

CommonSpot™
Winter 2012
Getting Started Guide

This CommonSpot Content Server Getting Started Guide, as well as the software described in it, is provided under the CommonSpot Evaluation 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 Verity, Inc.

This document was last updated February 16, 2012.

Chapter 1 Introduction.....	4
Chapter 2 Understanding the CommonSpot Site Hierarchy	5
Chapter 3 Creating Your Site.....	6
Chapter 4 Understanding Templates	10
4.1. Template Overview	10
4.2. Base Templates.....	11
4.3. Navigation Templates	11
4.4. Content Templates.....	11
Chapter 5 Developing the Base Template	12
5.1. Base Template Introduction.....	12
5.2. Importing HTML.....	13
5.3. Replacing Content with CommonSpot Elements.....	17
5.4. Adding Content to the “base +1” Template	21
5.5. Advanced Base Template Development	26
Chapter 6 Developing Content Templates	28
6.1. Creating a Content Template	28
6.2. Changing Ghost text.....	29
6.3. Setting Element Inheritance Restrictions.....	29
Chapter 7 Next Steps	30
Chapter 8 PaperThin Resources.....	32
8.1. CommonSpot ADF and the commons	32
8.2. Contacting PaperThin.....	32
8.3. Contacting Technical Support	33
8.4. Contacting Sales	33

Chapter 1 Introduction

Thank you for your interest in CommonSpot Content Server. This guide is for first-time developers and administrators looking for step-by-step instructions for quickly getting a site up and running. This document assumes a basic understanding of ColdFusion development, web application development, and CommonSpot and its interface. This guide covers the fundamentals of building a new site using CommonSpot and provides instructions for migrating your web site's design into CommonSpot. This guide does not replace CommonSpot fast-track training or product documentation. See the *CommonSpot Contributor's Reference* and the *CommonSpot Administrator's Reference* for usage and implementation details.

Please refer to the following additional resources to learn more about CommonSpot:

- *CommonSpot Administrator's Reference*
- *CommonSpot Content Contributor's Reference*
- *CommonSpot Developer's Guide*
- *CommonSpot Demo Guide*
- *CommonSpot Elements Reference*
- *CommonSpot Installation Guide*
- *CommonSpot Template Developer's Guide*

Visit <http://www.paperthin.com/support/> to access the:

- CommonSpot Document Library
- Knowledge Base Articles
- Product downloads and updates

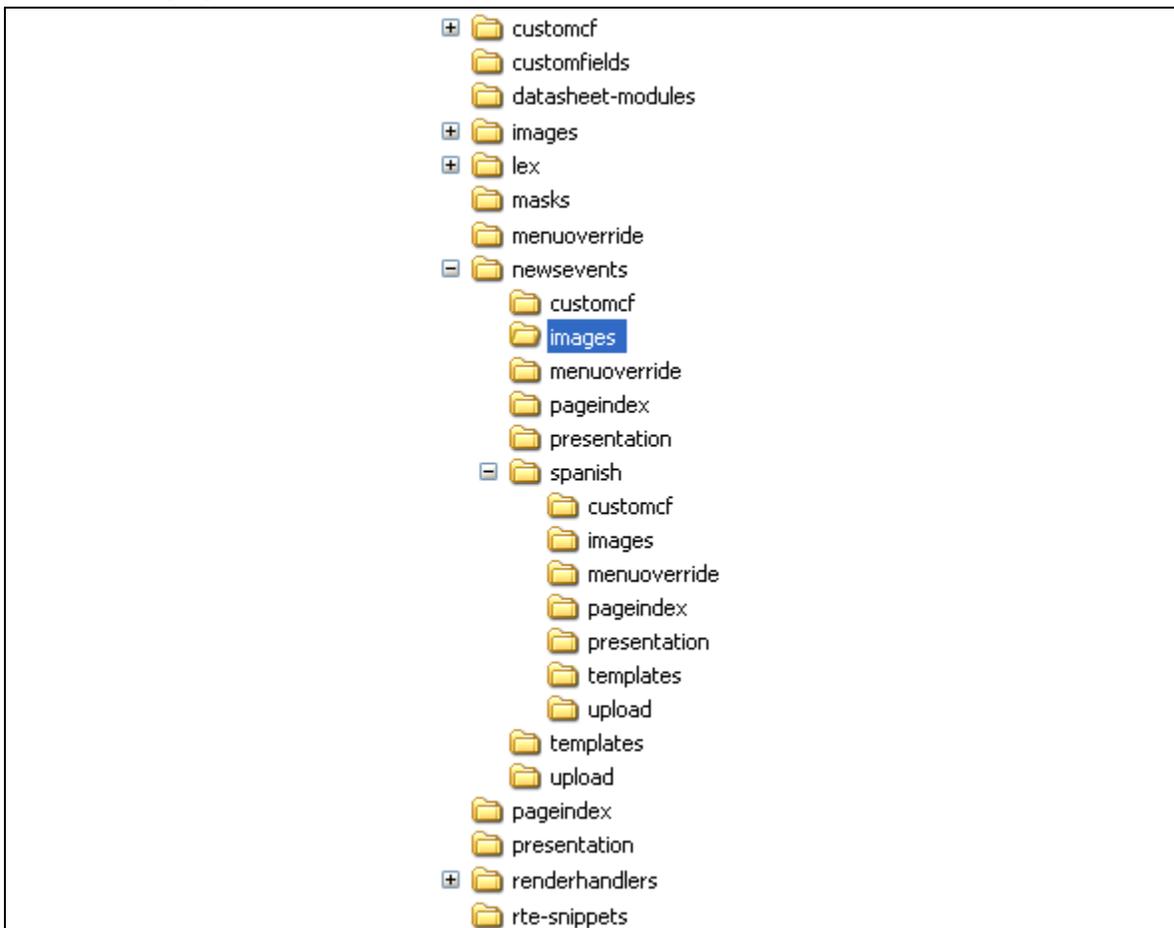
The CommonSpot community site, the | commons, is a valuable resource for CommonSpot users and developers. Visit <http://community.paperthin.com> to learn about the Application Development Framework, PaperThin's tool for rapidly developing rich applications in CommonSpot. This site is a platform for sharing information about CommonSpot and features applications built by CommonSpot customers and the PaperThin professional services team, articles on everything from content strategy to technical architecture, an interactive version of the latest API.

Chapter 2 Understanding the CommonSpot Site Hierarchy

When CommonSpot creates a site or subsite, it builds a directory hierarchy. Every CommonSpot site is a stand-alone entity with its own content database and hierarchy.

Note: CommonSpot sites do not share search indexes with one other, nor do they share content or templates. It is important to keep this in mind when creating separate CommonSpot sites.

Each subsite adds to this directory hierarchy. When adding a new subsite, it is important to note that the site's structure affects URLs, breadcrumbs, security, content scheduling, and searches. The following figure shows an example of this structure after a *newsevents* subsite is added.



Some of the directories such as `customcf` and `images` exist in all subsites. Others such as `datasheet-modules` and `renderhandlers` exist only at the site's root.

Chapter 3 Creating Your Site

After deciding what your site's structure will be, you can create it. CommonSpot walks you through the site creation process, automatically generating a file structure and populating databases. If you are using Oracle or MS SQL Server, you need to manually create the site's *Users* database ahead of time.

After creating the database file, navigate to CommonSpot Server Administration located at `{yourserver}/commonspot/admin/index.cfm`.

CommonSpot: Content Server admin-commonspot | Profile | Logout

Admin Help

Server Administration

[+] Expand All [-] Collapse All

- Configuration
 - Server Information
 - Databases
 - Authentication
 - Client Variables and Cookies
 - Directories
 - Session Timeout
 - Email Notifications
- Customers
 - Manage
 - Add New
- Utilities
 - Server Tools
 - Scheduled Jobs
 - Running Jobs
- Security
 - General Security

CommonSpot Server Information

Licensed To:	550QACF8-Customer 2
Server Key:	s-507549-550QACF8.cfm
CommonSpot Version:	Internal Build 6.0.0.374
Sites Datasource:	commonspot-sites
Database Version:	SQLServer 9.00.4035
UTF8 Enabled:	No
Sites Schema Version:	2009-09-23 09:00:00
Authoring State:	Enabled Change

Server Permissions

- ✓ Authoring Enabled
- ✓ Hosting (up to 10 customers)

Software Availability

- ✗ Microsoft Word
- ✗ Microsoft PowerPoint

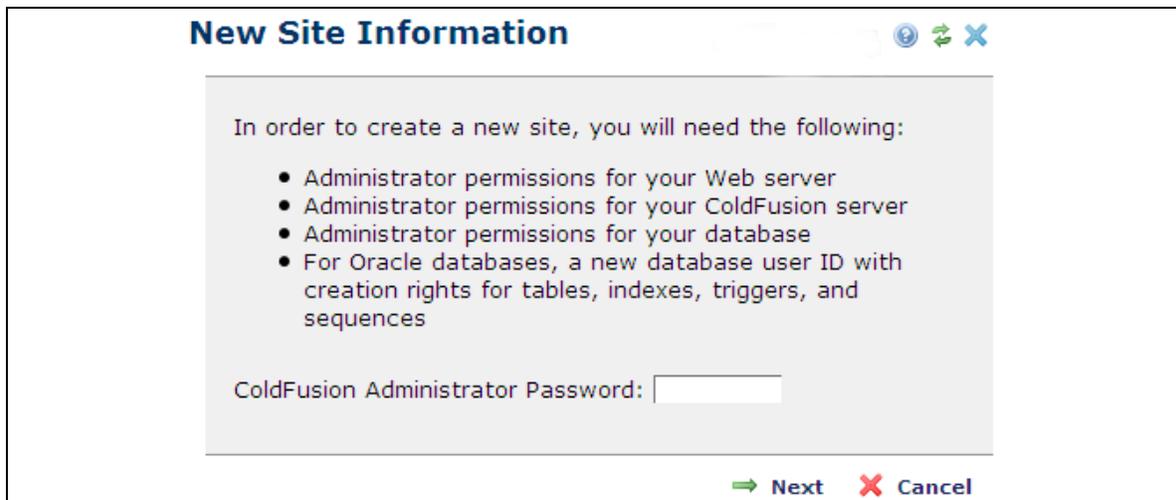
Sites on this Server

Site Name	Customer	Active	Actions
issues1	550QACF8-Customer 2	✓	
issues2	550QACF8-Customer 2	✓	
issues3	550QACF8-Customer 2	✓	
root	550QACF8-Customer 2	✓	
demo	CommonSpot Demo	✓	
Create New Site			

Installed Patches

Patch ID	Details
No patches installed	

Click the **Create New Site** link at the bottom of the **Sites on this Server** section.



New Site Information

In order to create a new site, you will need the following:

- Administrator permissions for your Web server
- Administrator permissions for your ColdFusion server
- Administrator permissions for your database
- For Oracle databases, a new database user ID with creation rights for tables, indexes, triggers, and sequences

ColdFusion Administrator Password:

→ Next ✕ Cancel

The **New Site Information** dialog lists the permissions you need to create a site. It also asks you to enter your ColdFusion administrator password. Enter the password and click **Next**.



New Site Name

Please enter the following information to create a new site.

Customer: Choose the customer of the site. If the customer does not exist, close this dialog and create the customer first.

Name: The name for the new site.

Description: A brief description of the new site.

Display Name: The name to display in dynamically rendered elements.

→ Next ✕ Cancel

Use the **New Site Name** dialog to enter the following information:

- **Customer** – If your license lets you have more than one customer, select the customer's name to identify the customer. This tells CommonSpot where it will create the site.
- **Name** – This text sets the site's directory name. This is where CommonSpot places all of your site's files and directories.
- **Description** – CommonSpot displays this text only within administrative and authoring pages when it displays information about a subsite.

- **DisplayName** – The text that CommonSpot uses within many pages such as links, breadcrumbs, etc.

After filling out these fields, click **Next**.

New Site Configuration

Please provide the following information to create a new site.

Root Directory:
Specify the fully qualified root directory for the site fundforhaiti (i.e. c:\inetpub\wwwroot\{sitename}, or the appropriate UNC path).

Web Server Mapping: **Site is at Web Server root.**
Check this box if the site will be seen by the web server as '{servername}/' instead of '{servername}/fundforhaiti'.

Site IP Address:
Enter the IP address (or alias) which will be used for this site.

Data Source:
The name of the data source to be created for this site.

Database Type: Oracle
SQLServer
MySQL
The database type for the new site.

Administrator Password:
The password for the 'admin-fundforhaiti' user.

Retype Password:
Retype the password for the 'admin-fundforhaiti' user.

← Prev
Next →
✗ Cancel

The **New Site Configuration** dialog gathers the information that CommonSpot needs to configure your site. The **Root Directory** names the place where CommonSpot installs your site. Each site must have its own directory. By default, CommonSpot installs new sites within this directory.

The **Web Server Mapping** checkbox tells CommonSpot if this site is located at the web server's root. If checked, CommonSpot installs the site within your web server's root directory. For example, if your domain is <http://www.MyDomain.com> and *mainsite* is your site's root, users display your site's pages by typing <http://www.MyDomain.com/> in their browser. If the site you are creating is not a root site, its address would be <http://www.MyDomain.com/mainsite/>. This checkbox also lets CommonSpot know how it will store internal path information.

Other controls let you:

- Set the new site's IP address or alias.
- Name the ColdFusion data source that CommonSpot creates.

- Provide the password for the user that will have administrative privileges for the site.
- Select the database software.

The remaining steps are installation dependent. If you are using SQL Server or Oracle, you must configure the new site's database settings. If you are using SQL Server, the **New Site Database Information** dialog displays as shown below.

New Site Database Information

Please enter the following database information:

Note: CommonSpot requires the creation of an empty SQL Server database for this new site. If you have not created an empty SQL Server database for this site, please create one now before proceeding. As part of the site configuration process, CommonSpot will create all of the necessary tables and indexes in the empty database.

Database Name:
Enter the SQL Server database name for the commonspot-site-fundforhaiti data source

Database Server:
Enter the server name where the SQL Server database resides

Database Port:
Enter the IP port for the SQL Server database connection

Database User ID and Password are REQUIRED:

Database User ID:

Database Password:

Use the Web Server Administrator application to:

- Verify that 'Script' and/or 'Execute' permissions are granted for **C:/web/fundforhaiti**

➔ Next

After CommonSpot creates your site, it displays the **New Site Created** dialog, which has three options:

- **OK** – Returns you to the Server Administration dashboard.
- **Configure** – Displays the CommonSpot Site Administration Dashboard. Here you can create new subsites, set site and content security, define templates, and set other site parameters.
- **Create First Page** – Displays the Create New Page dialog so you can start creating content for your site.

Chapter 4 Understanding Templates

Before implementing your web site's design, you need to understand how CommonSpot uses templates. This chapter discusses templates and how they help you create a site.

The contents of this chapter are:

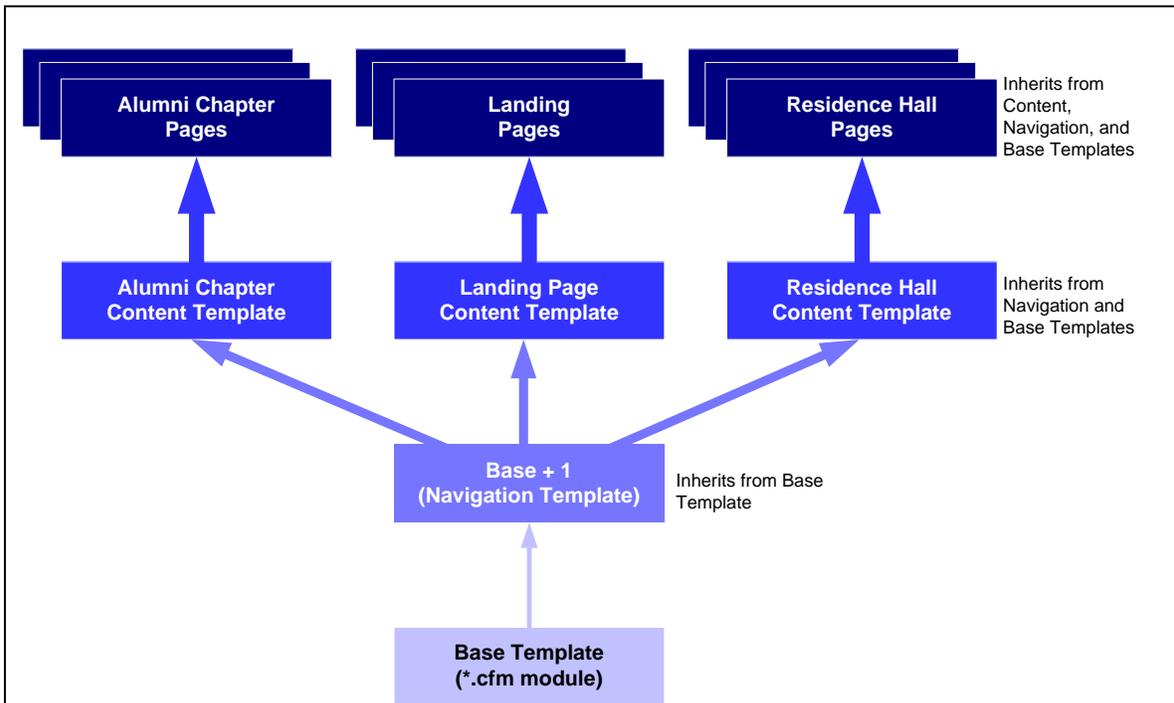
- [Template Overview](#)
- [Base Templates](#)
- [Navigation Templates](#)
- [Content Templates](#)

4.1. Template Overview

Every CommonSpot page derives from a template hierarchy. CommonSpot uses three kinds of templates:

- Base
- Navigation
- Content

Your site's look and feel is defined within these templates. Think of these templates as transparencies where lower-level templates show through to higher ones. At the page level, you are seeing all templates in a hierarchy. This hierarchy is called an inheritance structure.



Notice that all pages and templates inherit from the “base” and “base + 1” templates. This is good CommonSpot site design practice. The example above shows a three-level inheritance structure, with all pages derived from one of the three hierarchies.

4.2. Base Templates

A base template is your site’s blueprint and the starting point for the site’s structure and design. A base template only contains ColdFusion code created using a code editor. In contrast, you will create every other template within a browser using the CommonSpot page creation UI.

All other templates and pages are derived from the base template. The base template can contain a standard navigational bar, a hierarchical menu, a company’s logo, stock ticker, and other objects that appear on all pages. Because the base template is a ColdFusion module, you will create or customize it using CFML tags. You can also add web technologies that work with ColdFusion, for example, HTML, DHTML, and JavaScript.

4.3. Navigation Templates

Navigation templates immediately follow the base template in the hierarchy. A navigation template defines the “outside the white space” areas of your site; that is, these are the places on the page where contributors do not add information. While these two templates seem to do the same thing, their purposes differ. The base template defines your design’s structure; a navigation template defines the design’s substance. Most CommonSpot sites have one navigation template but some have more. In all cases, they are essential because they are the bridge between the base and content templates.

4.4. Content Templates

While the base and navigation templates work together to create the “outside the white space” areas of your site, content templates structure that white space. They define the type of content (CommonSpot Elements) that contributors can use when creating a page. For example, the base template may define a two-column design with a header and the navigation template sets the header’s content. In contrast, a content template contains the Elements that appear in the white space.

Typically, a site contains a number of content templates, each defining a page’s content. For example, your site may have About Us, Press Release, Blog, and Calendar templates. It is a best practice to design content templates so that all pages derived from it have a common look and feel and so that content contributors cannot change page design when adding Elements.

Chapter 5 Developing the Base Template

Modifying the base template that CommonSpot installs is the first step in building your site. It is where you define your site's foundation and your design's "skeleton." This chapter contains information you need to develop your site's base template. The contents of this chapter are:

- [Base Template Introduction](#)
- [Importing HTML](#)
- [Replacing Content with CommonSpot Elements](#)
- [Populating the "Base +1" Template](#)
- [Advanced Base Template Development](#)

5.1. Base Template Introduction

Note: While CommonSpot can create and register base templates, PaperThin recommends that you modify the base template that CommonSpot installs.

When CommonSpot creates a site, it defines a template structure, as follows:

- A base template – Its file name is `template-basepage.cfm` and resides in the `{yoursite}/templates/` directory.
- A navigation template – Its file name is `template-base-template.cfm` and it resides at your site's root. This template is referred to as the 'base +1 template'
- Content templates – You will name these files.

Before you begin, you should understand base template coding conventions:

- **The base template only defines text that exists within a page's <body> section.**
Although you are developing your site's structure using HTML and CFML, you are only creating the information that falls between the HTML `<body></body>` tags. Your code cannot include a starting `<HTML>` tag or a head section. This does not mean you cannot add information between a page's `<head></head>` tags because CommonSpot will honor the contents of a `.head` file. Other techniques also exist. For more information, see the *CommonSpot Developer's Guide*.
- **CommonSpot uses `enablecfoutputonly="yes"`**
CommonSpot uses the ColdFusion `enablecfoutputonly="yes"` setting. Consequently, you must place HTML and JavaScript within `<cfoutput></cfoutput>` tags. As a best practice, you should not execute CFML from within these tags.
- **Base templates execute only at render time.**
CommonSpot caches the results of base template execution only when it is first rendered,

or when you explicitly request this action. As this information exists within cache, CommonSpot does not execute the template when a browser requests a page. If the base template contains dynamic code, CommonSpot also caches the information created when the page executes. As a result, some variables, including time-dependent variables, may not contain the values you think are there. You should avoid using dynamic code within the base template and instead place it in other templates. Because caching is page-based, page and subsite variables now contain the correct information. Refer to the *CommonSpot Developer's Guide* for a more information.

- **You must clear the cache after changing the base template.**

Because CommonSpot caches the base template, you must clear the cache before your changes propagate to derived pages. However, you do not need to clear the cache for changes you make to templates and pages because CommonSpot does not cache these pages until it publishes them. You can clear the cache for base templates on a page-by-page basis by selecting Actions > Clear & Update Cache while viewing a page in author mode. You can clear it for a site by selecting Properties > Cache within Site Administration. Refer to the *CommonSpot Developer's Guide* for additional information.

5.2. Importing HTML

Note: This section uses the demo site's design as an example. This information is simplified as the demo site uses some advanced techniques. To learn more about advanced techniques, see the references in the [Advanced Base Template Development](#) section in this guide.

On the demo site, as with many sites, the home page's design differs significantly from the rest of the site. It is almost best to ignore the home page and instead concentrate on developing the site design.

Begin by analyzing the site's design and determining where you can replace HTML with CommonSpot Elements. The following figure shows the demo site's *About CSU* page, annotated to review how this page might be analyzed.

The screenshot shows the 'About Us' page of CommonSpot University. The page is annotated with ten numbered areas:

- 1**: The main header area containing the university logo and name.
- 2**: The top right navigation area with links like 'CSU Home', 'CSU Feeds', 'Community', 'Site Map', 'Contact Us', and 'My Profile', along with a search bar.
- 3**: The main banner image showing three students looking at a document.
- 4**: The left sidebar menu with links like 'About CSU', 'Welcome from the President', 'Fact Sheet', 'Employment at CommonSpot University', 'Your Feedback', 'Take Our Visitor Survey', and 'CSUTube'.
- 5**: The breadcrumb navigation 'Home > About Us'.
- 6**: The main content area containing the introductory text and several sections: 'Meet Our President', 'Feedback', 'CSU Fact Sheet', 'Visitor Survey', 'Employment at CSU', and 'CSUTube'.
- 7**: A small image of a city skyline.
- 8**: A small image of a globe with the text 'Additional Information', 'Boston.gov Website', and 'WCVB Boston'.
- 9**: The footer area with sections for 'Resources For' (Prospective Students, Current Students, Faculty & Staff, Alumni), 'Useful Links' (Feedback, Blog, Jobs, Site Map, CommonSpot, PaperThin, The Commons, ADF), and 'Connect With Us' (social media icons and a Facebook like button).
- 10**: The bottom footer area containing copyright information and the 'Powered by CommonSpot' logo.

All numbered areas with the exception of 6 are common navigation and design areas. Area 6 represents the page's white space. After this analysis, you should copy the page's HTML into the base template and then replace sections with CommonSpot Elements.

Start by opening the `{yoursitedir}/templates/template-basepage.cfm` file in a code editor. Because the base template works with the base +1 template, you should also open `{yoursite}/template-base-template.cfm` in a browser. After making a change to the base template, refresh the base + 1 template to see your changes.

Log in to CommonSpot and switch into author mode. The base + 1 template should look like this:

Base + 1 Template

 [Click here to define the Tabular Layout element](#)

This page was created using the default base template provided with CommonSpot. To remove this text and insert your own content, use a code-editor (such as ColdFusion Studio or Notepad) to edit the file `C:/web/cs50/testsite/templates/template-basepage.cfm` on your server.

1. After opening the base template in a code editor, notice that the file begins with a long comment. This text contains instructions and tips for developing the base template.
2. Below the instructions and tips, insert the following `cfmodule` call:

```
<CFMODULE TEMPLATE="/commonspot/utilities/ct-render-named-element.cfm
    elementtype="container"
    elementName="mainContent">
```

This instruction tells ColdFusion to call `ct-render-named-element.cfm`. You will always place this module within the base template. This call is unique as it is adds a CommonSpot Container Element and names it "mainContent". This element defines your site's white space (area 6 within the demo size analysis). All CommonSpot built-in content templates look for an Element with this name. Thus, you should build your navigational design and site structure around this Element.

3. Delete the descriptive text in the template, carefully retaining the mainContent Element.
4. You can now add HTML to this template. To get started, copy the HTML code within the `<body></body>` tags from the About Us page you analyzed and paste it into the base template. Because CommonSpot uses the `enablecfoutputonly=yes` setting, you must place this HTML within `<cfoutput></cfoutput>` tags. In addition, you must double up all `"#"` characters in the pasted text. This prevents ColdFusion errors.

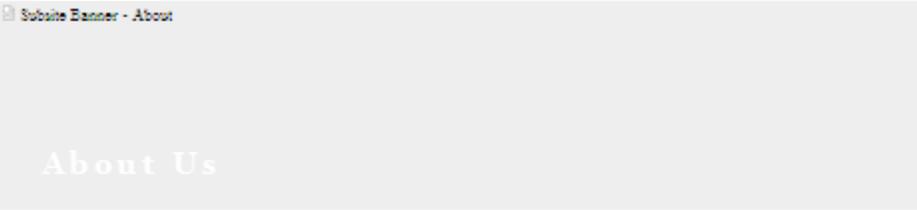
5. Browse to the base + 1 template, which should appear as follows:

[CSU Home](#) | [CSU Feeds](#) | [Community](#) | [Site Map](#) | [Contact Us](#) | [University Login](#)
 Search:



[Future Students](#) | [Current Students](#) | [Faculty & Staff](#) | [Alumni](#)

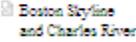
- [About CSU](#)
- [Academics](#)
- [Admissions](#)
- [Alumni](#)
- [News & Events](#)
- [Campus Life](#)
- [Safety & Security](#)

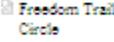


About Us

- [Welcome from the President](#)
- [Fact Sheet](#)
- [Employment at CommonSpot University](#)
- [Your Feedback](#)
- [Take Our Visitor Survey](#)
- [CSUTube](#)

[Home](#) [About Us](#)





Related Information

[City of Boston gov Website](#)
[WCVB Boston](#)

Perennially ranked the top public university in New England, CommonSpot University now stands among the best public institutions in the nation. CommonSpot University main campus in Quincy is admitting the highest-achieving freshmen in University history. Student diversity continues to increase, as does the number of honors students, valedictorians and salutatorians who consistently make CommonSpot University their top choice.

<p> Meet Our President A welcome message from our president </p> <p> CSU Fact Sheet A listing of facts about CommonSpot University </p> <p> Employment at CSU A listing of both full and part time employment opportunities at CommonSpot University </p>	<p> Feedback Using this form please provide us with any feedback you may have regarding the university, the website or your visit </p> <p> Visitor Survey Once you have visited CommonSpot University let us know what you liked most by completing our survey </p> <p> CSUTube Videos from the CommonSpot University Community. </p>
--	--

While paths to the page's images are not correct, the template's main structure is now in place.

5.3. Replacing Content with CommonSpot Elements

You are now ready to replace the content of your design with CommonSpot Elements. You will leave the page's structure in your code. For example, a `spacer.gif` in a structural table can remain. In contrast, you should upload images used in navigational areas into CommonSpot. There are many reasons why you should do this. The most important reason is that it eases updating, avoids base template caching, and incorporates the images into CommonSpot so that when you move or change pages or images, CommonSpot can change links to them contained within other CommonSpot pages.

1. Create your "white space." You will do this by deciding which parts of the design are consistent on your site's pages and which parts contain content that changes from page to page. In the demo site's About Us page, this is area 6. Create this area by replacing its HTML code with the mainContent Container element:

```
</cfoutput><CFMODULE  
  TEMPLATE="/commonspot/utilities/ct-render-named-element.cfm"  
  elementtype="container"  
  elementName="mainContent"></cfoutput>
```

The replaced section within the base template should look like this:

```
</div>  
<div id="TEXT" class="normal">  
  <div class="BlockWrapper"></div>  
  <!-- BEGIN MAIN CONTENT -->  
  </cfoutput>  
  <CFMODULE  
    TEMPLATE="/commonspot/utilities/ct-render-named-element.cfm"  
    elementtype="container"  
    elementName="mainContent">  
  </cfoutput>  
  <!-- END MAIN CONTENT -->  
</div> <!-- TEXT -->  
<br clear="all" />
```

2. Next, save the base template and reload the base + 1 template. The page should look like this:



Area 6 is now a Container Element. This is the only Element that you may not edit in the base + 1 template; instead, contributors edit it in pages derived from a content template.

The `cfmodule` call to `ct-render-named-element` inserts CommonSpot Elements into a page. The `elementType` attribute tells CommonSpot which Element to insert. (A list of these Elements is in the base template's commented area.) Use the `elementName`

attribute to name the Element. After executing this call, the base template contains a new Element instance with this name.

- Continue by replacing other HTML with Elements. For example, the top row of links contains the site's navigation links. Begin by deleting its HTML code and replacing it with `linkbar` and `customcf` Elements.

```
</cfoutput>
  <cfmodule template=
    "/commonspot/utilities/ct-render-named-element.cfm"
    elementName="headerlinks"
    elementType="linkbar">

  <cfmodule template=
    "/commonspot/utilities/ct-render-named-element.cfm"
    elementName="userlogin"
    elementType="customcf">
</cfoutput>
```

The replaced code should look like this:

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
2 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
4 <head>
5 <title>About Us</title>
6 <meta name="Description" id="Description" content="About Us" />
7 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
8 <link rel="stylesheet" href="/demo/style/main.css" type="text/css" />
9 </head>
10 <body>
11 <div id="HEADER">
12   <div class="Wrapper">
13     <div class="Links">
14       </cfoutput>
15       <cfmodule template="/commonspot/utilities/ct-render-named-element.cfm"
16         elementName="headerlinks"
17         elementType="linkbar">
18       <cfmodule template="/commonspot/utilities/ct-render-named-element.cfm"
19         elementName="userlogin"
20         elementType="customcf">
21     </cfoutput>
22   </div>
23 </div>
24 </div>
25 <div id="BRANDING">
26   <div class="Wrapper">
27     <div class="Logo">
28       <a href="/demo/index.cfm" style="text-decoration:none"></a></div>
30     <div class="Right">
31       <a href="/demo/futurestudents/index.cfm">Future Students</a>&nbsp;&#124;&nbsp;&nbsp;&nbsp;<a href=
32 "/demo/currentstudents/index.cfm">Current Students</a>&nbsp;&#124;&nbsp;&nbsp;&nbsp;<a href="/demo/facultyandstaff/index.cfm">
33 Faculty & Staff</a>&nbsp;&#124;&nbsp;&nbsp;&nbsp;<a href="/demo/alumni/index.cfm">Alumni</a>
34     </div>
35   </div>
36 </div>
```

4. The following figure shows what the page should look like after you refresh the base + 1 template:



You can now use the CommonSpot UI to populate the LinkBar Element.

5. Repeat steps 3 and 4 above for each section of this design. For example, you could replace HTML in the demo site with the following Elements. Notice that several Elements are used more than one time as this page contains multiple sets of links:
- Link Bar
 - Single Image (without header)
 - Page Index
 - Breadcrumb Links
 - Image Grid

- Simple Text Block (without header)

As you replace content with CommonSpot Elements, you should refresh the base +1 template and verify that you are correctly replacing your content.

5.4. Adding Content to the “base +1” Template

You are now ready to add new content to the base + 1 template. You will do this using the CommonSpot UI. While the design’s structure and CommonSpot Elements exist within the base template, you need to add content to these Elements. Since the default has content templates inheriting from this navigation template, content defined here displays on all of your site’s pages.

The following set of steps shows one way to do this:

1. Since the demo site uses CSS, you need to incorporate style sheets into CommonSpot. Begin by uploading them to your web site. From the **Manage** menu, select **Style sheets**, and check **Enable Style Sheets**.

Style Sheet Sets

Check or uncheck the box below to enable or disable use of style sheets for this page.

Enable Style Sheets Save

Class to use on Body tag: ---Use default---

The following list shows the Style Sheet Sets registered for this page. Each set contains one or more style sheets, and can be scheduled based on criteria such as subsite, audience, time of day, etc. CommonSpot uses the first set in the list whose criteria match. Click and drag to set priority, or select one or more items and click an Insert  icon to move them.

This page has its own style sheet set information (names, schedules and order). Stylesheets themselves inherit independently for each set, as indicated below.

Style Sheet Sets: [Select All](#) [Deselect All](#)

Name	Style Sheets	Schedule	Actions
Home (stylesheets from page)	/demo/style/home.css	Schedule: All Dates Groups: All Groups SubSites with URLs containing: '/demo/index.cfm'	   <input type="checkbox"/>
Student Pages		Schedule: All Dates Groups: All Groups Subsites: All Subsites	   <input type="checkbox"/>
Default (stylesheets from template)	/demo/style/main.css /ADFdemo/apps/pt_profile/style/profile.css	Schedule: All Dates Groups: All Groups Subsites: All Subsites	   <input type="checkbox"/>
Faculty Pages		Schedule: All Dates Groups: All Groups Subsites: All Subsites	   <input type="checkbox"/>

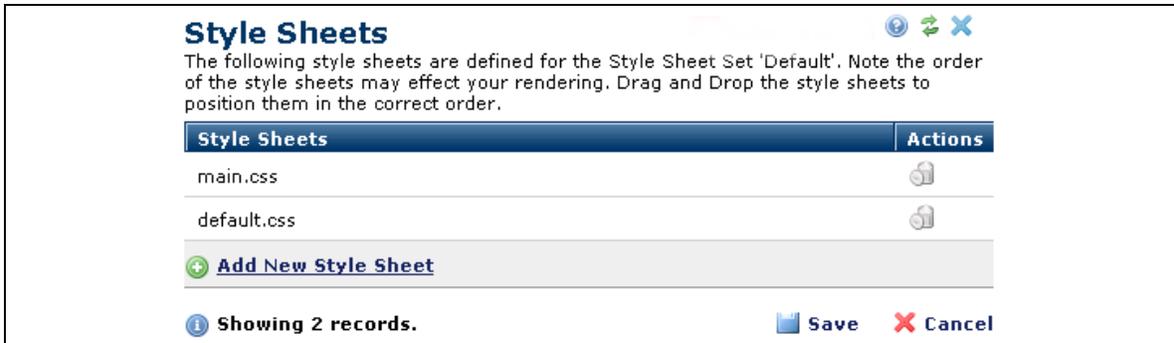
Showing 4 records.  [Add New Style Sheet Set](#)

 **Close**

2. The demo site uses CSS to set most Element colors, alignments, and other properties. Their definition is in the `default.css` file that CommonSpot added to your site when it was created. By default, CommonSpot includes this style sheet in all templates when style sheets are enabled. You can add style sheets to the default style sheet set and create

additional sets. For example, the Style Sheets dialog contains the Student pages style sheet set.

3. To add additional styles sheets:
 - a. Upload the style sheet into the `{yoursite}/style` directory
 - b. Select **Manage > Style Sheets**.
 - c. After CommonSpot displays the **Style Sheet Sets** dialog, click the pencil icon for the Default Style Sheet Set, and then select **Style Sheets** from the drop-down menu.



- d. Click **Add New Style Sheet** and enter a name and description. Click the pencil icon and select **Style Sheets**. In the Add Style Sheet dialog, select **Explicit**, then enter the style sheet's logical path and file name.

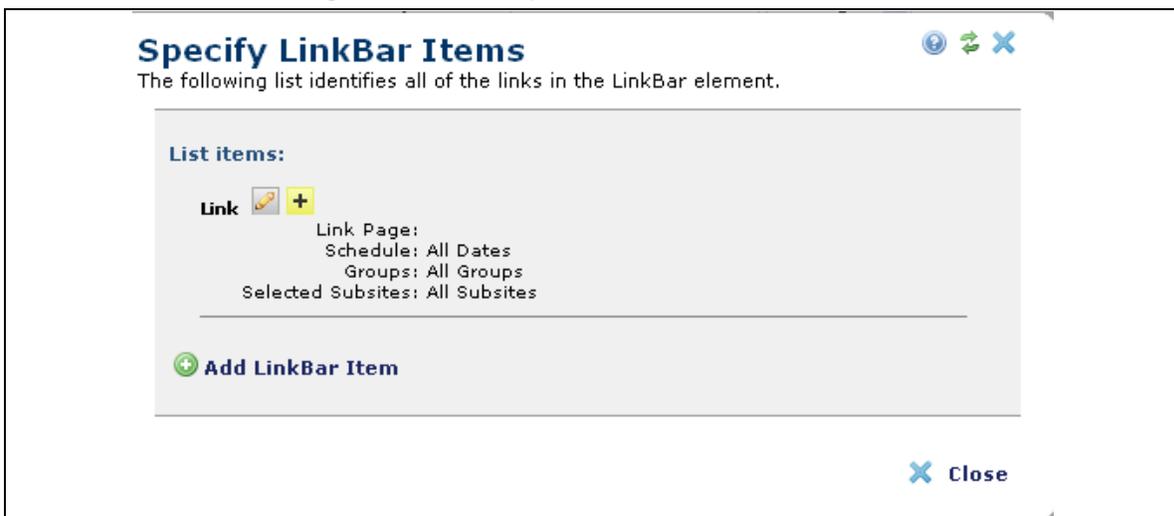


CommonSpot will now include your CSS file in templates, pages, and custom code.

4. As you add content to an Element, refresh the base +1 template. (You should view your changes as you populate an Element, rather than waiting until you add all Elements, as this greatly simplifies debugging.) For example, populate the LinkBar Element containing the demo site's top-most links by clicking on the ghost text (the grayed text) that is a placeholder for the Element. CommonSpot will now assist you in creating the Element. For more information, see the LinkBar Element section within the *CommonSpot Elements Reference*.



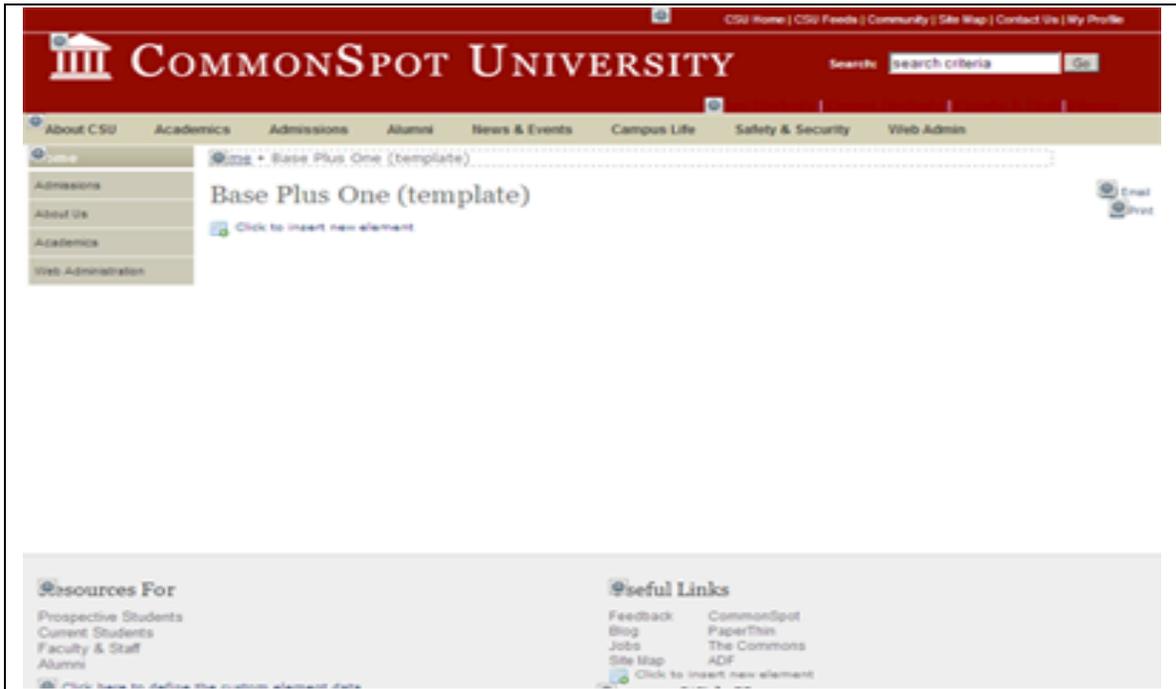
Here is what this dialog looks like after you enter an item:



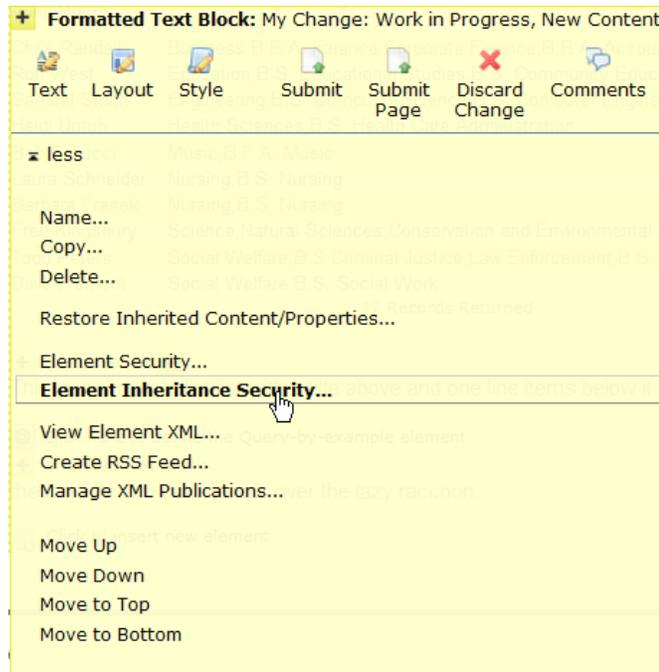
You can now add links:



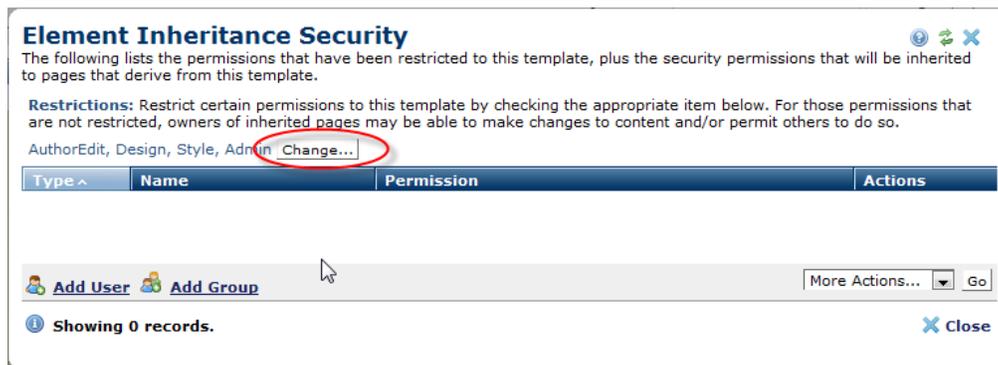
5. When you are done, redisplay the base + 1 template. The following figure shows a base + 1 template after the LinkBar Elements are populated:



6. You are now ready to populate other Elements in the base + 1 template. You will also need to know how each Element relates to the rest of your site. For example, the main Content Container Element is on all of your site's pages. It cannot contain content at the template level and must be fully functional at the page level. In contrast, you must add content to the LinkBar Element at the template level and not allow pages at lower levels to make changes. This strategy maintains design integrity. You will need to stop contributors from changing an Element by locking it. Do this by clicking the LinkBar's Element edit icon, select **more**, and then choose **Element Inheritance Security**.

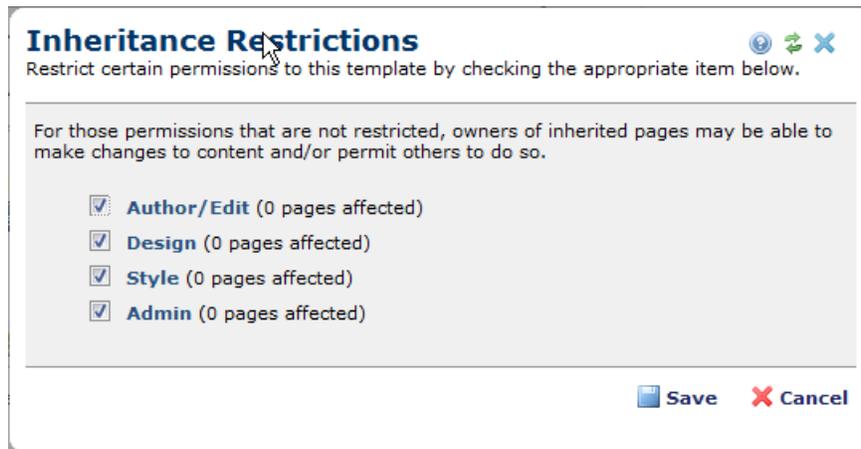


7. Within the **Element Inheritance Security** dialog, select **Change**.



The **Inheritance Restrictions** dialog shown in the following figure lists the Element's permissions. Each check box you select prevents a contributor from making changes to derived templates or pages. If you check all boxes, no one can edit the Element within derived templates and pages. For example, selecting **Author/Edit** prevents contributors

from changing an Element on the current page. If this is set for a LinkBar Element, no derived template or page can edit its links.



8. Continue adding Elements and setting their properties until you are done replacing text with Elements.

5.5. Advanced Base Template Development

Some designs cannot use the techniques described in Section 5.4. Navigation that changes by subsite, content that uses dynamic code, or other similar techniques may lead you to believe that you need to create multiple base templates. This is a mistake. Having more than one base template complicates inheritance and leads to a messy, hard-to-maintain site. In almost all cases, multiple base templates are unnecessary as CommonSpot lets you make your base template flexible. Here are some of the more popular techniques:

- **Head section customization**

If you need to place information within a page's `<head></head>` tags, place it instead in a `.head` file. For more information, see "Customizing the `<head>` Section" in "Chapter 3 APIs and Interfaces" in the *CommonSpot Developer's Guide*.

- **Style Sheets**

Most web sites now use cascading style sheets. CommonSpot features for integrating cascading style sheets are robust and include conditional style sheet use and class registration. See the *CommonSpot Administrator's Guide* for details.

- **Custom Metadata**

Sometimes a site's layout must be flexible; for example, navigation links appear on the left on one page, on the right on another, and not appear at all in yet another. Using custom metadata lets CommonSpot adjust attributes on the fly. See the *CommonSpot Administrator's Reference*, *CommonSpot Developer's Guide*, and *CommonSpot Template Developer's Guide* for details.

- **Template Variables**

Template variables are another way to add flexibility. You can add subsite-specific variables that an Element can use to automatically adjust your base template. See the *CommonSpot Developer's Guide* for details.

- **ColdFusion variables and CommonSpot metadata**

Using ColdFusion and CommonSpot variables adds flexibility to your design. See the *CommonSpot Developer's Guide* for details.

- **Scheduled Content**

Navigation that changes by subsite or by user is a common challenge when you are creating a base template. There are several ways to do this, with the scheduled Element being the most popular and effective. See the *CommonSpot Element Reference* for details.

Chapter 6 Developing Content Templates

With navigation and base templates done, the next step is creating content templates that your site's contributors will use to add new pages. A content template guides contributors through the process of creating pages and inserting content. CommonSpot installs a number of content templates that contributors can use immediately. Because you used the `template-basepage.cfm` and `template-base-template.cfm` files and the Container Element, these templates already inherit your design.

The content templates that CommonSpot installs may be all that your site needs. Other sites may only need a few of these templates. In almost all cases, however, you will create content templates that meet your site's requirements.

This chapter is an outline of how you create content templates. The contents of this chapter are:

- [Creating a Content Template](#)
- [Changing Ghost Text](#)
- [Setting Element Inheritance](#)

6.1. Creating a Content Template

To create a content template means creating a CommonSpot page and adding Elements and content to it. Begin by creating a page using the template that most closely matches your content's needs. Usually this means selecting a fairly simple design and then expanding on it. See the *CommonSpot Content Contributor's Reference* for information on creating pages.

The second step is adding the Elements that your contributors will use when they add content. For example, a news article template may need a title, an image, and a body. This means that you would drop in a Simple Text Block, a Formatted Text Block, and an Image Element. Refer to the *CommonSpot Content Contributor's Reference* for more information on adding content.

You will now set attributes for these Elements, as follows:

- **Properties:** The layout, fonts, and other properties that an Element should have.
- **Default Content:** Is the Element pre-populated with content. If it is, will this be default content that a contributor can change or is it unchangeable?
- **Styles:** The classes that browsers apply to content.

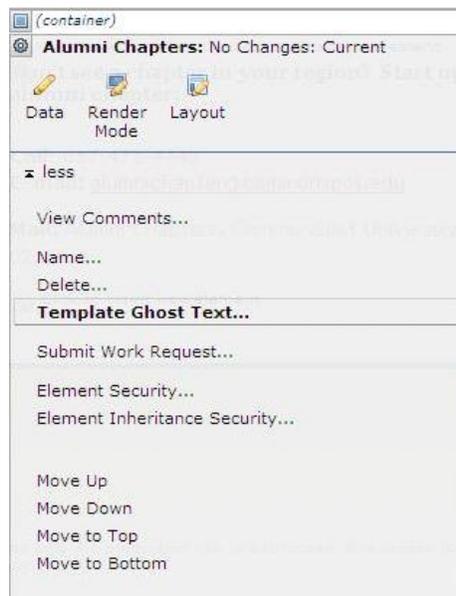
When you finish adding and positioning Elements, you must publish the page and then make it a template. Begin by selecting **Templates > Save as Template** while in page view.

A wizard will help you convert the page. The *CommonSpot Template Developer's Guide* has details.

6.2. Changing Ghost Text

Now that the template is created, you can change it so that it has the features that should be in derived pages.

Each CommonSpot Element has ghost text that prompts the contributor to enter content. For example, a textblock's ghost text is "Click here to enter textblock content". (This is similar to the ghost text that other Elements display.) To change this ghost text — for example, if you want it to be "Click here to enter the article's text" — click the Element's properties icon, choose **Template Ghost Text**, and then change the text.



6.3. Setting Element Inheritance Restrictions

With properties and ghost text set, you need to decide which of the Element's features contributors can use after they derive a page from a template and which are locked down. You can set restrictions either by selecting Element Inheritance Security from the Element's properties menu or by selecting **Templates – Element Inheritance Security** from the My CommonSpot dashboard.

Finally, With a template displayed in Author Mode choose **Templates – Submit Template for Public Use** to make it available to your site's contributors. You may want to use **Manage – Security** to name the users and groups who can use the template.

Chapter 7 Next Steps

With your base template in place, the base +1 template populated, and content templates ready to go, what else is there? Depending on your site's complexity, there may be many things left before contributors can begin authoring pages. CommonSpot documentation will help you with all those tasks. Here are a few of the more common ones:

- **Users and Groups**
One of the most important goals of a content management system is to let non-technical staff assume responsibility for creating and publishing content. However, you must add users and groups of users to CommonSpot before they can create content. See the *CommonSpot Administrator's Reference* for details on creating and managing users and groups.
- **Security**
From read permissions to author permissions to admin rights, CommonSpot security determines which users and groups interact with content and to what extent. See the *CommonSpot Administrator's Reference* for information on setting up CommonSpot security.
- **Personalization**
From individual Elements to entire pages, CommonSpot has robust mechanisms that let you serve up individualized and targeted content. See the *CommonSpot Administrator's Reference* and the *CommonSpot Content Contributor's Reference* for more information.
- **Content Reuse**
Although most content is bound to the pages to which it was added, Custom Elements allow you to share and syndicate content. See the *CommonSpot Administrator's Reference* and the *Elements Reference* for more information.
- **Templates and Elements**
As you develop your site, you almost certainly will create templates that enforce the site's look and feel. See the *CommonSpot Template Developer's Guide* for more information. CommonSpot provides over fifty Elements that you can add to templates and pages. These Elements make it easy to set your site's look and feel. They include Text Block, Simple Forms, Datasheets, Image, Page Set, Flash, MS Word Elements, and many more. You can simply drop these items into templates and pages. See the *CommonSpot Element Reference* for information on Elements.
- **Managing and Creating Pages**
The most basic CommonSpot activity is creating pages. See the *CommonSpot Content Contributor's Reference* for information on managing and creating pages and authoring content.
- **Approval Process**
If your site needs to define a workflow process that controls how content is published, you will need to set up an approval process. For example, you may want all pages edited

before they are published. See the *CommonSpot Administrator's Reference* for more information.

Chapter 8 PaperThin Resources

PaperThin offers a full range of professional services to help you leverage your investment in CommonSpot. Whether it is architecting and designing your site, customizing CommonSpot's capabilities, building specialized components, integrating critical business applications, or helping you with installation and training, PaperThin has the experience and knowledge to assist you with the following resources:

- [CommonSpot ADF and the | commons](#)
- [Contacting PaperThin](#)
- [Contacting Technical Support](#)
- [Contacting Sales](#)

For more information about PaperThin Professional Services visit us at <http://www.paperthin.com>.

8.1. CommonSpot ADF and the | commons

The CommonSpot Application Development Framework is a tool for rapidly developing rich applications in CommonSpot. As of release 6.1 a demo version is available for installation as part of CommonSpot.

<http://community.paperthin.com>, or the | commons, is a resource for the CommonSpot community for:

- Questions, answers, and feedback.
- Downloading applications built by other CommonSpot customers, as well as the PaperThin professional services team
- Articles on everything from content strategy to technical architecture

8.2. Contacting PaperThin

PaperThin, Inc.
300 Congress Street, Suite 303
Quincy, MA 02169
Phone: 617.471.4440
Fax: 617.471.4465

<http://www.paperthin.com>

8.3. Contacting Technical Support

Technical Support is available Monday through Friday 7:00 a.m. to 7:00 p.m. EST, except on holidays. Full details of support plans and options are on our web site at:

<http://www.paperthin.com/support>

8.4. Contacting Sales

Phone: 617.471.4440 (option 1)

Fax: 617.471.4465

E-mail: sales@paperthin.com