

Preview Editor  
**Plug-in Guide**  
trunk-SNAPSHOT

# Table of Contents

<a href="#">1 Introduction</a>	3
<a href="#">1.1 Intended Audience</a>	3
<a href="#">2 Using The Preview Editor</a>	4
<a href="#">2.1 Editing Text</a>	4
<a href="#">2.2 Editing Images</a>	5
<a href="#">3 Installation</a>	7
<a href="#">3.1 Conventions</a>	7
<a href="#">3.2 Preview Editor Installation</a>	7
<a href="#">3.3 Verify the Installation</a>	8
<a href="#">3.4 Modify and Deploy Your Publications</a>	9
<a href="#">4 Configuration</a>	10
<a href="#">4.1 layout-group Resource Changes</a>	10
<a href="#">4.2 content-type Resource Changes</a>	10
<a href="#">4.3 JSP Template Changes</a>	12
<a href="#">4.3.1 Enabling Editing at Field Level</a>	12
<a href="#">4.3.2 Enabling Section Page Image Editing</a>	13
<a href="#">4.3.3 Highlighting Editable Texts and Images</a>	13
<a href="#">4.4 Enable Preview Editing</a>	14
<a href="#">4.5 Application Configuration Layer Changes</a>	14

# 1 Introduction

Preview Editor is a plug-in for Content Studio that adds preview editing capabilities to Escenic publications. With the Preview Editor plug-in installed, editors and journalists can edit both section pages and content items while previewing them in Content Studio. Editing previews in this way is a much more immediate way to make final layout adjustments: you can immediately see the results of your changes.

The standard Content Studio preview function allows you to see how your content will look on various devices. When you switch between devices, the preview area changes shape accordingly, and if your publication is set up to omit some content on certain devices, or use different image versions, then these rules are also applied. With the addition of the Preview Editor, you can easily make minor adjustments (such as shortening a headline to fit iPhone page widths) without having to switch back and forth between preview mode and editing mode.

Preview editing is very simple - you just open the preview tab of the section page or content item you are working on and enable the editing facility:

<screenshot>

You will then see that parts of the page you are previewing are highlighted, indicating that they are editable:

<screenshot>

Two kinds of content are editable: plain text and images.

You can change plain text content in the following ways:

- Delete, insert and modify
- Apply bold and italic styles
- Change the font size

You can modify images in the following ways:

- Zoom in and out
- Pan (if the image is zoomed in)
- Change the aspect ratio (if the image version allows it)

All of Preview Editor's features are described in detail in [chapter 2](#).

## 1.1 Intended Audience

This guide is of interest to the following kinds of users:

- Editors and journalists responsible for editing section pages.
- Template developers who want to create Preview Editor-enabled templates.
- Administrators responsible for managing Escenic sites.

## 2 Using The Preview Editor

Depending on how the Preview Editor has been set-up, preview editing may not be available in all sections of your publication, or for all content types. If preview editing is available for the content item or section page you are working on, then you will see a **Preview Editor** panel on the right of the **Preview** tab:

<screenshot>

To start preview editing, first display the preview by clicking on the **Preview** tab at the bottom of the editor, and then check **Enable Editing** in the **Preview Editor** panel.

<screenshot>

When you enable preview editing, you will see that all editable components in the preview are highlighted in some way. In the screenshot shown above, the editable text is displayed on a yellow background, and the editable image is ... ???. Different highlighting conventions may be used in your publication, but you should see that some of the content changes appearance when you switch preview editing on and off.

Once you have logged in, you can edit section pages in the following ways:

- Delete, insert and modify text
- Apply bold and italic styles to text
- Change the font size of text
- Zoom images in and out
- Pan images (if the image is zoomed in)
- Resize images (if the image version allows resizing)

How you do these things is described in detail in the following sections.

Changes you make to a preview are handled in exactly the same way as changes made in the normal editor view:

- The content item field or section page area you have edited is locked, and remains locked until you save the change.
- Changes are invisible to everyone but you until you **Save** or **Publish** them. Saving makes the changes visible to other Content Studio users, publishing makes the changes visible to web site visitors.

### 2.1 Editing Text

You can edit any highlighted text on the page, using your standard keyboard shortcuts for copy (**Ctrl+C** or **Cmd+C**), cut (**Ctrl+X** or **Cmd+X**) and paste (**Ctrl+V** or **Cmd+V**).

You can also use the following formatting tools:

Tool	Function	Shortcut
	Move slider to the right to increase the size of currently selected text.	None
	Move slider to the left to decrease the size of currently selected text.	None
	Bold on/off for currently selected text.	<b>Ctrl+B</b> (Windows) <b>Cmd+B</b> (Mac)
	Italic on/off for currently selected text.	<b>Ctrl+I</b> (Windows) <b>Cmd+I</b> (Mac)

You can combine the above formats in any way you like.

You can use the Preview Editor to apply formats to text in plain text fields such as **Title** and **Lead text** fields that do not normally support text formatting. If you do so, then you will notice that when you return to the **Editor** or **Desk** tab, some HTML mark-up has been inserted into the field. Be careful not to modify this code. If you don't feel comfortable editing text containing HTML mark-up, and you need to make further changes to the field, switch back to the **Preview** tab and make your changes there.

## 2.2 Editing Images

You can edit images displayed in content summaries in the following ways:

- You can zoom in and out.
- You can pan the image.
- You may be able to change the shape (aspect ratio) of the image by adjusting its height or width.

Any changes you make in this way only affect what is displayed on the page you are editing. The original image is not affected in any way.

### Zooming

To zoom an image:

1. Click in the image to select it.
2. Rotate the scroll wheel on your mouse: forward to zoom in, backward to zoom out.

### Panning

To pan an image:

1. Click in the image to select it.
2. Place the cursor anywhere in the image.
3. Hold down the left mouse button and drag.

You cannot pan an image if it is zoomed all the way out.

## Adjusting the height

To adjust the height of an image:

1. Click in the image to select it.
2. Drag the bottom border of the image frame up or down.
3. When you are satisfied with the frame height, release the handle.

You may not be able to adjust the height of some images. You can only adjust the height of an image if its content type specifies the use of a "free crop" representation (an image version with an adjustable aspect ratio).

## 3 Installation

The following preconditions must be met before you can install the Preview Editor trunk-SNAPSHOT plug-in:

- The Content Engine and Escenic assembly tool have been installed as described in the [Escenic Content Engine Installation Guide](#) and are in working order.
- You have the required plug-in distribution file `preview-editor-trunk-SNAPSHOT.zip`.

The Preview Editor trunk-SNAPSHOT plug-in depends in general on Content Engine 5.7.??, you must have version 5.7.?? or a later version.

### 3.1 Conventions

The instructions in the following section assume that you have a standard Content Engine installation, as described in the [Escenic Content Engine Installation Guide](#). *escenic-home* is used to refer to the `/opt/escenic` folder under which both the Content Engine itself and all plug-ins are installed.

The Content Engine and the software it depends on may be installed on one or several host machines depending on the type of installation required. In order to unambiguously identify the machines on which various installation actions must be carried out, the [Escenic Content Engine Installation Guide](#) defines a set of special host names that are used throughout the manual.

Some of these names are also used here:

#### **assembly-host**

The machine used to assemble the various Content Engine components into an enterprise archive or .EAR file.

#### **engine-host**

The machine(s) used to host application servers and Content Engine instances.

#### **editorial-host**

**engine-host**(s) that are used solely for (internal) editorial purposes.

The host names always appear in a bold typeface. If you are installing everything on one host you can, of course, ignore them: you can just do everything on the same machine. If you are creating a larger multi-host installation, then they should help ensure that you do things in the right places.

### 3.2 Preview Editor Installation

Installing Preview Editor involves the following steps:

1. Log in as **escenic** on your **assembly-host**.
2. Download the Preview Editor distribution from the Escenic Technet web site (<http://technet.escenic.com>). If you have a multi-host installation with shared folders as described in the [Escenic Content Engine Installation Guide](#), then it is a good idea to download the distribution to your shared `/mnt/download` folder:

```
$ cd /mnt/download
$ wget http://user:password@technet.escenic.com/downloads/release/57/preview-
editor-trunk-SNAPSHOT.zip
```

Otherwise, download it to some temporary location of your choice.

3. If the folder **/opt/escenic/engine/plugins** does not already exist, create it:

```
$ mkdir /opt/escenic/engine/plugins
```

4. Unpack the Preview Editor distribution file:

```
$ cd /opt/escenic/engine/plugins
$ unzip /mnt/download/preview-editor-trunk-SNAPSHOT.zip
```

This will result in the creation of an **/opt/escenic/engine/plugins/preview-editor** folder.

5. Log in as **escenic** on your **assembly-host**.
6. Run the **ece** script to re-assemble your Content Engine applications.

```
$ ece assemble
```

This generates an EAR file (**/var/cache/escenic/engine.ear**) that you can deploy on all your **engine-hosts**.

7.  If you have a single-host installation, then skip this step.

On each **engine-host** you wish to run the Preview Editor plug-in, copy **/var/cache/escenic/engine.ear** from the **assembly-host**. If you have installed an SSH server on the **assembly-host** and SSH clients on your **engine-hosts**, then you can do this as follows:

```
$ scp -r escenic@assembly-host-ip-address:/var/cache/escenic/engine.ear /var/
cache/escenic/
```

where *assembly-host-ip-address* is the host name or IP address of your **assembly-host**.

8. On each **engine-host** you wish to run the Preview Editor plug-in, deploy the EAR file and restart the Content Engine by entering:

```
$ ece deploy
$ ece restart
```

### 3.3 Verify the Installation

To verify the status of the Preview Editor plug-in, open the Escenic Admin web application (usually located at <http://server/escenic-admin>) and click on **View installed plug-ins**. The status of all currently installed plug-ins is shown here, and indicated as follows:



The plug-in is correctly installed.



The plug-in is not correctly installed.

## 3.4 Modify and Deploy Your Publications

In order to enable Preview Editor features in your publications you need to make some modifications to them. You must:

1. Configure the publications as described in [chapter 4](#).
2. Rebuild and deploy your publications as described in the [Escenic Content Engine Installation Guide](#).

## 4 Configuration

In order to be able to use Preview Editor after installing it (see [chapter 3](#)) you must:

- Add **preview-editor** decorators to your publication **layout-group** and **content-type** resources.
- Add image alternative fields to content type summaries in the content-type resource (to support image editing on section pages)
- Make a number of small changes to your publication templates
- Possibly create an application configuration layer for the Preview Editor web application, and add one file to it

### 4.1 layout-group Resource Changes

To enable preview editing for your publication's section pages, you must add a **decorator** element with the name **preview-editor** to each root group defined in the publication's **layout-group** resource. For example:

```
<group name="content" root="true">
  <ui:label>Content</ui:label>
  ...
  <ui:decorator name="preview-editor"/>
  ...
</group>
```

Once you have added this decorator to a content type, you can enable preview editing for individual fields on section pages by adding preview editor classes to the field definitions in your group JSP templates, as described in [section 4.3](#).

### 4.2 content-type Resource Changes

To enable preview editing for a content type, you must add a **decorator** element with the name **preview-editor** as a child of the **content-type** element:

```
<content-type name="news">
  <ui:label>Story</ui:label>
  ...
  <ui:decorator name="preview-editor"/>
  ...
</content-type>
```

Once you have added this decorator to a content type, you can enable preview editing for individual fields in the content type by adding preview editor classes to the field definitions in your article JSP templates, as described in [section 4.3](#).

To enable image editing on section pages, you will probably need to modify your image content type definitions. The "crop" or "alternates" field in your image content type definition needs to be available in the content type summary as well. You can achieve this by copying the whole field definition into

the summary, but the easiest way of doing it is to make use of the **rep:representation** element's **inherit** attribute. The highlighted code in the following example illustrates the technique:

```
<content-type name="picture">
...
  <ui:decorator name="preview-editor"/>
  ...
  <panel name="default">
    <field name="title" type="basic" mime-type="text/plain"/>
    ...
    <field name="binary" type="link">
      <relation>com.escenic.edit-media</relation>
      ...
    </field>
    ...
  </panel>
  <panel name="crop">
    <field name="alternates" type="basic" mime-type="application/json">
      <representations xmlns="http://xmlns.escenic.com/2009/representations"
type="image-versions">
        <representation name="BASE_LANDSCAPE">
          <output width="1140" height="712"/>
          <crop/>
          <resize/>
        </representation>
        <representation name="BASE_LANDSCAPE_SMALL">
          <output width="560" height="350"/>
          <crop/>
          <resize/>
        </representation>
        ....
      </representations>
    </field>
  </panel>
  <summary>
    <field name="title" type="basic" mime-type="text/plain"/>
    ...
    <field name="alternates" type="basic" mime-type="application/json">
      <ui:hidden/>
      <representations xmlns="http://xmlns.escenic.com/2009/representations"
type="image-versions" inherit="true"/>
    </field>
  </summary>
</content-type>
```

You will also need to make corresponding changes to your template code, to make sure that image alternatives are retrieved from the content item summary, rather than from the content item itself. For details, see

Note that the change described here will have no effect on how images behave in normal Content Studio editing, since Content Studio section page editors do not allow the summary fields of a content item's related images to be edited. It is therefore a good idea to hide the field by adding a **ui:hidden** element, as shown in the example.

## 4.3 JSP Template Changes

To enable preview editing in general, you need to make sure your JSP templates include the following code:

```
<c:if test="${requestScope['com.escenic.context.isPreview'] == 'true'}">
  <c:url var="js" value="/js/cross-site-preview.min.js" scope="request"/>
  <script
    type="text/javascript"
    src="${js}"
    data-studio-host="http://editorial-host:port"></script>
</c:if>
```

where *editorial-host* is the domain name or IP address of your editorial host. For more information about this code and its purpose, see [http://docs.escenic.com/ece-server-admin-guide/configuring\\_cross\\_site\\_preview.html](http://docs.escenic.com/ece-server-admin-guide/configuring_cross_site_preview.html). Note, however, that this code is **required** by the Preview Editor, even on single-host installations.

### 4.3.1 Enabling Editing at Field Level

You also need to enable preview editing at field level. For every field that you want to be able to edit:

1. Find the JSP code that displays the field.
2. Add a reference to the Preview Editor CSS classes.

The Preview Editor CSS classes are provided by the **preview-editor** decorator, which adds them to the **PresentationArticle** and **PresentationElement** beans representing content items. They are added to the **options** property of each field in a content item, using the key **preview-editor-classes**. You can therefore typically access them with the following JSTL expressions:

- `${article.fields.field-name.options['preview-editor-classes']}` for basic fields (plain text only) on content item pages
- `${element.fields.field-name.options['preview-editor-classes']}` for basic fields (plain text only) on section pages
- `${image.fields.alternates.options['preview-editor-classes']['crop']}` for images (where *crop* is the name of the required image variant)

**article**, **element** and **image** in the above examples are variable names: different variable names may be used in your JSP files.

Here, for example, is a simple JSP file for displaying teasers on section pages, with the preview editing additions highlighted:

```
<!--@elvariable id="element" type="neo.xreditsys.presentation.PresentationElement"--%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<div class="row">
  <c:if test="${not empty element.fields.images.value}">
    <div class="col-md-4">
      <c:set var="image" value="${element.fields.images.value[0]}" />
      <!--@elvariable id="image"
type="neo.xreditsys.presentation.PresentationElement"--%>
      <!--@elvariable id="crop" type="java.lang.String"--%>
      

    <p class="{image.fields.caption.options['preview-editor-classes']}">
        {image.fields.caption}
    </p>
</div>
</c:if>
<div class="col-md-8">
    <h4 class="{element.fields.title.options['preview-editor-classes']}">
        {element.fields.title}
    </h4>

    <p class="{element.fields.leadtext.options['preview-editor-classes']}">
        {element.fields.leadtext}
    </p>
    <a href="{element.content.url}">Read more</a>
</div>
</div>

```

### 4.3.2 Enabling Section Page Image Editing

In order for the image representations on section pages to be edited independently of the original image representations, the image content type must have a copy of its "alternatives" field in its summary, as described in [section 4.2](#). You then also need to modify your section page template code to access these summary representations rather than the original image representations.

This JSP code for displaying an image, for example:

```

<c:set var="imageRep"
    value="{summaryElement.content.fields.alternates.value['BASE_LANDSCAPE_SMALL']}" />


```

would need to be modified to:

```

<c:set var="imageRep"
    value="{summaryElement.fields.alternates.value['BASE_LANDSCAPE_SMALL']}" />


```

### 4.3.3 Highlighting Editable Texts and Images

When preview editing is enabled, the Preview Editor adds CSS classes to editable text and image elements, allowing them to be highlighted. You then need to add some CSS definitions to add the highlighting you want. Including the following code in your JSP templates, for example, will add a yellow background to editable text and a yellow border around editable images. If an image is selected then its border will turn into red indicating it is now able to editing, mouse out will make it to non-editable mode with yellow border:

```

<style type="text/css">
    .preview-editor[contenteditable=true] {
        background-color: #ffff00;
    }
    .preview-editor-image-editable {
        border: solid #ffff00 2px;
    }
    .preview-editor-zooming-enable {
        border: solid #ff0000 2px;
    }

```

```
}
</style>
```

## 4.4 Enable Preview Editing

By default preview editing is disabled for all the section pages. To enable it needs to configure as followed:

1. Create `PreviewConfiguration.properties` file

```
mkdir -p studio-config/com/escenic/studio/preview/config/
vi studio-config/com/escenic/studio/preview/config/PreviewConfiguration.properties
```

2. Set `configurations.preview.editor.sectionUniqueName.enable` to `true`. Where `sectionUniqueName` is replaceable with the unique name of the section which will have preview editing enabled. For example:

```
configurations.preview.editor.ece_frontpage.enable=true
configurations.preview.editor.news.enable=true
```

## 4.5 Application Configuration Layer Changes

The `cross-site-preview.min.js` script mentioned in [section 4.3](#) is generated by a Java servlet called `PreviewScriptServlet`. This servlet generates `cross-site-preview.min.js` by merging a number of different Javascript scripts into a single file. The merged scripts include a number of well-known open source packages such as JQuery, so it is quite possible that you already use some of them on your site and load them yourself elsewhere on the page.

In order to avoid scripts being loaded twice, you need to check what is included in `cross-site-preview.min.js` by default, and if necessary configure `PreviewScriptServlet` to exclude scripts that you are loading yourself.

You can find out what scripts are being loaded by default as follows:

1. Start a browser and go to your escenic-admin webapp (`http://host:port/escenic-admin/`)
2. Click on the **Component Browser** option
3. Click **Browse other scope**
4. Pick your publication web app from the displayed list
5. Pick **com > escenic > servlet > PreviewScriptServlet**

You can also get straight to this page using the URL `http://host:port/escenic-admin/browser/Webapp%20publication-webapp-name/com/escenic/servlet/PreviewScriptServlet`.

You should now be able to see the default set of merged scripts listed under **resources**, something like this:

```
resource.0.1.0-jquery=/webjars/jquery/1.10.2/jquery.min.js
resource.0.1.1-jquery.mousewheel=/webjars/jquery-mousewheel/3.1.11/
jquery.mousewheel.js
resource.0.1.2-jquery.panzoom=/webjars/jquery-panzoom/2.0.5/jquery.panzoom.min.js
```

```
resource.0.2.1-jquery.ui.core=/webjars/jquery-ui/1.10.2/ui/minified/  
jquery.ui.core.min.js  
resource.0.2.2-jquery.ui.widget=/webjars/jquery-ui/1.10.2/ui/minified/  
jquery.ui.widget.min.js  
resource.0.2.3-jquery.ui.mouse=/webjars/jquery-ui/1.10.2/ui/minified/  
jquery.ui.mouse.min.js  
resource.0.2.4-jquery.ui.resizable=/webjars/jquery-ui/1.10.2/ui/minified/  
jquery.ui.resizable.min.js  
resource.5.3.1-controls-editor=/js/preview-editor.min.js
```

If you already load any of these scripts, then you will need to create an application configuration layer and add a **PreviewScriptServlet.properties** file that prevents them being included in **cross-site-preview.min.js**.

To create an application configuration layer, simply create a folder called **localconfig** in your publication webapp's **WEB-INF** folder. You then need to add a **PreviewScriptServlet.properties** file to the layer in the correct location (i.e *publication-webapp/WEB-INF/localconfig/com/escenic/servlet/PreviewScriptServlet.properties*).

To prevent a particular script being included, you simply need to set the appropriate property to an empty string. Your setting will then override the default configuration. To prevent JQuery from being included in **cross-site-preview.min.js**, for example, you would add the following line to your **PreviewScriptServlet.properties** file:

```
resource.0.1.0-jquery=
```

Remember that you will need to add a similar **PreviewScriptServlet.properties** to the application configuration layer of every publication in which you use the Preview Editor.