3. The most recent EditLive Cumulative Fix Release download from http://ephox.com/download

Prerequisites

The use of the EditLive Drupal integration requires that you have the WYSIWYG module installed and running on your Drupal server.

This integration has been tested only on Drupal 7.

Note: When EditLive encounters local image files it will automatically upload those files to the Drupal server when you save the content item (node). EditLive will store all of these files in a series of folders inside the Drupal server's public files directory.

*To determine your public files directory you can look at the File System settings in the Configuration/Media page (*http://<server>/admin/config/media/file-system*). After any user saves the first local image you will find an* ***`ephox_files`*** *directory in public files location. Inside this* ***`ephox_files`*** *directory will be a single folder per user. Each user's uploaded files will be stored in their user-specific folder.*

Installation Steps

1. Install the ephox_editlive module

The **`ephox_edilive`** module needs to be installed in order to allow local images to be uploaded to the Drupal server. Installing this module should be no different than installing any other Drupal module. To install the module…

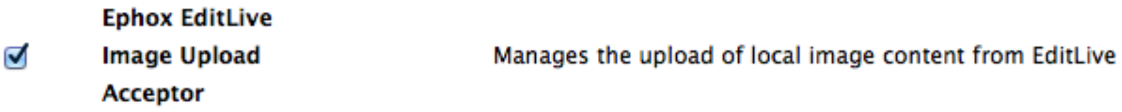
a) Unzip the **`ephox_editlive.zip`** file – this will result in the existence of a single **`ephox_edilive`** folder that contains 2 files.

b) Copy the **ephox_edilive** folder to the **/sites/all/modules** folder.

Note: If you use another location for contributed modules place the ***ephox_edilive*** *folder in the location where you install contributed modules*

c) Login to Drupal

d) Go to the Modules page (*http://<server>/admin/modules*) and enable the module called "Ephox EditLive Image Upload Acceptor". The module should appear in the User Interface grouping in the modules list.



☑ **Ephox EditLive Image Upload Acceptor** Manages the upload of local image content from EditLive

e) Click Save to accept the enabling of the Ephox EditLive Image Upload Acceptor module.

2. Add the EditLive editor definition files to the WYSIWYG module

Implementing EditLive requires you to have the WYSIWYG module previously installed. In this step we are going to add 2 files to that module that define EditLive to the WYSIWYG module.

a) Locate the WYSIWYG module in your Drupal server installation. Modules are typically installed in the **/sites/all/modules** folder. The folder name for the WYSIWYG module is **wysiwyg** so the complete path would be **/sites/all/modules/wysiwyg**

Note: If you use another location for contributed modules the ***wysiwyg*** *folder will be in the location where you install contributed modules*

b) Locate the **editlive.inc** file from your download and place this file in the **editors** folder within WYSIWYG module root folder.

c) Locate the **editlive.js** file from your download and place this file in the **editors/js** folder within WYSIWYG module root

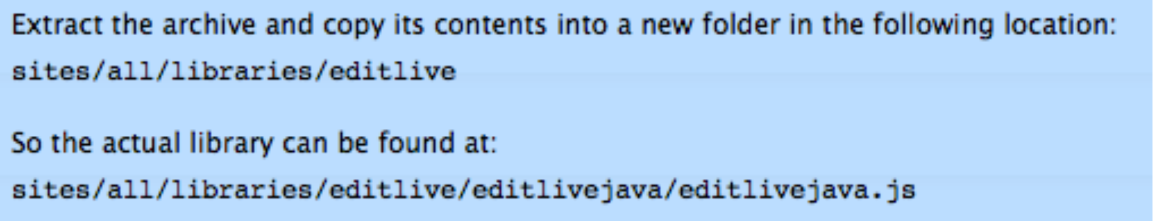
3. Configure WYSIWYG to use EditLive

In order to use EditLive as your editor we need to perform several configuration steps to make EditLive available within the WYSIWYG module.

a) Navigate to the configuration screen for the WYSIWYG module (*http://<server>/admin/config/content/wysiwyg*)

b) Click on the Installation Instructions link – you should see EditLive as an editor but it should show "Not Installed"

c) Follow WYSIWYG module admin screen instructions to get the editor "installed" properly. The module should provide you some details on where to place EditLive's files in order to make them available to the WYSIWYG module. On our test system the instructions were as follows:



Extract the archive and copy its contents into a new folder in the following location:

`sites/all/libraries/editlive`

So the actual library can be found at:

`sites/all/libraries/editlive/editlivejava/editlivejava.js`

(We will assume these paths for the remainder of these instructions)

d) Create a directory named **editlive** in the **/sites/all/libraries** directory.

e) Copy the EditLive runtime files to the **/sites/all/libraries/editlive** directory.

    a. When you download the EditLive SDK from the Ephox web site you get a ZIP file.

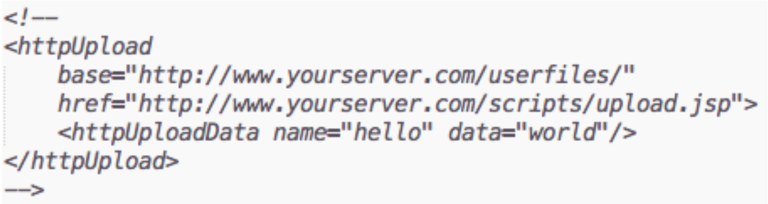
    b. When you unzip that file navigate to the **/webfolder/redistributables** directory.

    c. Copy the **editlivejava** folder into the **/sites/all/libraries/editlive** folder

    **Note: If you download the update SDK file the editlivejava folder is directly in the root of the unzipped content.*

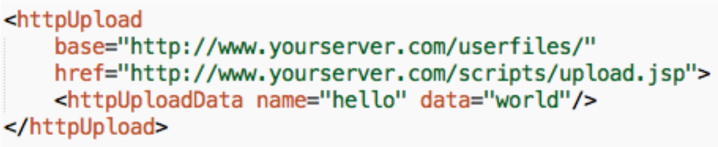
f) Navigate to the **/sites/all/libraries/editlive/editlivejava** directory. Copy the EditLive sample configuration file (sample_eljconfig.xml) and make a new file named **configuration.xml**.

g) Open **configuration.xml** with a text editing tool and locate the <httpUpload> element in the file. Uncomment the <httpUpload> element by removing the starting and ending comment tags.



```
<!--
<httpUpload
    base="http://www.yourserver.com/userfiles/"
    href="http://www.yourserver.com/scripts/upload.jsp">
    <httpUploadData name="hello" data="world"/>
</httpUpload>
-->
```

...becomes...

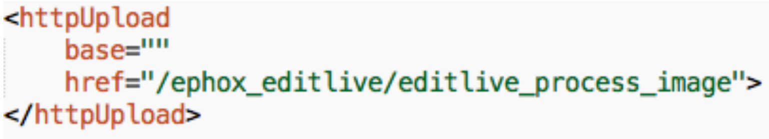


```
<httpUpload
    base="http://www.yourserver.com/userfiles/"
    href="http://www.yourserver.com/scripts/upload.jsp">
    <httpUploadData name="hello" data="world"/>
</httpUpload>
```

h) Edit the HREF and BASE attribues of the <httpUpload> element

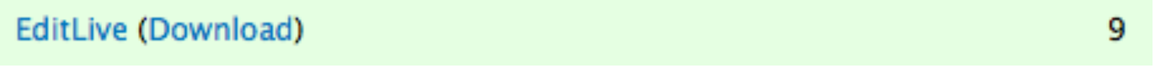
  a. The BASE attribute should be the empty string
  b. The HREF attribute must point to the EditLive module's editlive_process_image target. The module defines this as the path **/ephox_editlive/editlive_process_image**.

     This leads to the following configuration settings:



```
<httpUpload
    base=""
    href="/ephox_editlive/editlive_process_image">
</httpUpload>
```

i) Refresh the WYSIWYG admin screen and click on the Installation Instructions link – now EditLive should show as "installed" and a version number should appear on the right side of the row



EditLive (Download) 9

This signifies that EditLive was "installed" to the correct location and that the WYSIWYG module has located the files.

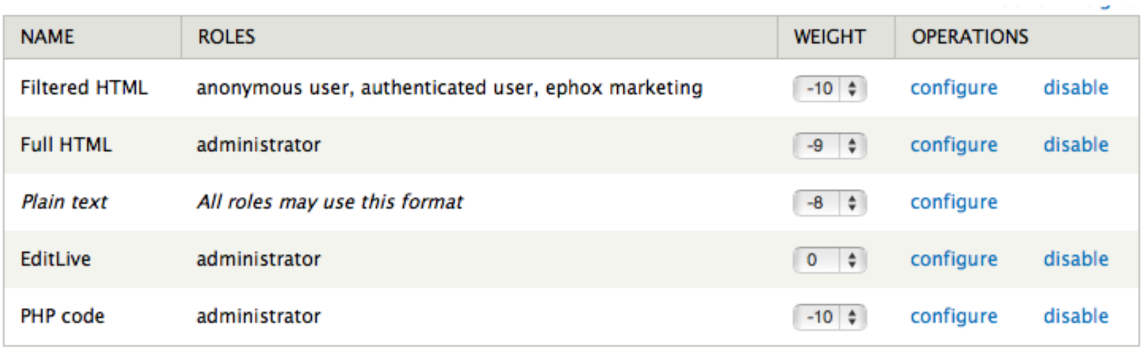
4. Create a Text Format that uses EditLive

The final step in getting EditLive to work in Drupal is the creation and configuration of a Text Format. The Text Format defines which Editor is to be used while editing content.

a) Navigate to the Configuration / Content Authoring / Text Formats screen (*http://<server>/admin/config/content/formats*)

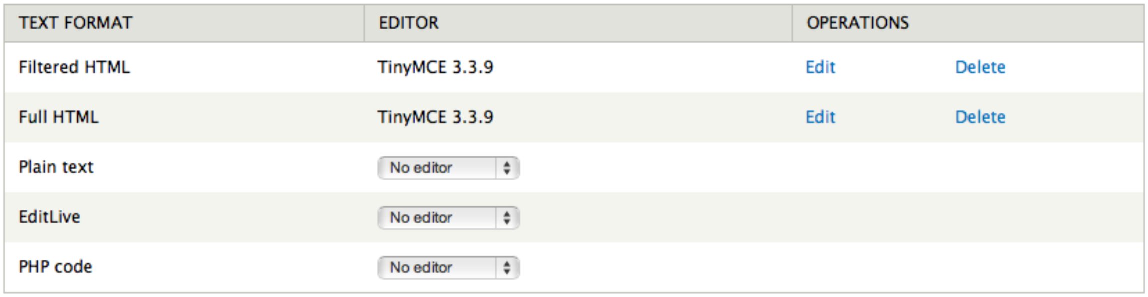
b) Click on Add Format and enter the following values:
   a. Name = EditLive
   b. Roles – select whatever roles are appropriate for your usage
   c. Ignore the Enabled Filters (not supported at present)
   d. Click Save Configuration

c) Return to Configuration / Content Authoring / Text Formats (*http://<server>/admin/config/content/formats*) and confirm your new Text Format is in the list



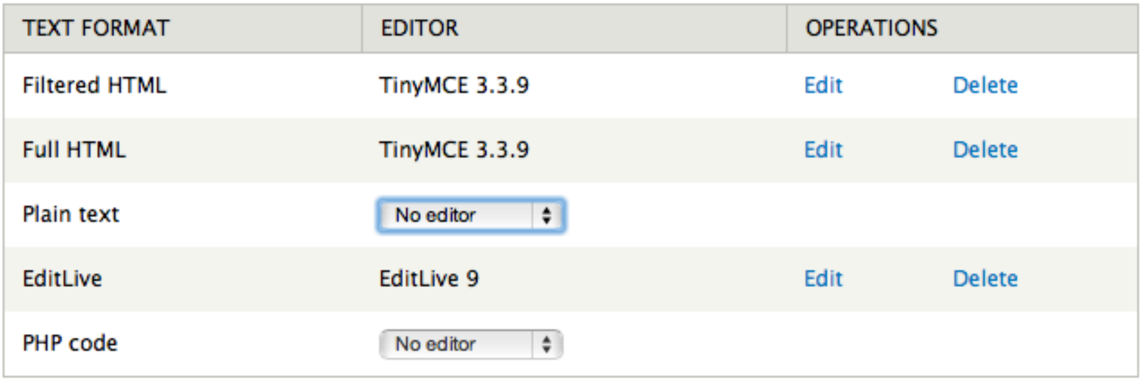
NAME	ROLES	WEIGHT	OPERATIONS	
Filtered HTML	anonymous user, authenticated user, ephox marketing	-10	configure	disable
Full HTML	administrator	-9	configure	disable
Plain text	*All roles may use this format*	-8	configure	
EditLive	administrator	0	configure	disable
PHP code	administrator	-10	configure	disable

d) Navigate to the Configuration / Content Authoring / Wysiwyg Profiles page (*http://<server>/admin/config/content/wysiwyg*) and confirm your text format is in the list



TEXT FORMAT	EDITOR	OPERATIONS	
Filtered HTML	TinyMCE 3.3.9	Edit	Delete
Full HTML	TinyMCE 3.3.9	Edit	Delete
Plain text	No editor		
EditLive	No editor		
PHP code	No editor		

e) Select EditLive from the Editor select list and click Save button. The list of profiles should now show EditLive is the editor for the EditLive Text Format:



TEXT FORMAT	EDITOR	OPERATIONS	
Filtered HTML	TinyMCE 3.3.9	Edit	Delete
Full HTML	TinyMCE 3.3.9	Edit	Delete
Plain text	No editor		
EditLive	EditLive 9	Edit	Delete
PHP code	No editor		

f) Now you can test to see if things are configured properly. Create a new content item and use the EditLive text format. If everything is configured properly you will see EditLive load.

Note: If you have issues at this point please register on our support site at http://support.ephox.com and open a support ticket.

5. *(optional)* Configure CSS to be passed into EditLive

Note: Test that EditLive loads BEFORE you try to add this CSS step. If the path to the CSS is wrong EditLive will not load. If you test the remainder of the work first you will know that any issues at this point are CSS related!

Once you have EditLive loading you will want to make sure that our design view provides a proper representation of what the content will look like when rendered on the web site. In order to accomplish this task you will need to configure the WYSIWYG module to pass an appropriate CSS file to EditLive when EditLive loads.

We recommend you create a specific CSS file for EditLive that includes only those classes needed to render and edit content. For details on how EditLive leverages CSS please see http://docs.ephox.com/display/EditLive/Using+CSS+in+the+Applet.

a) Place your CSS file in your **DEFAULT** theme at
`/sites/all/themes/<themename>/css`

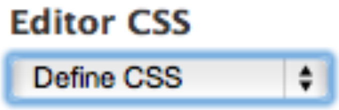
(we will assume the file is named editor.css for this document)

b) Navigate to the Configuration / Content Authoring / Wysiwyg Profiles page (*http://<server>/admin/config/content/wysiwyg*).

c) Click the Edit link on the EditLive profile line

d) Click on CSS to open the CSS section

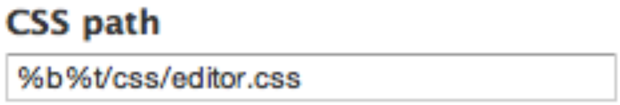
e) Select Define CSS in the Editor CSS section



Editor CSS

Define CSS

f) Enter a path to the CSS file. (Since we placed it in the Default Theme we can use substitution variables to help create the path:



CSS path

%b%t/css/editor.css

g) Click Save to save the EditLive profile.

h) Click Save to save the wysiwyg profile page changes.

i) Now you can test to see if things are configured properly. Create a new content item and use the EditLive text format. Test to see that the CSS you provided is coming through and that you see the correct fonts/sizes/colors/etc in EditLive.