

# CommonSpot™

## Release 9.0

### Demo Site Guide

This CommonSpot document, as well as the software described with it, is provided under the CommonSpot License Agreement and may be used, copied, and distributed only in accordance with the terms of the license. The content outlined in this manual is for informational purposes only and is subject to change without notice. By no means should the information contained herein be construed as a commitment by PaperThin, Inc. PaperThin assumes no responsibilities or liability for errors or inaccuracies found in this documentation.

ColdFusion, Acrobat, and Flash are registered trademarks of Adobe Systems Incorporated. Microsoft, Windows, Microsoft Word, Excel and PowerPoint, FrontPage and Windows NT are all registered trademarks of Microsoft Corporation. Solaris is a trademark of Sun Microsystems, Inc., and Verity is a trademark of Autonomy, Inc.

This document was last updated April 16, 2014.



<b>Getting Hands on With CommonSpot Using Commonsport University .....</b>	<b>8</b>
Get CommonSpot University .....	8
Existing On-Premises Customers .....	8
Existing Cloud Customers .....	8
Potential Customers.....	8
Read Demo Site Guide .....	8
Getting Hands On .....	9
Login to CommonSpot University .....	9
Edit Existing Web Page .....	10
Insert Link .....	12
Add an Image .....	14
Other notable Options in the Rich Text Editor.....	16
Preview and Workflow .....	17
View History .....	19
Adding a New Page .....	21
Working with Other Types of Content.....	23
<b>Demo Site User Map .....</b>	<b>25</b>
Demo Site Users and Groups .....	25
Contributors and Admins.....	25
Basic Users .....	25
Additional Users .....	26
Groups .....	26
Additional Groups.....	27
Settings .....	27
<b>CommonSpot Base Site .....</b>	<b>29</b>
<b>Application Development Framework (ADF).....</b>	<b>30</b>
<b>Demo Site Solution Map.....</b>	<b>31</b>
Blogs .....	33
Container Layouts .....	33
Content Freshness, Publication, and Expiration .....	37
Content Personalization: Audience-Specific Content .....	38
Content Personalization: Explicit (Known-User) .....	38
Content Personalization: Implicit (Contextual) .....	39
Content Reuse .....	42
Custom Apps: Simple No-Code App: Jobs .....	45
The Jobs Element .....	45
Jobs Pages .....	46
Jobs Manager.....	49
Field Level Security on the Job Form .....	50
Only Out of the Box Tools .....	51
Custom Element Examples: .....	51
Custom Element Example: Local Custom Element (News) .....	51
Custom Element Example: Local Custom Element (Residence Halls) .....	52

Controlling the Layout of Residence Halls.....	54
To change the List page from one layout to the other.....	56
To change the Detail page from one layout to the other.....	58
Residence Hall Document Libraries.....	58
Custom Element Example: Global Custom Elements.....	59
Custom Metadata.....	60
Custom Metadata to Classify Content.....	60
Custom Metadata to Tag Content for Content Reuse.....	62
Custom Metadata to Capture Additional Page Content.....	63
Custom Metadata to Define Layout Options.....	64
Custom Script (University Bookstore).....	65
Datasheet.....	67
Visitors Survey (Simple Form data in a Datasheet).....	67
All Apps (Custom Element data in a Datasheet).....	67
University Bookstore (Custom Query in a Datasheet).....	68
Document and Asset Management.....	69
Events Calendar.....	70
External User Authentication.....	71
Forms.....	72
Simple Form Example.....	72
Simple Form with Custom Layout.....	73
Simple Form using Custom Element.....	74
Field Level Security.....	75
Enable Calls to Action Field.....	75
CSUTube Approved and Featured Checkboxes.....	76
HTML Design and CSS.....	77
Images.....	78
Carousels.....	79
Other Custom Content Types: News, Blogs and Residence Halls.....	81
In-Context Content Object Authoring.....	83
Landing Pages.....	84
Building Blocks of a Good Landing Page.....	84
Landing Pages on CommonSpot University.....	85
Configuration Layout of Landing Pages.....	85
Built-in Form vs. Custom Form.....	86
Color options via style selector.....	87
A/B Testing on Landing Pages.....	89
Variation A.....	90
Variation B.....	90
Layouts (Display Templates and Render Handlers).....	91
Lead Management.....	91
Visitor Survey.....	91
Marketing Landing Page.....	91
A/B Testing.....	91

Mobile Optimization .....	92
Multi-Language Content.....	92
Multi-Language Pages .....	92
Foreign Language Characters .....	93
My CommonSpot Customization .....	93
My Tags.....	94
Navigation Using Tree Element.....	96
Main Navigation Mega-Menu.....	96
Left Navigation Accordion Menu.....	97
Subsite Home Page Link List.....	99
Navigation Using Linkbar:.....	99
News, Press Releases and Third Party News.....	100
News (Page with Local Custom Element).....	100
Press Releases (Page with Formatted Text Block) .....	100
In the News (Registered URL's) .....	100
RSS Reader .....	100
RSS .....	100
News .....	101
Press Release.....	101
In the News.....	101
Scheduled Content .....	101
Schedule/Personalize Element.....	101
Container with Scheduling Enabled.....	101
Search .....	101
Site-Wide Search.....	102
Specific Subsite Searches .....	102
Social Media .....	102
Social Links.....	102
Social Media Manager .....	102
PT Social Media App (Social Media Stream) .....	103
Styles and CSS .....	104
Choosing a Registered Style from within the Rich Text Editor.....	104
Using the Element Menu to change styles.....	104
Taxonomy and Facet Navigation (Document Library).....	105
The DocLibrary metadata form.....	105
The “documents” subsite (/doclibrary/documents/).....	105
Page Index on Document Library home page.....	106
Facet-Based Navigation on Document Library home page.....	106
Template Overview .....	106
Base Template .....	106
Base Plus One Template .....	107
Soft Templates.....	107
API Only Soft Templates.....	108
Blank Template .....	108

Users and Groups .....	108
Refer to the Administrator Guide for all information regarding users and groups. ....	108
URL Management.....	109
Human Readable URLs .....	109
Redirect URLs.....	109
Video .....	109
Native Video Integration.....	110
Visitor Contributed Video Example .....	110
Web Administration .....	111
Workflow .....	112
WYSIWYG Editing .....	112

# Chapter 1

## Getting Hands on With CommonSpot Using Commonspot University

### 1.1. Get CommonSpot University

There are several ways you can obtain access to CommonSpot University, the CommonSpot demonstration site.

#### 1.1.1. Existing On-Premises Customers

Existing customers with an on-premises license of CommonSpot already have access to the software. CommonSpot University can be installed at the time of the initial installation or it can be installed at a later date. Please see the Installation Guide for more information. We do not recommend installing the demonstration site on your production server.

#### 1.1.2. Existing Cloud Customers

CommonSpot University is not automatically installed into a standard cloud instance. Please contact your account manager for more information about gaining access to CommonSpot University.

#### 1.1.3. Potential Customers

Potential customers have two choices for evaluating CommonSpot, cloud or download.

- **CommonSpot Cloud Evaluation:** PaperThin can create an instance of CommonSpot Cloud. Please contact your sales associate for more information. The cloud instance of CommonSpot University is a fully functioning installation of CommonSpot and can be made available to you for a limited time. This option is best for evaluators who are interested in getting acquainted with the overall feature set. Because there is no access to the file system, customization options are limited to the capabilities of the CommonSpot user interface.
- **CommonSpot Downloaded Evaluation:** PaperThin can provide a fully featured license of CommonSpot for a limited evaluation time period. Please contact your Sales Executive for more information. This version is a fully functioning installation of CommonSpot that will expire after 30 days. Users have access to the CommonSpot user interface and the back-end file system and database. Developers may try out customizations using this model.

### 1.2. Read the Demo Site Guide

Read this guide to fully acquaint yourself with CommonSpot University. Really.

## 1.3. Getting Hands On

The following is a step-by-step guide for touring CommonSpot University. Some steps assume that you have already taken the previous steps.

### 1.3.1. Login to CommonSpot University

There are several user accounts available for use in CommonSpot University. Refer to the section labeled “Demo Site Users and Groups” later in this document for an explanation about each user and the role the user plays in managing content in CommonSpot University. The main administrator user is as follows:

username: webmanager

password: password

To log into CommonSpot University, navigate to any page on the site. Let’s start at the home page. There are two ways to log in:

- Type login.cfm after the final slash “/” in the URL. This might mean replacing the filename. For example, change `http://{yoursite}/demo/about/index.cfm` to `http://{yoursite}/demo/about/login.cfm`
  - Or, from the main drop down menu of CommonSpot University, go to Resources -> Login



You will be presented with a login form. Enter the appropriate username and password.

A screenshot of the CommonSpot Content Server login form. The form has a white background with a blue header that reads 'CommonSpot™ Content Server'. Below the header, there is a message: 'You need to login to access this page. Please enter your username and password and click Login.' The form contains two input fields: 'User ID:' and 'Password:'. Below the 'User ID' field, there is a note: 'Enter the User ID assigned to you.' Below the 'Password' field, there is a note: 'Enter your password.' There is also a checkbox labeled 'Open My CommonSpot'. At the bottom of the form, there are two buttons: 'Login' (with a key icon) and 'Change Password...' (with a key icon).

You have the option of opening My CommonSpot, a dashboard personalized for each user where you will find a list of Pending Page Activities and Notifications that may require your attention. To launch My CommonSpot now, you may opt to click on the checkbox labeled “Open My CommonSpot”. However, to continue back to the website once logging in, leave the box unchecked.

Once you log in, you will arrive back at the CommonSpot University home page, or the default page of the subsite you last visited.

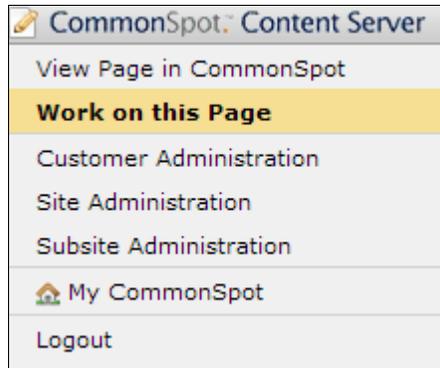
Notice nothing much has changed. You will not see the CommonSpot user interface yet, with the exception of the dimly lit icon (“Entrance Tab”) in the upper right corner of the page.  We'll get to this in a moment.

## 1.3.2. Edit Existing Web Page

1. If you haven't found a page to edit, let's locate a good one now. You may continue to navigate around the site as needed. Let's use the site's navigation to land on the President's Welcome page. This page is located at /demo/about/our-president.cfm



2. Once you find the page you wish to edit, you may invoke the CommonSpot menu by hovering over the entrance tab and click. This will reveal the entire entrance menu. The first two options, “View Page in CommonSpot” and “Work on this Page” will both reveal the CommonSpot menus surrounding the page you are on. The main difference is that “Work on this Page” will also reveal in-context menus, overlaid directly on top of your page for quick access to editing. In our example, we wish to edit this page, so that's exactly the option we want to select. Also, notice that I can get to some of the Admin areas (depending on my permissions) or I can get to the My CommonSpot pages from this menu as well.
3. We came here to edit this page, so select “Work on this Page”.

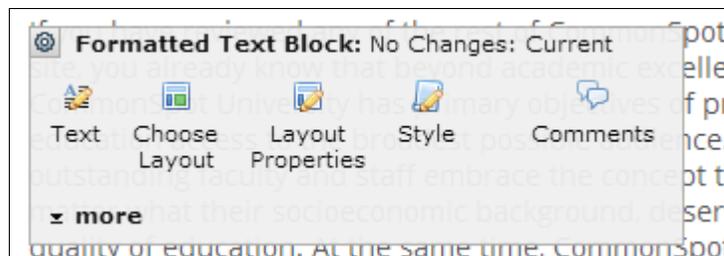


4. This reloads the browser and places the CommonSpot menu across the top and reveals an "element menu" icon directly on the page content.

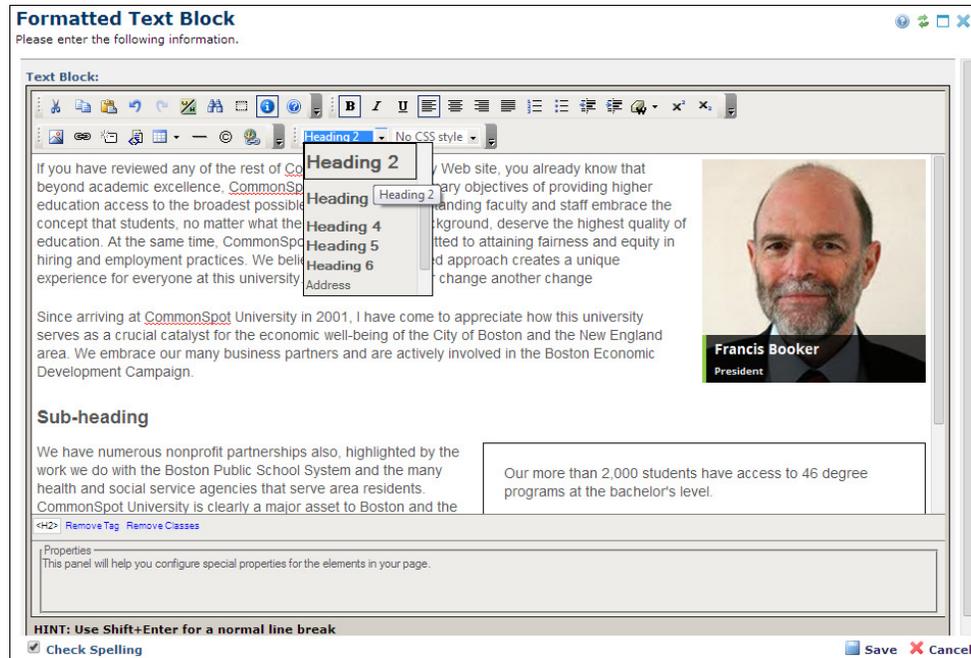


Clicking on the gear reveals the options you have for this particular element. In this case, it's a "Formatted Text Block" element. This element allows you to access the rich text editor and directly edit and save rich text to the page.

5. All element menus will reveal the most common tools directly on the first row when you click on the gear icon. For this element, the most common functionality is the rich text editor, so the "Text" button is the first on the list. Click the text button.



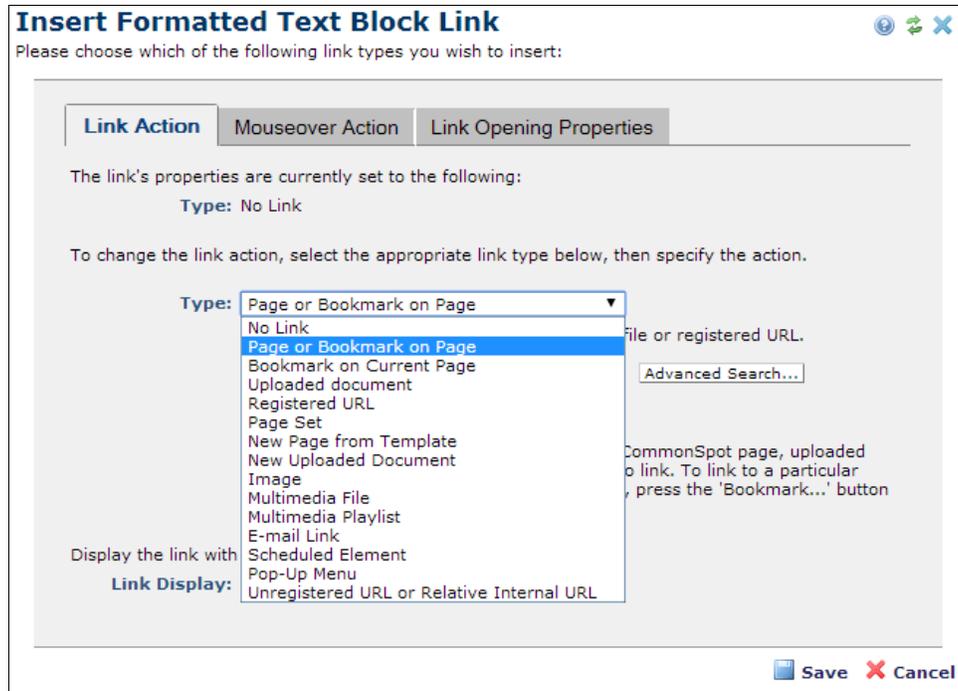
6. This will open up the rich text editor. You can open the editor full screen by clicking on the full screen button  in the upper right corner of the dialog.
7. Now, simply add text by typing into the editor. You can apply styles from your CSS by selecting from the dropdown boxes.



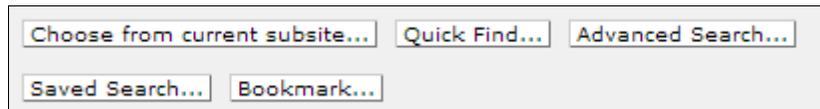
### 1.3.3. Insert Link

Keep the editor open. Now we're going to add a link to the content. CommonSpot has a number of ways to add links to a page. There is an element designed specifically for links, but you can also add links inside the rich text editor.

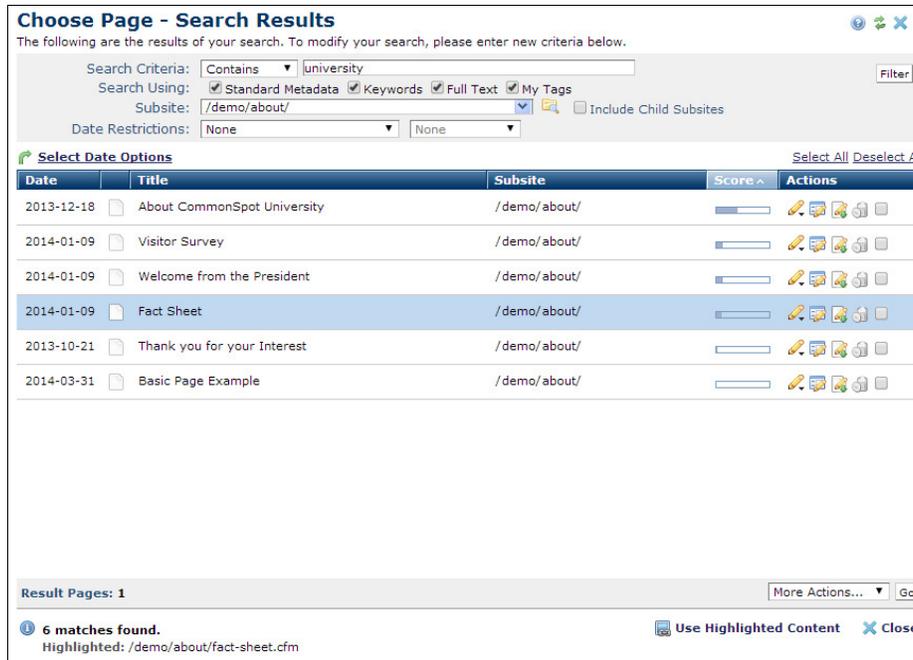
1. To create a link, highlight the text you wish to link and click on the Link tool  in the tool bar.
2. This presents the link wizard interface. You may now choose whether to link to pages, uploaded documents or previously registered URL's. You can also link to pages or documents you haven't yet set up. Select your choice in the "Type" field. For this example, let's link to an existing page.



3. You may now use any of several options for locating a page.



4. Choose the one that helps you most easily locate the page you intend to link to. Let's start with Quick Find, but any option will work.
5. You will be presented with a Choose Page dialog which can help you search your site. Perform a search for a word. I've chosen the word "University".

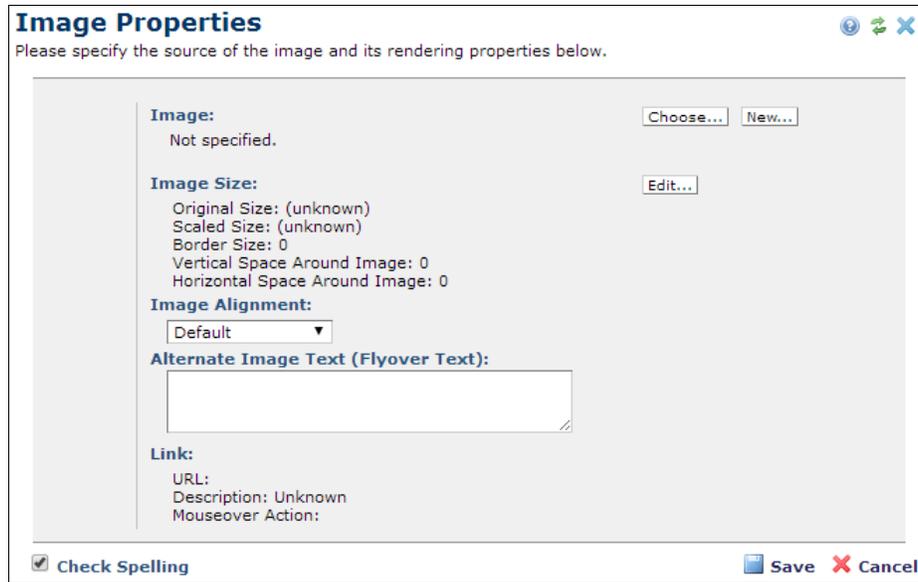


6. Highlight the page you are linking to, and choose Use Highlighted Content in the lower right.
7. Notice how the target page is now listed on the wizard dialog. Save the wizard dialog to return to the page.
8. Repeat these same steps to create other links, including links to uploaded documents or registered URL's.

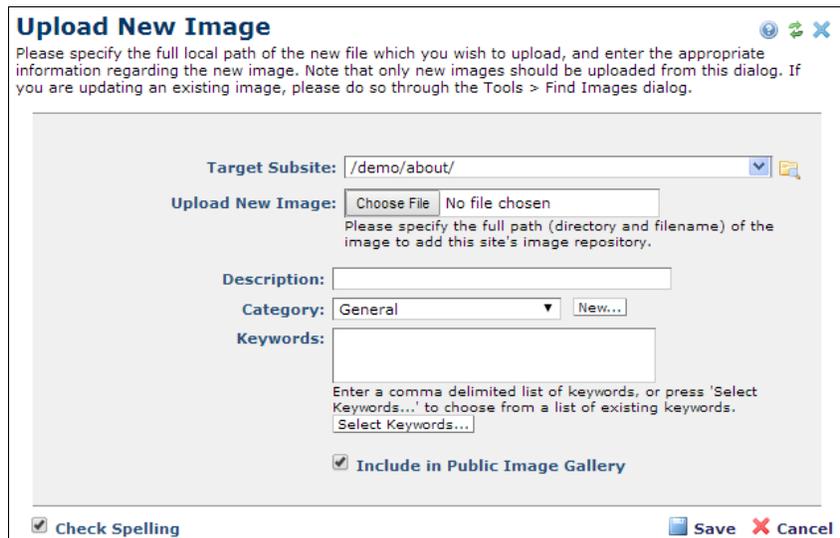
### 1.3.4. Add an Image

Keep the editor open. Now we're going to add an image to the content. CommonSpot has a number of ways to add images to a page. There is an element designed specifically for images, but you can also add images inside the rich text editor.

1. To insert an image, place a cursor where you want the image to go, and then click on the Image tool  in the tool bar.
2. You will be prompted to either **Choose** an image from the existing Image Gallery or upload a **new** one here.



3. Choose New.
4. You will be presented with the Upload New Image dialog. The first step will be to tell CommonSpot where you want the image to go. You do this by selecting a subsite.



5. Now click Upload New Image and locate the image you wish to upload.
6. Select the file you wish to upload.
7. Once selected, you can now fill out the rest of the form. This includes complete descriptive text, a category (from the dropdown of categories created by your admin), and keywords. Also note the "Include in Public Image Gallery". This will allow others to use the same image on their pages.
8. When complete, click "Save".
9. This will complete the task of uploading the image, but not complete the task of adding it to your page. Before you hit Save again, make sure to add the "Alt text", noted with the label "Alternate

Image Text (Flyover Text)” in the dialog.

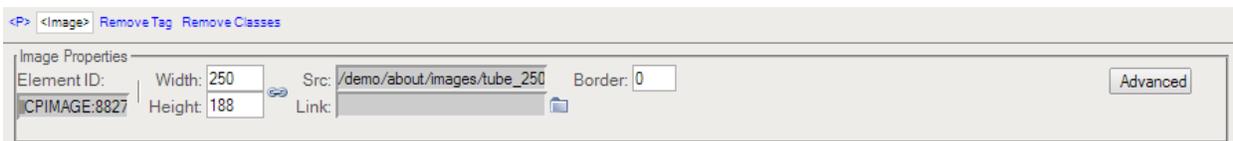
10. For now, let’s leave all other choices as-is. There are some styling choices you can make here and you can also invoke the Link wizard if you wish to turn this image into a link. Let’s just click Save now. Your image will now appear in the content.
11. You can also repeat these steps for adding an existing image to the page. Instead of clicking “new”, you can select “Choose”. It will present a search window, similar to the link wizard, where you can select from existing images.

## 1.3.5. Other notable Options in the Rich Text Editor

While we’re here, let’s take a look at a few components of the rich text editor that may be helpful to you. For each of these, please refer to the documentation for help on how to use or configure them.

### 1.3.5.1. Properties Inspector

First, the region at the bottom is called the Properties Inspector. If you select something in your content, this region may change based on what HTML tag is referenced. In the screenshot below, this is what I see when I select the image in the text. The fields here correspond with the html attributes for the image tag. If I select something else, it will display the attributes for that particular tag. Also, note that the fields are either editable or greyed out. You can directly modify the editable attributes. But the greyed out ones are not editable, as they represent attributes that are computed dynamically by CommonSpot.



### 1.3.5.2. Style Dropdowns

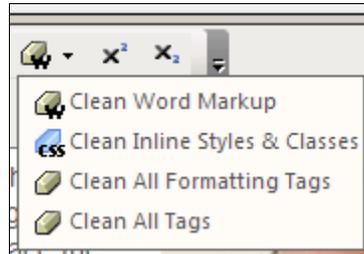
The two dropdown boxes in the toolbar represent style classes. The one on the left is for standard HTML styles, such as heading and paragraph. The one on the right is a list of style classes direct from my CSS style sheet. Not all classes from my stylesheet show up in the list. The admin has to register the class and then bind it to the rich text editor. But once that is done, it appears in this list and is available to the user. Also, in both dropdowns, not only does it list the class by name, but it also displays the class based on the specific style attributes it represents.



### 1.3.5.3. Cleanup Tools

If you paste content from somewhere else, you may have noticed that the raw html may have some additional code that is not critical to the content and may conflict with the styles on your page. This can yield some unpredictable results. CommonSpot has some cleaning tools that you can tap into to avoid this. There are a few ways to use these cleaning tools. One way is to configure paste options so that it is stripped out while it’s being pasted. But if the extra code shows up in your content, you can use the

cleanup tools in the rich text editor as well. The options include just markup coming from Microsoft Word, inline styles and classes, all formatting tags and all tags.



### 1.3.5.4. HTML WYSIWYG Toggle

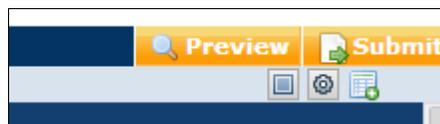
You can easily switch between the WYSIWYG view and the HTML view in the rich text editor. This is done by clicking on the toggle button  in the first row of the toolbar. This button is controlled by permissions. You can control who gets to see the button.

## 1.3.6. Preview and Workflow

CommonSpot supports any workflow approval process that you want to configure and it can be assigned across the entire website, for a specific subsite/folder, or based on an individual page template.

CommonSpot University has a workflow established that requires all content to be approved by a group of approvers. Now that we've made a change to our page, we can see how that workflow works.

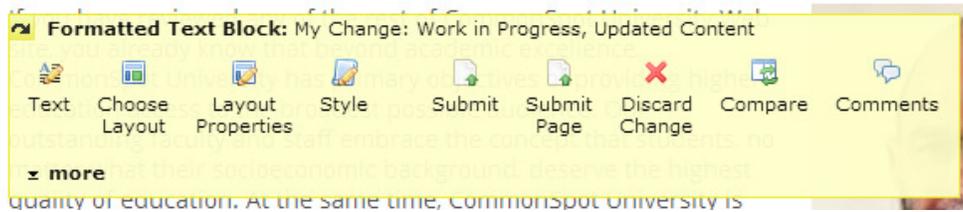
Before we publish our content, let's see what it looks like without all the in-context menus. To remove the menus from the page, you could switch out of the "Work on this Page" mode. But doing so would show you the way the page is currently published, without any of your proposed changes. Instead, there is a Preview button in the upper right corner of the page.



Clicking preview will hide all element tools on the page, giving you an unfettered view of the proposed change in the context of the page. Also note the three icons right underneath. These let you toggle components of the in-context menus one by one. For example, the one in the middle (gear icon) will hide the main element menu link for an element, while the first one (square) will hide a container element. You can either choose any of these toggles independently, or use the "Preview" button to hide them all at once.

Also note, the Submit button appears there too if there are unsubmitted elements on the page. Below, read the description of the "Submit Page" link in the element menu, which accomplishes the same thing as this Submit button does in the upper right.

If you return to the element menu, you'll see some changes. First, the icon has changed from the gear icon to a yellow arrow. If I click on it, I see a refreshed set of options.



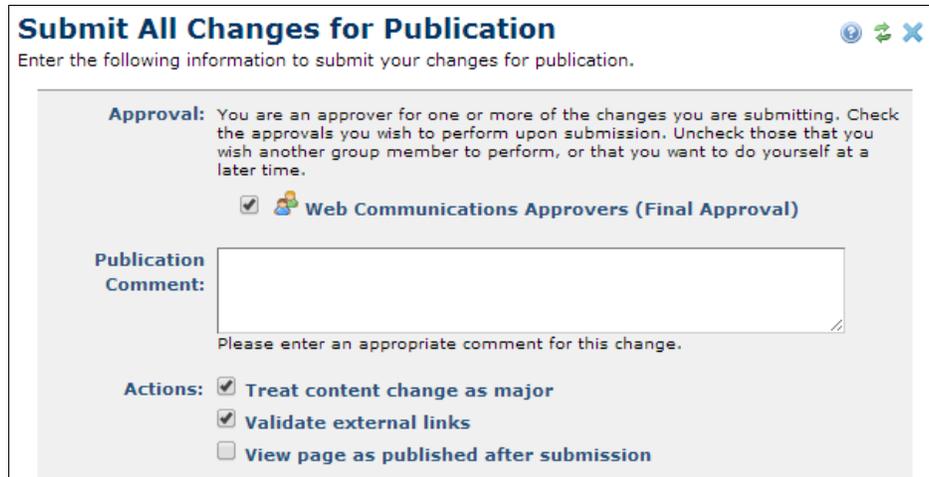
The new options are Submit, Submit Page, Discard Change and Compare.

- Submit: This option will take the content from this one element and push it through to the next step of the workflow.
- Submit Page: This too will submit the content from this element toward the next step of the workflow, but it will also do the same for any other unsubmitted element on the page. This is the same as clicking the orange Submit button in the upper right corner of the page.
- Discard Change: This will undo any changes you made to the element since the last time it was published.
- Compare: This will invoke our visual comparison tool, putting the old version of this element up against the proposed, unsubmitted, changes.

First, let's see the comparison tool. Click on Compare. You will see the Visual Difference tool. It displays a new window with the proposed change (with only the content from this element) on the left and the currently published version on the right. Click the check box at the bottom labeled "Scroll compare windows simultaneously" and then scroll up and down. Note that you can see the changes side by side. Also, note the other tab at the top labeled "Red-lined Comparison". Selecting this option renders all of the content, new and old and highlights the changes accordingly. A red line appears over the deleted content, and yellow highlights over anything that's been added.

Now let's walk through the process of publishing our change.

1. Choose either Submit or Submit Page from the element. You will be presented with the following dialog.



**Submit All Changes for Publication**

Enter the following information to submit your changes for publication.

**Approval:** You are an approver for one or more of the changes you are submitting. Check the approvals you wish to perform upon submission. Uncheck those that you wish another group member to perform, or that you want to do yourself at a later time.

 **Web Communications Approvers (Final Approval)**

**Publication Comment:**

Please enter an appropriate comment for this change.

**Actions:**  **Treat content change as major**  
 **Validate external links**  
 **View page as published after submission**

2. Enter comments that describe the changes made to the content. Refer to the documentation regarding what other options are on this dialog.

Note the Approval section at the top of the dialog. This is telling me that this particular piece of content is subject to approval and the group of approvers is the group called “Web Communications Approvers”.

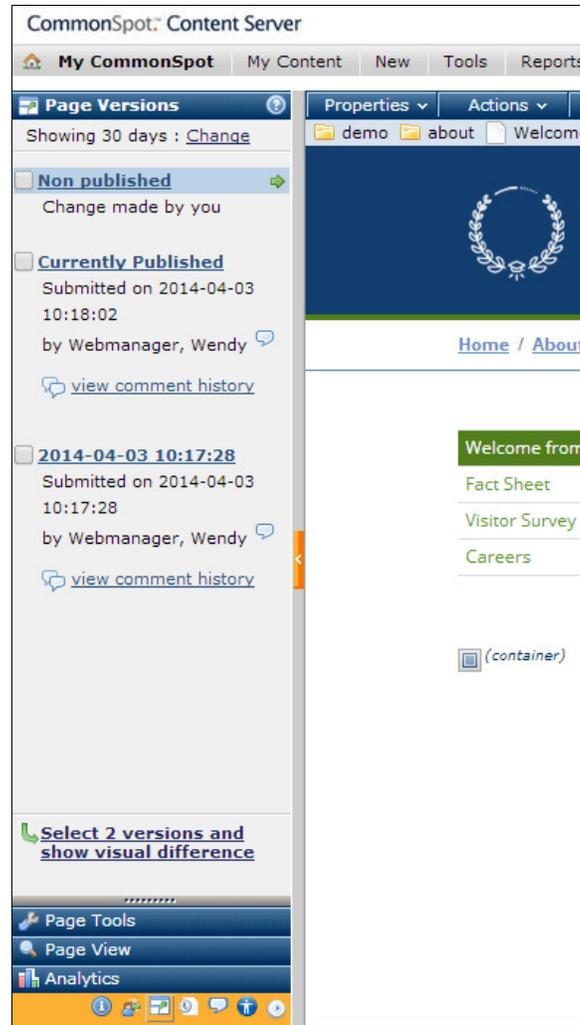
However, I happen to be logged in as Wendy Webmanager. This user is in that group. This gives us the little check box allowing me to instantly approve this content. Under normal circumstances, a non-approval user wouldn’t see the check box.

If I want to approve this right now, I can check that box and hit Submit. If I want someone else in that group to view it, I can uncheck that box and hit Submit.

## 1.3.7. View History

CommonSpot University initially ships with much of the page history removed. This is for efficiency of the installation download. So if you view the history right now, you may not see much. But it’s worth checking out, so repeat the above steps a few times so that you have fully published several changes to this page.

Once you have some changes saved, expose the left panel. Do this by clicking on the orange handle at the far left of your CommonSpot window.



The history of the page will show up. Note that you can see everything with time stamps, including past versions, currently published version, as well as the work in progress version. These are all clickable, so you can opt to view any of these versions simply by clicking on it.

If you view a past version, you will notice the element icon differs. The grey arrow represents the “Revert Content” tool. Choosing this will make a new work-in-progress version of this element and allow you to publish this version again as a brand new version of the page.

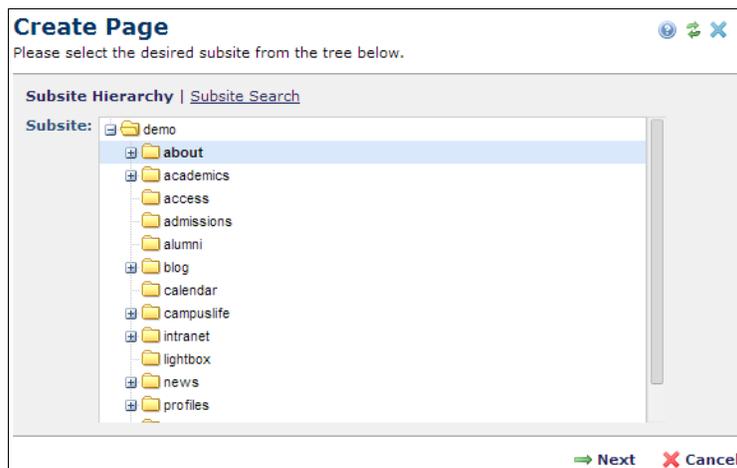


Alternatively, you can invoke the visual difference tool by selecting two versions and clicking the Visual Difference link at the bottom of the panel.

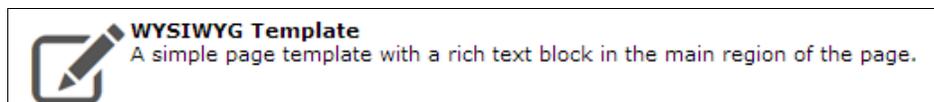
## 1.3.8. Adding a New Page

You are not limited to the pages we provided in CommonSpot University. You can add pages too. Follow these steps to add a page.

1. Go to the “New” menu in the grey menu bar.
2. Select “Page...”
3. Before CommonSpot can build your page for you, it needs to know some basic properties. The very first thing it wants to know is the subsite location of your. You will be presented with a list of subsites available to you. Select the About subsite and click Next.



4. Now, CommonSpot needs to know what template to use. For this example, we'll use the WYSIWYG Template, which is categorized in the Simple Layout Templates.



5. The next thing it needs is all the remaining “Standard” properties for the page you’re creating. Fill out this form to the best of your ability. You can return to this at any time by choosing to edit “Standard Properties”.

6. After filling out the form, click Next.
7. If the “Check Spelling” checkbox is checked, it may interrupt you with potential spelling errors. Correct them, if necessary, and then proceed.
8. Depending on what template you’re using, what subsite you’re in, or a number of other variables, you may be presented with one or more additional dialogs containing “Custom Properties”. In our example, pages built with our base template require some additional properties and we’ve created a form that collects them. This form is presented to you now.

9. Items such as number of columns, displaying the title or header container, are all options that directly impact the way the base template renders. Feel free to experiment with this dialog. Make

a basic selection here, and then click Save. You can always return to this dialog by choosing “Custom Properties”. Upon clicking save, CommonSpot will create the page and present it to you in “Work” mode, ready for your input.

## 1.3.9. Working with Other Types of Content

A page with wide open text region is one type of content you can create. Here is a quick summary of the other content types and how to create them:

- **Uploaded Document:** An uploaded document is a file that you wish to make available through your website. To upload a document, follow the steps above, but instead of “Page...” select “Uploaded Document...” A great example of uploaded documents is the Document Library in CommonSpot University. For more info about how the Document Library is configured, see the section titled “Taxonomy and Facet Navigation (Document Library)” later in this document.
- **Registered URL:** A registered URL is a path to an external website that you can register into a centrally managed gallery of useful websites. Users can link to them and if anything ever changes, you can modify it centrally instead of updating many pages. Plus, they can be displayed in dynamic lists. A great example of registered URL’s is the Press Release section of CommonSpot University. To create a registered URL, follow the steps above, but instead of “Page...” select “Registered URL.” For more info about how these URL’s are managed, see the section titled “In the News (Registered URL’s)” later in this document.
- **Page with Structured Content:** Structured content means that instead of a wide-open formatted text block element to work with, the page template contains an element with a form into which you can specify content in a more structured way. A great example of this is the News articles in CommonSpot University. To create a news article, follow the above steps, but select the News Article template instead. For more info about the News pages, see the section titled “News (Page with Local Custom Element)” later in this document.



## Chapter 2 Demo Site User Map

The CommonSpot demo site ships with an independent Users datasource, which allows the demo site to contain its own user logins and security settings. The sections that follow describe the users in the demo site, list additional users authenticated through an external database, and review the roles of these users in the context of the CMS and the CommonSpot University Web site.

### 2.1. Demo Site Users and Groups

The demo site will not respond to the logins used in the rest of your CommonSpot installation. Instead, use the following logins for the demo site:

#### 2.1.1. Contributors and Admins

The following users have contributor or administrative access to the site.

User ID	Password	User	Role
webmanager	password	Wendy Webmanager	Manager and admin of our site
approver	password	April Approver	Member of the group who approves changes to web content
marketer	password	Mary Marketer	Another content manager, responsible for marketing/admissions content

#### 2.1.2. Basic Users

The following users are not configured as contributors, but have specific purposes when logged in.

User ID	Password	User	Role
ghayes	password	Glen Hayes	Professor
admission	password	Ed Meshin	Has access to Landing Page Lead manager in Webadmin
alumni1	password	Karen Watkins	Alumnus from the Northeast Alumni Association
alumni2	password	Arnold Whitley	Alumnus from the Southeast Alumni Association

alumni3	password	Norma Warren	Alumnus from the Midwest Alumni Association
alumni4	password	Abe Watts	Alumnus from the West Coast Alumni Association

## 2.1.3. Additional Users

The following users are set up in the site for specific functionality that is in a pre-release mode and not yet ready to be used on this site: campuslife, hiringmanager, hr and admissions. You may see these in the list of users. They can be ignored.

## 2.1.4. Groups

The following is a list of groups configured for control of access and permissions.

Group Name	Members	Purpose
Alumni	<ul style="list-style-type: none"> <li>● Alumni – Northeast</li> <li>● Alumni – Southeast</li> <li>● Alumni – Midwest</li> <li>● Alumni – West Coast</li> </ul>	A collection of regional Alumni groups. Membership in this group, indirectly or directly allows access to content on the Alumni page.
Alumni – Northeast	alumni1	Members of this group see regionally specific content on the alumni page.
Alumni – Southeast	alumni2	Members of this group see regionally specific content on the alumni page.
Alumni – Midwest	alumni3	Members of this group see regionally specific content on the alumni page.
Alumni – West Coast	alumni4	Members of this group see regionally specific content on the alumni page.
Web Communications Approvers	<ul style="list-style-type: none"> <li>● Web Managers</li> <li>● approver</li> </ul>	People or groups who can grant final content approval in the workflow
Web Communications	contributor	People or groups who can create

Editors		content but may not have the rights to publish
Web Managers	webmanager	Users who can manage all aspects of the site

## 2.1.5. Additional Groups

The following users are set up in the site for specific functionality that is in a pre-release mode and not yet ready to be used on this site: Admissions Department, Hiring Managers, Human Resources, OHSU Navigation Administrators, Polls Administrators, PT Blog Administrators, PT Multimedia Administrators, PT Calendar Administrators, PT Photo Gallery Administrators, PT Profile Administrators, PT Social Media Administrators, Staff,

## 2.2. Settings

Security in the CommonSpot demo site is handled mostly through groups and their memberships. One area of the demo site is secured from anonymous users and requires authentication, the Web Administration subsite. This subsite is located in /demo/webadmin/ and will require a login by someone in the appropriate administrator group.



## Chapter 3 CommonSpot Base Site

CommonSpot University was built using a vanilla instance of CommonSpot often used by PaperThin Services to help quickly build and deploy CommonSpot sites. This site image is referred to as the CommonSpot Base site, and it includes all the common features and apps typically needed by a CommonSpot implementation.

However, not all of the standard apps have been activated or fully configured in CommonSpot University. Due to this, you may discover the existence of subsites or elements that are not fully configured or populated yet. Future iterations of CommonSpot University may use these apps or have additional apps installed. If you would like to know more about one of these apps, feel free to read up on them on the Community site, and set them up in CommonSpot University. The Community Site, “the | commons” is located at <http://community.paperthin.com>

# Chapter 4 Application Development Framework (ADF)

CommonSpot University takes full advantage of the Application Development Framework (ADF). To find out more about the ADF, please visit the Community site at <http://community.paperthin.com>

Please note that the Demo site runs in a space isolated from other real production sites you may have on your server. In order to accomplish this goal, the CommonSpot University's ADF installation and apps refer to an altered version of the ADF. Instead of finding the ADF in a mapping/folder called "/ADF", CommonSpot University uses a mapping/folder called "ADFdemo". All references in the code for the ADF and its apps have been changed to "ADFdemo". As such, a regular ADF app must be updated in order to work in CommonSpot University. The apps are 100% identical, with the only exception being the reference to the ADF mapping.

# Chapter 5 Demo Site Solution Map

CommonSpot's robust feature set and functionality make it easy to create and manage Web content throughout the entire content lifecycle. These solutions enable faster and easier publishing, making staff more productive, operations more efficient, and content more relevant and up-to-date. The demo site solution map presents multiple content management solutions, the features and functionality that support them. The overview of each solution below is not meant to replace more in-depth product documentation. References to additional documentation provide more information about the CommonSpot features outlined in a solution.

## Alert System

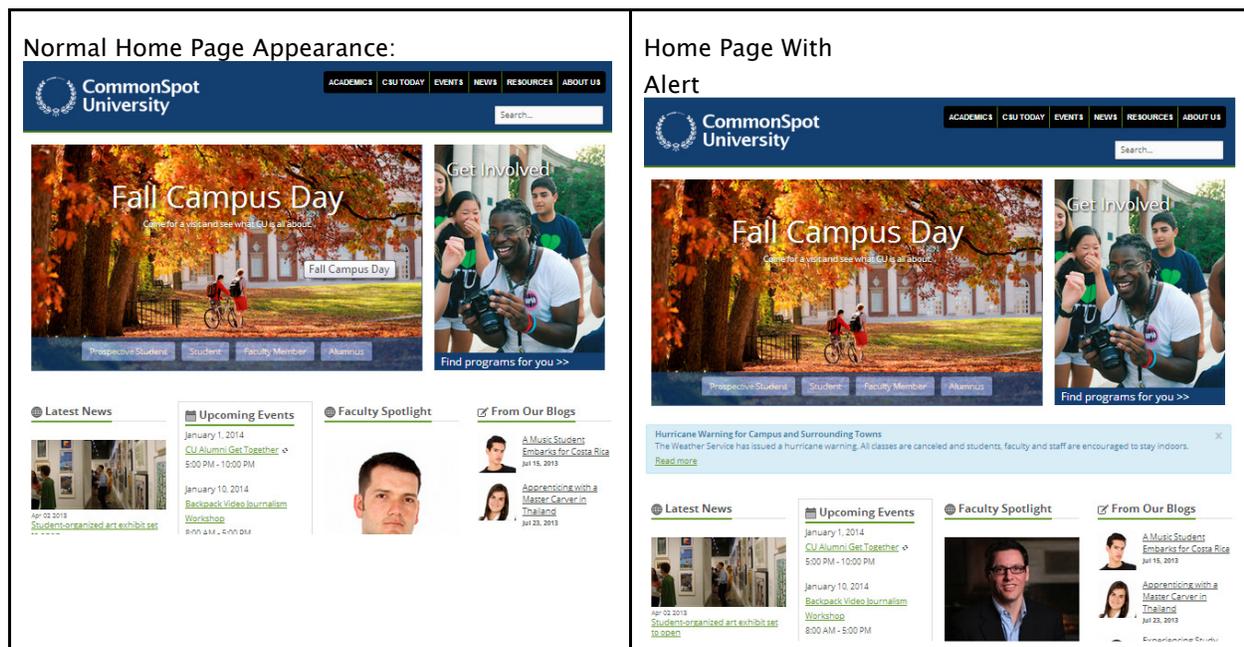
Features: Global Custom Element, Datasheets, Simple App, Content Reuse, Dynamic Self-Updating Lists, RSS

URL's:

- /demo/news/alerts.cfm
- /demo/webadmin/alerts/

Many organizations want a simple way to get urgent messages out to their community. In addition to in-house systems that may incorporate social media and SMS alerts, it is important to make an emergency or urgent message available on the web site in key locations quickly and effortlessly.

In our use-case, the organization required a simple way to add alerts to the system and have them automatically appear in a logical, obvious, and normally hidden location on the home page.



One such way is to create an alert management system using a global custom element with content scheduling. Our example is made up of the following components:

- Global Custom Element: The Alert custom element contains basic fields for an alert message, including title, alert level (low, medium, high), summary, detailed description, status (active or not active) and start and end times.
- Management page with datasheet and form: In the Web Admin subsite, there is an alerts management page located at /demo/webadmin/alerts/ From here, those with appropriate access can add new alerts using a simple form, or update existing ones. Three datasheet views were created to accommodate various needs, including All Alerts, Active Alerts and Inactive Alerts
- Dynamic Filter on home page: When one or more alerts meets the display criteria, they will show up on the home page. We have placed the Alert element on the home page and changed Render Mode to display one or more **existing** element data records. We have created a simple render handler which displays the alert using a different style class depending on the level (low, medium, or high) of the alert. The filter is looking for any item that is tagged as “Active” and where the current time falls within the start and end dates. The filter looks like this:

Filter:  All records  
 Subset of records

Alerts.Activated Equals 1

AND

(  Alerts.Start Time Less Than {now()}

AND

Alerts.End Time Greater Than {now() }

- Detail viewing page: If there is more information about particular alert, visitors can have the option of clicking through to a detail page. This page is located at /demo/news/alert.cfm and will dynamically display the requested alert.
- Alert Home Page: All active alerts are also rolled up into one main page which can be optionally publicized as an aggregate alert source page. This page is located at /demo/news/alerts.cfm This is also the location of the Alerts RSS feed.

While this can be the foundation for an organization’s alert system, it can be enhanced even further. For example, new alerts can be fed into a social media channel and automatically queued as a social media post using CommonSpot Social Media Manager capability. Additionally, post-form hooks can be invoked to push alert messages to third party systems such as SMS.

**Note:** Scheduling content on the home page obviously has some cache implications. Review “Performance Options” and “Stale Cache Handling” in the *Administrator’s Reference* for best practices.

## 5.1. Blogs

Features: PaperThin Blog Application, Templates, Global Custom Element, Datasheets, Custom Elements, Dynamic Self-Updating Lists

URL's:

- /demo/blog/
- /demo/webadmin/blog/blog-dashboard.cfm

The demo site implements a blog system developed by the PaperThin Professional Services team using CommonSpot's Application Development Framework. Several blogs have been added to the site, including:

- Backpack Journalism: A blog about one of the Journalism Department programs
- President's Blog: A personal blog
- School of Business: A blog about one of the many schools
- School of Visual and Media Arts: A blog about one of the many schools
- Studying Abroad: A blog about one of the University's programs
- Web Design and Technology: A blog about one of the many schools

A blog has two components. The first is a 'Blog Manager', which exists for each blog where users write and review posts, add categories, and manage comments. The second is a 'Blog Renderer', which essentially implements a page where the blog posts are displayed for reading and commenting. The Blog Manager and Blog Renderer are terms used in this and other guides.

In the Blogs section of the site, announcements and information are presented as a blog, with various categories linked in the right column to allow users to filter blog posts. Clicking on a blog title brings you to the blog post where you can also comment on the post. Some comments have been created for some of the posts, and visitors are welcome to add more.

The Blogs are managed inside the Web Administration section under Active Blogs. This section is not linked and must be accessed directly via the URL /demo/webadmin/blog/. Once logged into this secure area as a staff member (refer to the Demo Site User Map for information on which users are staff members), you will see a link to 'Active Blogs'. Staff members can then write posts, review their previous posts, review their comments, and add/edit categories.

## 5.2. Container Layouts

Features: Container Element, HTML5, Layouts, Render Handlers

URL's:

- /demo/blog/
- /demo/webadmin/blog/blog-dashboard.cfm



Container layouts are actually Render Handlers which could be as simple as rendering the element as-is, or wrapping the entire set of contents with some sort of layout widget, such as accordions, tabs or carousels.

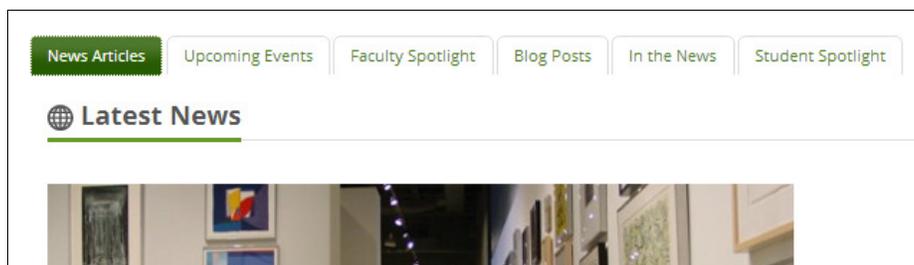
To demonstrate this, we have included in CommonSpot University several container render handlers. Feel free to experiment with these on the home page. To change them, simply select “Choose Layout” from the container element menu.

### 5.2.1.1. Accordion

Displays all the elements in an expandable and collapsible region.

### 5.2.1.2. Tabs

Displays each elements in its own tab



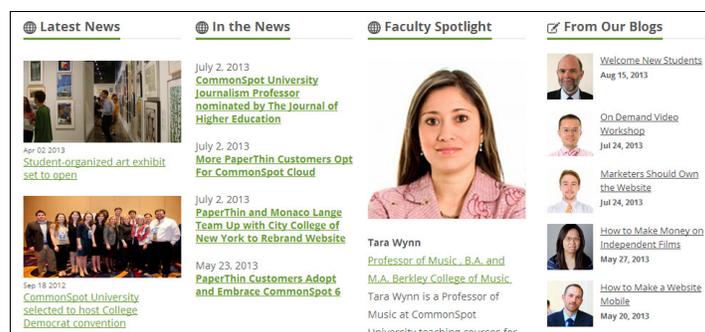
### 5.2.1.3. Carousel

While this one is not optimal for the home page region, it is something to be aware of. Each element can render as a page in a slider carousel.

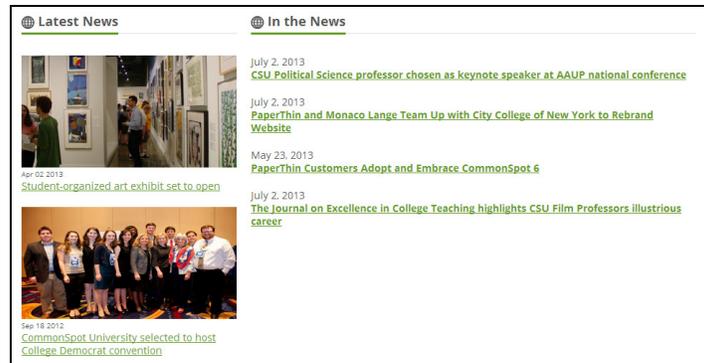
### 5.2.1.4. Grid

Displays a preset quantity of elements, adhering to the underlying grid used in the design. There are several variations, each having the proper combination of grid widths to accommodate the desired look. Examples are:

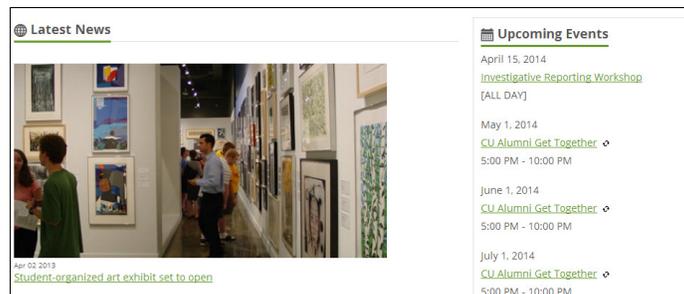
- Grid (<4) Equal Widths: Displays one, two, three or four columns of equal widths. Quantity of columns depends on number of elements. If more than four, it will only display the first four.



- Grid (2) Thin/Wide: Displays only two columns containing the first two elements. The first column is one-third of the width while the second column is two-thirds.



- Grid (2) Wide/Thin: Displays only two columns containing the first two elements. The first column is two-thirds of the width while the second column is one-third.



- Grid (3) Quarter/Half/Quarter: Displays three columns with one wide column in the middle.



- Grid (3) Quarter/Quarter/Half: Displays three columns with larger column to the right.

**Latest News**

July 2, 2013  
[CommonSpot University Journalism Professor nominated by The Journal of Higher Education](#)

Apr 02 2013  
[Student-organized art exhibit set to open](#)

July 2, 2013  
[CSU Political Science professor chosen as keynote speaker at AAUP national conference](#)

July 2, 2013  
[CSU Website Redesign Receives High Grade](#)

May 23, 2013  
[PaperThin Customers Expand Web Presences Using CommonSpot](#)

Sep 18 2012  
[CommonSpot University selected to host College Democrat convention](#)

**In the News**

**Faculty Spotlight**

**Myra Garza**  
Professor of Writing Studies, B.A. University of Connecticut, M.A.

- Grid (3) Half/Quarter/Quarter: Displays three columns with larger column to the left.

**Latest News**

Apr 02 2013  
[Student-organized art exhibit set to open](#)

July 2, 2013  
[CommonSpot University Journalism Professor nominated by The Journal of Higher Education](#)

July 2, 2013  
[CSU Political Science professor chosen as keynote speaker at AAUP national conference](#)

July 2, 2013  
[PaperThin and Monaco Lange Team Up with City College of New York to Rebrand Website](#)

May 23, 2013  
[PaperThin Customers Adopt and Embrace CommonSpot 6](#)

**In the News**

**Faculty Spotlight**

**Tara Wynn**  
[Professor of Music, B.A. and M.A. Berkley College of Music](#)  
Tara Wynn is a Professor of Music at CommonSpot University teaching courses for both music and music history, specializing in American style. Being one of CSU's original music teachers, Wynn has...

## 5.3. Content Freshness, Publication, and Expiration

Features: Freshness Reminder, Content Scheduling, My CommonSpot  
URL's: various

CommonSpot pages are set up to publish on the date and time content is created, pending any workflow. However, pages can also be set to publish at a specific date and time, as well as expire on a specific date. Publication and expiration dates are configured in the page creation process or edited at any time by updating the page's Standard Properties. Any time content is scheduled for publication at a future date or time, CommonSpot tags it as 'scheduled for release' (and includes a visual toolbar at the top of the page as a reminder). These pages then appear in your My Pages report when filtering by 'state.' Expired content is automatically removed from any self-updating page lists, and is no longer browsable. If there is concern

with expiring pages that might have been bookmarked by site visitors in their client browser, there is an option upon expiration to redirect to another URL or to present a warning message, rather than to just deny access.

Freshness Reminders can also be set for any CommonSpot user to review and update content. The user can then receive an email notification reminder and/or view a listing of reminders in his/her the Reminders section of My CommonSpot, a personalized report presented to every user upon logging in. Log in as one of the demo site users to view their version of My CommonSpot and some of the Reminders pre-configured for the demo site. Then go in and create your own!

## 5.4. Content Personalization: Audience-Specific Content

Features: Content Scheduling, Users and Groups, Security, Visitor Contextualization

URL's:

- /demo/alumni/ (Explicit personalization)
- Various (Implicit/Contextual personalization)

CommonSpot can help deliver highly targeted content that anticipates a website visitor's needs. This means giving personally-relevant content to the right audience at the right time for a more compelling Web experience that increases engagement and helps marketers improve their overall chance of success.

CommonSpot achieves this in one of two ways, using both explicit personalization (where you know who the user is and what they are likely to want to see) and implicit (also known as contextual, where the web site makes assumptions based on the user's behavior).

### 5.4.1. Content Personalization: Explicit (Known-User)

If CommonSpot knows who the visitor is, dynamic content can be delivered to them based on things that we know about them, including preferences, group memberships, etc. In order for CommonSpot to know who the visitor is, they would have to authenticate either using CommonSpot's built-in user authentication or via third party authentication. Once CommonSpot knows who they are, it can dynamically display content using the Scheduled Element or Container Element or any other element that supports scheduling.

An example of explicit content personalization is as follows:

The user is a member of:	CommonSpot displays this:
None of the "alumni" specific groups.	On Alumni page:

	<p style="text-align: center;">Sign In</p> <p>If you don't have an account, use this form to contact your regional Alumni representative.</p> <p>First Name <input type="text"/></p> <p>Last Name <input type="text"/></p> <p>Email Address <input type="text"/></p>
<p>Northeast Alumni Association</p>	<p>Same location on Alumni page now show this:</p> <p><b>Northeast Alumni Association</b> Wendell Edison, President</p> <p>Come visit us at our next event on the 25th at 7:00 PM. To find out more, call or email</p> <p>617-555-1212 naa@commonsspot.edu</p>
<p>Southeast Alumni Association</p>	<p>Same location on Alumni page now shows this:</p> <p><b>Southeast Alumni Association</b> Wendell Edison, President Congratulations to our recent graduates!</p> <ul style="list-style-type: none"> <li>• Angelita Harrington</li> <li>• Gilbert Dyer</li> <li>• Dale Mills</li> <li>• Cleo Martinez</li> <li>• Kareem Rush</li> <li>• Octavio Hensley</li> </ul> <p>Come to our welcome mixer on the 18th at the Congress Street Pub.</p>
<p>Any Alumni group</p>	<p>On the news listing page, they would see this a link to "View the Alumni Calendar"</p>

## 5.4.2. Content Personalization: Implicit (Contextual)

If you don't happen to know exactly who the user is, you can still present entirely different Web pages with changes to site design, content and offers, for example, based on other factors. These factors include the

mobile device being used, geo-location, website activity, areas of interest, content viewed, searches made, and form conversions—to name a few.

CommonSpot can gather information from each visitor during every session, whether they are logged in or not, so Web content can dynamically adjust, delivering personally relevant content at exactly the right moment.

CommonSpot does this using a combination of content scheduling and group management. Using its scheduling capabilities, web managers can determine which audiences should see what content. Audiences are grouped together into Visitor Context Groups, which allow web managers to create various personas that can benefit from targeted content.

The other component is a “rules engine” which lets web managers determine the specific behavior that would qualify a web visitor to become a part of a Visitor Context Group.

The site has several examples of personalization. Some examples are laid out here. The following rules are configured in the site. If a user does one of the examples on the left, CommonSpot will tag them by adding them to an ad hoc “group”.

Visitor Does This:	CommonSpot Classifies them into a Visitor Context Group called:
<ul style="list-style-type: none"> <li>● Submits a form to register for a Campus activity</li> <li>● Visits the “Fall Campus Day” landing page /demo/admissions/fall-campus-day.cfm</li> </ul>	Campus Life Activity
<ul style="list-style-type: none"> <li>● Performs a site search for the term “journalism”</li> <li>● Visits an article about journalism /demo/news/8-students-earn-trip-to-australia.cfm</li> </ul>	Journalism
<ul style="list-style-type: none"> <li>● Visits any page within the “Admissions” subsite or its children subsites</li> </ul>	Prospective Student
<ul style="list-style-type: none"> <li>● Visits the “Current Students” persona page /demo/students.cfm</li> </ul>	Student
<ul style="list-style-type: none"> <li>● Arrives from a search engine after using a search term that contains “study abroad”</li> <li>● Performs a site search for the term “study abroad”</li> <li>● Navigates to a page in subsite “/blog/studyingabroad/” or below</li> <li>● Visits an article about journalism /demo/news/8-students-earn-trip-to-australia.cfm</li> </ul>	Travel

Using the ad hoc groups established above, the following “effects” can be carried out on various pages and templates throughout the site. Several School pages as well as the News Listing page will deliver the following content based on these rules.

The user is a member of:	CommonSpot displays this:
<p>Journalism + Prospective Student + Travel</p>	 <p>A banner for the Backpack Journalism Program. The background shows a person in a dark cap taking a photo with a camera. The text 'Backpack Journalism Program' is overlaid in white. At the bottom, a blue bar contains the text 'Learn more &gt;&gt;' in white.</p>
<p>Student + Travel</p>	 <p>A banner for 'Design Your Own Year Abroad'. The background shows a group of students walking in front of the Colosseum. The text 'Design Your Own Year Abroad' is overlaid in white. At the bottom, a blue bar contains the text 'Learn more &gt;&gt;' in white.</p>
<p>Prospective Student</p>	 <p>A banner for Campus Tours. The background shows students walking on a tree-lined campus path. The text 'Campus Tours' is overlaid in white. At the bottom, a blue bar contains the text 'Sign up now &gt;&gt;' in white.</p>

Campus Life Activity	<div style="border: 1px solid gray; padding: 10px;"><p><b>See Campus Events</b></p><p>Student events are happening weekly. Concerts, mixers, comedy and sports!</p><p><a href="#">Check the calendar</a> for the latest events.</p></div>
----------------------	---

Below is an example of the scheduling rules that establish some of the dynamic display scenarios from above. This is taken from the News Listing page.

Scheduled Element Timeframes/Audiences:
<p><b>Backpack Journalism Link</b> </p> <p>Schedule: All Dates Groups: Journalism, Prospective Student, Travel (All) Selected Subsites: All Subsites Rendered For: Full-Text Search, Print</p>
<p><b>Study Abroad Link</b> </p> <p>Schedule: All Dates Groups: Student, Travel (All) Selected Subsites: All Subsites Rendered For: Full-Text Search, Print</p>
<p><b>Alumni Calendar</b> </p> <p>Schedule: All Dates Groups: Alumni, Alumni - Midwest, Alumni - Northeast, Alumni - Southeast, Alumni - West Coast, Alumnus (Any) Selected Subsites: All Subsites Rendered For: Full-Text Search, Print</p>
<p><b>Campus Tour</b> </p> <p>Schedule: All Dates Groups: Student (Any) Selected Subsites: All Subsites Rendered For: Full-Text Search, Print</p>
<p><b>Campus Life Calendar Link</b> </p> <p>Schedule: All Dates Groups: Campus Life Activities (Any) Selected Subsites: All Subsites Rendered For: Full-Text Search, Print</p>

## 5.5. Content Reuse

Features: Page Index, Custom Element, Custom Render Handler

Sections: This solution is implemented across all sections

Users: webmanagers

References: Administrator's Reference, Elements Reference, Developer's Guide

CommonSpot supports content reuse in two ways by allowing end users to repurpose page metadata and content object metadata. There are many examples of content reuse in the demo site across the various demo site sections.

A linked list of pages can be easily defined using a Page Index Element, whose menu toolbar enables end users to define the 'Filter' criteria, based on the metadata of specific pages and/or documents to be listed. Some great examples of the Page Index Element can be found in the "Press Release Example of the Example Solution" section, later in this guide.

- `/demo//news/press-releases.cfm`
- `/demo/news/inthenews/index.cfm`

Once the filter criteria are defined and the list rendered, you can alter the Page Index presentation by editing its 'Layout' from the element toolbar. Select which standard metadata properties you want to display (for example, title, description, publication date). Once defined, the list will automatically self-update when any page matches the filter criteria, and the page list will be re-cached.

Content object reuse is also implemented in the demo site. Content objects metadata is defined within a 'Custom Element' in CommonSpot, whose interface can be configured to display existing data using the 'Render Mode' option on its toolbar menu. Once configured for reuse, the options for the custom element filter are either to show all of its data, or define filter criteria to display only a subset of its data. Similar to Page Index filters, which use page metadata as the criteria for rendering their pages, a Custom Element can utilize any of its own custom metadata field definitions in a filter to choose which data to render. The difference between the Page Index and Custom Element is determined in how the presentation is controlled for the data being displayed. The Page Index 'Layout' options are limited to a handful of predefined views using only a subset of page standard metadata (that is, you have no control over their placement), whereas a Custom Element's layout can be configured with an unlimited number of 'Display Templates' which expose all of the element's defined fields for defining its presentation. You can also define a customized HTML presentation by registering a 'Custom Render Handler' for both a Page Index and a Custom Element.

Content Reuse using Custom Elements is used throughout the site. While there are many great examples, one specific one is the News articles. This app goes beyond the standard page-based news article, adding additional metadata such as tags, people and standardized images. Unlike the Press Releases which provide just a formatted text block element on a page, the News element provides a more comprehensive form for creating and modifying News Articles.

## 5.6. Course Catalog App

CommonSpot University contains an installation of the CommonSpot Course Catalog app, available from the Community site. The course catalog app contains a collection of elements and code which enable you to publish a complete course catalog on your website.

Nearly everything in the Academics section of CommonSpot University is powered by the Course Catalog app, including many of the specific pages. Here is a rundown of the types of content found in the app, and how it integrates into CommonSpot University:

- **School:** At the highest level, a school represents a division of the institution dedicated to a particular area of study. This is often referred to as a school within a University, but the term school can be replaced by whichever terminology is appropriate for the organization. Schools are managed in a Global Custom Element. In CommonSpot University, there is a “Schools” page(/demo/academics/schools.cfm), and each school has its own dedicated subsite, such as /demo/academics/journalism/ The school’s home page has its own special template, which automatically populates the page with various items related to the school, such as events, blog posts, news as well as links to department pages within that school. This subsite is automatically created when you add a school using the app.
- **Department:** Schools are divided into one or more “departments”, which further define a particular area of study. Departments are managed in a Global Custom Element. Each department has its own dedicated subsite, which is located inside of its specific school. For example, /demo/academics/liberalarts/ The department home pages have their own template, which allows for an image, description and links to Programs within that department.
- **Program:** Each department can have one or more “programs” which further define an area of study, and specify the degree type, term, and a collection of required and recommended courses needed to complete that program. For example, the Bachelor of Science in Accounting is a Program in the Business Administration Department for the Fall Term. Each program has its own page within the department subsite. For example, the Language Arts program is located in the Language Department, within the School of Liberal Arts:  
/demo/academics/liberalarts/language/language-arts.cfm By binding the Program to a Term, it allows schools to evolve their programs over time, while still maintaining an accurate historical record of what was contained in each program.
- **Course:** Courses are linked to Departments and can span multiple Programs, as necessary. Courses can even be added to Programs in different departments. Courses are managed in a Global Custom Element. A course is bound to a Department and Term. This allows a Department to alter course attributes for each new term if needed. Courses can also stipulate prerequisite courses. The user interface for a course contains the ability to add the class schedule without having to go to a different form. When a course is created, a page for that course is created in the Department’s subsite. The course’s page contains the class schedule.
- **Class Schedule:** The schedule for the classes is managed in a Global Custom Element. The app does not provide, nor do we recommend, direct editing of class schedule records. Instead, schedule data is entered on the Course form. Multiple variations of a schedule can exist, so that it can support a class that meets regularly on specific days of the week at the same time, but also meets one day a week on a different time. The schedule record also contains building and room information.

- **Term:** The term allows the entire catalog to be bound to a specific semester, or whatever term length the school requires. When new terms are planned and rolled out, it allows the programs and course descriptions to evolve over time, while still maintaining an accurate record of historical programs and courses. Terms are managed in a Global Custom Element.
- **Degree:** The degrees are managed in a Global Custom Element and allow schools to create various Degrees, such as Bachelor of Science or Bachelor of Arts, which can be used to further define the Programs offered.
- **Campus:** You can specify one or more campuses to be used to help specify buildings. Campuses are managed in a Global Custom Element.
- **Building:** All class schedules make reference to a building. All buildings are managed in a Global Custom Element.

Everything in the course catalog should be managed entirely in the Webadmin section of the site, located at `/demo/webadmin/coursecatalog/`

## 5.7. Custom Apps: Simple No-Code App: Jobs

URLs:

- Main Careers Page: `/about/careers/`
- Student Jobs: `/about/careers/student-jobs.cfm`
- Academic Careers Page: `/academics/academics-jobs.cfm`
- Job Management Dashboard: `/webadmin/jobs/index.cfm`

Features used:

- Global Custom Element, display template, simple form, datasheet, security

A good way to demonstrate CommonSpot's out-of-the-box capability of creating custom apps is by looking at the career section of the site. Here we have a simple app built almost entirely with out-of-the-box components, including custom elements, datasheets, display templates and simple forms.

### 5.7.1. The Jobs Element

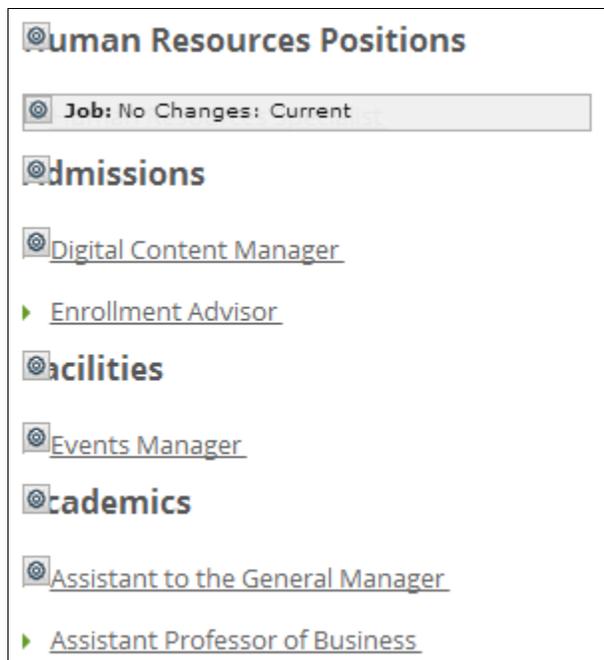
The Job Listings element is a global custom element. It contains several fields which store various attributes about each job, such as job title, description, department, location, etc.

Field Name	Label	Field Type
<a href="#">id</a>		Hidden
<a href="#">jobTitle</a>	Job Title	Text
<a href="#">datePosted</a>	Date Posted	Calendar
<a href="#">published</a>	Published	Radio Buttons
<a href="#">department</a>	Department	Selection List
<a href="#">location</a>	Location	Selection List
<a href="#">jobType</a>	Job Type	Selection List
<a href="#">creator</a>		Hidden
<a href="#">description</a>	Description	Formatted Text Block

## 5.7.2. Jobs Pages

There are two public-facing pages that display content from the Job element. The first is the Job Listing page, which displays a list of all published jobs in the system, categorized by department. To accomplish this we simply placed the Job element to the page. It is added several times so that each section can be filtered according to department.

In the following screenshot, the titles live in a Formatted Text Block element, while the list of jobs come directly from the Jobs element.

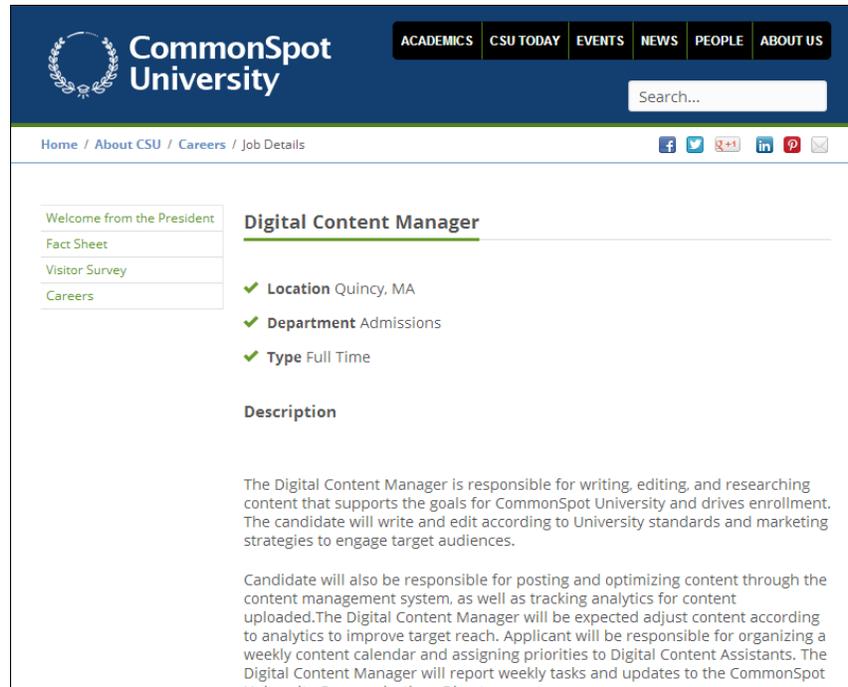


The filter on the first Job element in the list looks like this. Notice that it is only looking for the Jobs tagged as Published = “Yes” and Department = “Human Resources”. The other Jobs elements on the page have similar filters, using the name of the appropriate department.

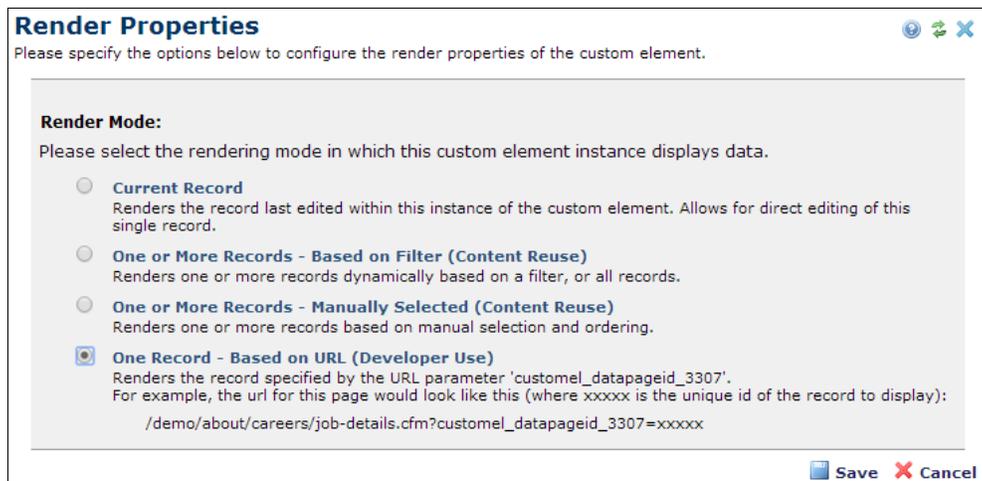
The resulting page looks like the following screenshot.

We have used a simple Display Template which displays the job title as a link in a bulleted list. Display templates are layout instructions that can be created using the CommonSpot rich text editor. They are accessible through the layout tab in the element menu.

The job detail page also contains the Job element. However, on this page, it uses a different Display Template which shows all of the job's information optimized for a full page.



This page is set up dynamically in the Render Properties dialog. By selecting the fourth option in the list, it tells CommonSpot to not render any particular job here unless one is requested by the referring page. The referring page, in our example, is the Careers page, which shows a dynamic list of published jobs. The Display Template handles the creation of the dynamic URL's from which this page figures out which job to display.



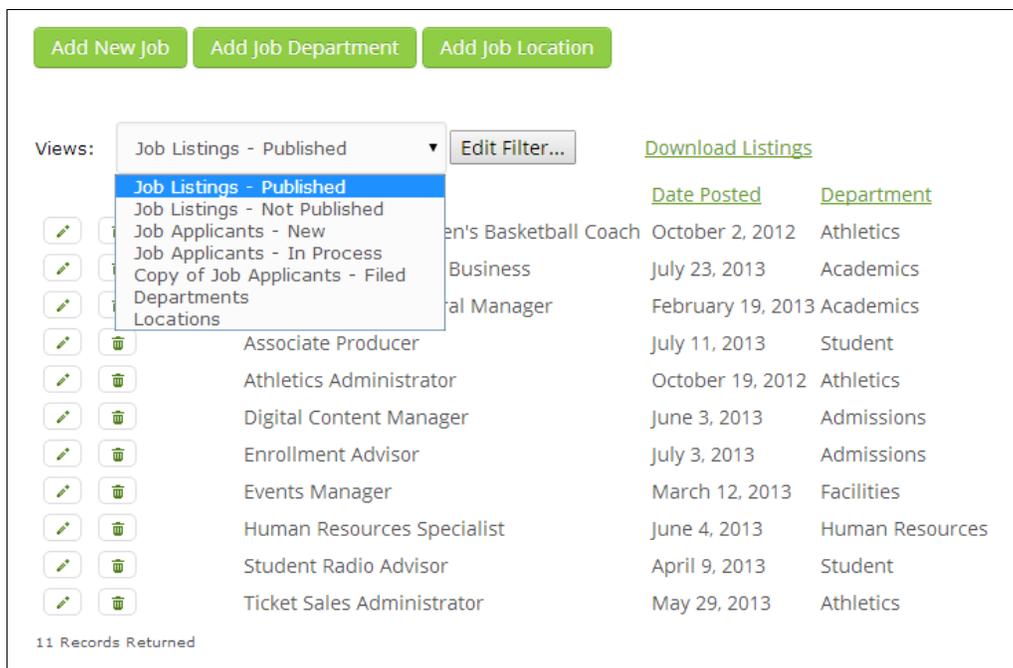
These same job listings show up on different pages with different filters. For example, student jobs are listed on the Student Jobs page, located at /demo/about/careers/student-jobs.cfm while academic jobs are located on the Academic Jobs page, located at /demo/academics/academics-jobs.cfm. Each uses the same element and Display Templates, but a different filter.

## 5.7.3. Jobs Manager

To manage jobs, go to the Web Administration page located at /demo/webadmin/jobs/index.cfm Here, we have created a dashboard for all Job related activity.

This page behaves differently depending on your login. Two user groups have been created for this page. One for Human Resources, who can see everything. And another for Hiring Managers, who can only submit job requests and check their status.

If I'm logged in as someone from the Human Resources team, I see a view with a datasheet element displaying several views, edit and delete buttons on each row, and buttons at the top for creating new jobs, departments or locations.



Alternatively, if I log in as just a Hiring Manager, my view will look considerably watered down, showing only the link for adding a job, as well as a list of Jobs without edit or delete links.

[Add New Job](#)

Views:

[Download Listings](#)

<a href="#">Job Title</a> ▲	<a href="#">Date Posted</a>	<a href="#">Department</a>
Assistant Men's/Women's Basketball Coach	October 2, 2012	Athletics
Assistant Professor of Business	July 23, 2013	Academics
Assistant to the General Manager	February 19, 2013	Academics
Associate Producer	July 11, 2013	Student
Athletics Administrator	October 19, 2012	Athletics
Digital Content Manager	June 3, 2013	Admissions
Enrollment Advisor	July 3, 2013	Admissions
Events Manager	March 12, 2013	Facilities
Human Resources Specialist	June 4, 2013	Human Resources
Student Radio Advisor	April 9, 2013	Student
Ticket Sales Administrator	May 29, 2013	Athletics

11 Records Returned

### 5.7.4. Field Level Security on the Job Form

When using the Add New Job form, the “Published” field will only show to Web Managers and those in the Human Resources group. Using Field Level Security, only those groups have edit capability on this field. This means that if a hiring manager fills out the form, they won’t be able to automatically publish their job listing to the website.

**Add New Job**

Job Title:

Date Posted:

Published:  Yes  No

Department:

Location:

Job Type:

Description:

**Rich Text Editor:**

- Toolbar: Bold, Italic, Underline, Bulleted List, Numbered List, Link, Unlink, Image
- Style: Paragraph
- Text: No CSS st...
- Buttons: Remove Tag, Remove Classes
- Properties: This panel will help you configure special properties for the elements in your page.
- HINT: Use Shift+Enter for a normal line break

### 5.7.5. Only Out of the Box Tools

The Jobs section of the site was built using only out-of-the-box tools with only minor exceptions. The user's ability to upload their resume is actually using a custom field type from the ADF. Also, the Edit and Delete buttons on the datasheet are using a custom datasheet column render handler to achieve the consistent look with the other datasheets. The latter could have been done with out-of-the-box datasheet capability however.

## 5.8. Custom Element Examples:

CommonSpot provides a capability to create user-defined custom objects called Custom Elements. Please refer to the documentation for an explanation of custom elements and the difference between a "Local" custom element and a "global" custom element. Below, please find some good examples of Local and Global elements found in CommonSpot University.

## 5.9. Custom Element Example: Local Custom Element (News)

Using the PT News app, the News custom element contains all the information needed for rendering a polished news article on the page, along with links to related news articles. Please see the Community site for more information about the News app.

## 5.10. Custom Element Example: Local Custom Element (Residence Halls)

In the Campus Life section of the site, we display a list of student residence halls. Because we needed each residence hall to have a consistent appearance, we took the approach of using a custom element, which allows the content to be entered onto the page via form, rather than a free-form rich text box. We chose a “local” custom element so that each residence hall could have its own page, unique URL and be maintained by the standard workflow configured for the site.

The element contains fields for the name of the hall, capacity, category, location, description, image and amenities. The form looks like this:

The screenshot shows a web form titled "Enter Custom Element Data" for the "Residence Halls" element. The form is divided into three tabs: "Residence Halls" (selected), "Images", and "Description". The "Residence Halls" tab contains the following fields and options:

- Name:** Text input field containing "Monty Hall".
- Capacity:** Text input field containing "1000".
- Category:** Dropdown menu with "Freshmen" selected.
- Location:** Radio buttons for "On Campus" (selected) and "Off Campus".
- Rooms:** Checkboxes for "Single", "Double", and "Triple", all of which are checked.
- Amenities:** Checkboxes for "Athletic Facilities", "Grocery Store", "Wi-Fi", "Cafeteria", "Pets", "Convenience Store", and "Public Transportation". "Athletic Facilities", "Grocery Store", "Wi-Fi", "Cafeteria", "Convenience Store", and "Public Transportation" are checked. "Pets" is unchecked.
- Director:** Text input field containing "Alec Peruker".
- Email:** Text input field containing "aperuker@commonspot.edu".

At the bottom left of the form, there is a checked checkbox for "Check Spelling". At the bottom right, there are "Save" and "Cancel" buttons.

The resulting page looks like this:

CommonSpot University

ACADEMICS CSU TODAY EVENTS NEWS RESOURCES ABOUT US

@search...

Home / Campus Life / Monty Hall

Residence Halls  
Apply for Campus Housing  
Campus Life Calendar  
CSU Tube

### Monty Hall

Capacity: 1000 Rooms: Single, Double, Triple

Monty Hall, formerly Montgomery House, was named in 1984 in honor of the Montgomery's, benefactors of CSU. It is comprised of some of the earliest buildings in the Residence Hall system. Ballot Hall, built in 1901, is the oldest of Monty's buildings and one of the most beautiful. Five other stately brick dormitories and the Dining Hall surround four sides of the Quadrangle, which is the largest enclosed open "green" at CSU. Following the renovations in 1987, a brightly-lit tunnel system connecting Monty Hall and Ballot Hall on the east side of the Quad and Monty Hall on the west, was created.



Residence Hall Director: Alec Peruker

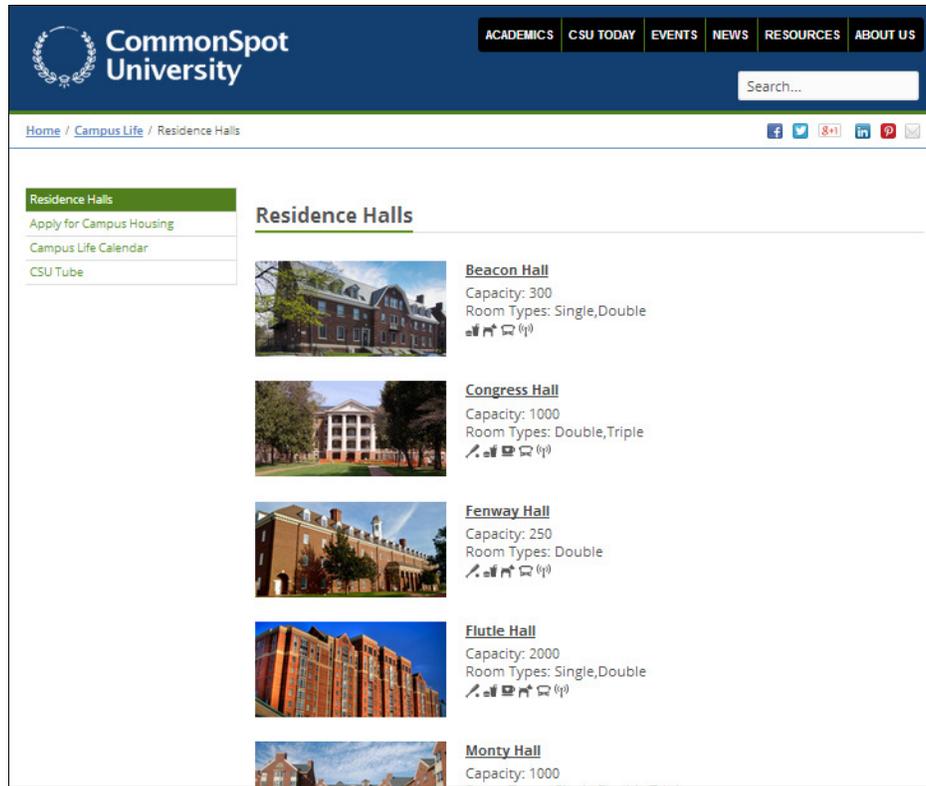
- Category: Freshmen
- Location: On Campus
- Amenities: Athletic Facilities, Cafeteria, Convenience Store, Grocery Store, Public Transportation, Wi-Fi

Related Documents

- Campus-Wide Residency Application
- Montgomery Floor Plans
- West Campus Map
- Click to insert new element

To create a new residence hall, all that is needed is to create a page using the Residence Hall template. This template contains the Residence Hall custom element. To edit a residence hall, you can simply go to its page, and open up the element for editing.

A list of residence halls is also provided at /demo/campuslife/res-halls.cfm



## 5.10.1. Controlling the Layout of Residence Halls

CommonSpot supports two ways of controlling the custom element records are laid out on a page. Display Templates are layout definitions that can be created by non-technical users using the CommonSpot rich text editor. Render Handlers achieve the same goal, but are created using custom code, giving them a great deal of flexibility, but requiring more technically skilled resources to build.

To illustrate these two techniques, the Residence Hall area of the site has been configured to demonstrate each. The tradeoff is as follows:

- Display Templates let non-technical users easily and quickly create layouts without writing code or waiting for developer resources. However, the layouts are limited to simply displaying content. No conditional formatting can take place.
- Render Handlers provide you the ability to add conditional component to the layout, but it comes with the cost of relying on a technical resource to get it done.

The display template for a residence hall list simply shows the name of the hall, the text for the description and a textual list of amenities:

**Beacon Hall**

Beacon Hall was named after the famous Beacon family, which includes two presidents, a famous historian, several ambassadors, and one of the great women of the Colonial and Federalist periods.

**Amenities:** Cafeteria,Pets,Public Transportation,Wi-Fi

Alternatively, the render handler for the residence hall list adds a little more flair. Notice how the thumbnail image is displayed and the list of amenities has been converted to a row of representative icons:



**Beacon Hall**  
 Capacity: 300  
 Room Types: Single,Double  
 🍽️ 🐾 🚗 📶

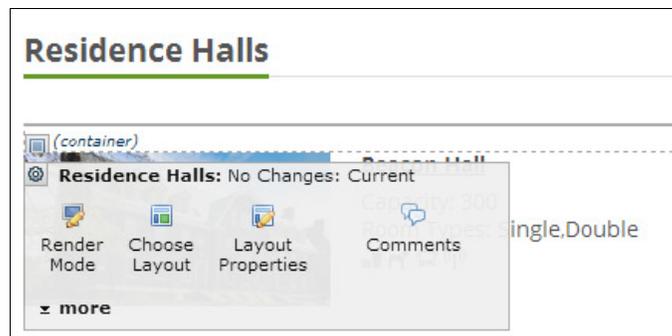
The same can be said for the detail pages. The following screenshots show the different Residence Hall layouts side by side.

<p><b>Residence Hall List using a Display Template</b></p> <p><u>Residence Halls</u></p> <p><b>Beacon Hall</b>          Beacon Hall was named after the famous Beacon family, which includes two presidents, a famous historian, several ambassadors, and one of the great women of the Colonial and Federalist periods.  <i>Amenities: Cafeteria,Pets,Public Transportation,Wi-Fi</i></p> <p><b>Congress Hall</b>          Congress Hall is named for Congress Street where many "learned, conscionable, and industrious men and women work". One of the first two Halls built under the Residence Hall system, Congress was opened in 1930.  <i>Amenities: Athletic Facilities,Cofeteria,Convenience Store,Public Transportation,Wi-Fi</i></p> <p><b>Fenway Hall</b>          Fenway Hall was named for the Fenway family, although recognition is generally given to A. Lawrence Lowell, President of CSU from 1909-1933. It was due to his efforts that the Hall system was established. Opened in 1930, the building has eighteen entries situated around two lovely courtyards.  <i>Amenities: Athletic Facilities,Cafeteria,Pets,Public Transportation,Wi-Fi</i></p> <p><b>Flutie Hall</b>          Flutie Hall was named in honor of Charles William Hardiman, President of CSU from 1869 to 1909, who broadened the undergraduate curriculum and encouraged graduate and professional instruction. One of the original Halls, Flutie is organized around three open courts, the largest of which was built to be reminiscent of 18th-century  <i>Amenities: Athletic Facilities,Cafeteria,Convenience Store,Pets,Public Transportation,Wi-Fi</i></p> <p><b>Monty Hall</b>          Monty Hall, formerly Montgomery House, was named in 1984 in honor of the Montgomery's, benefactors of CSU. It is comprised of some of the earliest buildings in the Residence Hall system.  <i>Amenities: Athletic Facilities,Cafeteria,Convenience Store,Grocery Store,Public Transportation,Wi-Fi</i></p> <p><b>Patriot Hall</b>          A graduate of the class of 1680 and Tutor for twelve years, John Patriot promoted a more liberal point of view in college instruction and was praised for his "enlarged spirit." After a notable career as a lawyer and judge, Patriot returned to CSU in 1708 as President.</p>	<p><b>Residence Hall List using a Render Handler</b></p> <p><u>Residence Halls</u></p> <p><b>Beacon Hall</b>          Capacity: 300          Room Types: Single,Double          🍽️ 🐾 🚗 📶</p> <p><b>Congress Hall</b>          Capacity: 1000          Room Types: Double,Triple          🍽️ 🐾 🚗 📶</p> <p><b>Fenway Hall</b>          Capacity: 250          Room Types: Double          🍽️ 🐾 🚗 📶</p> <p><b>Flutie Hall</b>          Capacity: 2000          Room Types: Single,Double          🍽️ 🐾 🚗 📶</p> <p><b>Monty Hall</b>          Capacity: 1000          Room Types: Single,Double,Triple          🍽️ 🐾 🚗 📶</p> <p><b>Patriot Hall</b>          Capacity: 2000          Room Types: Single,Double,Triple          🍽️ 🐾 🚗 📶</p>
<p><b>Residence Hall Detail using a Display Template</b></p>	<p><b>Residence Hall Detail using a Render Handler</b>          Notice the formatting of the amenities, capacity and room types at the top. Instead of merely listing the data, the render handler contains code that can associate the data values to images, making the layout more compelling.</p>

<h3>Monty Hall</h3> <p>Monty Hall, formerly Montgomery House, was named in 1984 in honor of the Montgomery's, benefactors of CSU. It is comprised of some of the earliest buildings in the Residence Hall system. Ballot Hall, built in 1901, is the oldest of Monty's buildings and one of the most beautiful. Five other stately brick dormitories and the Dining Hall surround four sides of the Quadrangle, which is the largest enclosed open "green" at CSU. Following the renovations in 1987, a brightly-lit tunnel system connecting Monty Hall and Ballot Hall on the east side of the Quad and Monty Hall on the west, was created.</p>  <p>Residence Hall Director: Alec Peruker aperuker@commonspot.edu</p> <ul style="list-style-type: none"> <li>• <b>Category:</b> Freshmen</li> <li>• <b>Location:</b> On Campus</li> <li>• <b>Room Types:</b> Single, Double, Triple</li> <li>• <b>Amenities:</b> Athletic Facilities, Cafeteria, Convenience Store, Grocery Store, Public Transportation, Wi-Fi</li> </ul>	<h3>Monty Hall</h3> <p> Capacity: 1000 Rooms: Single, Double, Triple</p> <p>Monty Hall, formerly Montgomery House, was named in 1984 in honor of the Montgomery's, benefactors of CSU. It is comprised of some of the earliest buildings in the Residence Hall system. Ballot Hall, built in 1901, is the oldest of Monty's buildings and one of the most beautiful. Five other stately brick dormitories and the Dining Hall surround four sides of the Quadrangle, which is the largest enclosed open "green" at CSU. Following the renovations in 1987, a brightly-lit tunnel system connecting Monty Hall and Ballot Hall on the east side of the Quad and Monty Hall on the west, was created.</p>  <p>Residence Hall Director: Alec Peruker</p> <ul style="list-style-type: none"> <li>• <b>Category:</b> Freshmen</li> <li>• <b>Location:</b> On Campus</li> <li>• <b>Amenities:</b> Athletic Facilities, Cafeteria, Convenience Store, Grocery Store, Public Transportation, Wi-Fi</li> </ul>
--	--

## 5.10.2. To change the List page from one layout to the other

Simply go to /demo/campuslife/res-halls.cfm and open it up into Work on this Page (My Changes). Then select "Choose Layout" from the element menu.



You will be presented with the Choose Layout dialog.

Name ^	Description	Actions
Detail View - No Code	Simple view built using a display template	
Detail View - With Code	Using a render handler, this detail shows some additional flair.	
List View - No Code	Simple listing of residence halls using a Display Template.	
<b>List View - With Code</b>	Using a render handler, this list shows some additional flair.	
Standard Rendering	Renders the element utilizing the standard method provided by CommonSpot.	

Notice that there are four pre-configured layouts. Two are for the Detail view and two are for the List view. Each one specifies the type of layout it is. The ones using a Display Template are labeled “No Code” while the ones using Render Handlers are labeled “With Code”.

For the List page, select one of the two List Views. Also, the Display Template layout can be altered by simply clicking the pencil icon to the right. Doing so will provide the following dialog, where you can modify the layout using the rich text editor:

### Element Display Template

Please provide the following information for the Display Template.

**Template Name:**  
  
Enter a name for this template.

**Template Description:**  
  
Enter a description for this template.

**Private**

**Preview Image:** /demo/webadmin/images/ScreenClip-3.png

**Template:**

Name

shortDescription

**Amenities:**

<H3> <Anchor> Remove Tag Remove Classes

Link Properties

Element ID:  Target: normal Title:

Link: /demo/campuslife/lcustom\_element\_data\_3712!

**HINT: Use Shift+Enter for a normal line break**

Check Spelling

### 5.10.3. To change the Detail page from one layout to the other

The detail pages are all built using a template, specific to those types of pages. Therefore, you can change the configuration in one place and it will update all the pages. Simply go to the template located at /demo/campuslife/template-Residence-Hall-Template.cfm and open it up into Work on this Page (My Changes). Then select "Choose Layout" from the element menu. Choose one of the Detail layouts from the list. Then save the change and go view any of the residence hall pages to see your change take effect.

### 5.10.4. Residence Hall Document Libraries

To supplement the residence halls, there is a document library containing related PDF documents for each residence hall, such as maps, application form, etc. The related documents appear in the right column of the residence hall template, using a page index element to pull in the related documents.

The screenshot shows a detail page for "Monty Hall". On the left, there is a title "Monty Hall" with a green underline. Below the title are icons for edit, delete, and other actions, followed by the text "Capacity: 1000 Rooms: Single, Double, Triple". A paragraph of text describes the hall's history and location. On the right, under the heading "Related Documents", there are three links: "Campus-Wide Residency Application", "Montgomery Floor Plans", and "West Campus Map", each with a small red icon.

These documents are all stored as PDF in a subsite created specifically for these residence hall documents. It's located at /demo/campuslife/reshalldocs/ subsite. A metadata form has been created (see Custom Metadata section for other metadata examples) so that the documents in the residence hall document library can be tagged according to which residence hall they apply to. The form looks like this:

The screenshot shows a "Custom Properties" form. At the top, it says "Please provide the following information." Below this, there is a checkbox labeled "Show in all Residence Halls". Underneath, it says "If not checked, select one or more of the following:". There is a section labeled "Residence Halls:" followed by two columns of checkboxes. The first column includes "Beacon Hall" (checked), "Fenway Hall", "Monty Hall", "Quincy Hall", and "River Hall". The second column includes "Congress Hall", "Flutie Hall", "Patriot Hall", "Quincy Towers", and "Socks Hall". At the bottom right, there are "Save" and "Cancel" buttons.

To add a document to the library and tag it for a specific residence hall, simply upload a document into the /demo/campuslife/reshalldocs/ subsite. After uploading the document, you will be presented with the Custom Properties dialog. Either select “Show in all Residence Halls” or select the specific one from the list.

Note: The Residence Hall element uses different techniques for gathering and storing images within the element. For more information about this, please refer to the section titled Images

## 5.11. Custom Element Example: Global Custom Elements

There are several examples of Global Custom Elements in CommonSpot University. Here is a list of examples and a brief description about them

- **Careers/Jobs:** *This is a great example of a Global Custom element and a back-end management dashboard, all written without the aid of custom code.* The Jobs element stores available jobs at CommonSpot University. There is a back end page where internal resources can view and update available jobs. On the public-facing side, changes to the job status affect which jobs show up on the Careers pages. See “Custom Apps: Simple No-Code App: Jobs” for more information about how the jobs element is configured.
- **CSU Tube:** *This is a great example of a Global Custom element and a back-end management dashboard, but also takes advantage of customizations using custom code.* CSU Tube uses a Global Custom element to manage references to third party video services such as YouTube and Vimeo. This differs from the built-in video capability in CommonSpot in that it allows visitors to submit videos, rather than limited it to only internal users. Instead of uploading the video like they would with CommonSpot Multimedia, they use a form provided by the Visitor Video element which merely requires a unique reference to the video on the third party site. By providing the video’s URL, CommonSpot University can display the video directly on a page. This is done with a render handler that dynamically displays the appropriate video player with a reference to the desired video. For more details, see Visitor Contributed Video example in the Video section of this document).
- **Alerts:** The Alert system is described in detail elsewhere in this document.
- **Event calendar:** Using the PT Calendar app, the Event Calendar custom element is a fully functional web calendar system, complete with a back-end dashboard for managing one or more dynamic calendars on a site. Please see the Community site for more information about the Calendar.
- **Course Catalog:** The course catalog is described in detail elsewhere in this document.
- **Profile:** Using the PT Profile app, the Profile custom element contains information about people in an organization. It can be used to manage one or more groups of people and provide individual profile pages and lists or directories of people, such as a doctor finder, faculty and staff lists, student lists, management teams, employees, etc. Please see the Community site for more

information about the Profile app.

- **Blog:** A fully functional blog application is built around a global custom element. Please see the Community site for more information about the Blog.
- **Calls to Action:** CommonSpot University contains several image carousels at the top of pages which are designed to attract attention and direct users to other parts of the site. These calls to action use any of the several global custom elements designed for this purpose. The calls to action are described in detail elsewhere in this document.

## 5.12. Custom Metadata

Features: Custom Metadata, Metadata Forms, Page Index Element

Section:

Users: webmanager, contributors

Reference: Content Contributor's Reference

Custom Metadata, also known as Custom Properties, is CommonSpot's capability for allowing administrators to tag content with their own vocabulary of categorizations. While all content is tagged with Standard Properties automatically, you can use Custom Properties any way you require.

The demo site has several examples of how CommonSpot's Custom Metadata can be utilized in one of four basic ways:

- to classify content
- to tag content for linking and content reuse
- to capture additional page content
- to define design layout options

Please refer to the *Administrator's Reference* for more information about setting up and configuring Custom Metadata in CommonSpot.

The following sections spell out where you can find one of the many examples of Custom Metadata in CommonSpot University.

### 5.12.1. Custom Metadata to Classify Content

There are several examples of classification of content using Custom Metadata. The best example is the document library which uses a taxonomy to classify content. The result is an easy way to deliver a taxonomy-based navigation view for easy discovery of relevant content.

The Document Library can be found in `/demo/intranet/doclibrary/index.cfm`. Here, you'll see a page featuring a Page Index displaying an unfiltered view documents in the library. To the left, you'll find the Facet Based Navigation element, which displays facets and terms, from the CommonSpot University taxonomy, as links. Clicking on the links will result in filtering of the page index in the main area of the

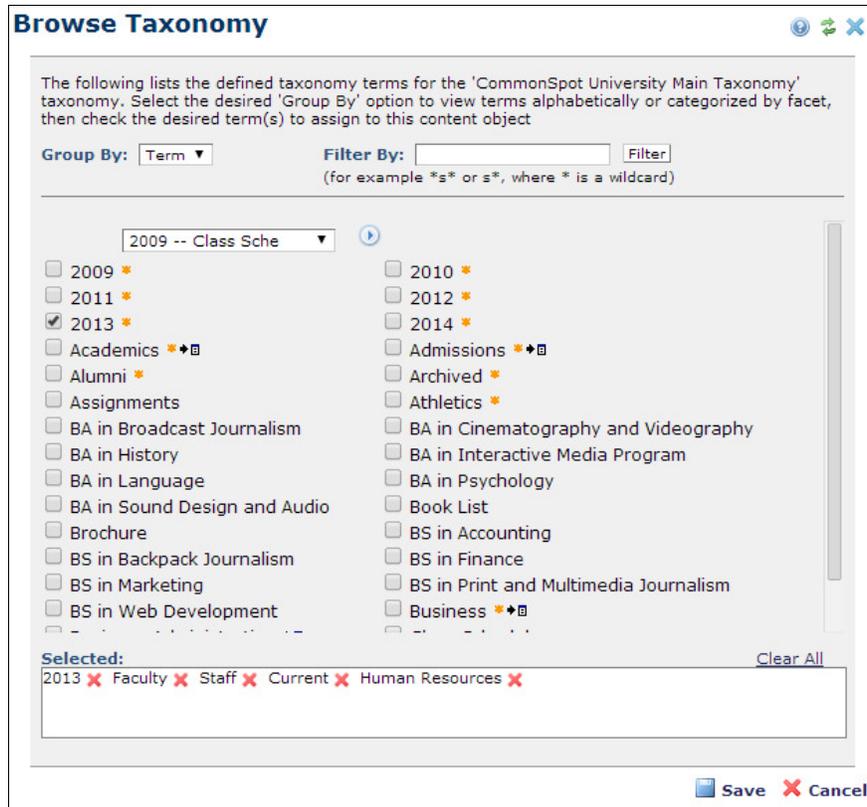
page. The more terms you click, the further filtered the list becomes. This is akin to online shopping, where you can do a broad search and then narrow down by price, manufacturer, color, etc.

The screenshot shows a 'Document Library' interface. On the left, under 'Narrow or Expand Results', there are two sections: 'Browse Academics' and 'Browse Audience'. 'Browse Academics' includes links for Business (5), Design and Technology (1), Journalism (2), Liberal Arts and Sciences (3), and Visual and Media Arts (3). 'Browse Audience' includes links for Faculty (5), Staff (6), and Students (7). Below these is a 'Browse Status' section with links for Current (11) and Draft (2). On the right, under 'Showing Results', there is a heading 'Browse topics at left to drill down and discover relevant documents.' followed by three document entries: '2013 Employee Holiday Calendar (23k)', '2014 Employee Holiday Calendar (0k)', and 'Admissions Application (0k)'. Below these are 'Backpack Journalism Brochure (0k)' and 'Benefits Overview (23k)'. Each entry includes a document icon and a brief description.

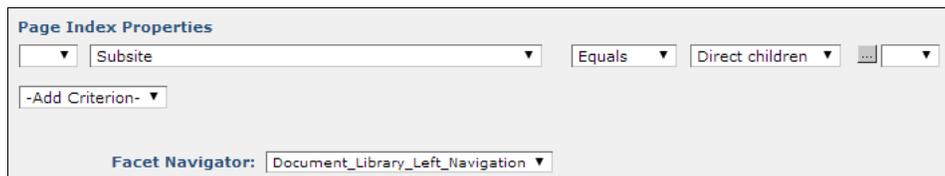
In order for the proper documents to show in the Page Index, we needed a way to easily identify them. One simple way to do this is to place all Document Library documents into a special subsite. We created a subsite located at /demo/intranet/doclibrary/documents/. Then, we created a metadata form that contains the taxonomy field type. We bound that form to the above subsite, so that any time a document is placed into that subsite, the form is automatically presented. The metadata form looks like this.

The screenshot shows a 'Custom Properties' dialog box. It has a title bar with a close button (X) and a refresh button. Below the title is the text 'Please provide the following information.' The main area contains a 'Classification:' label followed by a list of terms: '2013', 'Human Resources', 'Staff', 'Current', and 'Faculty'. Below the list are two buttons: 'Browse Taxonomy...' and 'Clear'. At the bottom right of the dialog are 'Save' and 'Cancel' buttons.

Clicking on "Browse Taxonomy..." will open up the Classification field dialog which uses the Taxonomy Browser field type. That dialog looks like this, and can be filtered by either term or facet.



Now that the metadata has been tagged for each document, the Page Index can easily be configured to display items based on selections in the Facet Based Navigation element, which are in that subsite.



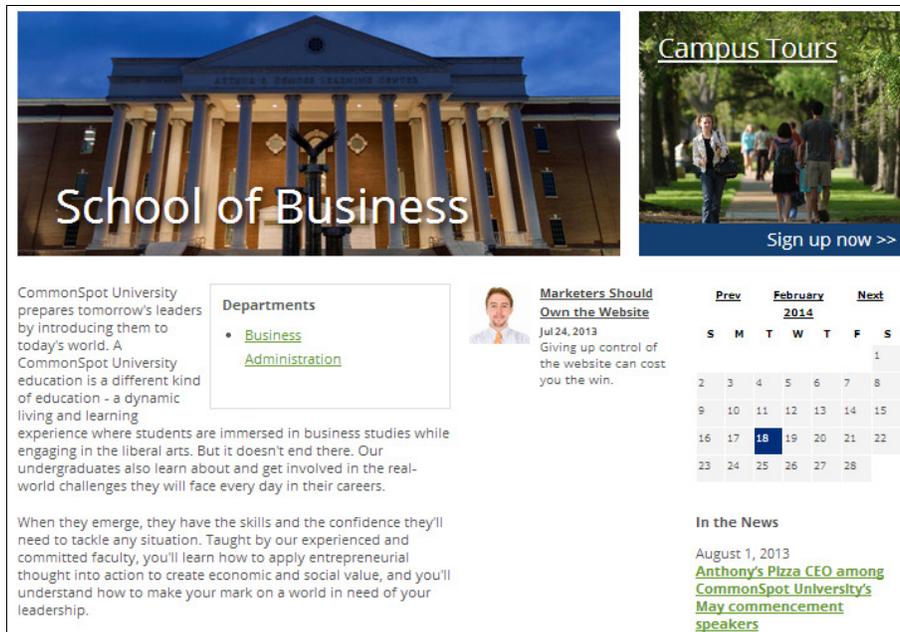
For more information on how the Document Library is configured, please see the section named “Taxonomy and Facet Navigation (Document Library)” later in this document.

## 5.12.2. Custom Metadata to Tag Content for Content Reuse

While classifications can be used for a number of things, one of the common uses is for the classification to directly drive the reuse of content on various sections of the site. One such example is the metadata form field “Related Schools” which gets presented for both News Articles and Press Releases.



Selecting a school will allow that article to show up in a dynamic list of links to related news articles on the appropriate School's home page. Notice the "In the News" section on the School of Business home page, below.



The screenshot displays the School of Business website layout. At the top left is a large image of a classical building with columns and the text "School of Business". To the right is a "Campus Tours" section with a photo of students walking and a "Sign up now >>" button. Below the main header is a "Departments" section with a list containing "Business" and "Administration". Next to it is a featured article titled "Marketers Should Own the Website" dated Jul 24, 2013, with a small profile picture of the author. To the right of the article is a calendar for February 2014, showing the 18th as the current date. Below the calendar is an "In the News" section dated August 1, 2013, with a link to "Anthony's Pizza CEO among CommonSpot University's May commencement speakers".

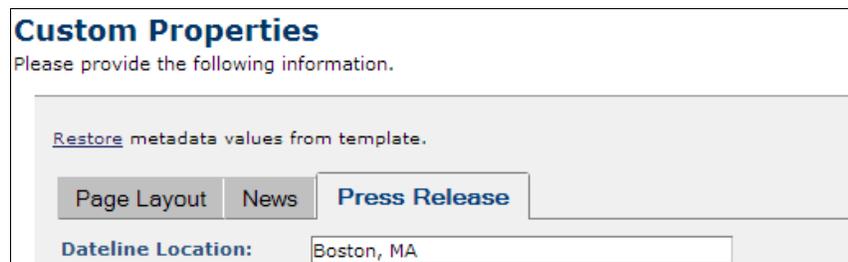
Another great example is the list of documents available for each Residence Hall on campus. Please refer to Residence Hall Document Libraries for more information on how these are configured.

### 5.12.3. Custom Metadata to Capture Additional Page Content

In addition to using your own fields to classify or drive content reuse, you can use the metadata value as content too. When content is created or edited directly on a page using a specific type of element, you still may require some piece of information in a very structured or consistent format. For example, the Press Release template in CommonSpot University allows the page author to add the content to the page directly into a Formatted Text Block element. However, the dateline information needs to be formatted a special way each time, even though the actual information may vary from page to page.



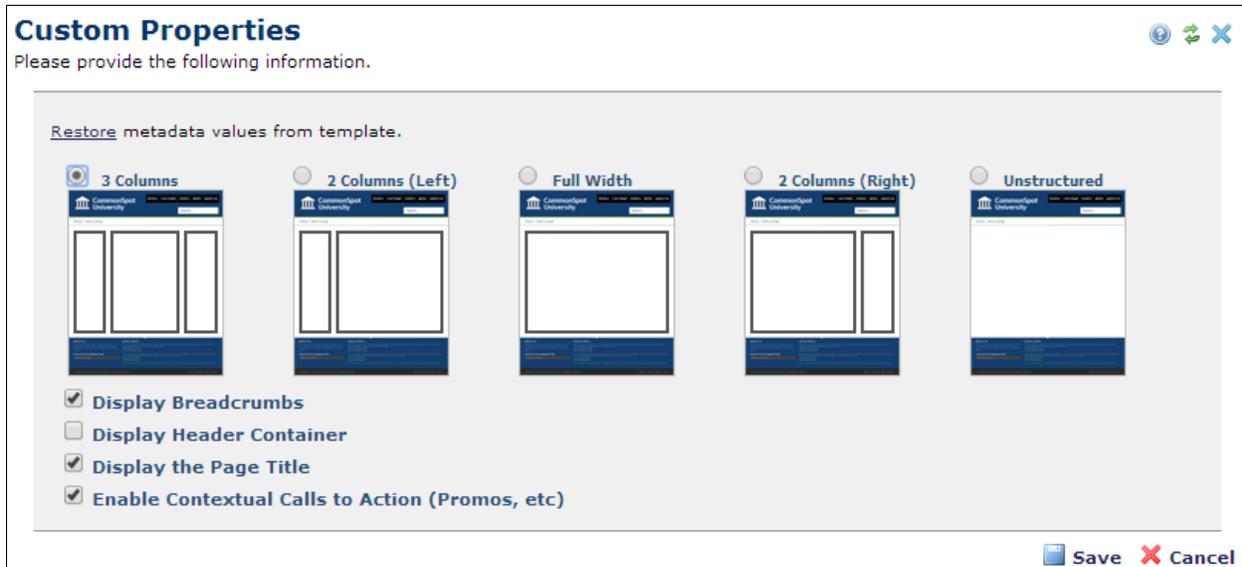
The dateline section, in the image above, says “Boston, MA, August 1, 2013”. Immediately following the date is where the author’s contributed text begins. But the dateline comes from two places. First, the date itself comes from the page’s standard properties (publication date). That was easy because that information is already available in the Standard Properties for the page. But the location is not a standard field, and may vary depending on how you manage your content. So, a metadata form was created that collects the location in a structured and consistent way.



## 5.12.4. Custom Metadata to Define Layout Options

Another way to use Custom Metadata has less to do with the content itself as it does with the way the content is presented. Designs of websites may vary greatly from site to site. And within the design, the designer may have allowed for variations. Site administrators may lock down some of these variations for specific templates, but it also may be completely within the style rules of the site to allow the content author to make some decisions on how the content gets presented.

The best example is whether or not to display supplementary regions of the page, such as left navigation, right column content wells, breadcrumbs or page title. The following metadata form is presented for every page on the site. It is bound to the base template. It contains rules such as how many columns should the base template present, whether or not to show the breadcrumbs (useful when making landing pages), and whether or not to show the page title, or the much larger header container.



## 5.13. Custom Script (University Bookstore)

CommonSpot supports the ability to add custom code to any page or template by using the Custom Script element.

One such example on CommonSpot University is the University Bookstore page. Since the fictitious bookstore uses a fictitious third party e-commerce system used at the University, the content is not available using standard mechanisms in CommonSpot (such as content reuse of CommonSpot content). But the administrators of CommonSpot University still want up to date information available on this page to advertise specials and to direct traffic to the bookstore home page.

### CommonSpot University Bookstore

---

The CommonSpot University Bookstore is owned and operated by a third party bookseller. Use this link to go directly to their home page.

[College Bookstore, Inc.](#)

Or, browse some of the latest specials directly from their inventory:

Sale Price	Dept. Code	Book
\$35 <small>In Stock</small> <a href="#">Buy Now</a>	BA	Pierstorff, <i>Advertising Basics</i> Save \$30 off regular price of \$65 <small>Creating and deploying effective advertising.</small>
\$30 <small>Backordered</small>	BA	Unruh, <i>Principles of Marketing</i> Save \$5 off regular price of \$35 <small>The marketing landscape of today.</small>
\$25 <small>In Stock</small>	IM	Hardiman, <i>Cinematography for Modern Production</i> Save \$40 off regular price of \$65

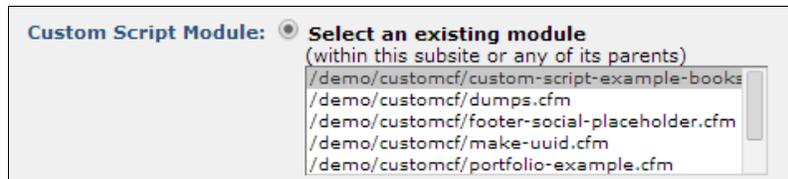
**Related Links**

- [Academics](#)
- [Bookstore Sales](#)
- [Course Catalog](#)
- [Faculty Directory](#)

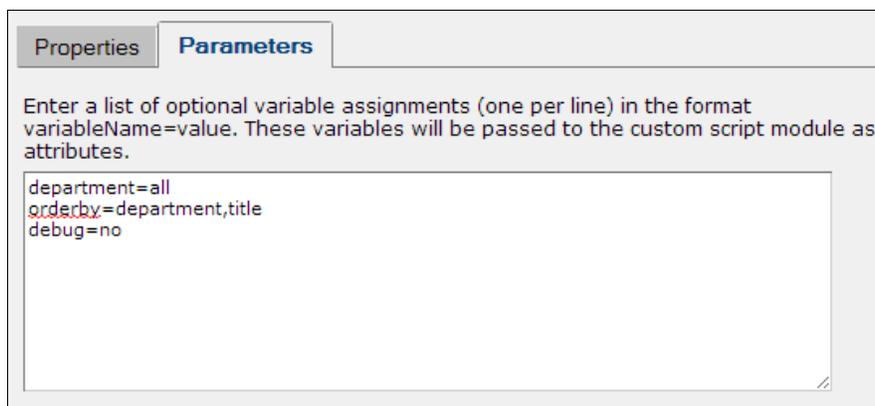
The developer has created a module and placed it into the /demo/customcf/ directory on the site. The file is located at /demo/customcf/custom-script-example-books.cfm This module does the following things:

- Queries the bookstore database, a non-CommonSpot-managed database that has a datasource accessible from the CommonSpot server.
- Loops through the query, displaying the content using the look and feel of the site
- It also takes into account some user-specified parameters or attributes such as filters and sort order so that the same script can behave appropriately on various pages. See attributes, below

Adding this to a page is as simple as adding the Custom Script Element to the page. Because the script itself is in the standard “customcf” folder, it is available in the dropdown list in the element dialog.



Dropping it onto the page will render out a list of bookstore specials. However, if you want to further control what shows in the list, there are some parameters that you can optionally specify using the Parameters tab on the form.:



These parameters are specific to this specific script example. Any script can be configured to accept and use any parameter any way that it sees fit. In this case, we have three parameters that we can specify. I can specify them each on their own line. They are optional, which means that if I leave one out of the list, the code will use whatever the developer specified as a default value. For developers, these variables will appear in the attributes scope when the script runs on the page.

In our example, the possible attributes are as follows:

- department: Either choose “All” or the two letter code for the department. The department codes are the same two letter codes used in the Department element, as part of the course catalog elsewhere on the site.
- orderby: This will let you specify what order you wish the results to appear in. The possible values are department, title and author. Specify a list of as many or as few as you wish. These values are not the exact values needed in the database query. In our script example, the values

get parsed and translated to the actual values so that there is no risk of an injection attack using this example.

- debug: This boolean will allow you to show or hide some cfdumps of the query and attributes scopes. This can be useful when learning more about the Custom Script element in general or this specific custom script.

## 5.14. Datasheet

Some CommonSpot content can be displayed in a table view using the Datasheet Element. There are several examples of using a Datasheet to display content in CommonSpot. Most of them can be found in the Web Administration section of the site at /demo/webadmin/

Some notable examples are listed here, each covering a different way to populate the datasheet, including Simple Form data, Custom Element Data, and a custom database query.

### 5.14.1. Visitors Survey (Simple Form data in a Datasheet)

The visitors survey is a simple form located in the About subsite. When visitors fill out the form, the data can be retrieved by internal resources in the Forms section of Web Administration. The datasheet is configured to display multiple views of the visitor data, a link to download the data set, a row for each record, filtered either by type of visitor (from the form field) or showing all records, an Edit link which presents the data in the original form for editing and a delete link which will allow the user to delete the record.

The screenshot shows a web interface titled "Survey Administration". At the top left, there is a "Views:" label followed by a dropdown menu set to "All Results". To the right of the dropdown is a green link labeled "Export Data to Spreadsheet". Below this is a table with the following columns: "Date Added", "First", "Last", "Email", and "I am a". The table contains two rows of data. The first row shows a record added on August 28, 2013 at 7:48 AM, with first name "Patricia", last name "Moreno", email "pmo@commonsspot.edu", and role "Student". The second row shows a record added on September 13, 2013 at 3:34 PM, with first name "Walter", last name "Springfield", email "walter@commonsspot.edu", and role "Student". Each row has an "Edit" link to the left and a "Delete" link to the right. At the bottom left of the table area, it says "2 Records Returned".

Date Added	First	Last	Email	I am a
<a href="#">Edit</a> August 28, 2013 7:48 AM	Patricia	Moreno	pmo@commonsspot.edu	Student <a href="#">Delete</a>
<a href="#">Edit</a> September 13, 2013 3:34 PM	Walter	Springfield	walter@commonsspot.edu	Student <a href="#">Delete</a>

### 5.14.2. All Apps (Custom Element data in a Datasheet)

The Web Administration area of the site contains links to all the custom built applications used in CommonSpot University. These apps all contain one or more Custom Element, therefore the content can be easily managed inside of a Datasheet. Refer to the Web Administration section of the site for examples, and refer to more complete app descriptions elsewhere in this document.

**Job Manager**

[Add New Job](#)
[Add Job Department](#)
[Add Job Location](#)

Views: Job Listings - Published [Download Listings](#)

	<a href="#">Job Title</a> ▲	<a href="#">Date Posted</a>	<a href="#">Department</a>
 	Assistant Men's/Women's Basketball Coach	October 2, 2012	Athletics
 	Assistant Professor of Business	July 23, 2013	Academics
 	Assistant to the General Manager	February 19, 2013	Academics
 	Associate Producer	July 11, 2013	Student
 	Athletics Administrator	October 19, 2012	Athletics
 	Digital Content Manager	June 3, 2013	Admissions
 	Enrollment Advisor	July 3, 2013	Admissions
 	Events Manager	March 12, 2013	Facilities
 	Human Resources Specialist	June 4, 2013	Human Resources
 	Student Radio Advisor	April 9, 2013	Student
 	Ticket Sales Administrator	May 29, 2013	Athletics

11 Records Returned

### 5.14.3. University Bookstore (Custom Query in a Datasheet)

A Datasheet can also display content from another data source, outside of CommonSpot . The University Bookstore, as explained above, is an example of a fully custom coded content block. But that same external database table can be used as a source for a Datasheet too.

There is a page called “Bookstore Sales” which does just that. It is located at </demo/academics/bookstore-sales.cfm>

On this page, there is a datasheet with various pre-defined filters, by school, which show a list of items from the bookstore database.

### Bookstore Sales

Here's what's on sale at the University Bookstore. Browse all, or choose a department:

Views:

[A](#) | [B](#) | [C](#) | [D](#) | [E](#) | [F](#) | [G](#) | [H](#) | [I](#) | [J](#) | [K](#) | [L](#) | [M](#) | [N](#) | [O](#) | [P](#) | [Q](#) | [R](#) | [S](#) | [T](#) | [U](#) | [V](#) | [W](#) | [X](#) | [Y](#) | [Z](#) | [All](#)

Sale Price	Regular Price	Author(s)	Title ▲
35.00	65.00	Pierstorff	Advertising Basics
25.00	65.00	Hardiman	Cinematography for Modern Production
25.00	45.00	Daigle	Discovering Journalism
40.00	60.00	Seeley	Feature Writing
30.00	59.00	Tobey	Intro to Web Development
25.00	40.00	Schneider	Introduction to Psychology
30.00	55.00	Kingsbury	Making Documentaries
30.00	35.00	Unruh	Principles of Marketing
35.00	45.00	Merrill	Web Development Languages

9 Records Returned

In this case, instead of specifying a form or element as the data source, we specified a query, such as in this dialog.

#### Datasheet Custom Query Information

Please provide the custom query information to use as the datasheet source.

**Data Source Name:**   
Enter the datasource name which connects to the desired database.

**User Name:**   
If the database requires authentication enter your username. (optional)

**Password:**   
If the database requires authentication enter your password. (optional)

**Custom Query:**   
Enter a valid SELECT SQL statement to run to retrieve the data.

## 5.15. Document and Asset Management

The demo site has a number of external documents and assets uploaded to the system for use as link targets. These documents and assets can be linked from any link, linkbar, or link inside a WYSIWYG Rich Text editor. For example, try browsing to a WYSIWYG page under About CSU, editing the Formatted Text Block, highlighting and linking some text, and linking to an 'Existing Page, Document, or URL'. Select the

Choose option, then select the 'Resources' subsite from the Subsite dropdown to see demo site documents listed as linkable assets.

Users can manage their own documents from the 'My Uploaded Documents' report. Try logging in as the webmanager and accessing the 'My Uploaded Documents' report from My Content, to see a number of uploaded and external documents (links to other sites) listed. Try logging in as a different user, and initiate an 'Upload Document' and/or 'Register External Page' to move some external documents and assets into the system. Then view the 'My Pages' report to see these listed.

Administrators can also use the Reports menu to manage all documents and assets in the site. From the Reports menu, use the Pages, Templates Uploaded Documents, & Registered URLs report to view document assets by type. Check Include Child Subsites to view a report for all subsites. Notice how this report gives you the option to navigate to a page to make changes and makes multiple other options available for page change and management in the Actions column. The Report of Images gives you similar options for managing this asset type. This is a great document management tool for administrators to maintain the various documents in the system.

Additionally, we have established a document library where visitors can browse and download documents. Please refer to "Custom Metadata to Classify Content" section of this document.

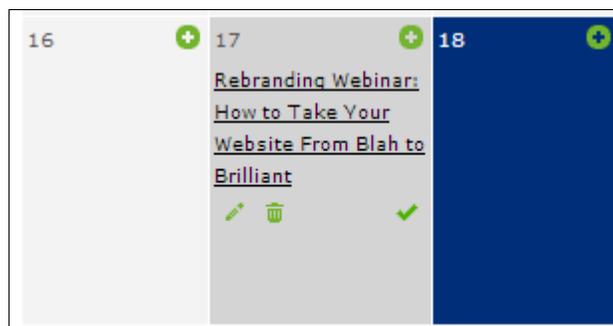
## 5.16. Events Calendar

The demo site implements an events calendar that is configured to display events by Day, Week, and Month views, as well as some 30-day calendar views. The primary events calendar is displayed under the Events section of the site. Events are defined as a 'Global Custom Element', a reusable content object.

This event calendar comes with some displayable elements which can be used on any page and configured to display existing events. The Events Calendar has a unique feature that allows users to define both specific event dates, and recurring event dates. Specific dates can be stored directly in a custom element, but the recurring dates are actually stored as configurations in a separate element. The corresponding dates from a recurring event rule does not actually have a record in the Event Calendar. This means that you can't simply display an accurate list of event dates if you merely work with the Event custom element. In order to get a complete list, you need the Events Calendar app to do some additional heavy lifting, aggregating regular events and recurring events into one list. The Events Calendar app contains some of these display elements which handle the aggregation. For example, the home page uses the element called "Calendar Widget Properties - Upcoming Events - Single", which is specifically designed to aggregate event listings and display them in a list.

The home page filter looks as follows. Notice how it allows the user to select multiple organizations and/or event categories in order to create the list of relevant events.

Management of the Events Calendar is done entirely within the Web Administration area of the site, located at /demo/webadmin/calendar/Calendar-Events-Manager-Grid.cfm Here, you can manage events organizations and categories, in addition to managing the events themselves. The user interface for managing events allows for direct editing of the event within an intuitive calendar interface. In the image below, note the Plus sign for adding an event, as well as the Pencil icon for editing and the Trash can for deleting an event.



## 5.17. External User Authentication

This example has been removed from the demo site for the 9.0 release. We plan to add this back into a future version.

## 5.18. Forms

CommonSpot's Simple Form element is the easiest way to present a form on a page for user or visitor input. A simple form can be created and configured for a specific purpose, or it can present fields from a Custom Element.

### 5.18.1. Simple Form Example

As an easy example of a Simple Form, we have created a Visitor Survey in CommonSpot University. This page is located at /demo/about/visitor-survey.cfm The form contains a few text fields, as well as a dropdown and radio button example. It also uses Captcha, which ensures the respondent is human, and not an automated robot.

**Visitor Survey**

 We want to hear from you. Thank you for visiting us today. Whether you came to our campus or are viewing us here on the web, your feedback is valuable to us. Please provide us with any feedback you may have regarding the University, the web site or your visit.

**First Name:**

**Last Name:**

**Email:**

**Where did you visit?:** Cafeteria ▼

**I am a:**  Prospective Student  Student  
 Alumni  Parent

**Comments:**

reCAPTCHA™ stop spam, read books!

The field definitions for this form are as follows.

**Edit Simple Form Fields** 🔍 ↻ ✕

Add, edit and delete fields as needed.

**visitorSurvey**

Field Name	Label	Field Type	
<a href="#">firstName</a>	First Name	Text	
<a href="#">lastName</a>	Last Name	Text	
<a href="#">email</a>	Email	Email	
<a href="#">campus</a>	Where did you visit?	Selection List	
<a href="#">classification</a>	I am a	Radio Buttons	
<a href="#">comments</a>	Comments	Large Text Area	

[Add New Field](#) [Add Shared Field](#) [Add New Tab](#)

✕ **Close**

In the Web Administration area, a Datasheet has been set up to view input from this form. For more information, please see Datasheet Element, elsewhere in this document.

## 5.18.2. Simple Form with Custom Layout

The standard layout for a Simple Form is to present the field labels to the left in approximately 25% of the usable space, and the fields themselves to the right in the remaining space. This may not always be practical. An alternative is to present the form using a Custom form layout.

For example, on the Alumni page, located at `/demo/alumni/index.cfm`, there is a form in the right column where the visitor can contact their representative. Since it's in the right column, there isn't enough space for the standard layout. We have created a custom layout by going to the Simple Form, choosing Layout Properties, and selecting Custom Layout. The resulting layout looks like this, with the field labels above the fields.

If you don't have an account, use this form to contact your regional Alumni representative.

First Name

Last Name

Email Address

Class of

Region

### 5.18.3. Simple Form using Custom Element

One way of populating records in a Global Custom Element is to present the element's fields on a page using the Simple Form element. Many of the Global Custom Elements in CommonSpot University have a management interface in the Web Administration section, which in some cases include pages with Simple Forms on them. Some notable examples are:

- The Jobs app contains a simple form for adding a new job listing, located at `/demo/webadmin/jobs/add-job.cfm`
- The Jobs app contains a simple form for adding a new Job Department, located at `/demo/webadmin/jobs/add-department.cfm`
- The Jobs app contains a simple form for adding a new Job Location, located at `/demo/webadmin/jobs/add-location.cfm`
- The CSUTube section of the site displays visitor contributed video. Each video is stored in a Global Custom Element record. While there is a management interface in the password protected Web Administration, there is also a publicly available version of the form located at `/demo/campuslife/csutube/submit-video.cfm`

### Submit A Video

**Title:**

**Video URL from service:**   
Fully qualified URL from YouTube or Vimeo

**Category:**

Features  Partners  
 Content Management  Marketing  
 Sports

**Description:**

## 5.19. Field Level Security

Using “field-level” security, CommonSpot can dynamically show or hide form fields depending on who is viewing the form. This allows you to create and deploy forms that can adapt to the situation and give you more control over what information is collected and who can provide it. Below are two examples of Field Level Security in CommonSpot.

### 5.19.1. Enable Calls to Action Field

This field is located in the Layout metadata form, which is presented for any page on the site. This field determines whether or not a region of the page, on the left column, will display promos. The Base Plus One template is configured to allow for personalized promos or “calls-to-action” (see Contextual Personalization). Administrators need to turn this on for any page wishing to have this feature displayed. Here is what an administrator would see. Note the last field on the form.

#### Custom Properties

Please provide the following information.

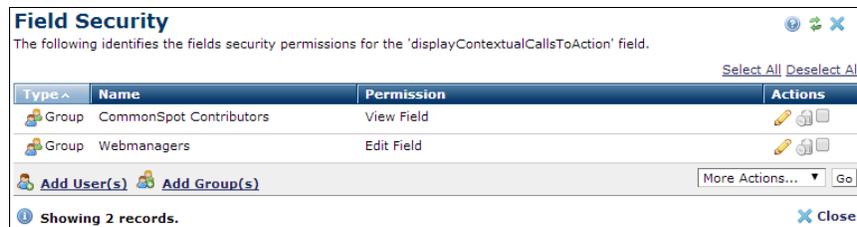
[Restore metadata values from template.](#)

3 Columns  2 Columns (Left)  Full Width  2 Columns (Right)  Unstructured

Display Breadcrumbs  
 Display Header Container  
 Display the Page Title  
 Enable Contextual Calls to Action (Promos, etc)

Save Cancel

The field controls whether or not the promo region of the page is displayed, and it is only clickable by Administrators. Here is the Field Security Settings dialog, found at Admin > Metadata Forms > Layout (form name) > displayContextualCallsToAction (field name) > Security > Edit Security Settings.



Note, the Webmanagers Group has full edit capabilities on this field, while the Contributors can merely view the field. For those users, they would see an uneditable version of the field, but properly indicating its current value.



## 5.19.2. CSUTube Approved and Featured Checkboxes

Another good example is the “Submit A Video” form in the CSUTube section. CSUTube is a library of videos submitted by the community, not just internal users. To enable this, CommonSpot University needed a way to manage and approve the video submissions.

Any anonymous visitor can submit this form, which puts that video into the library in an unpublished state. The records are stored in a global custom element, which has a field titled “Approved”. Only videos marked “approved” will show up on the publicly facing pages. This field has field-level security on it, only allowing internal users the ability to even see it. Anonymous visitors don’t see the field and upon submission, it remains unchecked until an internal user opens up the record and approves it.

Submit a video form, as it appears to anonymous visitors.	Submit a video form, as it appears to internal users. Note the additional checkboxes.
---	---

The image shows two identical 'Submit A Video' forms side-by-side. Each form contains the following fields and options:

- Title:** Text input field.
- Video URL from service:** Text input field with the note 'Fully qualified URL from YouTube or Vimeo' below it.
- Category:** Radio button options for Features, Partners, Content Management, Marketing, and Sports.
- Description:** Text input field with a small icon for text formatting.

The right-hand form includes two additional checkboxes at the top:  Approved and  Featured.

To control the display of videos on the publicly facing pages, we merely use the “Approved” value as filter criteria everywhere we use the CSUTube element.

The image shows a 'Dynamic Filter' dialog box with the following configuration:

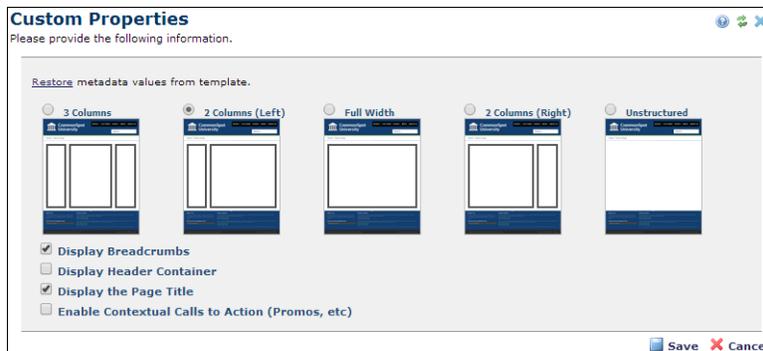
- Filter:**  All records,  Subset of records
- Criterion:** Visitor Video.Approved, Equals, Yes
- Sort Order:** Date Added, Descending
- Buttons:** Save, Preview Records, Cancel

Another field is “Featured” which is another checkbox that works just like “Approved”. The only difference is that this field is used to denote something the admins want to feature on the site. This gives them additional filter criteria for dynamic rendering.

## 5.20. HTML Design and CSS

The CommonSpot University design was created by an outside graphic designer, and implemented by PaperThin, as an example of how to migrate professional HTML and CSS-based designs into a CommonSpot Web site. The CommonSpot University Web site implements several layouts for different users, including one for the home page and others for various combinations of columns quantities and widths.

HTML design is managed outside CommonSpot using your HTML editing software of choice. The main HTML designs are stored in CommonSpot as Base Templates. Implementing and registering these HTML designs as a Base Template is fairly simple, requiring only a few basic steps. Implementing one is made easy in CommonSpot through its extensive support of custom metadata, enabling end users to select which design is implemented within a ‘Layout’ metadata form.



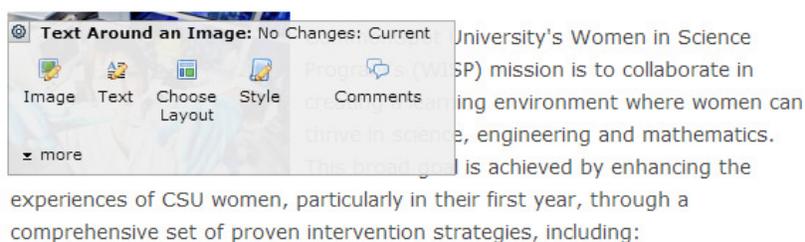
The base HTML design is saved on the Web site’s file system under ‘/demo/templates/’ and can be opened in the editor of your choice. For more information on how the demo site Base template was implemented, and how to leverage custom metadata to invoke different designs, refer to the Template Developer’s Guide and the Developer’s Guide.

Style Sheets are also managed in the CSS editor of choice. The process of migrating a Style Sheet file into a CommonSpot Web site simply requires saving the CSS file under your site’s ‘/demo/style’ directory and registering the style with the appropriate template. For example, the main CSS for the demo site is registered in the Base Plus One template under the ‘Style Sheets’ menu on the CommonSpot toolbar. The CSS file itself can be opened in the editor of your choice to manage the styles implemented on the CommonSpot University site.

## 5.21. Images

The demo site has a number of pages with images in the content. These images are managed by first adding an image element to a container or using the Insert Image tool in the rich text editor, and then selecting the image source from the Image Gallery. You can change the image being used, or even upload a new image from your local file system by selecting Edit on any image element’s toolbar. You can also crop, resize, or manipulate the image file using CommonSpot’s browser-based Image editor.

CommonSpot handles a variety of digital assets, including images. The digital asset management functionality allows users to easily edit and manage images, photos, and other graphics directly within the system. Several popular image functions are available, including image crop, rotate, zoom, compress, flip, adjust contrast and brightness, and more. For more information on the Image editor, see the Contributor’s Reference.



Additionally, some types of content in CommonSpot University are managed as fields in custom elements, giving you additional control over how they are displayed and what sizes are allowed. Here are some examples of using custom elements to manage images.

## 5.21.1. Carousels

Also known as photo galleries, slideshows, sliders, etc., carousels allow you to rotate or cycle through several images in one space on a page. These are used for highlighting images, or content or even links to other pages on a site. CommonSpot University has several image carousels on the site. A primary example is the home page and all subsite home pages.

Below, notice the two main image regions on the home page just beneath the header. The one on the left is for featured content, and it cycles through images along with some additional text including a title, subtitle and a link to the content. The one on the right is called a call-to-action, where we cycle through several promos or offers for the visitor to interact with the site. Both behave the same way, but have different layout, footprint and timing to differentiate it from the other.



The subsite landing pages all have similar configurations. The main image on the left of those pages is not used as a link, however. Instead, it's merely a backdrop for the subsite title. These are some good examples of using image carousels to make your pages more compelling and inviting.

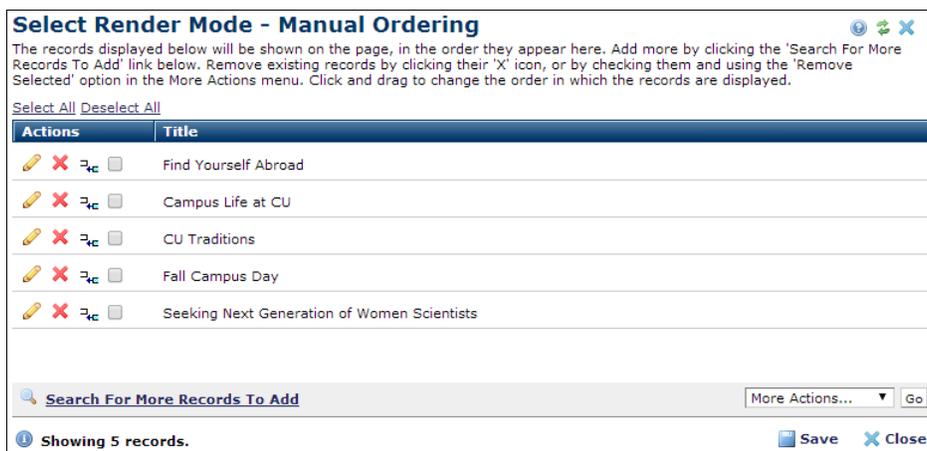


Each of the above examples are enabled using a simple technique. For each, a custom element was created that can manage both the image info along with the additional metadata such as title, subtitle and link. Additionally, the elements may also contain fields which capture layout information. For example, on the main home page, carousel, the image's title and subtitle can be placed anywhere on the image you wish. In order to accommodate this, the element's form also gathers positioning preferences, image by image, so that it can be used at render time.

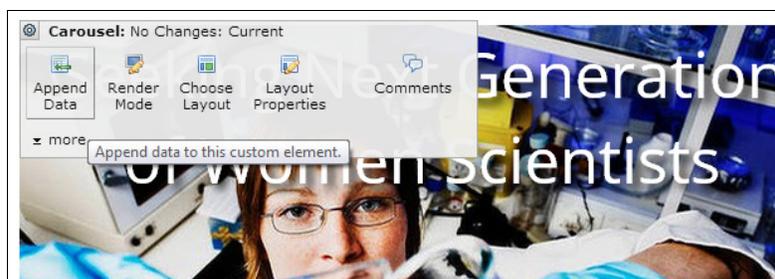
Additionally, the element does not use a standard image field. Instead, we have employed the use of the ADF's custom field "Photo Cropper", which allows the user to upload images with the exact required dimensions, every time. If it's not the right dimensions, a cropper tool will present itself at upload time. For more information, please refer to the Community Site for information on the ADF and its custom fields.

The image shows a screenshot of a web form titled "Datasheet Action - Edit Custom Element Data". The form is for editing a "Carousel" element. It contains the following fields and options: "Photo:" with a small image of a campus scene and a "[Remove]" link; "Title:" with a text input field containing "Fall Campus Day"; "Page Link:" with a text input field containing "Fall\_Campus\_Day" and "Select..." and "Clear" buttons; "Justify:" with radio buttons for "Left", "Center" (selected), and "Right"; "Top:" with a text input field containing "50" and a note "Distance from top of image, in pixels."; "Margin:" with a text input field containing "20" and a note "Distance from justified edge, in pixels. Ignored if center justified."; "Description:" with a text area containing "Come for a visit and see what CU is all about." At the bottom of the form, there is a "Check Spelling" checkbox (checked), a "Save" button, and a "Cancel" button.

In the above examples, we have turned on "Manual Selection" of records, allowing us to pick and choose the images we want at any given time. But because CommonSpot allows you additional rendering options, such as Dynamic rendering based on a filter, there is no limit to how you can configure your carousels.



To add to a carousel, the user simply adds a record to the custom element. They can simply choose the “Append Data” button to add records to an existing list of manually selected items. This makes it really easy to add items to the carousel.



To control the rendering of the carousel, site designers can use just about any technique, including javascript and css libraries which handle the fancy effects and layouts. CommonSpot supports these by allowing you to provide your own CSS, as well as injecting custom code into the rendering layer using what we call “Render Handlers”. For more information on render handlers, refer to the Developer Guide. Each carousel element in CommonSpot University has appropriate render handlers for displaying those images the way the designer intended. In the above example, the positioning of the title and subtitle is determined by the values of the “Top” and “Margin” fields.

## 5.21.2. Other Custom Content Types: News, Blogs and Residence Halls

Similarly, images are gathered for specific content types such as News and Blogs. By requesting them as fields rather than embedding them into the rich text of a page, administrators and template designers can have more control of their presentation, keeping you aligned with your brand at all times.

In the case of Residence Halls we have provided two alternate ways of managing images all within the same element. The purpose is more educational than practical. Typically, you’d settle on one of the following methods.

- **Image Field:** This is the traditional CommonSpot image field. Using this field allows you to input an image that can be referenced using the WYSIWYG Display Template editor. However, it lacks the ability to control image dimensions, and therefore can result in layout variations. Images uploaded in this manner are used by the Display Templates defined for the Residence Hall element. See [Residence Halls](#) for more information on how these layouts are defined.
- **Photo Cropper:** This is a custom field type provided by the ADF which allows the user to upload images with the exact proper dimensions, every time. This is because it uses some of the structure of the Photo Gallery app (see [Community Site](#)) for explicitly controlling image sizes for specific purposes. Using this allows you to be sure that an image is of the proper size and/or aspect ratio, so that it can be displayed in a common and consistent manner. Images uploaded in this manner as used by the render handlers defined for the Residence Hall element. See [Residence Halls](#) for more information on how these layouts are defined.

The Residence Hall form is as follows. Notice how it further explains the example concept presented here.

### Enter Custom Element Data

Please enter the following information for the 'Residence Halls' element.

Residence Halls **Images** Description

The Residence Hall element demonstrates two alternatives for displaying and managing images. The first one is easy to use in simple views and non-code based layouts. The second one gives administrators more control and is only visible in coded layouts. We include both here as examples.

This photo uses the image cropper, giving the site administrator explicit control over size and dimensions.

Photo:  [Remove]

This image field captures any image and places it into the Image Gallery. For best results, make sure your image is 460 x 230.

Photo: 

Select Image... Clear

Photo from the Image Gallery. For best results, make sure your image is 460 x 230.

Check Spelling

## 5.22. In-Context Content Object Authoring

CommonSpot's in-context editing goes beyond just WYSIWYG editing. CommonSpot allows you to capture content from users at a more granular level using a structured content object form interface. Examples of this can be seen in how many of the pages on the site are edited, including News and Residence Halls.

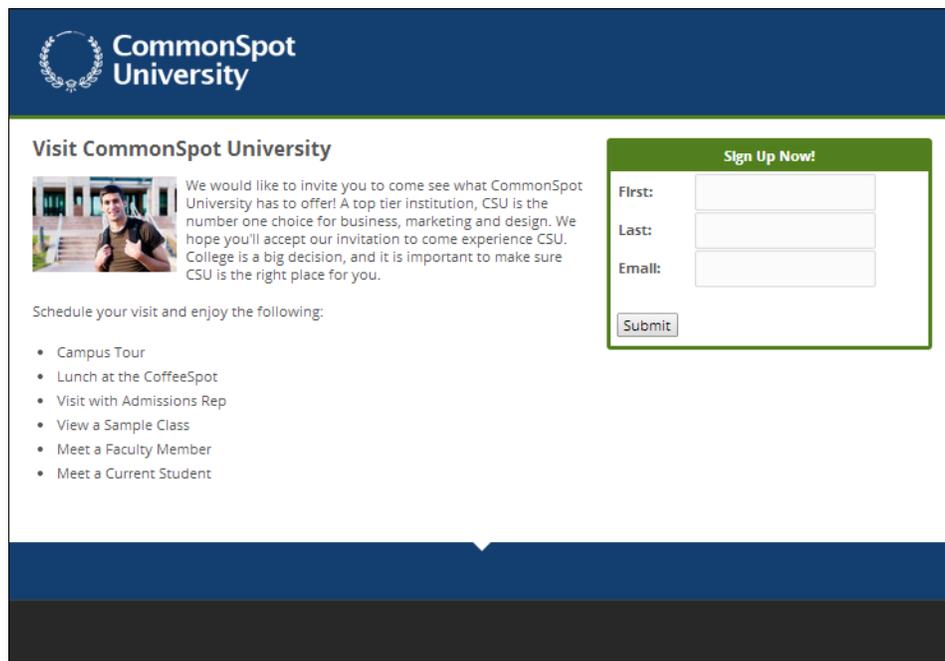
The content is captured inside a reusable content object (or content type), known as 'Local Custom Elements' in CommonSpot, which can then be reused on other site pages. The local custom element presentation is pre-configured and saved in a template so that when users create or author a content object page in the demo site, the local custom element content is captured in-context, and submitted back into the page in its pre-defined template presentation. This is a good example of how CommonSpot separates content from presentation.

In contrast, items like the Events Calendar or Blog use Global Custom Elements. In these examples, the content is not edited in the context of the page, rather it is editing in a centralized dashboard interface (using a Datasheet element) where users can view all the records in tabular form and then edit them.

## 5.23. Landing Pages

Converting visitors into customers isn't as simple as putting a form on a page. A good "landing page" will complement the form and give you your best chance at converting the visitor. CommonSpot University has several landing pages that show how you can use CommonSpot to create multiple, consistently laid out, but flexible landing pages.

A good example is the Visit CommonSpot University landing page located at </demo/admissions/visit-commonspot-university.cfm>



### 5.23.1. Building Blocks of a Good Landing Page

A good landing page has the following components:

- Headline (in the above screenshot, the headline is "Visit CommonSpot University")

- Hero shot, which is a photo or video of what you're selling
- Itemized benefits: A bulleted list supporting your main message
- Social Validation: Testimonials from customers or social media
- Additional Reinforcements: Video, images, lists, etc.
- Call to Action: Clear step toward conversion. In the above screenshot, the call to action is "Sign Up Now" followed by the form. Choose from:
  - Take them to something
  - Invite them to chat with you
  - Ask them to fill out a form

## 5.23.2. Landing Pages on CommonSpot University

Here is some basic information about how the landing pages are configured within CommonSpot University.

- All landing pages are in the Admissions subsite. This is likely where the "marketing" of our organization takes place.
- The landing pages are built off of a template called "Marketing Landing Page"
- The landing pages are not made available through the website's menus. Landing pages are intended as places for visitors to come too first. As such, they should not appear to visitors already on the site.
- The landing pages contain no built-in navigation. Landing pages are designed to not only attract and convert the visitor, they typically want to capture their information before letting them proceed. Therefore, the only obvious path to the site is to use the form. Notice how the header and footer are blank. There should be no link on the page that takes the user directly to the site without first filling out the form. The one exception is the CommonSpot University logo, which is linked to the home page. Normally, this would not be the case. Please forgive us for the oversight. We'll do better next time.
- To remove the navigation elements, including the menus, breadcrumbs and footer links, we have made certain regions of CommonSpot University pages configurable using the base template and custom metadata. See Custom Metadata elsewhere in this document for more information about how to configure this. The choices available to us in the landing page template are as follows:
  - The metadata form called Page Layout allows the user to specify whether or not to display the Breadcrumbs.
  - Another metadata form, called Landing Page Config, allows the user to hide the header and footer navigation.

## 5.23.3. Configuration Layout of Landing Pages

The landing pages on CommonSpot University have some additional configurable options, built using a combination of custom metadata and element scheduling. The Landing Page template has two columns with fixed widths, determined by the designer and administrator. In the template, we've used scheduling on the elements to show or hide based on the values selected in the metadata form.

- The left column contains a required region where the image (“hero shot”) and text can go, as well as a place for the bulleted list. See list of best practices, above.
- The right column contains a region for the call-to-action and form.
- Both the left and right columns can also support, at the discretion of the page builder, some additional regions. Using the metadata form, the user can select from a list of built-in elements (Video (Multimedia Element) and the custom Testimonials Element), or an empty container.
- There is also a full-width column at the bottom where they opt to expose an empty container as well.

The screenshot shows a 'Custom Properties' form with a title bar containing a refresh icon, a plus icon, and a close icon. Below the title is the instruction 'Please provide the following information.' and a link to 'Restore metadata values from template.' The form has two tabs: 'Page Layout' and 'Landing Page Config', with the latter being active. Under the 'Landing Page Config' tab, there is a checked checkbox for 'Hide Header and Footer Navigation'. Below this is a section for 'Call to Action Form' with a 'Form to Use:' label and two radio button options: 'Standard Call to Action' (selected) and 'Custom Form'. A note below the radio buttons says 'To change the color, use the Style button in the container. See Demo Site Guide for more information.' There are three sections for optional elements, each with a label and three checkboxes: 'LEFT COLUMN: Show the following optional elements:' (Video, Testimonials, Empty Container), 'RIGHT COLUMN: Show the following optional elements:' (Video, Testimonials, Empty Container), and 'BOTTOM REGION: Show the following optional elements:' (Empty Container).

## 5.23.4. Built-in Form vs. Custom Form

The landing page template also contains a built-in simple form so that you don't have to worry about setting it up each time. This form is configured as follows:

- Contains three visible fields, First Name, Last Name and Email.
- Contains one hidden field, Page Title, which defaults to the name of the current page. Because it's a standard form and will likely appear on several different landing pages, we may not automatically know where the user was when they filled out the form. By including this hidden field with the page name as its default value, records gathered using this form can be sorted according to which page they came from.
- Records submitted using this form can be viewed in the preconfigured datasheet located in the Web Administration section of the site at /demo/webadmin/leads/index.cfm

If the built-in form does not meet your needs, you have the option of choosing a custom form which you can configure on the fly. Using the metadata form (Custom Properties), seen above, select “Custom Form” next to “Form to Use”. Doing so will replace the built-in form with an unconfigured Simple Form element menu. Of course, if you choose this option, the datasheet page won't yet exist in the Web Administration section. Feel free to configure one as needed.

## 5.23.5. Color options via style selector

Varying the color scheme in your landing pages can also impact conversion rates. This is why the example landing page template also allows for easy modification of the landing page color scheme using out of the box CommonSpot functionality.

Changes can be made at the page-level, ensuring that you can choose the colors on a page by page basis. With the page in author mode, you can see the element menus that comprise the call-to-action box in the upper right corner of the landing page. The call-to-action box is inside a div using the class “.ctaform” which is defined in the style sheet. Inside the div, you’ll find a container, which contains a heading (the call to action), and a form.

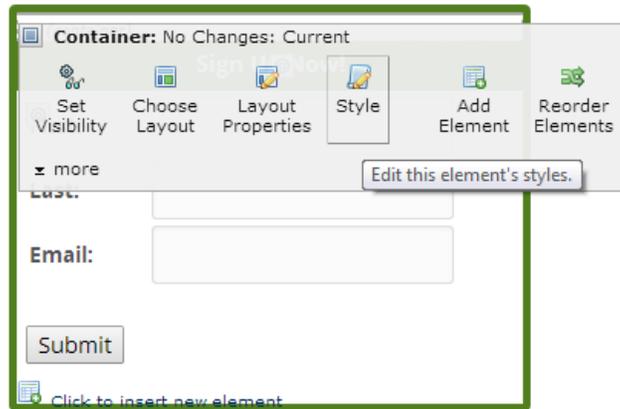
At the container level, we can choose from one of the many color-specific classes defined in the style sheet. Here is an example of one of the colors, as defined in the .css.

```
/* marketing landing page */
.ctaform {
    margin:2px 6px 10px 6px;
}

.ctabox-green {
    margin-top:20px;
    margin-bottom:20px;
    border: 5px solid #527F1E;
    -moz-border-radius: 4px;
    border-radius: 4px;
    padding: 0px;
}
```

To specify a color, simply click on the style link in the container element menu inside the call-to-action box on the landing page.

Note: This is a page-level choice, so make sure you are on the page and not the template. If you make this change on the template, all you are doing is defaulting the box to a certain color. Setting it at the page level ensures that you are making that page unique.



You will be presented with the Element Style dialog.

Element Style			
Component	Parent Component	Classes	Action
Container	None	ctabox-green	
Line Height for spacing between elements	Container	Use Default	

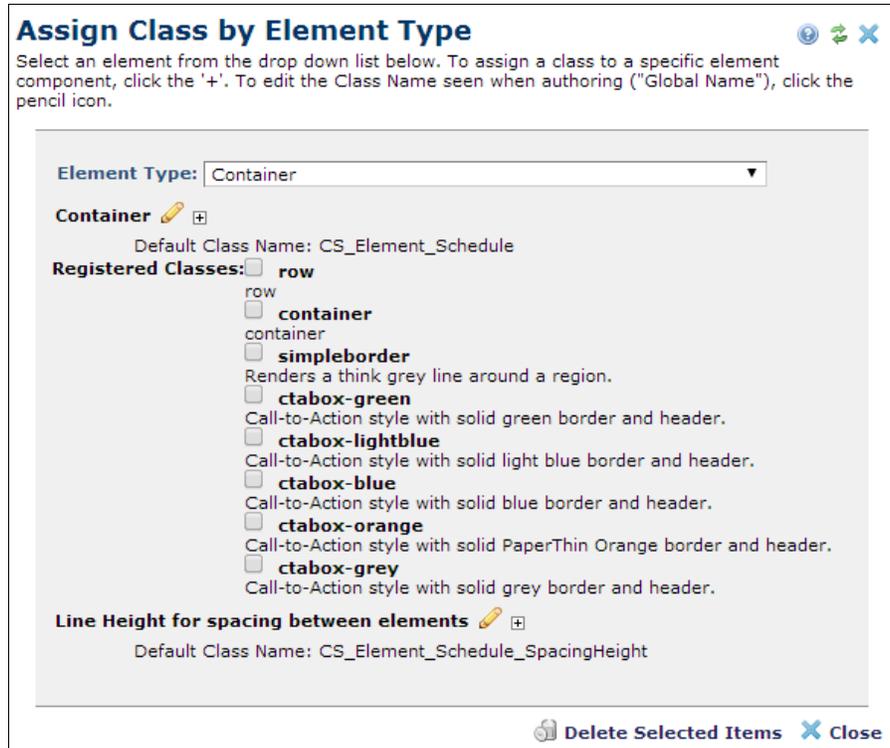
Showing 2 records. Close

Clicking on the pencil icon on the “Container” row will give you the following dialog.

Edit Element Component Style	
Class ^	Description
<input type="checkbox"/> container	container
<input type="checkbox"/> ctabox-blue	Call-to-Action style with solid blue border and header.
<input checked="" type="checkbox"/> ctabox-green	Call-to-Action style with solid green border and header.
<input type="checkbox"/> ctabox-grey	Call-to-Action style with solid grey border and header.
<input type="checkbox"/> ctabox-lightblue	Call-to-Action style with solid light blue border and header.
<input type="checkbox"/> ctabox-orange	Call-to-Action style with solid PaperThin Orange border and header.
<input type="checkbox"/> row	row
<input type="checkbox"/> simpleborder	Renders a think grey line around a region.

Notice how several color variations are listed there with the prefix “ctabox”. Selecting one will format the box according to that color. Each of these items has a corresponding entry in the .css file.

In addition to having them in the .css file, they also have to be registered in CommonSpot so that they show up in the dialog. Please refer to the documentation for how to register classes. The settings are in the Content Formatting section of the Site Admin. Here is what the dialog looks like when defining the Container element classes.



## 5.23.6. A/B Testing on Landing Pages

Here is how you can define a landing page in CommonSpot where you can test variations on your marketing messages. Technically, CommonSpot does not have specific functionality for launching a full scale multivariate campaign, but there are several building blocks that can be used to create A/B testing scenarios.

### 5.23.6.1. Our example

In our example, we have created two pages using the Landing Page template. We have created variations in each of the key landing page areas, including:

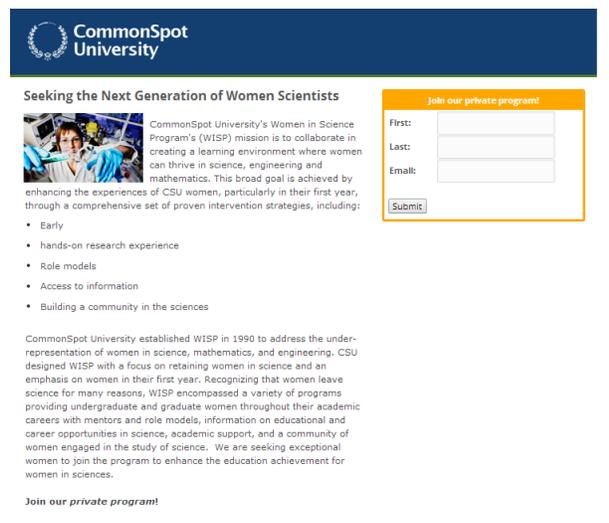
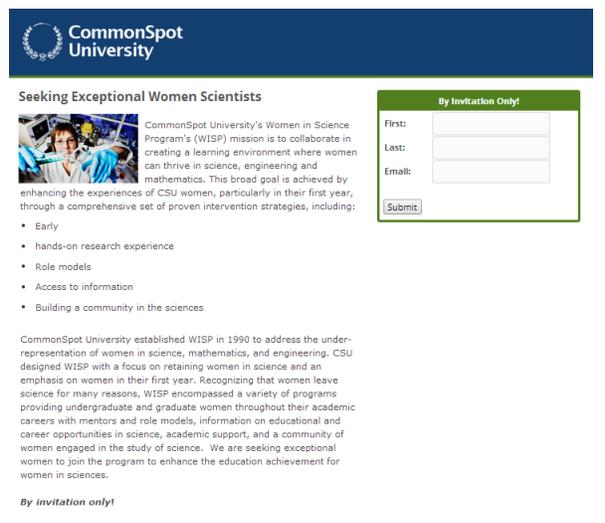
- Title
- Call to action
- Supporting info (itemized benefits)
- Color scheme

Additional variation options include:

- Custom form instead of the default form
- Additional items on the page such as a video, a photo carousel, testimonials, etc.

The variations are as follows:

- Variation A:
  - /demo/admissions/seeking-the-next-generation-of-women-scientists.cfm
- Variation B:
  - /demo/admissions/seeking-exceptional-women-scientists.cfm

5.23.7. Variation A	5.23.8. Variation B
 <p>The screenshot shows a form titled "Seeking the Next Generation of Women Scientists" for CommonSpot University. It features a blue header with the university logo. The main content includes a photo of students, a paragraph about the Women in Science Program's (WISP) mission, a bulleted list of benefits, and a paragraph about the university's history with WISP. A prominent orange button labeled "Join our private program" is positioned to the right of the text, with a "Submit" button below it. The form includes input fields for "First:", "Last:", and "Email:".</p>	 <p>The screenshot shows a form titled "Seeking Exceptional Women Scientists" for CommonSpot University. It features a blue header with the university logo. The main content includes a photo of students, a paragraph about the Women in Science Program's (WISP) mission, a bulleted list of benefits, and a paragraph about the university's history with WISP. A prominent green button labeled "By Invitation Only!" is positioned to the right of the text, with a "Submit" button below it. The form includes input fields for "First:", "Last:", and "Email:".</p>
<ul style="list-style-type: none"> <li>● Title: Seeking the Next Generation of Women Scientists</li> <li>● Body: contains slightly different text at the bottom</li> <li>● Call-to-Action: Join our private program!</li> <li>● Call-to-Action color: Orange</li> </ul>	<ul style="list-style-type: none"> <li>● Title: Seeking Exceptional Women Scientists</li> <li>● Body: contains slightly different text at the bottom</li> <li>● Call-to-Action: By invitation only!</li> <li>● Call-to-Action color: Green</li> </ul>

The plan is to publicize this campaign in several ways, including an email blast, posters, etc. In the distribution of this campaign, we can use various techniques to drive customers to one or the other page.

- Specifically mention one or the other URL in the email, poster or whichever medium is used. Alternatively, a vanity URL should be used to make it even easier for the respondent.
- Direct people to an interstitial page which can randomly (using custom code) redirect them to one of the two pages. If done properly, the visitor doesn't even know this has happened.

## 5.24. Layouts (Display Templates and Render Handlers)

CommonSpot makes it easy to select from multiple layout options for any type of content. Custom layouts can be defined as either “display templates” or “render handlers”.

A display template is a custom layout that can be defined by any user, even non-technical users, using CommonSpot’s rich text editor. Alternatively, a render handler is a snippet of code which allows a developer to programmatically define how a piece of content is laid out. Render handlers require programming skills, but can allow for more complex logic or embedding of additional user-interface components such as javascript or API interaction.

Since version 9.0, CommonSpot now displays both render handlers and display templates in one “unified layout” dialog box. Regardless of how the layout is defined, the user can now select from either a display template or render handler using the same unified dialog box. Refer to “Controlling the Layout of Residence Halls” for a great example of the difference between a display template and render handler and how to select them.

## 5.25. Lead Management

Organizations often want to provide forms on the site for visitors, and then track these “leads” in a centralized dashboard or export them for use in other systems. CommonSpot University has some examples of simple lead management.

### 5.25.1. Visitor Survey

The visitor survey contains two components.

- Visitor Survey form, located at `/demo/about/visitor-survey.cfm` This is a simple form with some various demographic and feedback questions.
- Visitor Survey Dashboard, located in the password protected Web Administration at `/demo/webadmin/survey/index.cfm` This is a datasheet containing multiple views of the incoming visitor survey submissions, including all, prospective students, current students, alumni and parents. It also contains a download link for exporting as a comma delimited spreadsheet.

### 5.25.2. Marketing Landing Page

The landing pages contain a region using either a stock simple form or a placeholder for a custom simple form. Using this template you can publish forms and track the incoming leads. Please refer to “Landing Pages” for more detail on how these are configured.

### 5.25.3. A/B Testing

Similarly, the Marketing Landing Page template provides a structure for simple A/B testing. Please refer to “Landing Pages” for more detail on how CommonSpot can be configured to manage A/B Testing.

## 5.26. Mobile Optimization

Features: Style Sheets, Templates

Sections: All pages

Users: All users

References:

CommonSpot University is built from the ground up to be viewable on devices as small as standard smartphones and as large as a standard desktop. All pages in the site adapt to the dimensions of your browser. This is accomplished using the principles of “responsive design”, which is a technique that combines a set of size-specific style parameters in CSS using “media queries”, providing the ability for a browser to dynamically use different sets of styles depending on environment parameters, such as the current size of the browser window. Sometimes javascript is used to better accomplish this effect, such as in the example of the various image carousels found on the home page and throughout the site.

Another technique historically used by mobile-compatible web sites is to redirect visitors to a mobile-optimized subsite if a mobile browser or device is detected. Such subsites would use a very specific mobile optimized template and style sheet, and it would reuse content from other areas of the site. At this time, the site does not have such a subsite. Responsive design is the most popular option at the time of this release, and sites built from the ground up in responsive design typically no longer require a mobile optimized subsite. The industry is trending toward responsive design, which has several advantages over a mobile optimized site, including development effort and SEO considerations.

## 5.27. Multi-Language Content

CommonSpot provides good support for mapping content to multiple languages.

### 5.27.1. Multi-Language Pages

The News & Events section of the demo site utilizes CommonSpot Multi-Language support, effectively mapping and linking English news pages to their corresponding pages in Spanish. English News Articles can be found on the News Listings page </demo/news/news.cfm> while the Spanish versions of some of those articles can be found on the CSU Ahora page, located at </demo/news/spanish/index.cfm>.

To get a feel for it, go find a Spanish news article. Note the link in the right column labeled “View Article in English”. This link takes you to the English version of this news article. The English page has a link labeled “Articulo en Espanol”, which leads back to the Spanish version.

Note: While the link to the corresponding Spanish page will always be presented, this does not ensure a page in Spanish exists. Should a page in Spanish not exist, CommonSpot’s link management utility will simply redirect back to the English page. This solution could be customized using code to only render a link when a Spanish page exists.

Each page was built using a language specific template, so that the surrounding content is appropriate. However, they both use the same News Element for the actual content.

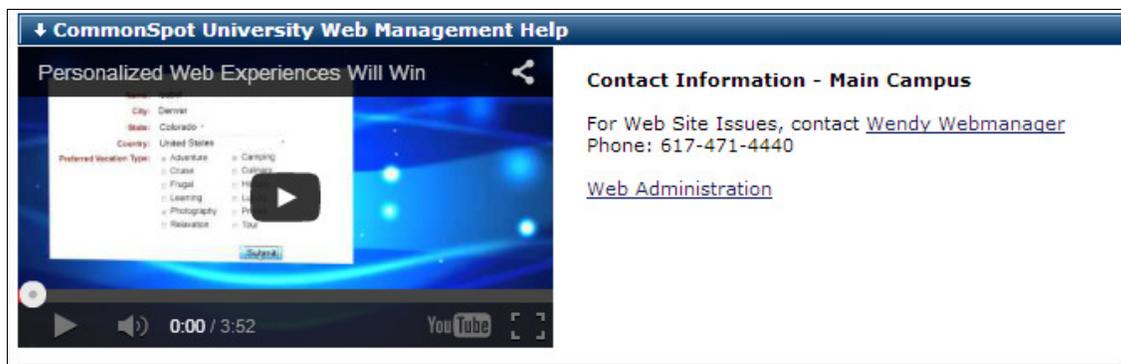
## 5.27.2. Foreign Language Characters

CommonSpot uses UTF8, supporting foreign language characters. You can see an example of content with Chinese characters by visiting the blog post “Experiencing Study Abroad in China” located at /demo/blog/studyingabroad/2013/Experiencing-Study-Abroad-in-China.cfm

## 5.28. My CommonSpot Customization

There is a customization in the My CommonSpot available to all users. This is an example of how you can create custom modules and present them to your users. Using this capability, you can extend the CommonSpot Interface and provide additional tools or resources. Please refer to the documentation for more information.

The sample customization displays some basic info about the internal resources available to contributors at CommonSpot University. The default section looks like this.



Even this is customizable. In our scenario, the user may be located at one of the several CommonSpot University campuses. To change the view, click customize in the upper right corner of the My CommonSpot notification region.



Here you can control parameters for all of the visible regions in your My CommonSpot view. Some sections may have additional parameters, like the section in our example. Simply change the field labeled Campus, save it, and notice how the section changes.

**Customize My CommonSpot**

Click and drag to reorder My CommonSpot sections. Click the icon in the first column to show or hide sections. Enter the number of items initially displayed per section, and set whether each section is expanded by default. If enabled, optionally enter custom settings.

↑	Title	Items	Expand	Customize
✓	My Changes	5	✓	
✓	Notifications	5	✓	
✓	Changes Pending My Groups' Approval	5	✓	
✓	CommonSpot University Web Management Help		✓	Campus: Main
✓	Changes to My Content	5	✓	East
✓	Changes to My Groups' Content	5	✓	West
✓	Changes Pending My Approval	5	✓	

## 5.29. My Tags

My Tags allows users to tag content for their own personal management and organization. Personal tags are only visible to the user who created them. For more information, please refer to the documentation.

Here is a partial list of tags created by the webmanager user.

- AB Test
- academics
- admissions
- campus life
- important
- video

These tags are used in various locations through the site. To see the effect, log in as webmanager and do a simple find for one of the terms. Notice the “Search Using” checkboxes in the Find dialog. Uncheck all but “My Tags” and use one of the above tags as the keyword to search.

**Find**

Please enter the criteria by which you want to search, and any other options as desired.

[Quick Find](#) | [Advanced Search](#) | [Saved Searches](#)

**Search Criteria:** Contains | important

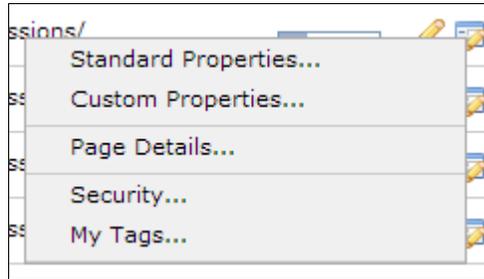
**Subsite:** /demo/

Include Child Subsites

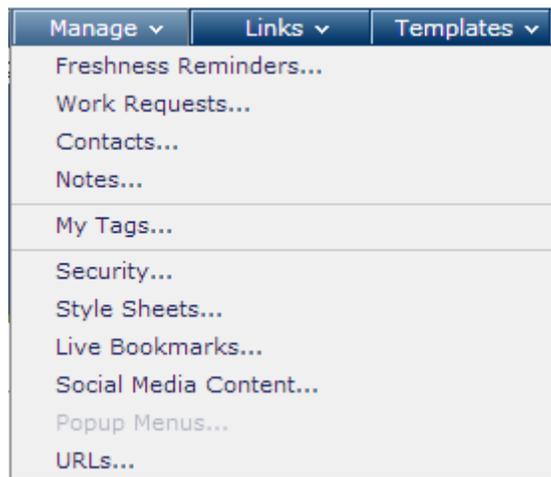
**Search Using:**  Standard Metadata  Keywords  Full Text  My Tags

[Show More](#)

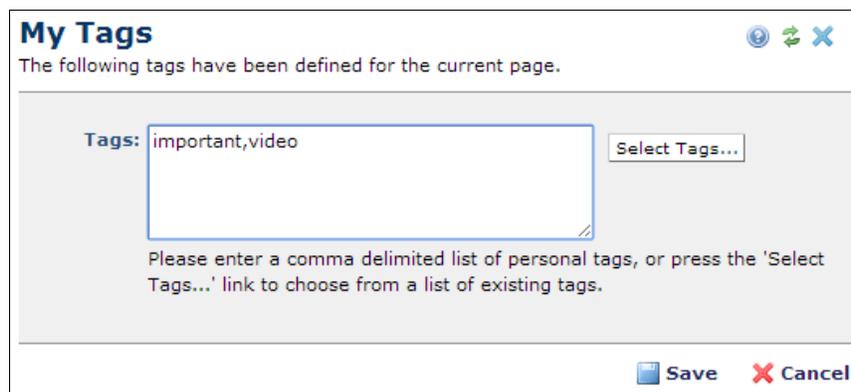
The above search will yield several pages. You can access the tags in one of two ways. Either use the Properties button on the search results and choose My Tags.



Or go to the page under the Manage menu.



The dialog looks like this.



There is also a Saved Search for "Important Pages" which automatically searches for any page tagged Important.

Imagine a scenario where this user is responsible for all the landing pages as well as anything related to them. This person is the only person who needs to work with this particular collection of page and

nobody else would draw the same relationships. Therefore, there may not be any organic way for this user to locate the pages they need to work with. But if they tag their pages using the personal tags, then they can easily retrieve their pages at any time without cluttering other users with irrelevant metadata.

## 5.30. Navigation Using Tree Element

URLs:

- Any page (main navigation)
- Several pages have left navigation
- Subsite home pages have list of links

Webadmin URL: none

Features used: Tree navigation builder

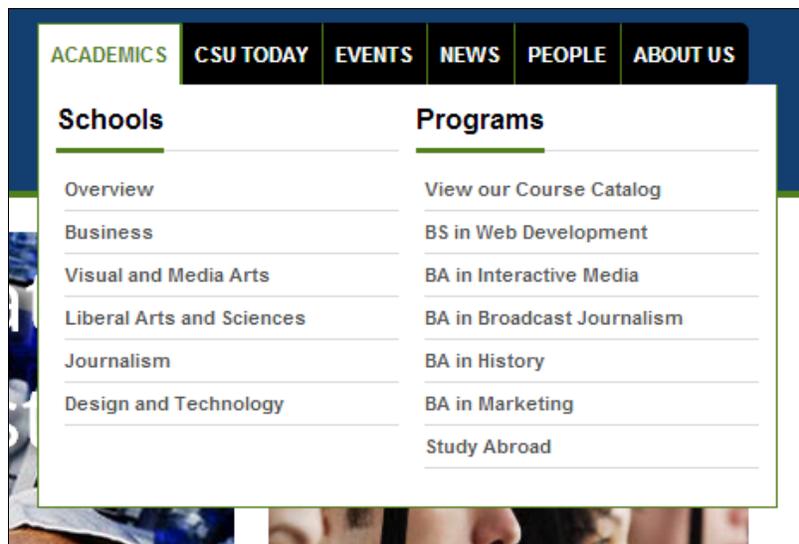
Documentation: Administrator's Reference, Elements Reference Guide

The main navigation on the site utilizes CommonSpot's Tree Navigation Builder. The Tree Builder is a tool which lets you create structured hierarchical collections that can be displayed in a variety of ways throughout your site. The site uses these "trees" to build several variations of navigation for the visitors to use, including drop down menus ("mega menu"), accordion menus and basic lists.

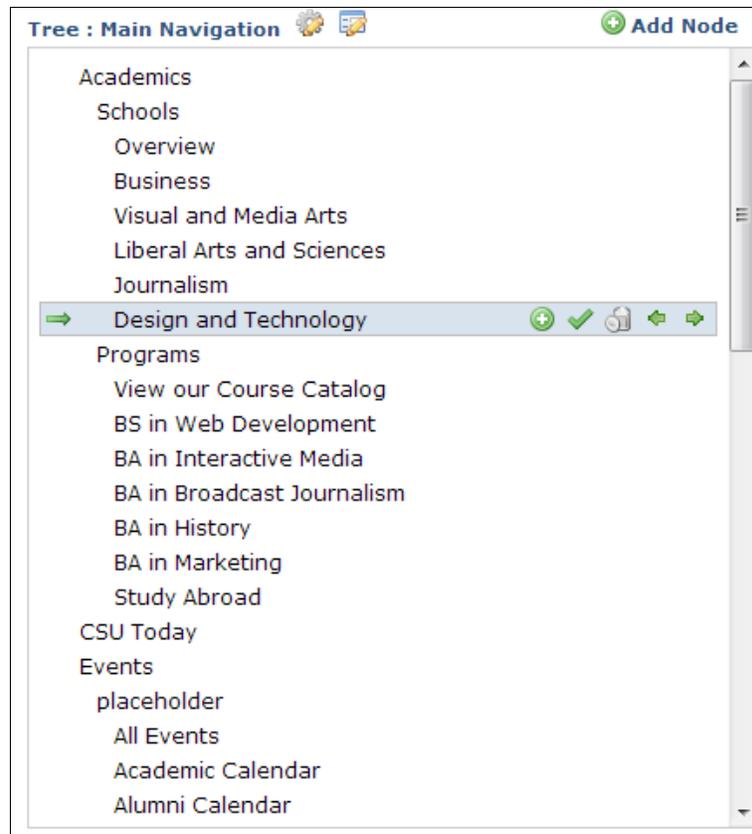
### 5.30.1. Main Navigation Mega-Menu

The main navigation throughout the site is a mega-menu that presents an easy-to-use site-wide navigation with an expanding and collapsing dropdown region and several layout options.

The menu looks like this.



The underlying structure of the menu comes from the CommonSpot Tree Navigation Builder. A portion of the tree behind the above navigation looks like this:



Notice that the tree structure is a simple hierarchical view of the site's navigation. Each entry in the tree is considered a "node", which carries some basic parameters such as title, description and link target information. The tree itself can be modified independent of the layout of the page. Like other CommonSpot elements, users have many options when it comes to the layout of content.

In the site, the main navigation tree is being displayed using a Layout which distributes the nodes across an easy to use mega-menu.

One of the nodes in the main navigation calls upon a custom script element. See Custom Script section for more information.

## 5.30.2. Left Navigation Accordion Menu

Additional trees are used throughout the site to define the navigation that optionally appears in the left column of many pages. The layout used here is capable of sub-nodes which can expand and collapse accordion-style.

An example of the left navigation appears here:



### 5.30.2.1. Use of Tree Families

Like the main navigation in the header, the left navigation is used on a majority of the pages throughout the site. But unlike the main navigation, the left navigation needs to be contextual, and change its contents based on where you are in the site. To accomplish this, the site uses “tree families”, which enable administrators to associate collections – or families – of trees to a particular region of a template, instead of just one. Once associated within the template, administrators can specify which trees appear in which subsites.

This site makes use of tree families. Most of the subsites have their own tree where administrators can independently manage its navigation. Each of these subsite navigations are grouped under the “Left Navigation” tree family, and each is mapped to its appropriate subsite. In the template, the left-hand region has been defined to display the “Left Navigation” family, and CommonSpot chooses the appropriate tree based on the subsite mappings.

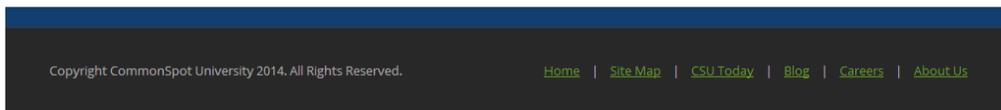
For more information on trees and families, please consult the *Administrator’s Reference*.

### 5.30.3. Subsite Home Page Link List

The trees used to define the Left Navigation are also used in the subsite home page template. Here, instead of displaying them as a mega-menu or accordion, the nodes are displayed as basic links with descriptions. This is an example of the versatility of the tree navigation builder.

### 5.31. Navigation Using Linkbar:

The footer of the site contains basic links to the home page, a site map and various other basic pages on the site.



This is accomplished using the Link Bar element.



## 5.32. News, Press Releases and Third Party News

News items in CommonSpot University are separated into three distinct areas: News, Press Releases and In The News. These different types of news items are built in three very different, but uniquely useful ways.

### 5.32.1. News (Page with Local Custom Element)

News articles can be found in the News Subsite and the Spanish News Subsite. English and Spanish news articles are related to each other using CommonSpot's multi-language functionality. News articles are created using the News template, which uses a Custom Element named "News Article". To add, choose New -> Page and then select the News Article template. The News element uses a predefined set of categories for tagging the news articles. Use the News Category Manager to manage these entries.

### 5.32.2. Press Releases (Page with Formatted Text Block)

Press Releases can be found in the News Subsite and are built using the Press Release template. There is no custom element involved. Instead, this template makes use of out of the box "Formatted Text Block" element combined with a Custom Metadata (Custom Properties) form with the page. To add, choose New -> Page and select the Press Release Template.

### 5.32.3. In the News (Registered URL's)

Items in the In the News section represent mentions of CommonSpot University from third party web sites or literature. The items are recorded here as external URL's or uploaded documents. Simply adding an external URL or uploading a document into the /news/inthenews subsite will get the item to appear in this list. To add, choose New -> Uploaded Document, then select the /news/inthenews subsite.

## 5.33. RSS Reader

CommonSpot University has a sample piece of code that helps to consume an RSS feed and display the links on a page. You will find it on the page located at /demo/news/college-sports-news.cfm

## 5.34. RSS

CommonSpot natively supports presentation of structured data as an RSS feed. This applies to both lists of pages as well as lists of user-defined custom element data. The following RSS feeds are configured in CommonSpot University. Please refer to the documentation for more information about how to enable RSS feeds from the elements listed here.

## 5.34.1. News

- Uses the News Article element, located on /demo/news/news.cfm
- The feed is located at /demo/news/news.cfm?xml=CommonSpot%20University%20News,RSS2.0

## 5.34.2. Press Release

- Uses the Page Index element, located on /demo/news/press-releases.cfm
- The feed is located at /demo/news/press-releases.cfm?xml=CommonSpot%20University%20Press%20Releases,RSS2.0

## 5.34.3. In the News

- Uses the Page Index element, located on /demo/news/inthenews/index.cfm
- The feed is located at /demo/news/inthenews/index.cfm?xml=CommonSpot%20University%20Mentioned%20In%20External%20News%20Sources,RSS2.0

## 5.35. Scheduled Content

CommonSpot is capable of dynamically scheduling content based on a number of factors, including time of day, day of week, location on site, membership in a group, etc. Please refer to the *Contributors* and *Elements Reference* for more information on scheduling. There are two elements that can be used, the Schedule/Personalize Element and the Container. CommonSpot University has some good scheduling examples.

### 5.35.1. Schedule/Personalize Element

On the Alumni subsite home page, you will find the Schedule/Personalize element. This element is designed to show one, and only one, element based on the scheduling criteria. For further detail, please visit the section titled “Content Personalization: Explicit (Known-Visitor)”

### 5.35.2. Container with Scheduling Enabled

In various areas of the site, you will find a Container element that is configured for scheduling. The main difference between this element and the Schedule/Personalize element is that a Container element is capable of showing more than one element that matches the scheduling criteria. For this reason, we use it for the “implicit” personalization within CommonSpot University. For more information, please visit the section titled “Content Personalization: Implicit (Contextual).”

## 5.36. Search

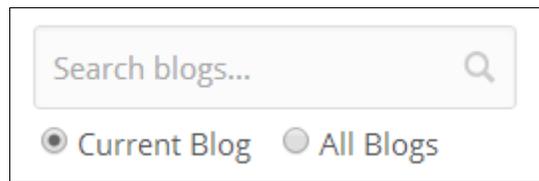
CommonSpot University leverages the built-in search capability supported by CommonSpot. There are two types of searches enabled in the site. Site-wide search and specific subsite search.

## 5.36.1. Site-Wide Search

The site-wide search can be invoked by typing a search term into the Search field in the header on every page. Search results appear on /demo/search.cfm

## 5.36.2. Specific Subsite Searches

Two subsites also have their own search capability with search results limited to each respective subsite. These can be found in the Careers section ( /demo/about/careers/ ) as well as within any Blog ( /demo/blog/ ). In the case of the blog search, the visitor has a radio button where they can select between “Current Blog” or “All Blogs”.



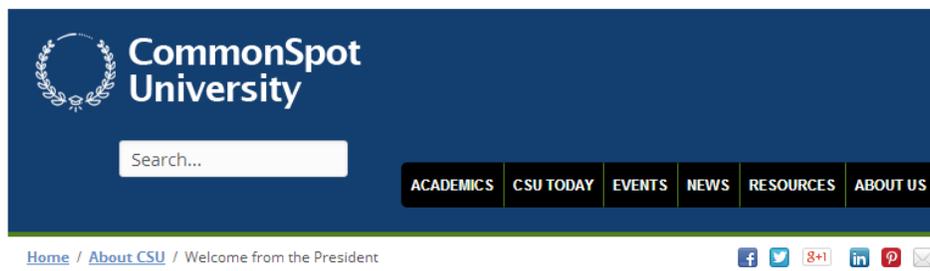
## 5.37. Social Media

CommonSpot University covers social media options in several ways, including social links, Social Media Manager functionality, and the Social Media Stream app. The three options are described here:

### 5.37.1. Social Links

As a simple example of how to provide social functionality on your website, we have placed a small component on the Base Plus One template that displays links to social media tools in the breadcrumb bar on every page in the site. It uses a third-party service, ShareThis (sharethis.com). We have configured a simple example that contains links to Facebook, Twitter, Google Plus, LinkedIn, Pinterest and an email share.

These buttons will disappear (see Responsive Design) due to spacing issues on smaller devices or browser widths.



### 5.37.2. Social Media Manager

CommonSpot contains social media management capabilities allowing contributors with proper permissions to manage social media posts from within the CommonSpot user interface. This means they

can post from within CommonSpot, post content from CommonSpot (such as pages, element data, images) manage and schedule the timing of social media posts, as well as review data from past posts. All this can be done without leaving the CommonSpot interface.

Because the configuration of these tools requires access to credentials from various social media services, CommonSpot University does not come pre-configured with these tools enabled. It is highly recommended that you get acquainted with this functionality by configuring the social media manager. Please refer to the *Administrator's Reference*.

### 5.37.3. PT Social Media App (Social Media Stream)

The Social Media stream capability of the PT Social Media App enables you to easily fire up a web page containing a stream of timely and relevant posts from various social media services. More information on this app can be found in the Community site at

[http://community.paperthin.com/projects/pt\\_social\\_media](http://community.paperthin.com/projects/pt_social_media)

Because the configuration of this application requires access credentials from various social media services, CommonSpot University does not come pre-configured with these tools enabled. It is highly recommended that you get acquainted with this functionality by configuring the social media app.

An facsimile of social media stream output can be viewed at the bottom of each page in CommonSpot University.



To get started using this capability, visit the app's dashboard in the Web Administration, located at `/demo/webadmin/socialmedia/index.cfm`. All the tools are installed awaiting your own custom configuration. This includes a subsite named Social Media, located at `/demo/socialmedia/`

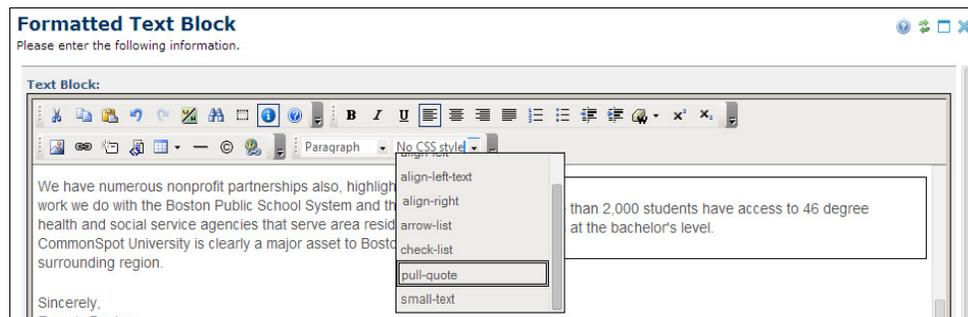
Note: The app is capable of supporting a moderation option, which allows you to blacklist a post so it no longer appears in the stream. In our example, we have turned off caching on the `/socialmedia/` page to best demonstrate how the app works. However, in a production environment you will typically leave caching on and then create an identical page, with caching turned off, in the Web Administration area. This enables you to get the best performance on your visitor-facing pages while providing the moderation capability.

## 5.38. Styles and CSS

CommonSpot University uses various style sheets in conjunction with the main theme used in the templates. Most content in CommonSpot will adhere to the default styling as defined by these style sheets. However, there are ways that users can explicitly call upon specific style classes in order to get the desired effect. Administrators can register style classes and configure specific elements to be able use them in their own context. Two examples are below:

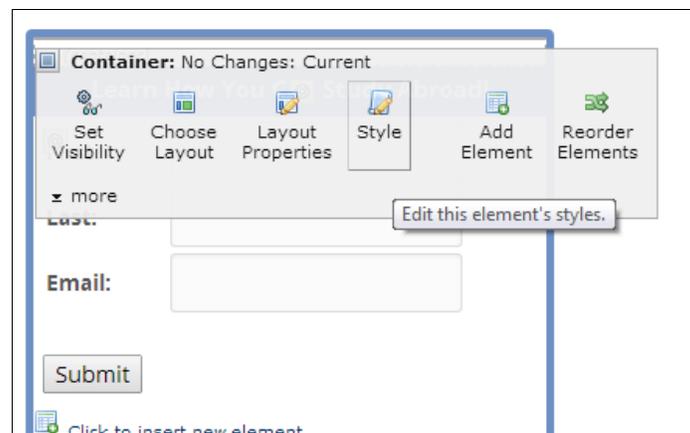
### 5.38.1. Choosing a Registered Style from within the Rich Text Editor

Specific style classes can be selected from a drop-down in the rich text editor, as seen in this screenshot taken from the Welcome from the President page (/demo/about/our-president.cfm). For more information, refer to the section titled “WYSIWYG Editing” in this document.



### 5.38.2. Using the Element Menu to change styles

Within the element menu, users also have the option to override default styles for that specific element. Here is an example of specific styles being selected in a Container element that drives the call-to-action box on any marketing landing page. This one is located at /demo/admissions/study-abroad-program.cfm



The resulting dialog provides choices that allow you to change the background color of the call-to-action box:

Class ^	Description
<input type="checkbox"/> container	container
<input type="checkbox"/> ctabox-blue	Call-to-Action style with solid blue border and header.
<input type="checkbox"/> ctabox-green	Call-to-Action style with solid green border and header.
<input type="checkbox"/> ctabox-grey	Call-to-Action style with solid grey border and header.
<input checked="" type="checkbox"/> ctabox-lightblue	Call-to-Action style with solid light blue border and header.
<input type="checkbox"/> ctabox-orange	Call-to-Action style with solid PaperThin Orange border and header.
<input type="checkbox"/> row	row
<input type="checkbox"/> simpleborder	Renders a think grey line around a region.

## 5.39. Taxonomy and Facet Navigation (Document Library)

CommonSpot University comes with a very basic built-in taxonomy. This demonstrates the use of a taxonomy to categorize content and the use of facet-based navigation to search and discover content. The example used in CommonSpot University can be found in the Document Library located at </demo/intranet/doclibrary/>

The major components of this example are as follows:

### 5.39.1. The DocLibrary metadata form

This contains one field labeled "Classification". This field contains the Taxonomy Browser field type, allowing users to tag uploaded documents against the CommonSpot taxonomy. This metadata form is bound to the Documents subsite (</documents/> subsite/) so that it only appears when uploading content into this specific area.

### 5.39.2. The "documents" subsite (</doclibrary/documents/>)

This subsite was created to be the repository for all documents in the doc library. By using a specific, isolated subsite for the purpose of containing the uploaded documents, the administrator could easily bind the appropriate metadata form to only those items in the document library.

### 5.39.3. Page Index on Document Library home page

To display a dynamic list of links to documents in the Document Library, we employed a Page Index element. This element shows all items in the documents subsite, and is configured to respond to the selections the visitor makes in the Facet Based Navigation element on the same page.

### 5.39.4. Facet-Based Navigation on Document Library home page

In order to browse items in the document library, the visitor can make selections on the left side of the page. This section is powered by the Facet Based Navigation element. See Element Guide for more information.

Narrow or Expand Results	Showing Results
Browse Academics:	Browse topics at left to drill down and discover relevant documents.
<a href="#">Business</a> (5)	 <a href="#">2013 Employee Holiday Calendar</a> (23k) 2013 Employee Holiday Calendar
<a href="#">Design and Technology</a> (1)	
<a href="#">Journalism</a> (2)	 <a href="#">2014 Employee Holiday Calendar</a> (0k) 2014 Holidays for Faculty and Staff
<a href="#">Liberal Arts and Sciences</a> (3)	
<a href="#">Visual and Media Arts</a> (3)	 <a href="#">Admissions Application</a> (0k) Apply to CommonSpot University via hard copy.
Browse Audience:	
<a href="#">Faculty</a> (5)	 <a href="#">Backpack Journalism Brochure</a> (0k) CommonSpot University features a program in the School of Journalism that has become an emerging trend for most media
<a href="#">Staff</a> (6)	
<a href="#">Students</a> (7)	

## 5.40. Template Overview

CommonSpot University takes full advantage of CommonSpot's highly configurable template architecture. Refer to the *Template Developer's Guide* for more background on how templates work in CommonSpot. The templates in CommonSpot University are architected according to best practices set forth by PaperThin Services.

The specific templates in CommonSpot University are described here:

### 5.40.1. Base Template

All standard web pages derive, either directly or indirectly from this base template. The layout and design utilizes a theme with a responsive grid and contains basic page regions intended to support main areas of a page, including header, footer, left and right columns and main body area. The base template is defined at the file system level and is stored in a file called template-basepage.cfm. This file can be updated as needed and can contain ColdFusion code. This template cannot be explicitly selected when creating pages. The only template that derives from this template directly is the Base Plus One template.

## 5.40.2. Base Plus One Template

The Base Plus One template derives its base attributes from the Base Template. This is where all the site's navigation is defined, along with other site-wide content such as header (logo, menu), footer (link bars, boilerplates), and navigational areas (left column navigation).

## 5.40.3. Soft Templates

Anything other than the Base Template is considered a soft template because it can be edited entirely within the CommonSpot user interface and it does not require manual editing of a file on the file system. All other templates either inherit their base attributes from the Base Plus One template or from other soft templates. Each soft template adds a specific layer of detail intended to create a unique look or layout required for that type of content.

Here is a list of useful soft templates available in CommonSpot University.

Title	Subsite	Description
Base Plus One Template	/demo/	See above.
News Article Template	/demo/news/	Supports creation of news articles along with the dynamic lists for discovery of related or recent news.
Press Release Template	/demo/news/	Similar to news, but optimized for a standard press release with appropriate dateline and boilerplate.
Marketing Landing Page Template	/demo/admissions/	A flexible, user-configurable template optimized for creating successful marketing landing pages.
Residence Hall Template	/demo/campuslife/	A type of content powered by a local custom element which contains unique pages and descriptions for each residence hall.
Subsite Home Page	/demo/	Designed for the default look and feel of a subsite within CommonSpot University.
WYSIWYG Template	/demo/	A simple layout, derived off

		the base plus one, with a built-in Formatted Text Block element for rich text content.
Web Administration Template	/demo/webadmin/	Optimized with the Web Administration navigation on the left side for use with various custom apps and datasheets in CommonSpot University.

## 5.40.4. API Only Soft Templates

Some templates are only intended to be used by automated page creation steps in various CommonSpot apps. While the soft template exists and may be visible, it is not intended to be directly chosen by contributors. As such, these templates should be hidden from view from standard contributors. To use any of them, users can go to the specific app pages, typically located in the Web Administration (/webadmin/) section of the site. When visible in the template gallery (by admins), these templates should have a thumbnail that says "API's Only"  API's Only, as an indicator that they are not intended to be selected by users.

- Blog templates
- Calendar templates
- Course Catalog templates
- Profile Templates
- Social Media app templates

## 5.40.5. Blank Template

The Blank Template is another hard template that has no styling or layout components to it. While this might be useful in some content situations, there aren't any real pages in CommonSpot University that use this template. However, it is useful when constructing simple dialogs for custom applications. As such, this template is used in a few cases for this purpose.

## 5.41. Users and Groups

**5.41.1.** Refer to the Administrator Guide for all information regarding users and groups. Refer to first section of this document (Demo Site User Map) for information about users and groups in CommonSpot University.

## 5.42. URL Management

CommonSpot provides the ability to manage URL's in several ways, described here.

### 5.42.1. Human Readable URLs

CommonSpot allows you to create pages using any filename you wish. This means that the value you provide for "name" when creating the page will automatically be used to create the file name for that page. Because of this, the actual URLs for any page can be understandable and contain the keywords necessary to be useful to both users and search engines.

### 5.42.2. Redirect URLs

In addition to the actual file name and location, CommonSpot allows you to create custom URL redirects, which are alternate paths to a particular page. You may create as many

- **Vanity URLs:** A vanity URL is an alternate path to a page which may be easier to remember or use. A 301 redirect is created and all requests for that URL are redirected to the actual location of the page. CommonSpot University contains several vanity URL's.
- **Legacy URLs:** Similar to vanity URLs, these 301 redirects allow administrators to intercept traffic bound for paths that are no longer in use. These paths can be from legacy sites or systems, retired CMS's, or in the event that a page is moved to a new location or renamed. CommonSpot will automatically create a legacy URL record when a page is moved or renamed.

Some example Vanity URLs in CommonSpot University.

<b>Redirect URL</b>	<b>Directs Browser To this Location</b>
/journalism	/demo/academics/journalism/index.cfm
/liberalartsandsciences	/demo/acadmics/liberalarts/index.cfm
/marketing	/demo/academics/business/marketing.cfm
/studyabroad	/demo/admissions/study-abroad-program.cfm
/visit	/demo/admissions/visit-commspot-university.cfm
/yourfeedback	/demo/about/visitor-survey.cfm

## 5.43. Video

CommonSpot supports the use of video in two ways. The first is a full-featured integration with third-party video services such as Brightcove and YouTube. The second is to simply manage a library of references to third-party services. These methods are explained here.

## 5.43.1. Native Video Integration

CommonSpot's native video capability supports locally hosted video as well as a full integration with third-party video services. This allows video assets to be managed just like and alongside other CommonSpot content, using all the same capabilities such as security, scheduling and metadata. Obviously, the Demo site does not ship with a default connection to any proprietary third party account. But it's a breeze to set up a connection to either YouTube or Brightcove.

And due to space constraints, we also do not include a library of locally hosted video assets. There is one video uploaded to the site, located in `/demo/admissions`. It is titled "Personalized Experiences Will Win". The video discusses the personalization capabilities in CommonSpot and features outstanding voice talent. This video has been applied to a page in the admissions subsite located at `/demo/admissions/personalized-web-experiences-will-win.cfm`

Either way, locally hosted video such as this one, or videos hosted with YouTube and Brightcove, can be completely managed the same way you manage other CommonSpot content such as pages and documents. Refer to the *Administrator Reference* for more information about configuring an integration.

## 5.43.2. Visitor Contributed Video Example

The options explained above allows you to manage video content just like regular web content in a centrally managed library. That is a very important characteristic so that you can control video assets and manage who can manage and use those videos. But in the event you require a more community-based approach to video content, there are other options. One such option allows you to provide visitors the ability to nominate and submit references to video content into a library of video titles that can be viewed by other members of the community.

In our example, we have created a public video library titled "CSU Tube", located at `/demo/campuslife/csutube/` This is a page where site visitors, not necessarily administrators, can publish videos for others to see. It supports videos from YouTube and Vimeo. When anonymous visitors submit a video, it is hidden from the public view of the library until a web manager approves the video here.

Here you will find a display of videos directly from the video library. None of the videos live in CommonSpot. Instead, it is simply a database of references to YouTube and Vimeo videos, submitted by the Community of visitors to CommonSpot University. A simple form is used to collect nominated videos from visitors. Each newly submitted video goes into a holding area, and doesn't show up on the site until some internal resource approves it.

This app uses the following components:

- The Visitor Video custom element is a global custom element where the video references live. The records in this element make up the library of videos and several render handlers, described below, have been created so that these items can be displayed in any number of ways.
- The CSUTube home page is broken up into three regions, using three specific render handlers that do specific things such as display the video name, description, video in a player and thumbnails

from the respective third party service.

- A featured video at the top rendered inside a live player, alongside a description of the video and other basic info about the page.
- A list of “Featured Videos”, which are basically videos from the library tagged as “Featured” by some internal resource. These videos are displayed with their title, description and a thumbnail.
- A list of the most recently added videos. These videos are displayed with their title and a small thumbnail.
- The Submit a Video page, where we have placed a simple form letting any visitor, including Anonymous visitors, submit information about a video they are nominating. It contains two fields that are only visible if you are logged in as an administrator.
- The Web Admin page, located at /demo/webadmin/csutube/ , is where the videos can be managed. Select from several datasheet views, including
  - All Videos
  - Approved Videos
  - Approved and Featured Videos
  - Unapproved Videos

## 5.44. Web Administration

Web Administration is a secure subsite within CommonSpot University where web managers and other appointed faculty and staff can manage structured web content. Many of the items managed here are used in several locations across the site. Web Administration is located at /demo/webadmin/

While most CommonSpot content can be managed directly on its page, some content might be better managed in a dashboard interface, such as can be found in the Web Admin subsite.

Many of the content types you find here are bundled as CommonSpot Apps, and are made available in our Community Site, at [community.paperthin.com](http://community.paperthin.com).

There are several ways to get to the Web Admin home page.

- It is located at /demo/webadmin/
- There is a link in the site’s main navigation under “Resources -> Faculty and Staff
- In My CommonSpot, there is a custom panel that displays by default to all CommonSpot University contributors. It is called CommonSpot University Web Management Help. On this panel, you will find a helpful video, as well as contact info for help resources and the link to Web Administration.

Access to Web Administration is limited to internal resources. Anonymous access is disabled, and only certain contributor groups can access this subsite. Those groups are Staff and Webmanagers.

The tools available in Web Admin are listed here, but described in more detail in other sections of this document.

- Alerts: A system where users can create and manage emergency alerts which appear prominently on the homepage of the site.
- Blog Dashboard
- Calendar Events Manager
- Course Catalog Manager
- Forms
  - Surveys
  - Lead Management
- Job Manager
- News and Press Releases
- Photo Manager
- Profile Manager
- Social Media Stream Manager

## 5.45. Workflow

CommonSpot can support your workflow business rules. The Demo site demonstrates a Single Level Approval, although you are not limited to the number of levels. To see workflow and approval in action, log in as marketer and create or update a page. When you submit the page, the Submit Page dialog will notify you that the Web Communications Approvers have Final Content Approval. Add comments and submit. Log out, then log in as webmanager, and go to the My CommonSpot page. You will be able to quickly browse to the page under the 'Changes Pending My Approval' list by clicking on the page title you just submitted. Switch to Approve Mode to see the change and use the Element Toolbar to Approve Page or try out other Approval functions available in the toolbar. Once approved, the change will be live.

Note: You can add additional levels of Approval from Site Admin Approval Levels. See these or any other levels of approval in action by assigning them at other areas of the Content Security structure. Approvals can be assigned at the Subsite, Page, Template, or Content Element level. For more information on workflow and how to set up approvals refer to the Administrator's Reference.

## 5.46. WYSIWYG Editing

Features: Formatted Text Block Element, Formatted Text Block Field

URL's:

- various

A number of pages on the site implement WYSIWYG editing using CommonSpot's Rich Text Editor (RTE). For more information on the tools and features available in the Rich Text Editor and how to use them, refer to the *Content Contributor's Reference*. We recommend reviewing the more advanced features such as the Tag Selector, Inspector Panel, and Clean HTML Content tool. The Rich Text Editor includes two

dropdowns for applying your styles to your content. The first one has basic HTML tags, rendered according to the definitions in your CSS style sheets. Most style sheets define the appearance of these tags, such as Heading 1 (H1) or Paragraph (P). The second dropdown allows site administrators to present their own style classes for use on the site. Once they are defined in a style sheet, the administrator can register them in CommonSpot and determine which ones should be available to users. The Demo Site has a few of these custom styles defined and visible in the Rich Text Editor. All styles are defined in `/style/main-rte.css`:

- `align-left`: Floats content, such as an image, to the left allowing other content to flow around it. Try applying this to an image.
- `align-right`: Floats content, such as an image, to the right allowing other content to flow around it. Try applying this an image.
- `align-left-text`: Same as `align-left`, but better for text.
- `small-text`: Renders a small variation of a simple sans serif font.
- `pull-quote`: Highlights a block of content by placing a border around it and floating it to the far right of the containing region. This has the effect of calling out the content for emphasis. See below: