

XLsuite

A Designer's Guide
to the XLsuite Content Management System

About XLSuite

XLSuite is a fully hosted CRM/CMS/ERP web-based application built using Ruby on Rails.

The XLSuite Contact Management System (**CMS**) is a feature rich, powerful sub-application that helps developers, designers, and business owners rapidly deploy and manage web content.

The unique, nested structure of the XLSuite CMS is based around three basic components – **Layouts, Pages, and Snippets** – a system that offers unparalleled flexibility within a highly-structured publishing environment.

We hope this guide will be of use to web designers – hopefully already familiar with **HTML, CSS, and JavaScript** – in setting up and maximizing the potential of XLSuite-driven websites.

Part 1: Content Management System

XLsuite Menu



The majority of XLsuite functions are accessed through drop-down menus stemming from the horizontal navigation bar that dominates the top of the window.

Almost all of the CMS applications are located under RESOURCES>

Common interface elements

There are several elements that are common both to the XLsuite interface and to the specific areas under RESOURCES:

| | |
|--|---|
| | <p>XLsuite uses a tab system for various applications. Close the tab by clicking on the X button on the right-hand side of any tab.</p> |
| | <p>Columns on any Index Page can be sorted and organized to the user's preference.</p> <p>Click on the blue arrow icon within the column heading to access the column preferences.</p> |
| | <p>At the bottom of the interface there are options to navigate through multiple pages, select the amount of items per page, and a button that will refresh the current page (shown).</p> |
| | <p>Under RESOURCES > CMS > Pages, there are icons (shown) that trigger the EDIT mode for the selected page.</p> |
| | <p>The "bug Buster" button (shown), lower right-hand of the interface allows users to report bugs and issues to the XLsuite team.</p> |

CMS

Under RESOURCES > CMS there are the following menu items:

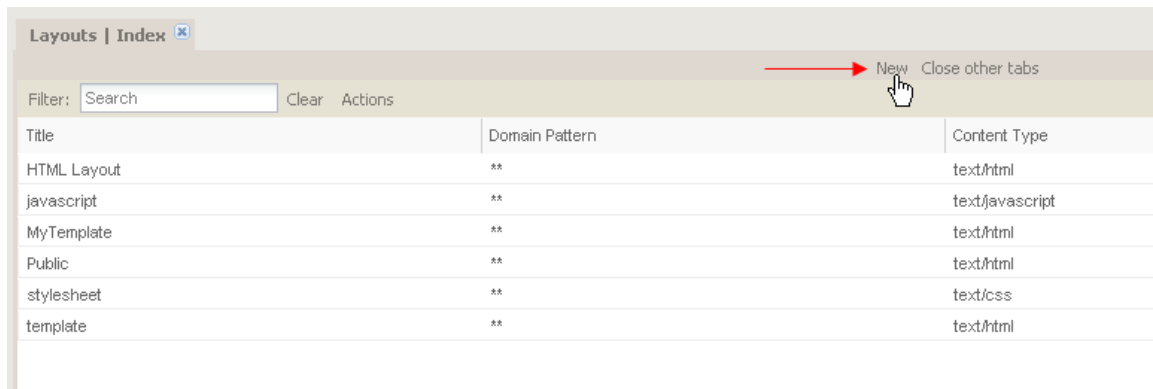
- **Layouts** – HTML Layouts that govern the structure of individual pages.
- **Pages** – where content is stored.
- **Snippets** – small sections of code and content that can be repeated across a multitude of pages.

Layouts

Layouts are the foundational HTML code that presents content to the viewer. One layout can be attached to many different pages. Designers and administrators can create page content and then assign a layout to particular pages. Common application elements such as Forums, Contact Pages, Catalogue etc., for example, are by default (as of the time of writing this manual) assigned the PUBLIC layout.

Creating New Layouts

- Navigate to RESOURCES > CMS > Layouts to open the **Layouts Index**
- Click on the “New” link to open up a brand-new layout.



| Title | Domain Pattern | Content Type |
|-------------|----------------|-----------------|
| HTML Layout | ** | text/html |
| javascript | ** | text/javascript |
| MyTemplate | ** | text/html |
| Public | ** | text/html |
| stylesheet | ** | text/css |
| template | ** | text/html |

The **Layout Edit** view is fairly straightforward. Users can fill in the following fields:

| | |
|--------------|---|
| TITLE | Assigns a title to the layout, this title will be referenced by pages, so it is best to choose something descriptive and obvious such as "DefaultLayout". |
| Content Type | Can be one of several options, and dictates how the layout, and child pages are interpreted by the browser. Standard uses are: <ul style="list-style-type: none">• text/html – for normal HTML pages• text/css – necessary to create CSS (Cascading Style Sheets)• text/javascript – necessary to create pages that contain javascript. |
| ENCODING | Character set for the code that follows – best left at default value, as most HTML, CSS and Javascript use UTF-8 as a base character set. |
| BODY | Where the code goes! |

There is another field for **Domain Patterns**, which is not covered here, as well as **Security Settings** which are also not covered here. These areas can, generally, be left as they are without impeding the functioning of an XLSuite website.

Preparing Layout pages to receive content

Every layout page MUST contain the following **Liquid Tag** (Code used by XLSuite for rendering) in order to function:

```
{{page.body}}
```

This tag attaches any **Page** content directly to the layout. A simple HTML layout might look like this:

```
<html>
  <head>
    <title>A Simple HTML Layout for XLSuite</title>
  </head>
  <body>

    {{page.body}}

  </body>
</html>
```

Saving this **Layout**, and then assigning it to a **Page** would render a very basic HTML page in any web browser. Note the inclusion of the `{{page.body}}` **Liquid Tag**.

Preparing Layout Pages for titling

XLsuite has a handy feature that will automatically assign a title to a page (from the Pages application), but in order for this to work the following **Liquid Tag** must be inserted in the **Layout**

```
{{page.title}}
```

This tag **MUST** be positioned in the HEAD of a standard HTML layout in order to function. This replaces the need to assign a title to each page: saving a **Page** will assign its title automatically.

```
<html>
  <head>
    <title>{{page.title}}</title>
  </head>
  <body>

    {{page.body}}

  </body>
</html>
```

Adding standard HTML to a layout page

Standard HTML can be applied to **Layout** pages. From a design perspective, it's best to think of **Layouts** as templates, where content can be added to a fixed, repeated structure. Keep in mind that **Snippets** (covered further on in this manual) can be added to a **Layout** to enhance its flexibility.

This is an example of what a standard **Layout** might look like (this code would be inserted into the BODY field on the Layout | Edit page):

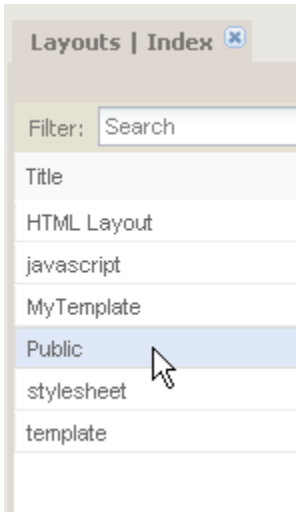
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html;
charset=utf-8">
    <title>{{page.title}}</title>
  </head>
  <body>
    <div id="header">
      <h1>MyTemplate</h1>
    </div>
    <div id="navigation">
      <a href="/">Home</a>
    </div>
    <div id="content">
      {{page.body}}
    </div>
    <div id="footer">
```

```
<p>Copyright 2008 IXL Media</p>
</div>
</body>
</html>
```

Public Layouts

Currently (March 2008) there are a number of modules that ship with XLSuite that, by default, look for a Layout named "Public" in order to render on a browser. Examples would be Forums, Warning Pages (404 error, etc), Contact returns, and so on.).

When constructing a site using XLSuite this layout needs attention: whatever conceptual design is used for the overall site must be applied to this layout in order to achieve continuity.

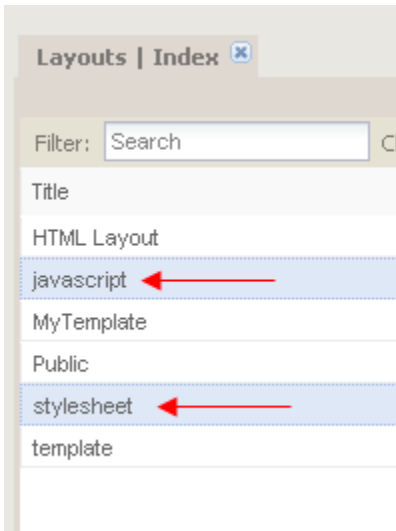


The default XLSuite setup includes a **Layout** with the title "Public". Simply apply your design to this template:

- Navigate to RESOURCES > CMS > Layouts
- Double click on the "Public" row
- Replace the code in the BODY field with whatever layout you wish to use for default XLSuite module pages.

This layout will, by default, be applied to any embedded XLSuite modules that can be used in a website.

CSS and JavaScript layouts



These are included by default in the XLSuite package and do not need any modification. When you create a CSS or JavaScript page (covered later in this manual) you simply assign that Layout (e.g. When creating a new CSS page be sure to assign the "Stylesheet" Layout to that Page).

Pages

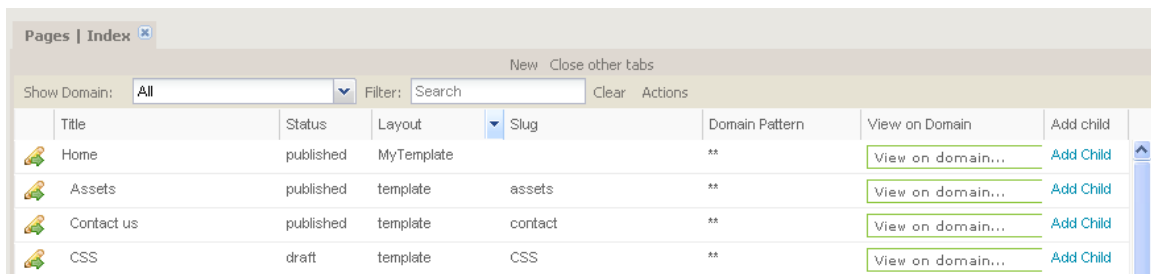
Pages contain content – text, HTML or XLSuite **Liquid Tags** – that will be rendered and presented on the web. In XLSuite, **Pages** have several unique characteristics:

- Pages can be assigned to different layouts
- Pages can be published (on the website) or remain in draft form
- Pages can be created and edited using rich text (WYSIWYG) or directly from the source code
- XLSuite site administrators can control individual user access to page edits

Pages Index

- Navigate to RESOURCES > CMS > Pages

The **Pages Index** lists any pages associated with the domain XLSuite is currently running on:



| Title | Status | Layout | Slug | Domain Pattern | View on Domain | Add child |
|------------|-----------|------------|---------|----------------|-----------------------------------|---------------------------|
| Home | published | MyTemplate | | ** | View on domain... | Add Child |
| Assets | published | template | assets | ** | View on domain... | Add Child |
| Contact us | published | template | contact | ** | View on domain... | Add Child |
| CSS | draft | template | CSS | ** | View on domain... | Add Child |

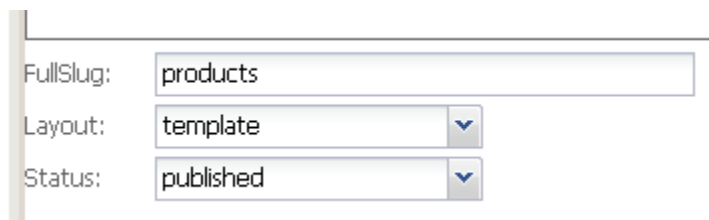
XLSuite site management is based on parent-child relationships. In the above illustration the first page (Home) is the *parent* page; anything that follows is a *child* (indicated by a slight left indent). As long as an XLSuite site has a parent page published there any number of cascading relationships can ensue.

New Page

To create a **Page** click on the *Add Child* link to the far right of any row with an existing page (add a child page) or simply click the *New* link on the tab bar.

The following fields are available:

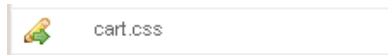
| | |
|----------------------------|---|
| TITLE | Assigns a title to the page which will display: <ul style="list-style-type: none">• In the browser• In the Pages index |
| PLAIN TEXT or HTML CONTENT | Can be entered as either title suggests. Plain text is displayed “as is” and HTML tags are rendered by the browser in a standard fashion. |
| FULLSLUG | Where the page resides in the parent/child relationship. <ul style="list-style-type: none">• The “Home” page is left blank• A <i>child</i> of the “Home” page would be entered as titled, WITHOUT a slash preceding the text: e.g. <code>contacts</code>• A child of a child requires a slash to establish hierarchy: e.g. <code>contacts/thankyou</code> |
| LAYOUT | Choose a layout from the drop-down list which will then be assigned to the page. |
| STATUS | <ul style="list-style-type: none">• Draft – will not be displayed but can be further edited.• Reviewed – ??• Protected – will be displayed but editing privileges are delineated. (security)• Published – displayed. |



A screenshot of a form with three fields. The first field is labeled 'FullSlug:' and contains the text 'products'. The second field is labeled 'Layout:' and contains the text 'template' with a small downward arrow icon to its right. The third field is labeled 'Status:' and contains the text 'published' with a small downward arrow icon to its right.

Domain patterns and security settings and are not covered in this document.

Edit Page



To open any page in **Edit Page** view, click on the “edit” icon to the left of any page from the Pages Index.

Refer to the **New Page** field chart for details; in **Edit Page** you have access to the same options.

Creating a CSS or JavaScript page

```
#cart thead td {  
padding: 10px;  
background: #464646;  
color: #EEE;  
text-align: center;  
}  
  
#cart a {  
color: #333;  
text-decoration: underline;  
font-weight: bold;  
}  
  
FullSlug: template  
Layout: stylesheet  
Status: published
```

CSS and JavaScript can be entered in the PLAIN TEXT or HTML CONTENT field of the Pages Edit view.

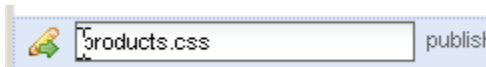
A layout needs to have been created for CSS and JavaScript (see above), which must then be referenced from the Layout drop-down menu.

Editing Status, Layout, Slug from the Page Index

| | | |
|-----------|-------------|-------------|
| published | Stylesheet | CSS/produ |
| published | Html Layout | CSS/profile |
| published | Javascript | CSS/style.x |
| published | My Template | domains |
| published | Public | feeds |
| published | Stylesheet | layouts |
| published | Template | |

Certain Page parameters (Status, Layout, and Slug) can be set directly from the Page Index.

To make changes single-click on a field and edit the text, or make a selection from the drop-down menu.



Snippets

Snippets are a powerful feature of the XLSuite CMS that allow chunks of code and/or content to be referenced from any Layout or Page and rendered in the browser. Using snippets, designers can deploy a single block of code on a multitude of pages across a site – snippets are affected by security controls as well, meaning a designer can allow or restrict access to specific code or textual content.

Snippets Index

To access the **Snippets Index**:

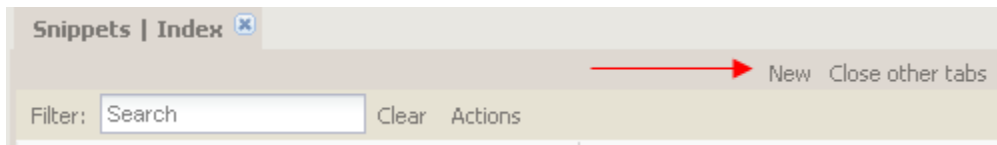
- Navigate to RESOURCES > CMS > Snippets

Select column sorting options by single clicking any column header and accessing the drop down menu on the right-hand side (See “Common Interface Elements” above).

Snippets have no cascading order and no parent/child relationship and each **Snippet** is unique, though it can be used an infinite number of times throughout the site.

Create New Snippet

Create a **New Snippet** by clicking on the “New” button in the **Snippets Index** view.



The following fields are available:

| | |
|----------------------------|---|
| TITLE | Assigns a title to the snippet which will be used as a reference when calling it from a layout or page. |
| PLAIN TEXT or HTML CONTENT | Can be entered as either title suggests. Plain text is displayed “as is” and HTML tags are rendered by the browser in a standard fashion. |

Domain Patterns and Security Settings are not covered in this document.

Edit Snippet

From the **Snippets Index** view simply single-click on the title of the **Snippet** you wish to edit.

See the above chart (create New Snippet) for field references.

Referencing Snippets from Pages and Layouts

The code or text in a **Snippet** can be called from any **Layout** or **Page**, any number of times, by using the following **Liquid Tag**:

```
{% render_snippet title: 'mySnippet' %}
```

where 'mySnippet' is replaced by the EXACT title of the Snippet. Snippets can be enclosed in HTML tags:

```
<ul id="navigation">
  {% render_snippet title: 'mySnippet' %}
</ul>
```

Whatever content the Snippet contains will be displayed from the Page or Layout that references it.

Snippet Strategy

The content of any Snippet is rendered by the browser "as is", without any additional styling or markup. Therefore one ought to consider how Snippets will be used within the greater context of a site.

For snippets that are accessible by site editors who are not concerned with HTML code, let's say, it would be safest to "wrap" the Snippet Liquid Tag in HTML in order to attach CSS to the content of the Snippet. If I create a Snippet called "landing_page_greeting" that is then filled with plain text by a content editor:

```
Hi welcome to our site. Have a look around and we hope you
have fun.
```

I would want to make sure that that Snippet is accessible to any CSS I may want to apply to the site:

```
<p class="greeting">{% render_snippet
  title="landing_page_greeting" %}</p>
```

Snippets that have a more practical use as shortcut or re-used code (such as navigation) can contain HTML. In this case it is easiest to think of **Snippets** as **server-side includes**.

Assigning Permission controls to Snippets

While assigning **Permissions and Security** on a global (site-wide) level are not covered in this particular manual, it should be noted that *each particular Snippet* can be access-controlled by checking off the applicable security groups:

These groups can edit this snippet:



A screenshot of a web interface showing a list of security groups. The groups are: Access, Admin, and New Listing List. Each group has a checkbox to its left. The 'Admin' group is highlighted with a light blue background.

In this example on the right, ticking one or more of the items in the list gives anyone from that group permission to edit the snippet

This will not affect anyone who has ALL permissions for the site, but rather for

people who have been assigned to a SPECIFIC group.

In this way one can limit editing access to very particular areas of a site.

Use Snippets across multiple domains

Multi-Domain management, while not extensively covered in this manual, can allow Snippets to be reused across several XLSuite sites at once (as includes). Thus a Snippet that is saved on one site can be assigned to multiple domains and sub-domains. This feature makes it very easy to deploy many similar sites using a limited amount of code.



A screenshot of a web interface showing a text input field labeled 'Domain Patterns:'. The field contains the text '**'.

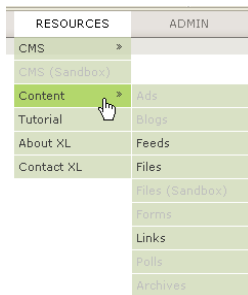
Using the **Domain Pattern** input for a particular Snippet, one could allow this Snippet to be viewed on, for example, www.hotelmanagement.com, www.hotelbooking.com, and www.hotelconstruction.com (provided one owned all those domain names and they were built on XLSuite!). For this one would set the Domain Pattern for this snippet to:

Hotel*.*

Now I can reference that Snippet from any of my "hotel*.com" sites without having to re-write or copy it into the Snippets Index.

More details on Domain Patterns can be found on the XLSuite [wiki](#).

Part II: Content



XLSuite Content applications can be accessed from **RESOURCES > Content** in the XLSuite menu.

Under Content the following features are available:

- Feeds – display RSS feeds directly on an XLSuite site
- Files – store, manage, and reference files: images, video, PDF etc.

Feeds

XLSuite includes its own [RSS](#) feed aggregator which can be used to display, publish, or disseminate frequently updated content.

Feeds Index

To access the **Feeds Index**, navigate to:

- **RESOURCES > Content > Feeds**

Each feed is displayed across the following columns:

- Action – The feed can be refreshed by clicking on the green arrows (may be necessary if the feed is not displaying, or is displaying improperly) or deleted by clicking on the red circle.
- Label – Displays the title of the feed.
- Feed URL – displays the URL of the feed.
- Tags – displays tags that are associated with the feed.



New Feed

Add a **New Feed** by clicking on the “New Feed” link.

The following fields are available:

| | |
|-------------|---|
| Label | The title of the feed. Used to reference the feed from the CMS. |
| URL | The URL (address) of the RSS feed. |
| DESCRIPTION | Optional – Add a description of the RSS feed. |
| TAGS | Optional but recommended – can be used to reference the feed from the CMS |

Edit Feed

From the Feeds index page double-click on the feed URL. Edit fields are referenced in the chart above, as they are the same as the New Feed view.

Publishing a feed

Feeds can be published from a Layout, Page, or Snippet by using the following Liquid Tag syntax:

```
{% render_feed |selection options| |other options| %}
```

The following options are available:

| Selection options | Description |
|-------------------|--|
| named: | Displays a feed with the exact title (label) specified |
| tagged_any: | Displays content from feeds with ANY of the specified tags – separated by commas. |
| tagged_all: | Displays content from feeds with ALL of the specified tags – separated by commas. |
| Other Options | |
| display: | Displays the feed(s) based on ONE of three criteria: <ul style="list-style-type: none">• summary – displays the feed with the content truncated• full – displays all of the feed content.• title – displays only the title of the feed. Default: summary |
| truncate_to: | Sets the length (in characters) of truncated content when the “summary” display option is used. Default: 255 characters |
| count: | Sets the number of feed items that will be displayed. Default: 5 items |
| order: | Sets the order feeds will be displayed based on ONE of two options: <ul style="list-style-type: none">• random – no order• by_date – displays feed items from newest to oldest. Default: random |

Here is an example of a Liquid Tag using some of the options described above:

```
{% render_feed name:"myFeed" count:7 order:"random"
↳truncate_to:100 display:"summary" %}
```

In this example I am rendering a feed with the title “myFeed”, which I’ve set to display 7 items at random, with each item truncated to a summary view of 100 characters.

The **Feeds Liquid Tag** can be enclosed in other HTML tags.

Combining Feeds using ‘tagged_any’

It is possible to combine input from several different feed sources by using the ‘tagged_any’ syntax in the **Feeds Liquid Tag**.

Let us suppose we have the following feeds in our Feeds Index:

| Feed URL | Tags |
|--|-------------------------------|
| http://feeds.feedburner.com/beyondrobson | beyond_robson, st_news |
| http://citizenvancouver.com/?feed=rss2 | citizen_vancouver, st_news |
| http://feeds.feedburner.com/philipstephen | philip_stephen, st_news |
| http://downtowneastside.blogspot.com/feeds/posts/default?alt=rss | oldtown_news, st_news |
| http://stephenrees.wordpress.com/feed | stephen_rees, st_news |

Note that each feed has been assigned the tag **st_news**, among others. We can now use that tag to reference and amalgamate several feeds at once using this syntax:

```
{% render_feed tagged_any:"st_news" display:"title" count:7 %}
```

This Liquid Tag will display the title of any 7 feeds tagged “st_news”.

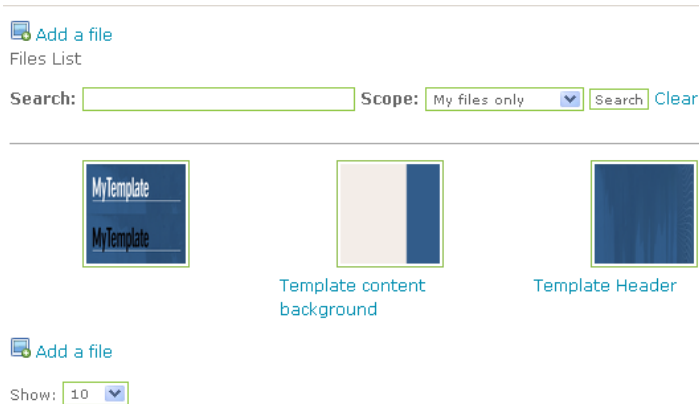
Assets Manager

The XLSuite Asset Manager is a database storage application that allows you to upload and store files such as MP3 audio, shockwave/flash, graphic files (.jpeg, .gif, .png), PDF and other documents (xls, doc, txt,) plus many other formats, and then also to reference, display, or publish files from a **Layout, Page, or Snippet**.

Assets Index

Access the **Assets Index** by navigating to:

- RESOURCES > Content > Files



The screenshot shows the 'Assets Index' interface. At the top left, there is a link 'Add a file' and the text 'Files List'. Below this is a search bar with the text 'Search:' followed by an input field. To the right of the input field is a dropdown menu labeled 'Scope:' with the value 'My files only' and a 'Search' button. Further right is a 'Clear' button. Below the search bar, there are three file thumbnails. The first is a blue square with 'MyTemplate' text. The second is a light blue square with the text 'Template content background' below it. The third is a dark blue square with the text 'Template Header' below it. At the bottom left, there is another 'Add a file' link and a 'Show: 10' dropdown menu.

From the **Assets Index** you can view a listing of all available files, with previews representing graphic files. There is a search field available to track down the exact file or exact range of files.

Add Assets

To add an asset to the database click on the “Add a file link”.

The following fields are available:

| | |
|-----------------------|---|
| FILE | Searches for the file on a local computer – cut and paste the location of the file on your hard drive into the field or click on “browse” to navigate to the file through your OS explorer. |
| UNZIP AFTER UPLOADING | Will automatically unpack files using the ZIP (.zip extension) format. |
| TITLE | The title of your file(s) – used to locate the file from the search field on the Assets Index page. |
| DESCRIPTION | Describe the file(s) – also used to locate a file or group of files using a search. |
| Tag List | Tag your files for easy reference |

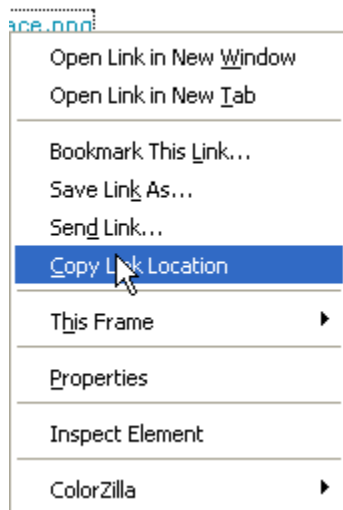
Security settings are not covered in this document.

Assets Detail/Editing Assets

Click on an asset to access the **Assets Detail** view.

From this view you can click on a link to **download** the file to your local hard drive, **delete** the file, and see some details about the file itself. You can click **edit** to alter the details for the file or to **upload** the file again (replacing the original).

Linking to assets from a **Layout**, **Page**, or **Snippet**



The easiest way to reference a file through the CMS is, in either the **Assets Index** view, or the **Assets Detail** page, is to right click on the link for the file and select “Copy Link Location”. You then will have the link location stored on your OS clipboard and can paste that link in the CMS to:

- Reference files for downloading/viewing through the <a> tag
- Reference images through the tag or in the CSS background-image property

Adding Images *En-Masse*

The XLSuite **Assets Manager** has a handy feature that allows a user to upload multiple image files all at once, in a ZIP file (.zip) that will be automatically unpacked on the server and assigned to a folder.

This is particularly useful for designers who build sites using a packaged application, such as Adobe Creative Suite.

File:

 Unzip after uploading

Files unpacked by XLSuite will be moved into the “z” directory. If, for example, you are building a site locally – using Adobe Dreamweaver – you would place all your images (including CSS background images) into a folder

named “z” and then reference that folder and the images within from your HTML using a site-relative URL:

```

```

Once you are ready to deploy your site package all of your images into a ZIP file (.zip) and upload them to XLSuite with the “Unzip after uploading” option ticked. Your images will now be stored in the “z” directory of the site.

Currently (April 2008), only images can be uploaded en-masse.

Epilogue

Well, that covers the fundamental operations in the XLSuite CMS. Further training can be had by downloading and following the XLSuite Template Tutorial, which narrates the step-by-step procedure for setting up a basic Layout and design elements.

Good luck and have fun managing your XLSuite site!

Further Information

HTML/CSS:

[W3C School](#)'s reference for **HTML** Tags
[SitePoint](#) **CSS** reference pages

XLSuite

[Quick Reference](#) for many XLSuite CMS features, plus downloadable [templates](#) for XLSuite sites.

[XLSuite WIKI](#) is a great place for documentation and reference.