

**Learning from the writer Mark Twain**

**Applying his writing advice today**



# Communicator

The Institute of Scientific and Technical Communicators  
Winter 2012

**Read about Technical  
Communication UK 2012**

**Develop your API and SDK  
knowledge**



**Manage projects with  
Scrum and PRINCE2**

**How to apply copyright law  
to technical communication**

# Avoiding casualties in content delivery

**Don Rasky** reviews how Mitchell enhances content quality and delivery by integrating MadCap Flare with Microsoft SharePoint and TFS.

If you filed an auto insurance claim in the United States, the chances that Mitchell International was behind the scenes are good. Our company is a leading provider of technology, connectivity, and information solutions to the property and casualty claims, the collision repair and the workers' compensation industries. Currently, we process over 50 million transactions annually for 300-plus insurance companies and for more than 30,000 collision repair facilities throughout North America.

To ensure that customers can effectively and efficiently use our software services to process their transactions, we deliver online Help and print documentation developed using MadCap Flare in combination with Microsoft SharePoint and Team Foundation Server (TFS). Together these products provide the single-source publishing, collaboration, and version control we need to deliver timely and accurate content across multiple formats.

Today, we have integrated Flare, SharePoint and TFS into a single system so it is now difficult to tell where one application stops and another begins. Implementing this integrated solution was a three-stage process for our team.

## Migrating from multiple publishing tools

For many years, we relied on two products to develop our online Help content and two additional software tools to create our print deliverables. It became obvious that manually updating four different systems and ensuring consistency was difficult and time-consuming.

A few years ago, we began working with fewer writers and faced a corporate mandate to reduce costs and improve efficiency. We had been evaluating different authoring and

publishing tools for three years, and we knew that it was time to decide on the one that would help us meet our company's objectives.

As part of our side-by-side testing of tools, we discovered that when we pulled in information from most products, we could not use the content right away. By contrast, MadCap Flare pulled in the information so fast and so cleanly that there was very little involvement needed from our part to make it work. This quick turnaround and the added efficiency of single-source publishing led us to choose Flare.

The initial transition reinforced our decision. We had more than 40 online Help projects alone to upgrade, and Flare allowed us to complete most of the conversion in a single business quarter while simultaneously managing our normal workload.

## Streamlining with single-source

The days of manually updating a variety of documents in different formats and programs throughout the year are now in the past. Through Flare's single-source publishing functionality, our technical writing team can now update information in a single Flare project and then use conditional tags to publish online Help and different print documents as PDF files.

We also have made extensive use of conditional text and snippets, so that changes can be made everywhere in the Flare project. This was particularly helpful when we had to change a logo. We were able to change one snippet, and automatically update more than 100 locations in the project.

Through such efficiencies, project tasks have gone from taking days to hours and from hours to minutes using Flare. As a result, we have been able to cut content updates for our different software versions in half.

## Collaborating with SharePoint

Once we had implemented Flare, we next wanted to eliminate the challenge of managing multiple copies and wondering whether we were using the most recent version. Therefore, the next step was integrating it with SharePoint, which we use to pull in information and map content.

The integration lets our writing team access external resources, explore synchronised files, and manipulate content externally. Once mapped with SharePoint, shared information can be pulled into Flare projects and published through SharePoint.

We can pull content in with a couple of mouse clicks and manage the mapping from the

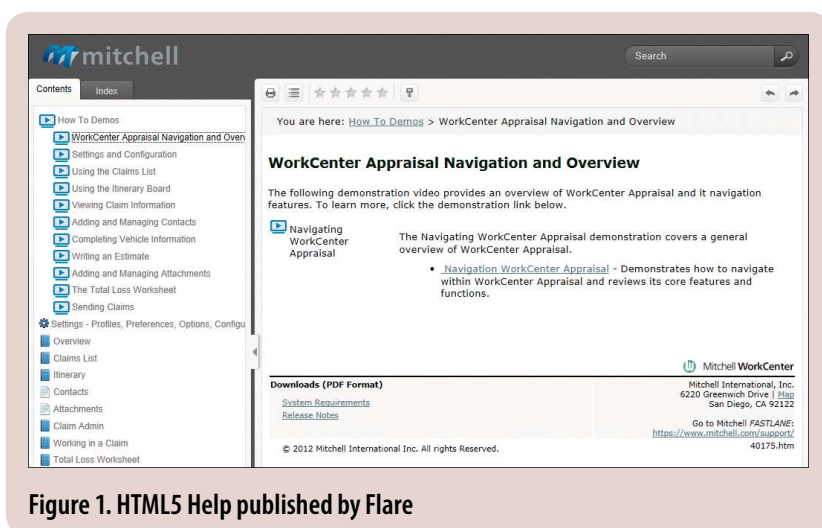


Figure 1. HTML5 Help published by Flare

**Editor's note:**

The case study chronicles events, advances, and implementations that took place over a three year period (Flare 2009-10, Feedback late 2010, SharePoint 2011, TFS and automated builds/publication 2012).

project toolbar. There is no need to question whether we have access to the most current content or if something is ready for production.

**TFS adds robust source control**

More recently, as the number of projects we manage grew, we added Microsoft TFS to handle version control. It was a smooth process to integrate it with Flare and SharePoint, and we quickly began seeing the benefits of source and author control.

Now, once an update is complete, an author uses Flare to check project changes into TFS. Should a problem arise, TFS also allows us to know what happened and who was working on it, so we can address the issue quickly.

We also realised notable efficiencies. With TFS and the automated publishing features in Flare, such as batch targets and command line builds, we can integrate Help with the product 30 to 60 days earlier in the product lifecycle. Because we publish directly to development, we are able to get our Help systems out faster than ever. We make sure all the critical pieces are in place up front and then we simply add content as we move along.

Another benefit with the integration of Flare, TFS and SharePoint is that this is giving our writers more freedom in publishing short how-to demos in the Help topics. Previously, a hosted external system did not allow for source control access. Today, our team can publish Help topics, which have demos mapped to Flare, and release them through TFS.

The integrated system has allowed for an automatic build and publishing option that is saving us considerable time. Recently our writing team presented an internal company demonstration where we were able to show:

- More than a 90% reduction in end-to-end processing times for building and deploying product Help to the Web.
- A 75% reduction in process time for producing our content set overall.
- An 80% reduction in processing times for

a complex reference set created by process partners and published by our team along with product Help.

- An overall reduction of 82% in our process times compared to process times prior to implementing Flare with TFS and SharePoint.

**Enhancing the user experience**

One benefit of the efficiencies we gained is that it also gave us more time to focus on enhancing the customer's experience.

Most recently, we began taking advantage of two new features in Flare 8.0: HTML5 output and a ribbon interface.

A new product our company launched provided the opportunity to build a new Help system, which we then published on the Web as HTML5 Help. The user interface of the HTML5 Help looks very much like the new product, providing a seamless experience.


We are also using the streamlined ribbon interface in Flare 8.0. Consistent with the Microsoft model, the ribbon has tabs at the top with commonly related commands and topics. The interface makes navigation faster and more intuitive.

Additionally, we use two other products from MadCap Flare to optimise the quality of our Help.

With MadCap Capture, we use different tools to add graphic elements that address the different learning methods of users. For example, additional images help the visual learner whilst bubble captions in the images benefit the reading learner, and pop-up thumbnails assist the kinaesthetic learner. It lets us make our Help visual without taking up a lot of screen real estate. Because Capture is integrated with Flare, it is easy to embed these graphics in our Help systems.

We also use MadCap Feedback Server to track users' experiences with the Help by reporting where it is getting the most hits, which topics people are going to, which search terms are not getting hits and need to be improved, and how much customers are using demos in the Help.

With Feedback Server, we are getting insight into what our customers really need, so we can then hone our content to support them.

To sum up, our integrated publishing solution is enabling us to put out more content, more completely, and more accurately than at any time in the history of Mitchell. Most importantly, it is allowing us to help our customers perform their jobs more efficiently using our products. 

**Don Rasky** is a senior technical writer at Mitchell International. With nearly 20 years of experience as a technical writer, he led the adoption, implementation, and automation efforts for MadCap Software at Mitchell. E: don.rasky@mitchell.com W: www.mitchell.com

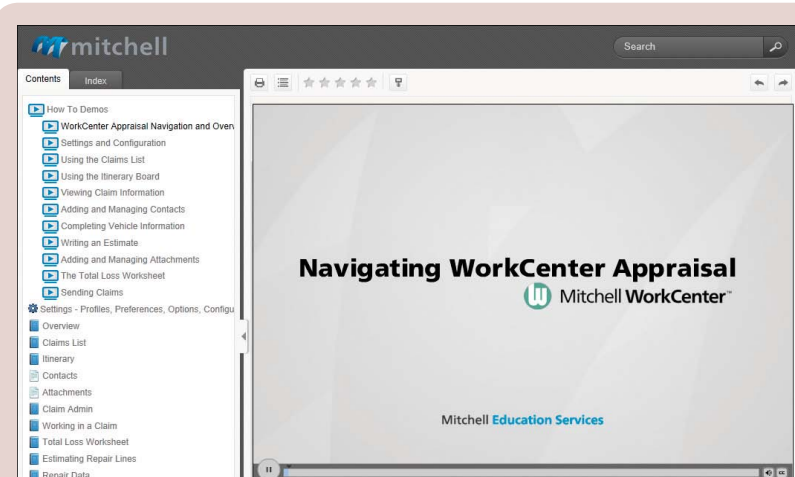


Figure 2. HTML5 Help with embedded How to Demo