Everything
You Need In One Box

ComponentOne Doc-To-Help 2009 has everything you need to author, edit, and publish documentation your end-users will love.

Author: Create content once using your favorite editor (e.g., Microsoft Word, Adobe Dreamweaver, or Doc-To-Help's new built-in XML-based editor).

Edit: Enjoy the convenience of visual project management features such as a topics organizer, related topics editor, and drag-and-drop linking. Plus, save time with automatic Table of Contents, Index, and Glossary generation.

Publish: Generate a combination of outputs with just one-click (online Help, platform-independent Web content, and press-ready manuals)—without changing or reformatting content.

This unique combination of editing, single sourcing, and publishing technology translates to significant time savings and the highest level of quality. Since Doc-To-Help is the complete documentation solution, your need for additional tools has been eliminated.

Key Authoring Features:

- *NEW* Built-in XML-based editor
- Ability to work directly in Microsoft Word, FrontPage, and Adobe Dreamweaver
- *NEW* Microsoft Word to XHTML converter
- Content variables (reusable content "chunks")
- Ability to add multimedia (especially ComponentOne DemoWorks™ Flash files)
- Style sheet / CSS editor
- Team Authoring Support
- Adobe RoboHelp, HTML Help, and WinHelp converters
- *NEW* Reusable project settings

Key Publishing Features:

- Standard Help formats (HTML Help, Help 2.0, WinHelp, JavaHelp)
- Best Web-based Help available (ComponentOne NetHelp®)
- Press-ready manuals (PDF or DOC)
- Storable output settings (i.e., themes, conditional content, custom TOC)
- *NEW* Embedded Help for .NET applications (ComponentOne DynamicHelp™)
- Error detection and warning

Add DemoWorks for More Multimedia!

ComponentOne DemoWorks, a great companion to Doc-To-Help, gives you the power to produce professional quality demonstration videos without multi-media expertise. You can record any screen activity and produce video in formats such as animated GIF and Flash. Unique frame-by-frame video and narration editing makes "one-take" demo creation a reality.

Modern Office 2007-Inspired Interface

Impressive Web-based output

Doc·To·Help 2009
ComponentOne®

brought to you by
Component**One**®

ComponentOne Sales | 1.800.858.2739 or 1.412.681.4343

Download free trials of Doc-To-Help and DemoWorks today.
www.doctohelp.com/everything

Embracing the un:

When the Community Runs the Event

By Anne Gentle, *Senior Member*
and Janet Swisher, *Senior Member*

hile Web 2.0 has many definitions, it is fair to say that Web 2.0 involves embracing user-created content and the communities that emerge around that content. One aspect of being a member of a Web 2.0-enabled online community is the real-world meetings that can happen at professional conferences or even networking events in your same town or city. User groups or focus groups are one type of real-world meeting, with a single goal in mind.

Unconferences and BarCamps are another type of real-world event where people with similar interests and goals can get together to share information. A third type of community event has just emerged and this article highlights aspects of this new BookSprint model because of its relevance to technical writing. This article describes the authors' experiences with participating in these types of events, and in particular the FLOSS Manuals BookSprint for the One Laptop per Child (OLPC) project held in August 2008. Anne Gentle was an organizer of that event, and Janet Swisher was a participant.

Unconference or BarCamp

An unconference or BarCamp is usually defined by its lack of definition—attendees gather at one place and time and offer their knowledge as presentations or demonstrations, which are then placed on a white board (or in a wiki) in certain time slots so that attendees can find sessions to attend.

BarCamps have started in different cities around the country and around the world as a way for like-minded technical people to give impromptu presentations about topics that the group votes they want to learn. These are user-generated conferences: if you attend one, you should be as ready and willing to stand up and present as you are to sit down, take notes for future or simultaneous blog entries, and listen. Anyone can start and organize a BarCamp by using the BarCamp wiki at *www.barcamp.org*.

The term BarCamp originated as an open-invitation alternative to the invitation-only participant-driven FooCamp sponsored by O'Reilly Media. In programmer slang, "bar" follows "foo" as automatically as B follows A.

How can you plan for such an unplanned event? The attendees are the first forming factor. For example, at a BarCamp that coincided with South by Southwest Interactive, it seemed that all the presenters knew one of the main organizers (open source advocate and strategist William Hurley).

Another major event involved planning an unconference as part of the DocTrain conference. Anne Gentle had little prior experience planning such an event, but she received encourage-

ment from Lisa Dyer, who was going to attend the unconference and encourage others to do so. With two people expanding the invitation list to include like-minded individuals, the unconference had some of the most interesting attendees and presenters in the field. One person even said it was the most useful session of the conference to him, a high compliment considering that it was embedded within DocTrain, an extremely valuable conference for a devoted group of conference-goers.

Originally the unconference started online with the presenters and presentation ideas listed on a wiki page. Then on the day of the unconference, manually transferring the "presentation" titles from the wiki page onto a white board put it in front of attendees by placing it outside the unconference room about an hour before the start time.

The unconference had about 15 attendees to start, with people coming and going, which is the expected ebb and flow for an unconference. There were presentations about wikis, open source for documentation, blog statistics, and even a mini-workshop session about wikis in the workplace, all infor-

mally presented with plenty of questions and answers flying. The only regret was that the time slot conflicted with the Vancouver STC chapter meeting being held in an adjacent conference room.

Another example of the unconference format is found in next year's Writing Open Source conference, a newly formed conference around open source projects and their documentation teams. While the first day's schedule contains talks by experts, the entire second day is dedicated to an unconference for all projects to share their experiences.

What Is a BookSprint?

A BookSprint is a community writing event where participants, either in one location or participating remotely, collaborate on a book or book set.

The term *sprint* comes from open source software development, where a *code sprint* is an event where programmers meet to focus on writing software for a project. The term is also used in *agile* software development processes, where it refers to one of a series of short phases of development, usually about a couple of weeks long. Sprints in open source development are typically shorter

(a day to a week) and more ad hoc than in "agile" commercial development.

Both an unconference and a sprint can be either a standalone event or connected to another conference or meeting. Both are driven by the community's need and interests. In an unconference, the agenda emerges directly from the community without mediation. But in a sprint, the agenda is usually outlined ahead of time by organizers.

The OLPC BookSprint was a weeklong concentrated effort of technical writers and subject matter experts getting together to create manuals for free, libre, open source software products: the "XO" laptop hardware, the "Sugar" user interface, and the "activities" or programs that come with the laptop. The BookSprint became a workshop where subject-matters experts (SMEs) shared their knowledge of the products, and writers shared how to write good user documentation. It was also a social experience where attendees formed a community of shared common goals and experiences. For the OLPC BookSprint, the attendees gave (up to) a week's time to be curators of information housed in wikis and websites everywhere, bring-

ing it all together in the FLOSS Manual TWiki implementation for online viewing or gorgeous print output.

Tools for a BookSprint

The tweaked-out TWiki implementation at FLOSS Manuals (*www.flossmanuals.net*) let us do collaborative authoring with a WYSIWYG HTML editor and a drag-and-drop chapter arrangement tool that also gave the status of each chapter, such as "Needs Images" or "Needs Proofing."

FLOSS Manuals also has a PDF engine that gives print-ready PDFs using CSS 3 styling, widow and orphan control, and a remix capability that enables any user of the wiki to drag chapters from multiple books and create their own book, either by outputting to PDF or HTML. HTML files can be copied onto a website or API code that can be embedded in a website.

Organizing a BookSprint

Much like planning a set of end-user documentation for any product, a BookSprint must have a known scope for the tasks and end-users that the documentation will target. We used an OLPC email list in the two weeks leading up to the sprint describing an outline and the tasks we would attempt. We discussed the scope—such as, whether to include programming examples for extending Sugar (we did), whether the School Server configuration was within our purview (it was not), or whether to include step-by-step disassembly and repair instructions for the laptop (we did not). This enabled the team to start writing after outlining. The team leaders, Adam Hyde and Anne Gentle, had a lot of discussion about who the target audience was for these books, which was an important part of scoping out our work for the week. After discussing via Skype and email, and vetting the decision with mailing-list interaction, Adam and Anne decided that the audience included the combination of students, parents, and teachers. Students, typically ages 6 to 16, may not actually refer to manuals, but they should be able to read them easily if they do. Parents and teachers are more likely to look at the manuals

and need orientation and reassurance about this unfamiliar computing environment. Other potential audiences, such as hobbyist enthusiasts, were not directly addressed.

By the start of the OLPC BookSprint, the selected participants had a definite plan and schedule in place. Monday, Tuesday, and Wednesday were heads-down writing days from about 9:00 AM until our stop time at 6:00 PM each day. Each day, both local and remote participants called in for a conference call, allowing the local writers to stay in touch with the remote contributors and find out if anyone was stuck or had questions. Most of the Sprint took place in one room where all the writers gathered and could ask and answer questions across the table. It was so valuable to have OLPC and Sugar experts on site, as will be described in the case study beginning on page 9.

The host location also provided other "quiet" areas that writers could go to when heads-down writing was needed. Local writers came to help as well, by working on the style guide, editing according to the guide, and also testing instructions after they were written.

To keep up the energy and workflow, the organizers made sure it wasn't all work and no play. Sunday night involved a coffee shop meeting with the Austin XO-users group (Austin locals who own XO laptops and support the OLPC project). Wednesday night the writers enjoyed a poolside cookout.

Nearly all the writers stayed late Thursday night, sprinting to the finish. Friday was a clean-up day, proofing print-outs of all seven books.

Funding

With attendees coming in from around the world and needing to sleep and eat while they wrote, it was necessary to finance their accommodations and travel. With three matching $1,000 support pledges from vested organizations (RedHat, OLPC, and an individual donation to SugarLabs), plus the fundraising efforts of FLOSS Manuals, we were able to provide travel and lodging and some meals for contributors. With five or six in-person contributors and at least

twice as many contributing remotely, we were able to fund the BookSprint for about $4,000 total. There are two books for sale as a result of the BookSprint, and a two-Euro markup on each book should help pay for such costs for future events.

Invite List

Selecting attendees by invitation or allowing for organic invitations is an important early decision. Selecting participants will drive the choice of a convenient date and location. In the case of our FLOSS Manuals BookSprint for OLPC, a flurry of activity on several mailing lists initiated the desire for a BookSprint. Invitations stemmed from those who were initially interested but also from community connections already made among OLPC enthusiasts.

Location and date selection go hand in hand because some locations are not going to be available for certain dates. Travel time, distance, and cost, as well as visa considerations are part of balancing this equation. When someone needs a visa to come to the United States to participate in such an event, at least a month's lead time is necessary.

All participants may be asked to sign a release by some host locations so that they will not be liable for any unfortunate result at an unconference or BarCamp. Signing a release should be fine—nearly all this planning is happening on good faith anyway, so suing would be poor form.

Why Hold an In-person Event?

It's an interesting day and age when technology like wikis and Second Life enable virtual meetings to be just as useful and productive as real space. But sometimes even an online community needs in-person interaction to be more efficient by communicating faster, and in person you can understand nuances only available with body language, personal space, and so on.

Also, when participants enjoy interaction with others with similar interests, these kinds of events really motivate and reward contributors. This interaction is likely to spur more activity later, by bringing together key players.

CASE STUDY

Participating in a BookSprint

The OLPC BookSprint brought together several overlapping communities:

- hard-core, full-time participants in the OLPC, SugarLabs, and FLOSS Manuals projects
- Austin XO enthusiasts
- Austin technical writers interested in open source software

Janet Swisher is in the third camp; I've blogged and presented on why technical writers might want to contribute to open source projects, and I work for a company that publishes some of its software as open source. I didn't have much prior experience with OLPC, other than having bought an XO for my nephews the previous Christmas. However, I knew that OLPC was a cool project and the BookSprint was in my hometown. On top of that, Anne Gentle, a blogging tech writer who I knew through the local STC chapter, asked me to participate. How could I not?

I walked in on day two of the sprint, having briefly seen my nephews' XO laptop nine months before, and having downloaded the XO emulator for Windows the previous day. Lacking domain expertise, what I could contribute was technical writing expertise. My first task was to improve the style guide for the XO manual. I reorganized it a bit and added a few items that others in the room said they wanted guidance on. My opinion about style guides for open source projects is that the fewer rules writers have to remember, the better. However, this parsimony must balance against readers' need for consistency.

After that, I reviewed chapters in the XO and Sugar manuals that were finished or nearly so, making sure they complied with the style guide, and editing anything I thought needed it. I tested procedures with one of the XO laptops scattered around the room. Apparently, this made an impression on David Farning, a Sugar programmer who attended the BookSprint and who later wrote:

> I realized this was not just a couple of programmers trying to throw together a wiki as I watched Janet Swisher intensely studying the XO's battery. Turns out she was trying to determine if the "installing the battery" section could be misread. From my experience, a programmer would have said, "If they can't figure out how to put the battery in, what's the point of a fine manual?"

One factor that I did not have to worry about for the BookSprint was writing tools. The FLOSS Manuals website, being a wiki with a WYSIWYG HTML editor, was dead simple to use. The site even kept track of which chapters were currently being edited by whom, so that we wouldn't step on each other's changes. There were times when a chapter that I wanted to edit was one of several that the system said was being edited by another person. Since we were sitting in the same room, I could just say, "Hey, are you still working on that chapter? If not, can I break your lock?" For certain sections, I decided that the most appropriate HTML structure was a definition list, which was not supported by the WYSIWYG editor. Being familiar with HTML, I was able to switch into "code" view and insert the tags manually. However, within a few minutes of my mentioning this to Adam Hyde, he had modified the editor, and buttons for the needed tags appeared on the editor's toolbar.

Another benefit of co-location was that when I came across a note that said, "Walter, check for accuracy," I could just say, "Hey, Walter," and Walter Bender (founder of SugarLabs, former president of OLPC for Software and Content, and former executive director of the MIT Media Lab) would walk around the table to look over my shoulder. Talk about immediate feedback!

Bringing together communities with related interests can lead to unexpected synergies. The Sugar API documentation was not within the scope of the BookSprint, but when it came up in conversation, I was able to point the Sugar folks toward a tool for wikifying it, to encourage programmers to contribute. Also, at the Wednesday cookout, my husband, who is a musician and a graduate student, got into a long discussion about computer-generated music with Adam Hyde, who is a "sound artist" as well as the FLOSS Manuals impresario. (OK, that has nothing directly to do with the BookSprint, but it wouldn't have happened without it!) In comparing this and other face-to-face meetings of virtual communities, I can see the following benefits:

- Community members get to know each other "for real." If you've ever worked on a virtual team, you know that you work much better with team members you've met face-to-face. Putting faces to names and personalities to email addresses helps the team or community work together virtually in the future.
- A concrete time and place to work increases productivity tremendously. Volunteers working asynchronously tend to lack urgency; there is always some other priority that pushes the volunteer work off to "someday." At a sprint, the work must be done here and now. And, as I mentioned, feedback can be provided within a few seconds instead of hours or days.
- The event helps put the project in perspective for participants. When a project is coordinated online, participants can get swamped by details. A sprint helps focus priorities and helps participants see how their pieces fit into the bigger puzzle.
- It's fun! People feed off each others' energy and excitement. Jokes are cracked, camaraderie develops, and friendships form.

Sure, maybe because extroverts and social animals really thrive on this kind of interaction, they will be the first to sign up. But there are good reasons for and benefits to occasional, planned events such as this one. There is a role for face-to-face interaction in online communities by improving their connections and communications.

Real-world Enthusiasm, Real-life Interaction

While gas prices peaked this summer, making travel more expensive, online interactions continue to get easier and faster all the time. Despite these seemingly related forces toward online-only collaboration, an in-person event enabled by the ease of coordination and organization that the Internet offers became an innovative method for creating a lasting set of content.

Completely experimental and plain fun, this Book-Sprint offered real results that you can hold in your hands and buy on *www.lulu.com*. A community Book-Sprint model can be shaped and formed to different projects, so be on the lookout for future BookSprints coming to a location near you. Be alert for opportunities to bring online communities together in the real world, and let them set their own agenda—maybe your next user group meeting? ●

Anne Gentle (annegentle@austin.rr.com) is a senior technical writer at Advanced Solutions International and blogger at www.justwriteclick.com. Advanced Solutions International provides management software for professional and social organizations such as STC. She also volunteers as a documentation coordinator and wiki gardener for One Laptop Per Child, an open source project dedicated to bringing kid-focused laptops to developing countries. She wrote a blog for her employer at BMC Software on http://talk.bmc.com and now writes her blog entries at JustWriteClick.com. She has been a technical writer for over ten years and has acquired many interests in that time, including structured authoring, social media, XML models for technical documentation such as DITA, blogging, wikis, collaborative documentation, online user assistance, and writing in an Agile development environment.

Janet Swisher (JMSwisher@gmail.com) is the sole technical writer for Enthought, Inc., which makes software for scientific data analysis and visualization using open source software, including the Python programming language. She has been a technical writer for ten years, including stints at Motorola (now Freescale Semiconductor) and Trilogy Software, and she previously filled other roles related to software development, including testing, technical support, and build management. She was a volunteer editor for OOoAuthors project, working on manuals for the OpenOffice.org open source productivity suite, and has now caught the Book Sprint bug. She blogs about topics related to technical communication and open source software on her Techie Tech Writer blog at http://www.janetswisher.com/.

**Structured content, Easy-to-use product.
Now you have it all.**

MadCap Software introduces the most advanced and comprehensive suite of tools supporting DITA authoring and publishing.

Learn more and sign up NOW:
www.madcapsoftware.com | + 1 (858) 320-0387