

Hannon Hill Corporation

800.407.3540 | www.hannonhill.com | info@hannonhill.com

Makers of the Award-Winning Cascade Server content management software

Cascade Server Glossary

A

Accessibility Checker

The Accessibility Checker (called the 508c checker in versions 3.7 and below) checks and reports errors that, when resolved, result in an enhanced accessibility for published content for the Web Accessibility Initiative (WAI) Guidelines. The accessibility checker, much like the spell checker and the link checker, is invoked after a page edit is committed by clicking the Submit button. If there are any elements that the CMS deems suspect, the user is forwarded to a Accessibility Checker page listing the errors sorted by the field on the page in which they occur. The checker will report on common errors such as <table> elements without summary attributes as well as tags without alt attributes among others. You can customize the accessibility checking for your own specific needs as well.

Administrator

Administrator is a role in the system that permits full, uninhibited access to any asset and/or area of the system. A user assigned the administrator role has access to both the standard " Home" area where web page assets are managed and the " Administration" area where advanced system entities and publishing can be configured. Folder access rights do not apply to the administrator role; therefore, a user with administrator rights can view, edit, copy, delete, or move an asset without restriction. Additionally, administrators have the ability to break asset locks anywhere in the " Home" area. Because an administrator has access to the everything within the system, the role should be assigned sparingly to only authorized users who require such access. Typically, administrator operations can be handled by users with the more restricted manager role, which is one level down from administrator. Managers have all of the same administrator type access and capabilities except that access rights apply to them. Therefore, a user with manager access can be effectively permitted or locked out from areas in both the " Home" and " Administration" sections of the CMS.

Advanced Search

Advanced Search offers a more fine-tuned search capability than the simple search. In addition to the standard search term field, Advanced Search includes check boxes to select types of content to search for as a way of narrowing down the results. When performing the search, only those assets for which the user has read or write access,

will display on the search results page. These results will stay in the system until your next search, which is useful if you need to look through multiple search result items.

Approver

Approver is a role in the system and can be assigned to a User or a Group. The Approver role allows the same actions as the Contributor, as well as allows the user to take part in the workflow process and review and approve (or reject) content that belongs to an active workflow. Approvers may be given read or write access to areas in the system. These access permissions are set by an Administrator at the folder level and are set for an individual User, Group or for All Users. An Approver level user may:

- Navigate through the site structure (read or write access).
- View content (read or write access).
- Edit content (write access only)
- Create new content using Asset Factories (determined by the listing under the New menu)(write access only).
- Copy items (write access only).
- Delete items (write access only).
- Approve or reject content in a workflow he/she is assigned to.

Actions taken by an Approver-level user within Cascade Server will go through a pre-determined workflow process similar to that of a Contributor-level user in order for the action to be completed, which may include approvals by other Approver-level users. A Group can also be assigned an Approver-level role with Cascade Server. When a Group has an Approver role assigned to it, any Users within that Group may approve or reject workflow steps assigned to the Group.

Asset

An Asset is any entity within the system that can be used to generate content. Many different types of entities can be assets: images, CSS stylesheets, XML blocks, pages and more. For ease of classification, assets are divided into several different groups:

1. Page - a mutable asset built by the CMS from other assets to create the finished product of published documents such as web pages or XML documents.
2. File - may be a jpeg image, CSS file, or any other indivisible and immutable asset.
3. Block - as the name suggests, these are XML building blocks from which other blocks and pages are built.
4. Stylesheet - an XSLT stylesheet is used to transform XML blocks.
5. External Link - a symbolic link to a specific URL.
6. Folder - a container within Cascade that acts a parent asset to other assets.
7. Template - the basic building block for a page's layout.
8. Reference - a special asset that is created to represent an existing asset in another location, allowing it to be indexed in multiple folders.

In simplest terms, assets are those entities within the CMS that represent and/or can be used to generate content. For example, a page (which itself is an asset) might derive some of its content from other assets: images displayed on the page, an XML index

block transformed by a stylesheet to create a navigation bar, a CSS file that gives the styles for the XHTML elements in the page's markup, etc.

Asset Archive

When a publishable Asset (Page, File, or Folder) passes its expiration date as specified in that asset's metadata, it will be archived. The archival process involves two steps:

1. first, the asset will be unpublished (removed from the remote server),
2. and then that asset will be moved in the CMS to the Expiration Folder defined in the asset's metadata.

If no expiration folder is specified, the asset will still be un-published however it will not be moved in the CMS.

Asset Factory

An Asset Factory is a way of providing a quick and easy way to create a specific type of asset or specially configured page - for instance, a new file upload or a press release. The asset factory is made available by clicking the "New" menu on the main toolbar at the top of the screen. Default asset factory elements included in Cascade Server are:

- Block
- External Link
- File
- Folder
- Page
- Stylesheet
- Template

New assets created via the default asset factory need not be based off of a particular asset, and may be placed in any folder. In addition, customized asset factories can easily be created by an administrator by selecting an example page upon which future pages should be based. New assets created via a customized asset will have the same template, block and stylesheet assignments, and a default region that must be populated with user-created content. The administrator may also require specific asset factory-created pages to be placed in a pre-selected folder. Which asset factories are available for each user is determined by user access rights. Each asset factory can be configured to display only for selected groups that already exist in the system. Likewise, asset factories can also be grouped into container folders that also allow for grouped access rights to be assigned to them. Containers for asset factories appear as new sub-menus underneath the "New" menu. An asset factory called 'Press Release' might be grouped into a container called 'Articles' and given access rights to Group A. Users that belong to Group A would see a new submenu called "Articles" when they click on the "New" menu. Inside of the "Articles" submenu would be menu item

Asset Tree

The Asset Tree refers to the folder navigation or hierarchical view of assets (files, pages, blocks etc.) that appears in the left side panel in Cascade Server. The asset tree is used for easy navigation and browsing in the content management system. The asset tree is permission sensitive and therefore will restrict unauthorized users from viewing or editing items in the tree.

Audit Trail

The Audit Trail is a tool for Administrators to see a summary of activities performed in the system by a particular user, group, role, or for the entire system. Selecting the audit trail for a group or roles will display the actions performed by all users belonging to that group or role. A date/time filter is provided as part of the Audit Trail view and is useful for filtering the results into a more specific timeframe. The record for each event includes:

- the username of the user who performed the activity
 - the time of the activity
 - the type of action performed (login, edit, publish etc.)
 - the information about that action (whether the asset was edited or published)
 - the IP address of the computer from which the user logged in
 - the asset type
 - a link to the location of the particular asset
-

Authentication

Login Authentication is the process of checking usernames and passwords provided at a login form against authorized credentials. It is accomplished by CMS in one of three ways:

1. Built-in authentication, which stores user information in the database used for other CMS data storage
2. LDAP authentication, accomplished by querying your LDAP-enabled directory server (Active Directory, OpenLDAP, etc.) to retrieve user information.
3. Custom login authentication. This authentication is determined by the administrator of the client network. Please consult your network or CMS administrator for more details.

With the built-in authentication, the administrator must create an account for each user from within the CMS before they can access the system. This is accomplished from the 'Users, Groups, and Roles' option in the 'Administration' area. Using LDAP authentication, on the other hand, the CMS authenticates users against a directory on an LDAP-enabled server (usually separate from the server housing the CMS), which could be a pre-existing directory server. This allows you to maintain users for the CMS and other purposes centrally, without having to create multiple accounts for each user. A custom authentication will vary depending on the particular circumstances.

Typically, there is a single sign-on service, which will authenticate a user against a system separate from Cascade Server. This is usually the same process a user might go through to check organizational email or to use other distributed software.

B

Base Folder

The Base Folder is the top-level folder for assets and folders located in the Home area. Within the Home area, all of the assets will have a parent folder, including folders themselves. The top-level folder called the Base Folder because it is the only folder that does not have a parent. Therefore the Base Folder is the root folder for all other Home assets in Cascade Server.

When multiple sites are managed within the same instance, the Base Folder may also be used to refer to the top folder for a specific site.

Basic Search

Basic Search provides an easy way to perform powerful searches within the CMS content repository. The single search field automatically searches all searchable assets across the name, content, and metadata fields. The content for each asset is typically the XML or XHTML field.

When performing the search, only those assets for which the user has read or write access, will display on the search results page. These results will stay in the system until your next search, which is useful if you need to look through multiple search result items.

Block

A Block is a "pluggable " piece of content that can be easily substituted with or without styling into any page region. There are five types of Blocks in Cascade Server:

- Index Block - used to dynamically generate system assets as XML
- XML Block - static XML content
- XHTML Block - static XHTML content
- XML Feed Block - dynamic XML from a third-party server (e.g. RSS feed)
- Text Block - static text content

Blocks are "pluggable" and reusable - one block can contain text or markup (for example, a copyright notice that should appear on more than one page) that can be plugged into a page region at three levels: the template level, the page configuration level, or the page level. When a block is plugged into a page region at the template level, the block content will appear in any pages using that template. Any blocks assigned at the page configuration level will appear in any pages using that configuration. Block assignments made at the template level can be overridden at either the configuration set or page level. Block assignments made at the configuration

set level can also be overridden at the page level. Whenever there is content that is common to many pages or regions, that content should be put in a block, allowing it to appear in many pages while being updateable in one place, making it easy to maintain that markup/text across many pages. A single block can be reused into an unlimited amount of page regions. Generally, when a block contains XML content (in the case of an index, XML, or XML feed block), that block's xml data is styled with the use of an XSL stylesheet before it is substituted into a page region. This also makes blocks reusable in the sense that one block can be styled in many different ways to output the same content with different looks and feels.

C

Check-In/Out

Often a user will want to retain control of an asset in the CMS, such that while that user has "locked" the asset, no other changes by other users may be made to that asset. This process of locking an asset is called "Checking Out", while releasing such a hold on the asset is referred to as "Checking In". If a user has checked out an asset and wishes to cancel his or her changes, that user would then break or cancel the lock, effectively discarding the changes. Checking in changes made to a locked asset commits the changes back into the CMS and breaks the lock. Assets are automatically checked out to a user any time edits are made in order to prevent multiple users editing the same asset. This lock is automatically timed out once changes have been submitted, unless the asset is sent into a workflow process. During a workflow process, assets are automatically checked in to the appropriate user for each step. If a user wishes to access a locked asset, s/he may choose to send a message to the appropriate user requesting that the asset be released. Manual locks may only be broken by the user to which that asset is checked, or by an administrator.

Component

A Component is any entity managed within the administration area of the system. Components have varied functions from facilitating the creation, management, and publishing of content to the management of system users. For ease of classification, components are divided into several different groups:

1. User - One who uses the content management system.
2. Group - One or more users with common permissions.
3. Role - Determines the abilities and permissions of a User or Group.
4. Asset Factory - Provides a quick and easy way to create a specific type of asset or specially configured page.
5. Configuration Set - Allows for grouping of Configurations, making them more manageable.
6. Data Definition - Generate forms-based input fields for the creation and editing of page content, as well as to generate XML data from the input provided by a user in said input fields.
7. Metadata Set - Provides the interface for customizing the kinds of metadata fields that can be visible and/or associated with an asset.

8. Publish Set - Allow groups of publishable assets to be published on-demand or on a schedule.
 9. Target - Represents a site or subsite and specifies the format in which content should be published (XHTML, XML, etc). A target acts as the glue that binds content to its destination(s).
 10. Destination - Links a target to a transport, specifies encoding (UTF-8 or ASCII), and allows for publishing content on a schedule.
 11. Transport - Stores data concerning how published content is pushed out to the final publish location.
 12. Workflow - A grouping of approval steps that ensure that content entered into the system is quality checked.
-

Configuration

A Configuration is a special component that is typically accessed from the context of a page asset. For example, a page might have "HTML," "Text Only," and "XML" configurations. This allows common content (defined in the page itself) to be rendered using different templates and output types. To allow for this flexibility, each configuration for a page requires a template assignment from which it will inherit the target in order to determine the output type. Blocks and stylesheets may also be assigned to page regions at this level. The configuration is able to override block and stylesheet assignments for each page region defined in the associated template.

Configuration Set

A Configuration Set is an Administration area component that allows configurations to be grouped in one place. Without configuration sets, if you had 100 pages that all needed the same five configurations, you would have to manage $100 * 5 = 500$ different configurations. If you had to change a page region assignment for the HTML configuration across all of these pages, this would have to be done 100 times; one for each page with that configuration. Using configuration sets, you define the five configurations at the configuration set level, and then each page which uses that configuration set inherits the configuration and page region assignments.

Contributor

Contributor is a role in the system and can be assigned to a User or a Group. The Contributor role is the most basic role and allows only the simplest actions within the CMS. Contributors may be given read or write access to areas in the system. These access permissions are set by an Administrator at the folder level and are set for an individual User, Group or for All Users. A Contributor-level user may:

- Navigate through the site structure (read or write access).
- View content (read or write access).
- Edit content (write access only)
- Create new content using Asset Factories (determined by the listing under the New menu)(write access only).

- Copy items (write access only).
- Delete items (write access only).

All of the actions listed are controlled by a workflow process for the Contributor role, meaning that any action taken by a Contributor-level user within the system will go through a pre-determined approval process in order for the action to be completed. A Group can also be assigned a Contributor level role with Cascade Server. When a Group has a Contributor role assigned to it, all Users within that Group are limited to actions permitted for the Contributor role, unless that user has a higher role.

CSS

The term CSS refers to a Cascading Style Sheet, which is the standard method for styling the appearance of a website. Cascading Style Sheets effectively separate the code needed to create the look and feel of a website from the actual XHTML tags that define the structure of its pages. When designing a website, it is important to separate the concepts of presentation and structure, because a site can quickly become very hard to maintain when a designer mistakenly combines them together. Because CSS properties reside inside an independent file, multiple pages within a single website can simply link to it to receive the style definitions that dictate the appearance and positions of elements on a page. When a website designer uses this modular approach to site development, it is easy to change the properties of all pages within a site by making a simple edit to the single CSS file to which all of the pages link.

Example XHTML and CSS

The XHTML below represents a simple webpage document that has a single paragraph in its body. Without any CSS styles, the page would have a plain, white background and the standard web browser font (Times New Roman in many cases).

XHTML Page (index.html) {CODE-1:xhtml}

The CSS file defined below contains two CSS definitions that could be applied to an XHTML document and any others in the website that contain the <body> and <p> elements. The highlighted, red <link> tag in the XHTML document demonstrates how a designer would link the particular page to the CSS file. Since the example XHTML document links with the example CSS file, the background of the webpage will be blue with the page text appearing in the Arial font with an 11-pixel letter height. Likewise, the paragraph in the body of the document would have a 10-pixel margin on all sides, and its text color will be white.

Cascading Style Sheet (styles.css)

{CODE-2:css}

CSS Class

A CSS class is a special component of Cascading Style Sheets that may be used to target unique elements within an XHTML document and assign CSS properties to them. In contrast to a typical CSS definition such as a rule that targets all <p> elements within a document, a class may be used to target a specific <p> element in the document

while skipping the others. Implementing a class within an XHTML document may be useful when an element or group of elements need to stand out on the page with different style properties.

Example XHTML and CSS

XHTML Page (index.html) {CODE-1:xhtml} Cascading Style Sheet (styles.css)
{CODE-2:css}

The XHTML document above contains three paragraphs within its body. The attached, example CSS file styles all <body> and <p> with their respective properties. The page background is blue and the font has been set to Arial with a letter height of 11 pixels. All paragraphs on the page will have 10-pixel margins, and their text color will be white by default except for the last paragraph with the class attribute "red." The name "red" in the paragraph class attribute corresponds to the CSS class definition ".red" in the CSS file. Since the class contains a foreground color property set to "red," the color of the paragraph text will consequently be red instead of white.

In Cascade Server, an administrator may configure a selection of CSS classes to be made available in a drop down box in the WYSIWYG editor. Users can then select the appropriate classes visually as they enter content.

D

Dashboard

The Dashboard, also referred to as the Home Area, is used to supply the user with a summary of the activities that require user attention. The dashboard displays automatically when a user logs into the system and can alternatively be accessed at any time by selecting "Home" from the main menu. The dashboard consists of 4 tabs:

1. Dashboard - offers quick access for creating content, outstanding items, and recently viewed assets.
2. Messages - used to display system messages as well as messages from other users to the user.
3. Workflows - displays all the users's active workflows.
4. Locks - displays a list of all the assets checked out by the current user.

The dashboard is also available as a personalized RSS feed so that the user can view their items from an RSS reader as well as a portal system (as a channel or portlet).

Data Definition

Data definitions are used by the CMS to generate forms-based input fields for the creation and editing of page content, as well as to generate XML data from the input provided by a user in said input fields. They consist of XML data authored either by hand or using the data definition builder (found in /common/data definition builder) that

describe how a certain set of data is structured. For example, an article in an online magazine might be described as follows:

```
{CODE-1:xml}
```

When applied to a page, you would see a series of form input fields: two text fields labeled 'Headline' and 'Location', a text field with a calendar chooser for 'Date', a file chooser for 'Image' and the multi-line WYSIWYG text field 'Main Content'. When the user fills out this form, the CMS again uses the data definition to make the data available as XML, something like this:

```
{CODE-2:xml}
```

This XML data can then be transformed by an XSLT stylesheet for the purposes of presentation or multi-purposing of content. Simply put, data definitions allow one to define a structure, which all data of a specific type should exhibit, making automation of presentation and repurposing possible.

Destination

A destination represents a location to which a target can be published, and allows scheduling as well as format (UTF-8 vs. ASCII) control over the published content. Destinations are useful for controlling individual sites that may need different settings or to publish on different schedules using different locations (represented by Transports). Each destination has a transport that represents a server where the published content will reside. Destinations have access rights restrictions by group membership. When publishing a target, Cascade Server allows the selection of the individual destinations that are associated with that target. This is generally used when content is to be published to a development or internal server for final Quality Assurance before being pushed to a live site. A target contains a collection of one or many destinations. These destinations defines the locations where the target can publish to. The location is defined by the destinations associated transport. The transport represents a server location.

Dynamic Navigation

Navigation menus on websites are typically a graphical view of a site's hierarchy. When pages are added, deleted, moved, or renamed, all corresponding links need to be updated on the navigation menus throughout the site. Dynamic navigation allows a navigation menu to update automatically, based on the state of a website's content. In Cascade Server, dynamic navigation menus are created through the use of Index Blocks, or blocks that index the content of a specified folder or portion of the site. An index block is then placed into a page region (at the template, page configuration, or page level), and styled with an XSLT stylesheet. Because index blocks dynamically update content, the navigation menu stays updated. Common types of navigation menus include main menus, sub-folder menus, breadcrumbs, last and next page, site maps, and site indices. Cascade Server provides many out-of-the-box Stylesheets that are helpful in creating dynamic navigation menus. For more information on creating dynamic menus, please consult the technical intro guides.

E

External Link

An External Link is an asset that points to a webpage outside the system. Because it is an indexable asset, an external link is useful when there is a need to include external pages in a dynamic navigation menu. The external link records the desired URL, and the index block, when rendering the external link, will write out the URL path in the rendered document.

F

File

A File is considered any sequence of bits stored on the server's database as a single unit. Images, PDF documents, and text files are some common examples of files used in Cascade Server.

Files are created by applications and conform to a particular file format (associated with a file extension). A file is characterized by its filename, file extension, file size etc. Files can be uploaded and managed with Cascade Server, either directly, or by creating an Asset and selecting the file override option. When file override is used, the contents of that asset is overwritten with the contents of the selected file.

File Upload

File Upload is an option available when creating or editing files, templates, stylesheets and XML blocks. File Upload allows a user to select a file to upload from their hard drive, which will then be used to fill in the content of that asset. When creating an asset, if no name has been specified, the name of the file from the hard drive will be inherited.

This is especially useful when working with large pieces of text or markup - sometimes the content may simply be too large or cumbersome to edit in the web interface. File Upload allows users to take that content, edit it on a local machine using any text or markup editor, save changes and then upload those changes back into the system.

Folder

A folder within the CMS is a way to group related assets. Folders may:

- be nested
- have workflows assigned for assets they contain (via the 'Workflows' tab)
- have access restrictions (configured via the 'Access' tab)

While there are no restrictions on what types of assets can be placed in a folder, it is important to note that a well-reasoned folder structure is important for organizing and regulating access to assets in the CMS. In Cascade Server 4.0 and above, folders outside of the 'Home' area are known as 'containers'.

G

Graphical User Interface (GUI)

A Graphical User Interface (GUI) is a design method letting the user view and interact with information via different graphical elements, such as windows, buttons, screens, wizards, etc. Generally graphical interfaces require the use of a mouse to navigate through the different elements. Currently, Graphical User Interfaces are omnipresent and users rarely encounter a situation where they cannot use one. Both the Microsoft Windows operating system and Mac OS X present information via graphical user interfaces. An example of a non graphical user interface is the command-line interface (CLI) which is found in Windows by using a command prompt, where the user primarily interacts by typing commands with the keyboard. Also, shells under Mac OS X or Linux operating systems are considered non-GUI or CLI interfaces. Cascade Server is inherently a Graphical User Interface due to the fact that it is a web application. All things viewed through a web browser are necessarily graphical.

Group

A Group is made up of one or more users with common permissions. Groups are created and populated by Administrators or via a third-party authorization system (e.g. [LDAP authentication](#)). Each user in the system must be given membership in one Group, and can be given membership to multiple Groups. Group role membership is passed on to the user, meaning that the user automatically belongs to any roles that are assigned to the group(s) that the user belongs to. This is a convenient and often preferred way to easily change role memberships across groups of users. In addition to providing a way of assigning roles to multiple users, the group also can be specified in the folder access rights interface, giving multiple users read and/or write access to folders and folder-contained assets.

H

I

Index Block

An Index Block is a special type of block asset that returns a listing of assets from the CMS directory structure in the form of XML data. Assets such as pages, files, folders, external links, and even other blocks can be returned as XML content that describe them. An index block can even return the data content of multiple pages within a directory for use on other pages within the system.

Index blocks may be configured to index an entire site or a specific folder. It can be limited by number or depth of folders to index, or by number of assets. It can also be configured to index parent and/or child folders relative to a specific location. Any time the data content in the system changes (a page is added, deleted, renamed, moved,

etc.), the index block automatically updates, and all pages using the index block are also updated.

An XSLT stylesheet can be applied to the XML data of an index block, which allows for HTML elements such as dynamic navigation toolbars or page listings to be created on pages throughout an entire site automatically. Examples of dynamic navigation include main menus, sub menus, breadcrumbs, previous/next links, site maps, and site indices. The powerful combination of the automatically populated XML data inside of index blocks and XSLT stylesheets provides unparalleled ease of use with managing large sites.

J

K

Keywords

Keywords are part of the metadata content for pages, files, folders, blocks and external links, and are used by search engines to determine the relevancy of individual pages or sites.

Typically, keywords are a comma-delimited list of words used on pages and generated in such a way that the page can receive higher points in a search engine query. For example, if a page exists that deals with cheap widgets produced by ACME in Kansas, the list of keywords might look something like this:

widgets, cheap widgets, ACME, Kansas, ACME widgets

In Cascade Server, keywords are typically typed by a user or selected from a pre-determined list, under the Metadata tab, when editing an asset. However, the CMS administrator may choose to restrict keywords for certain assets, or add a special keyword field inline with the editing screen instead.

While general CMS users need only enter the keywords into the appropriate metadata fields, Cascade Server administrators will determine how keywords are placed within the source code of published pages by creating the appropriate markup in the template. Keywords are usually placed close to the top of the page for the benefit of search engines. The finished code will look something like this:

```
{CODE-1:xml}
```

Keywords can also be searched when a user takes advantage of the Search function within the CMS. In Cascade Server 4.0 and above, users may search the keywords field using the advanced search by typing the "keywords:" operator before the keywords to be searched into the "Metadata" box.

L

Link Checker

The link checker ensures that all pathways from the current asset to other assets in the system are valid, and is invoked after an edit is committed by clicking the 'Submit' button. If the checker detects any broken links (which includes any relative links, as these are not valid within the CMS), it forwards the user to a page detailing these errors and providing several options for dealing with them.

The first field for each broken link shown on this page gives the value of the link as provided by the user. The second field, 'Content', gives the value of any 'alt' attribute within the <a> element. The third field, 'Spec Violation', details the specific type of violation that caused this link to be detected as broken by the link checker, these can include:

1. Invalid Image Link - The source specified for the image was not valid or was relative.
2. Broken Link - Either the target given by this link is relative or the link is targeting a nonexistent asset.

The user then has several options for handling each broken link:

1. Asset Location - A radio button and file chooser allowing the user to choose an asset within the CMS as the target of the link or source for the image.
2. File Upload and Destination - A radio button, file upload input field and chooser allowing the user to choose a file on their computer to upload to the folder specified using the chooser. This file will replace the target of the broken link and the link will be automatically rewritten by the CMS to point to the uploaded resource.
3. Ignore - A radio button allowing the user to disregard the broken link.

After deciding how best to handle each broken link, the user then clicks the 'Submit' button. At this point, the link checker rewrites links and uploads files as indicated and commits the user's edits.

Log In/Out

Logging in is mandatory for any user wishing to use Cascade Server. Once logged in, a user can access and edit assets inside the system. Upon logout, a user's session will be terminated. If a user wishes to view and/or edit assets inside the system, he or she will have to log in again using their login credentials (username/password).

If a user does not log out and the system remains idle, the CMS will automatically end the session after a specified time, determined by the Administrator.

M

Manager

The Manager role is the second highest role. It has many of the powers of the highest role, administrator, but those powers are more limited in scope. Generally, the manager role is used to assign administrator-level privileges to a site or sub-site. Since roles are cumulative, managers have all of the powers of the roles beneath them (publisher, approver, contributor) as well as the power to bypass workflow.

Another key difference between the manager role and the roles beneath them is that they have access to the Administration area to control items like configuration sets, metadata sets, publish sets, data definitions, and workflow definitions. Putting a user in the manager role and setting up the appropriate access rights provides for a user that is fully autonomous in the CMS across assets, sites (targets), and administration components.

For Cascade Server versions 4.0 and higher, Managers may also create users with the role of Manager or lower, and assign access rights within the manager's own area. In addition, Managers can cancel any publish job where the manager user has read access permissions to the asset associated with the publish job be it a Folder, Page, File, Target, or Destination.

Key aspects of the manager role:

- Limited by system access rights
- Access to the Administration area
- Ability to bypass workflow

Message

Messages are sent to users by other users as well as the CMS for a variety of reasons and take a number of forms, including informative reports and requests that the user perform a specific task. They are viewable from the "Messages" tab on the dashboard. Some of the most common messages are:

1. Workflow Notifications - Sent by the CMS when a workflow enters a specific stage or is completed, these messages are essential to the proper flow of work between users in the CMS.
2. Publish Reports - Sent following a publish job, these reports detail the successful and un-successful components of a publish, summarizing jobs that succeeded, failed or were skipped and reporting broken links.
3. User Created Messages - Users also have the option of creating a message via the 'New' menu, which they may send to any user or group within the CMS. They may contain any content a user might normally be able to insert into a WYSIWYG field.

Metadata

Metadata is simply data within the CMS that describes an asset. Common fields such as title, summary, or keywords provide quick information about the content contained

inside of a particular asset. Because the system manages all assets as XML, the metadata that a user enters for assets can be used for display purposes on site pages. When used in conjunction with index blocks, an XSLT stylesheet can be applied to the metadata to create common HTML content listings such as a listing of recent press releases including the title and a quick summary of the press release content. Cascade Server uses two types of metadata: wired and dynamic. Wired metadata fields are the default fields included in the system:

- Display Name - the short name of the assets (typically four or less words)
- Title - the title of the asset
- Summary - the synopsis of the asset
- Teaser - a short enticing phrase about the asset
- Keywords - words or phrases about the asset
- Description - the description of the asset
- Author - the person or organization responsible for the asset
- Review Date - the date and time for the content to be reviewed with a respective workflow
- Start Date - the date and time the content should be made public
- Expiration Date - the date and time the content should no longer be made public
- Expiration Folder - the folder the content should be moved to upon reaching the expiration date

All wired fields leading up to the date fields are standard text-based fields that accept normal text data. The wired date fields: review date, start date, expiration date, and the related expiration folder field provide an interface for date metadata to be entered that will trigger the system to perform actions on the asset. The review date, start date, and expiration date fields instruct the system to carry out activities based on the date entered such as prompting users to review the asset, permit an asset to be published to the live web server, or un-publish an asset after a particular date respectively. The expiration folder allows the user to select a folder into which the expired asset should be moved.

Dynamic metadata fields are custom fields that can be created by an administrator and can include:

- custom text fields
- drop down boxes with preset option
- Boolean Yes/No values

Dynamic fields are especially effective in allowing users to filter specific assets via stylesheets based on the metadata values that describe them. For example, a user could tag a particular asset with some metadata that places that asset into a certain category that could then be referenced in a stylesheet.

Metadata Set

A Metadata Set is a system component that provides the interface for customizing the kinds of metadata fields that can be visible and/or associated with an asset. Since metadata fields can be broken into two categories, wired and dynamic, metadata sets allow administrators to choose which wired fields can be made visible to end users

and which ones will be turned off. Likewise, the interface allows for dynamic fields to be created that help to further describe an asset beyond the scope of what regular wired fields allow. Metadata sets also allow for administrators to customize the wired and dynamic metadata fields so that they appear "inline" in the editing interface for an asset instead of appearing in the metadata pane, making the fields more accessible to end users. Additionally, fields can be set to 'required' so that end users must enter acceptable values for metadata before saving the asset.

N

O

P

Page

The Page is one of the core asset types in the system. It represents the grouping of several items together including content, template, blocks, and more. Pages are used to publish out HTML to the website as well other formats including PDF and XML.

Pages can represent many different types of content on a website. Pages can be frequently asked questions (FAQs), newsletters, press releases, employee profiles, news articles, simple content, and anything else you might find on online.

The system uses pages and their respective metadata to dynamically create navigation link menus, site maps, alphabetical indexes, RSS feeds, and many other automated items. The aggregating of page content is a crucial aspect of efficiently managing websites.

Preview

Preview is a view in Cascade Server that enables the user to view a page as it will appear once it has been published. This is very useful in cases when a publish job is about to be performed, or a page edit was recently completed, and the user wishes to verify the absence of possible rendering issues. Once the preview is activated (this is done by clicking on 'Preview' in the action bar), the final version of the page will appear.

Production Server

A Production Server is a web server that delivers what is often called the "live site." It is typically available to the entire web and houses the most recent version of its respective site.

As opposed to a staging server, which might house untested and, indeed, broken components, the files served by a production server are generally assumed to have undergone extensive testing.

The production server is set up in an environment separate from Cascade Server. Pages, images, documents and other web-related items are published out to a production server using the CMS. Typically, the production server contains a web services platform which allows the testing environment to be hosted and viewed within the confines of a company or other limited network. Please contact your network administrator about setting up a production server for your Cascade Server content.

Publisher

Publisher is a role in the system and can be assigned to a User or a Group. The Publisher role allows the same actions as the Contributor and Approver, as well as allows the user to push content out to a live site, either by completing workflow that contains a publish trigger upon completion or by selecting assets he/she has access to and selecting to publish them. Publishers may be given read or write access to areas in the system. These access permissions are set by an Administrator at the folder level and are set for an individual User, Group or for All Users. A Publisher level user may:

- Navigate through the site structure (read or write access).
- View content (read or write access).
- Edit content (write access only)
- Create new content using Asset Factories (determined by the listing under the New menu)(write access only).
- Copy items (write access only).
- Delete items (write access only).
- Approve, Publish, or reject content in a workflow he/she is assigned to.
- Bypass workflows to publish content directly upon editing.
- Cancel publishing jobs he/she has initiated (4.0 and higher).

A Group can also be assigned a Publisher-level role with Cascade Server. When a Group has a Publisher role assigned to it, any Users within that Group may publish content through workflow steps assigned to the Group.

Publish Set

Publish Sets are Administration area components that allow groups of publishable assets to be published on-demand or on a schedule. Publish sets may contain files, folders, and pages.

A common use of publish sets is to implement the workflow publish set trigger so that when a piece of content has been approved, it is published along with all the assets related to it (as defined by the set). Publish sets are controlled by container-level access rights and can be limited to certain users.

Q

R

Reference

A Reference is a special Home area asset that allows one to create shortcuts to non-reference Home area assets. The reference, when indexed by an index block, will render content for the linked-to asset. Deleting the reference will have no effect on the referenced asset.

References are a great way of creating index block renders, as one is able to create a folder of explicitly chosen content that can be quickly indexed by an index block that is set to render the folder of references.

References are often used to have links show up in navigation menus in other folders than where the page actually resides.

Relative Link

Relative links allow path references to other pages or files be written in a way so that the file can be located by navigating from the current page, as opposed to starting at the root directory and following a long path (an "Absolute Link").

Relative links are often written like this:

This link will be interpreted so that when the link is selected, the browser will move "up" two folder levels from the current location and then find the styles.css file at that new folder location. For example, if the current page were located at the URL "http://www.hannonhill.com/one/two/three/page.html" and this link were clicked, the browser would open the resource "http://www.hannonhill.com/one/styles.css". However, if the user were viewing a different page at "http://www.hannonhill.com/one/two/page.html", the link would take the user to "http://www.hannonhill.com/styles.css". Thus, we see a pattern emerge: for each "." in the link, one folder will be removed from the path.

Relative links are called relative because the resulting resource is "relative" to the current location of the user. Absolute links are called absolute because no matter where the user is, that user will always be taken to the same location represented by that location.

Cascade Server treats relative links as broken links, because only absolute or fully-qualified links are considered valid. Relative links are reported as broken by both the link checker and publish link checker. The standard link checker runs when a page is saved; when running this link check, users are able to fix the broken link before completing the save action. However, in the publish report link checker, broken links are listed at the bottom of the publish report and a link will be provided to the page, along with the broken link text.

Relative links can be used in the CMS if needed. Some websites require files to be managed only on the webserver, instead of being managed through Cascade. If this is the case, a relative link within the CMS will help link to this externally managed asset. This case works since links from Cascade Server, while absolute before publish, are converted to relative links after publish.

Role

In Cascade Server, the abilities of a user are often defined by the Role(s) to which that user is assigned. These roles may be either be assigned explicitly through the user management system, or they may be inherited from the groups to which that user currently belongs.

The five roles in Cascade Server, in order of increasing ability, are:

- [Contributor](#)
- [Approver](#)
- [Publisher](#)
- [Manager](#)
- [Administrator](#)

Note that each role inherits capabilities from the role beneath it. For example, the Manager may do anything that the Publisher, Approver, and Contributor role may do, but not everything that the Administrator is capable of doing.

Cascade Server, when considering if an action by a particular user is permissible, will examine the highest role that user has been given.

RSS Feeds

An RSS (Really Simple Syndication) Feed is a way of syndicating web content in an XML format that is understood by a wide variety of programs, such as news readers, web browsers, and mail programs. There are many different versions of RSS, with the most popular being 0.91, and the most recent being 2.0.

An RSS feed consists of channels, which are then broken up into individual items. Both channels and items have a title, link, and description. The format of a RSS 0.91 feed is as follows:

```
{CODE-1:xml}
```

For a discussion of the various versions of RSS, see the [article on XML.com](#).

Cascade Server makes it easy to author RSS feeds with out-of-the-box XSL stylesheets that can turn an index block into an RSS feed syndicating items such as news, press releases, or anything else that needs to be syndicated. In addition, external RSS feeds can be aggregated using Cascade Server's XML feed blocks.

Cascade Server also uses RSS feeds to syndicate workflow information from the user's dashboard.

S

Search and Replace

Search and Replace allows users to quickly changes multiple assets at once. While searching within the CMS, Administrators may choose to replace certain content (words, phrases, etc.) with other content.

When a search and replace is performed, all assets using the search term are listed, allowing the Administrator to perform a global replace, or to select individual instances to replace the term.

Spell Checker

The spell checker, much like the accessibility checker and the link checker, is invoked after a page edit is committed by clicking the 'Submit' button. If there are any words that the CMS deems suspect, the user is forwarded to a 'Spell Checker' page listing the errors sorted by the field on the page in which they occur. For each error, the user is presented with a link (so that the user can see the error in context), and radio buttons offering four options:

1. Change To - allows a 'change to' either a most-likely value as determined by the spell checker or a user-supplied alternative.
2. Suggested - allows the user to choose from a list of possible replacements suggested by the spell checker.
3. Ignore - allows the user to disregard the error altogether. This is the default option.
4. Add - allows the user to disregard the error AND add the term to the dictionary.

Once selections have been made for each error, the system makes the appropriate changes and forwards the user either to the next enabled checker or back to the layout view of the asset.

Staging Server

A Staging Server is a web server used to test the various components of, or changes to a web site before propagating them to a production server.

In the CMS, a Staging Server would be represented as one of several destinations to which content can be published. In practice, after integrating a site and preparing it for launch, the user might first publish the site to a staging server in order to perform quality assurance review. If the published site passes quality assurance review, it would next be published to a production server, otherwise it would then be reworked to correct any problems and republished to the staging server for more testing.

Syndication

Syndication refers to repackaging web content in a way that makes it available for others to display dynamically. The best example of syndication is news sites that rely on Atom, RSS, or simple XML formats to repackage their news headlines.

With Cascade Server, it is easy to syndicate your web content such as news articles, press releases, or other frequently updating content. Simply index those items using an index block then apply the supplied RSS XSL stylesheet to create an RSS-compatible feed.

Cascade Server itself syndicates workflow information to the user as well from the dashboard. Simply click the RSS link on the bottom of the page to open the Workflows Waiting RSS feed, which will display a user's workflows waiting for action.

T

Target

A Target is both a relational and a functional entity in that it ties things together and it provides a function of its own. A target is used to represent a site or subsite, and acts as the glue that binds content to its destination(s).

Most of the time one target will be created to represent a website, with subtargets (children of the master target) created for the different templates within that site (html, xml, handheld, etc.) By defining a base folder (the main folder that contains all the content for the site), the "master" target ties together the entire site - all the subtargets and their templates. Each subtarget ties an individual site template (xml, for example) to the pages (and the pages' related files) that use that template.

Template

A Template is a fundamental system entity that defines the HTML/XML structure of page asset in the CMS. Moreover, templates are essentially XHTML documents that contain the standard HTML tags and CSS that create what is commonly referred to as the "look and feel" of a web page. In addition to the normal HTML elements, templates also allow for special system tags to be embedded that define the content regions of the document. These content regions are the areas where new blocks of content can be placed in a "building block" manner to create a resulting XHTML web page. Likewise, XSLT stylesheets can also be applied to these regions against a block asset to create XHTML content as well.

Once a template has been created in the system, a user may choose to edit the template again where they will be presented with an interface to perform these block/stylesheets assignments to the various regions that might be available in the particular template. Each region is designated by a special HTML comment tag or XML tag where MY-REGION-NAME is replaced with the name of the region. An important note to consider is that all templates must contain a region called DEFAULT where a page's "default" content will appear from the data that the end user enters into the editing environment for a page that inherits the template. If a DEFAULT region has not been created, the system will automatically create the region immediately after the opening <body> tag of the template document.

All templates must adhere to the standards that define well-formed XML documents. This includes, but is not limited to, maintaining lowercase tag names (i.e. <P> becomes <p>) and always appending opening tags with their closing tag counterparts (i.e. <p> cannot be left alone and must have an ending </p> tag after the paragraph content) unless the document type definition allows for the tag to be self-closing (i.e. with the closing slash. Documents that are not well-formed can be passed through an optional XHTML filter that will perform a conversion of the document structure to adhere to the XHTML standard.

For more information about well-formed XHTML documents, consult the W3C website below:

Text Block

Text blocks are basic blocks of content that can be reused throughout a site much like an XHTML block. Text blocks are not as widely used as their XHTML counterpart, because text blocks lack the standard WYSIWYG editor contained inside of XHTML blocks that allow for the creation of rich content with images, links, and standard text formatting options.

More often, an XHTML block will be the desired solution; however, there are appropriate times to make use of a text block instead of an XHTML block. These less likely circumstances involve blocks of content where an administrator desires the user to enter plain text only, without any formatting or images. In this case, a text block is the appropriate solution for the content region. The administrator can then style the text block with an XSLT stylesheet. Like all blocks, a text block may be attached to a template, configuration set, or page, and may be reused across multiple pages. A single change to the text block will be present across all pages of the site that make use of the block.

Transport

A Transport is an entity in the CMS that stores data concerning how published content is pushed out to the final publish location. There are two types of Transports: file system and SFTP/FTP transports.

- File system transports can push out content to a location on the CMS server's hard drive or to a mapped network location. In order to set the transport to a mapped network connection, the account used by the CMS server must have the appropriate privileges to write to and create new files in the network publish location.
 - FTP/SFTP transports push content to a remote server via the SFTP or FTP protocol. The account specified in the transport's settings must have appropriate privileges on the remote server to navigate through the folder structure, write, and create to ensure publish operations do not encounter errors.
-
-

U

User

A user in Cascade Server is one who uses the content management system. Cascade Server requires individuals using the system to authenticate upon login for security, logging, and resource management purposes. Each user has a user account with a User ID (username), password, full name, and email as well as group and role membership.

Every user is identified by the system with his or her User ID. A password is then used for authentication and the full name and e-mail serve as contact information for the

user. The group and role membership help to determine the privileges a user has in the system and help establish his or her access to various system resources.

User ID

User ID is short for User Identification and is also referred to as username. A User ID is a unique, sequence of characters used to identify a user of Cascade Server. User IDs are used to authenticate the user upon login and for identifying the user for the purposes of logging, access, and resource management. Usernames are chosen upon user creation, and may consist of numbers, letters, and the underscore (_).

V

Version

As changes are made to any asset in the system, Cascade Server keeps track of the changes in separate copies of the asset called Versions. These are accessible through the Versions tab of any asset in the Home area.

Each version has a timestamp that allows the user to see when the change was made. This functionality is also available through a workflow trigger as well.

A version may be either previewed, which shows the user what the asset looked like at that point in time, or activated, which means that the current state of the asset takes on the content of the specified version.

The versioning system is similar to a multi-level undo whereby you can revert to a previous version at a later date.

W

Workflow

Cascade Server ensures that content entered into the system is quality checked by mandating that the entered content go through a series of approval steps. These steps can be configured through the web interface.

A workflow, then, is simply a grouping of these steps. A step might have a name such as "Managerial Approval", "Final Approval", and/or "Re-Edit". These steps are connected by actions that provide a pathway for moving from one step to another step. Note that one step may have multiple actions leading out of it.

Once the workflow moves to a step with no other actions leading out of it, the workflow is considered to be complete/finished. The asset associated with the workflow is then no longer to be considered in workflow any longer and is returned to a normal state.

The workflows are managed in the Administration area of the application and are defined as XML.

Workflow Action

Transitions between workflow states (or steps) are enabled by Workflow Actions. An action is defined as a directed path from a source step to a destination step. Actions can be enhanced by attaching one or more workflow triggers. The trigger executes code while the transition is occurring from the source step to the destination step. Cascade Server comes with a series of pre-defined triggers, and custom plug-in triggers may also be added to execute custom code during a step transition.

Workflow Trigger

A workflow trigger is a plug-in that enhances an action in the workflow process by executing code as the transition from a source step to a destination step occurs. It encapsulates some system logic to accomplish a non-workflow related function.

More than one workflow trigger may be added to an action in a workflow. A trigger may be enhanced by a parameter, which is an optional element that further specifies the system logic that should occur.

WYSIWYG

WYSIWYG is an acronym for "What You See Is What You Get," and refers to Cascade Server's ability to see formatting while editing in the word processor.

The WYSIWYG Editor is a tool that allows users create and edit web page content directly in the browser, in a familiar, Microsoft Word-like environment.

Common functionality such as bold, italics, underline, and even table and image management, are all visually previewed for the user as the edits are made.

If finer control is desired, the user may opt to see the source XHTML "behind-the-scenes" and toggle between the visual and source modes.

In addition, an administrator may set CSS classes for the WYSIWYG Editor, allowing users to select various styles from a drop down menu. Administrators may also limit formatting options available to users on specific assets.

X

XHTML Block

An XHTML block is a reusable block of content that provides a rich WYSIWYG interface for editing content. Unlike the text block counterpart, an XHTML block allows for a wide range of HTML elements to be created inside of its editing environment extending to items such as images, links, tables, and bullet points in addition to other standard text formatting.

XHTML blocks are particularly useful for fixed regions of content such as headers and footers that need to stay constant among all pages within the site. A single change to

one of these blocks will be present across all pages of the site that make use of the block. The basic level of modularity improves the scalability of the website in addition to allowing for easier "quick edits" that cannot be easily performed if the content were to remain in the body of the overall template.

XML Block

An XML block is a reusable piece of content stored as well-formed XML. XML blocks are particularly useful when there is a large amount of XML that must be styled and included on one or more pages of a site. Once the block is created, it can be assigned to a page region in a template, configuration set, or page, and, if desired, styled with an XSL Stylesheet. A single change to one of these blocks will be present across all pages of the site that make use of the block.

XML Feed Block

An XML Feed Block is a block whose XML content is pulled from a web location. This can be useful when aggregating outside RSS links, or receiving output from dynamic scripts or web applications that produce XML. This is the main method by which Cascade allows external content to come into the system and be used dynamically within system pages.

This block has one parameter, the feed URL, which is the location that will respond with an XML document. Cascade Server will then take that XML content and populate the block with it. The block can then be styled using an XSL stylesheet and included in a page region just like any other block in the system.

XSL Stylesheet

An XSL Stylesheet is written with a set of rules defined by the eXtensible Stylesheet Language, and is used to transform data from an XML document into an XML document that can be presented in some meaningful manner.

In Cascade Server, XSL stylesheets are assigned to specific regions on a page along with a block (XML, XML Feed, or Index) to style the XML content from the block in a manner appropriate for that page. By using different stylesheets on different pages, the same XML content can be customized for multiple purposes.

XSL stylesheets can also be assigned to the default region of pages using structured data definitions. In such a case, the XSL stylesheet styles the user-supplied structured content in a way that is meaningful for that page.

Example XML document:

```
{CODE-1:xml}
```

By applying the following stylesheet to the XML document above, an HTML/XHTML page will be created with the heading "Book library:" A table will be created on this page where each row represents a book from the XML document. For each row, the first column will display the book's title, the second column will display the book's author, and the third row will display the book's isbn.

{CODE-2:xml}

Y

Z

Zip Archive

An archive is a file that contains other files packaged together. There are many formats that are used to package files in such a manner, the most widely known are probably the ZIP file format (cross-platform) and the Stuffit file format (Mac only). There are many other archive file formats as well. An archive allows a computer user to organize and transfer files easily. For example, Cascade Server allows the user to upload a ZIP archive into the system and then unzip it. This is very useful when uploading content for integration: instead of uploading each HTML file and image individually using the File Override option, they can be packaged into one file, uploaded once, and then the system will unpack it as if each file were transferred individually.

© 2001-2008 Hannon Hill Corporation. All rights reserved.