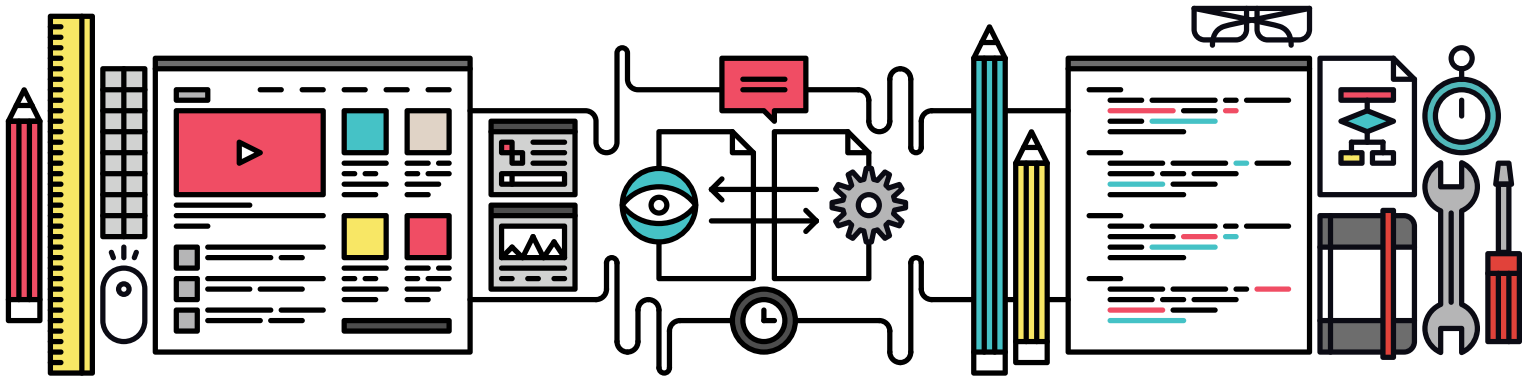
SEPTEMBER 2016

intercom

THE MAGAZINE OF THE SOCIETY FOR TECHNICAL COMMUNICATION



TOOLS OF THE TRADE

WHAT'S THE USE CASE FOR JEKYLL? 6

COLLABORATING AND CONTRIBUTING IN GITHUB 12

GIT HAPPENS 16

GENERATING API DOCS AUTOMATICALLY FROM THE SOURCE CODE 18

CREATING AUTOMATION TOOLS FOR WRITING TEAMS: A PROGRAMMER'S PERSPECTIVE 24

USING DATA-DRIVEN SYNTHESIS TOOLS TO AUTOMATICALLY GENERATE CONTENT 28

AN INTRODUCTION TO SPHINX AND READ THE DOCS FOR TECHNICAL WRITERS 34

TECHNICAL WRITING SUPPORTED BY A PRODUCT LIFECYCLE MANAGEMENT SYSTEM 40

THE BENEFITS OF DITA AUTHORING WITHIN A COMPONENT CONTENT MANAGEMENT SYSTEM 42

The Benefits of DITA Authoring WITHIN A Component Content Management System

shutterstock.com/ Flat Design

By KEITH SCHENGILI-ROBERTS

IT IS RARE for documentation teams to not use some form of database or repository for storing their technical content. A documentation team of any size will generate significant amounts of content over time that eventually outgrows shared network folders. This is doubly so for any documentation team that moves to a topic-based writing environment like DITA, as the concept of topic reuse relies upon there being a significant amount of previous content to leverage.

What I have observed in documentation groups over the years is that there often comes a point where any further growth depends on moving away from a simple file folder structure or database to a content management system; and more specifically for DITA-based content, a Component Content Management System (CCMS) where the "components" are individual topics.

When a documentation team is searching for a CCMS, they often do not know about all of the features that typical mature CCMSs have to offer. Moving to a CCMS is an opportunity for a documentation team to reinvent themselves and improve their processes, not to ossify existing and outmoded practices. When crafting a request for purchase of a CCMS, think about where you want your documentation team to be in five years and find a CCMS that can get you there. The following provides a brief survey of the types of features and benefits that can be gained by moving your DITA documentation processes over from a file system to a CCMS.

Versioning of Content

Content versioning is a feature common to most content management systems, although not to most file systems. Originally created for software development, versioning allows users to "check out" content for authoring and "check in" any changes. This practice is for programmers, as it allows them to revert to a previous version of their code to help root out bugs. Similarly, a typical CCMS will have versioning capabilities that ensures that only a single technical writer can work on a given topic at any time, and will also register who made an edit and allow for a quick comparison between versions. Just like a programming environment, any documentation "bugs" that are discovered in a later version of a topic can be reverted to an earlier version without problem. A CCMS should also include the ability for authors to comment on the topic- and map-level changes they are checking into the system so that someone else can tell at a glance what changes occurred and when.

Workflow

Another feature mature CCMSs provide is workflow. In addition to being able to check in/out versioned content, authors can also notify the system when they consider their work on a topic to be complete. Depending on how the workflow is set up, the topic can then be routed automatically for someone else to review and approve, such as an editor or product manager. Automating workflow introduces many ways to improve documentation processes and quality. In an engineering environment, subject matter experts (SMEs) may be assigned within the system to write content, which is then "polished" by a technical writer, and then sent to another SME for approval. This type of workflow implies that roles can also be assigned to contrib-

utors, helping define who can do what types of actions with particular content.

The ability to create workflows also sets up the possibility to schedule deadlines for when work is expected to be completed. A robust CCMS will be able to include not only the final deadline for when all of the topics within a document are due, but also various stages in between in order to help content contributors stay on track, and to notify them when they are falling behind schedule.

CCMS workflow can ensure that all topics within a document have been verified prior to publication, and that internal quality, external requirements—like a medical standard or legal standards—are met. In addition to being able to route content for review, a robust CCMS will have other workflow processes that can be triggered, such as automatically routing the finalized and approved version for localization purposes.

Measurement

Topic-based authoring allows for much more finely-grained measurement of content production. No longer are documentation managers restricted to only measuring when a document is completed and published, but they can now get a better understanding as to how long it takes to produce content. This is critical to know when planning ahead for future documentation releases. A good CCMS will enable a documentation manager to retrieve a wide range of metrics, allowing them to not only effectively measure the return on investment (ROI) for purchasing a CCMS, but also to understand ongoing production and quality issues. As an example of the latter, it should be possible to measure the topic reuse rate, providing the documentation manager information on whether existing content is being properly leveraged in the production of new content. Other types of production measurements, many of which relate to content quality, can also be measured effectively using a CCMS, including:

- Determining whether or not there is an optimal ratio of topic types being used (e.g., ensuring that there are enough task topics or troubleshooting topics in your user manuals)
- Establishing readability metrics for individual topics
- Tagging consistency throughout your document set
- Checking quality and consistent use of reused content

If a CCMS includes workflow features, it should also be possible to visually chart progress and how much work has been finished for a completed document. This type of function is critical in Agile-based development environments, making it possible for a documentation manager to attend Scrum meetings and show the velocity of documentation progress.

Localization

Localization savings are often a key driver in the move to DITA and a CCMS, so when choosing a system, you want assurance that it manages localization well. Content reuse in the source language equates roughly to localization reuse, guaranteeing significant cost savings from the start. But there are other key features that can help further maximize your localization budget. First, ensure that it works natively with Unicode character sets, as this reduces costs that could otherwise be incurred from having to deal with code-page issues when a character in one language is displayed as another. Confirm that the localization process is easy to use, so that packaging up content to be translated and importing the localized content back into the system can be handled by someone who may not necessarily be skilled in translation. In my experience, it is rare for a firm to have dedicated localization coordinators whose sole job is to handle this process, so it needs to be simple and straightforward while also providing tools for "power users" who can verify the quality of the translation. Look for flexibility in translation formats: DITA and XLIFF as packaging formats for translation firms to work with, and the ability to handle other XML formats such as MathML and SVG easily.

Similarly, beware of CCMSs whose processes are based on proprietary localization formats. This helps avoid the possibility of vendor lock-in, and any process that transforms your XML to another format and back is error prone. A CCMS-based localization process should ultimately deliver on the promise of cost savings and be easy to use and robust enough to grow with your localization needs over time.

Summing Up

A typical commercial CCMS will include most of the features mentioned here and more. While it is possible to use DITA within a small team using a file folder filled with topics, documentation teams really begin to shine when they have a CCMS optimized for use with DITA.

Read CCMS case studies, go to online DITA forums (such as those on LinkedIn and Yahoo!), and visit vendor websites to get a better understanding of what is available. There are several hundred firms using DITA worldwide, so reach out to your colleagues and learn what they have to say and what they recommend. And when selecting a CCMS, do not focus strictly on cool functions, the UI, or even the price of the system; instead think of where you need to take your documentation team and what your users will expect of you five years from now.

KEITH SCHENGILI-ROBERTS *is a DITA specialist at IXIASOFT, a DITA evangelist, and a technical writing nerd. Keith is an award-winning lecturer on information architecture at the University of Toronto's School of Continuing Studies. He is an active member of the OASIS DITA Technical Committee and can often be found presenting at conferences, working with customers, researching how DITA is being used, and sharing those results with the DITA community. Keith's popular industry blog* DITAWriter.com *has become a focal point on DITA resources and best practices. Connect with Keith on Twitter* @KeithIXIASOFT.

STC SUMMIT 2017

SAVE THE DATE

WASHINGTON, DC IN NATIONAL HARBOR, MD

7–10 MAY

TECHNICAL COMMUNICATION SUMMIT '17

STC's 64th Annual Conference