Confluence Site Setup

This document describes our web site setup: what is where and how it works.

Related Articles

- Developer Bible
- Version Numbers
- Developer Information
- Release Process
- Building Tapestry from Source
- Confluence Site Setup

Overview

Most of the web site and documentation (with the notable exception of the Javadoc API pages) are kept in Confluence.

Since the Confluence instance at https://cwiki.apache.org/confluence/ isn't capable of handling a lot of incoming requests, all wiki spaces are statically exported. The SiteExporter program is responsible for that. Once a page in Confluence changes, that page gets re-exported automatically.

How SiteExporter works

For more details see the SiteExporter README.

SiteExporter is a command-line Java program that is run hourly (currently at 19 minutes after the hour) from Apache's BuildBot. It makes a web service call to Confluence (to its RSS feed, actually) to get a list of pages that have changed since the last run, and the HTML-formatted export of those pages. For each, it post-processes the file (described below). Finally, SiteExporter commits all changed HTML files into Tapestry's part of the Apache Subversion repository, which (nearly instantly) makes it available to the public at http://tapestry.apache.org, and commit emails are sent to Tapestry's "commits" mailing list.

Attachments (to Confluence pages) are exported in roughly the same way.

Unfortunately, wiki pages with embedded images will export with <img> tags pointing to the origial cwiki.apache.org URLs, unless you manually change the image in the Confluence page to be linked to the exported static image file. This has been done for the banner image, but not every embedded image.

The time between saving a change in Confluence and seeing the result on the public site is at most 1 hour, depending on when you do it. If you save a change at 18 minutes after the hour you'll see the change in about a minute. If you publish it at 20 minutes after the hour then you'll have to wait almost an hour.

HTML files in SVN	https://svn.apache.org/repos/infra/websites/production/tapestry
Cache File	https://svn.apache.org/repos/infra/websites/production/tapestry/content/cache/main.pageCache
SiteExporter source	https://svn.apache.org/repos/asf/tapestry/tapestry-site/trunk
Velocity template	https://svn.apache.org/repos/asf/tapestry/tapestry-site/trunk/template/template.vm

Post-processing HTML Pages

HTML pages exported from Confluence are post-processed in several ways before being committed to SVN. Here are just a few of the things going on:

- Tagsoup is used to clean up the HTML.
- The breadcrumb links are updated.
- Empty paragraph (<p>) tags are removed from the top of the page.
- {code} macro output (code examples) are detected, and SyntaxHighlighter JavaScript links are added to the page when needed.
- {include} tags (when one Confluence page includes another) are detected, causing the *including* page to be regenerated autoamtically.
- {children} tags are also detected and handled

Manual Intervention

You can cause the *whole site* to be republished by deleting the main.pageCache file (above) in the subversion repo. This is usually only needed after changing the template.

Changing SiteExporter itself

Currently the SiteExporter source code is an unmodified copy of a program of the same name written by Dan Kulp for the Apache CXF project and also used by Camel, Geronimo, and Struts (and possibly others). It can be customized, but proceed with caution, because any customizations will make it harder to pull in future changes from the original CXF SiteExporter code. The CXF SiteExporter is likely to change as Confluence versions change.

To pick up changes to the original CXF SiteExporter code, just compare the Tapestry source code with the CXF source code.

Wiki Formatting Guidelines

- Precede annotation names with '@'. If the annotation name is hyperlinked, put the '@' character *outside* of the link: @[AnnotationType|http://... AnnotationType.html]

- The first reference to a type on a page should be a link to http://tapestry.apache.org/current/apidocs/... (or the component reference)
- Treat the page title as if it were an h0. element, and put top level sections within the page as h1.
- Page names as headings should have All Words Captialized.
- For other headings, only the first word of multi-word headings should be capitalized, e.g. "h2. Naming conventions" (following Wikipedia)
- Use `code` font for method and property names: `myProperty`, `someMethod()`.
- Use the default font for Class names (qualified or not).
- Use the default font for path names.
- Use {code} for listings, not {noformat}.
- Use {noformat} for console output.
- Images and diagrams should be small-sized thumbnails, centered, with no border.
- Use the Since and Deprecated macros to mark new or deprecated features.
- *Proposed: Each page should include explicit links to its child pages. Don't rely on the "Child Pages" links at the bottom, which don't carry over to the exported site.*
- *Proposed: In pages other than the User Guide pages, subsections that briefly discuss topics that are more fully covered in the User Guide should lead with a "Main Article: [Foo]" line, where Foo is the name of the page in the User Guide. Example: the "Template Localization" section of Component Templates*
- *Proposed: User Guide pages should generally start with a right-floated "Related Articles" box that provides links to related content in the FAQ, Cookbook, Cheat Sheets, etc. Example*
- *Proposed: The lead paragraph should generally lead with the title word or phrase in bold (following Wikipedia)*

Website structure

The Index page includes the Banner and Key Features pages as well as the blog posts. Most other pages are just plain pages and may or may not include other parts. In addition the Navigation, Small Banner and Footer pages exist.

Our SiteExporter template (described above) glues everything together. It adds the contents of the Navigation and Footer pages in the appropriate places and on all pages except the Index page. It also adds the contents of the Small Bannerpage as well as the breadcrumbs navigation.

> HLS: I've noticed that pages with footnotes that are combined with the {include} macro do not render correctly ... the footnote numbers and anchors reset back to 1 for each included page. Perhaps there's a way to fix that with the template?

Updating the template

You must be a Tapestry committer or otherwise have write access to the subversion repository (see link above).

To edit the template:

1. check out the SiteExporter source project (see link above)
2. find and edit the template.vm file
3. commit your changes