



merging usability practices

with Document Design
and Development

BY ELIZABETH FILIPPO, *Senior Member*

If I had a quarter for every time I've heard someone say "No one reads the documentation," I'd be a millionaire. Throughout my career, I have created paper documentation and online help for hardware, firmware, and software products. Everywhere I have worked there have been a few who justify their lack of review or response with the claim "No one uses the information anyway." To be honest, I have to admit that, in some cases, they are right.

One of my favorite books is Alan Cooper's *The Inmates Are Running the Asylum*. Although the book is intended primarily for interaction designers, technical communicators too can learn a lot from Cooper. Software engineers often create unusable products because they are designing for their own mind-sets, and technical communicators can fall into the same trap. We think we "know" our audience and what is best for them. So, as trained and specialized professionals, we implement sound document design principles to generate quality help or documentation that our target audience *should* be able to use. We document each product feature, translate that information into an interactive help system, and then attach it to our product in the belief that we have helped our users. We're done. The bear is dancing.

"The wonder isn't that the bear dances well, but that the bear dances at all."

—Alan Cooper, *The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity*

But as we begin to attend the design meetings for the next great feature or release, we fail to notice how that bear is waver. How he is lumbering from foot to foot, shuffling in his giant pink tutu. And how our users, staring at him with a skeptical eye, are beginning to think their time would be better spent at home watching *The Nutcracker*. In a time when outsourcing is second nature and technical communicators must constantly prove their value, this can be a fatal error. It is no longer enough merely to make the bear dance. We need to make him dance well. And the key to making that bear really leap is knowing our audience, knowing our internal stakeholders, and implementing an interactive design methodology—combined with sound usability methods—to generate flawless, truly usable document design.

Phases of Document Development

Traditional document design and development consist of the following tasks:

- Audience analysis
- Design
- Development
- Review
- Release

We begin by assessing our audience and how they will be using the product. We then combine that knowledge with product knowledge and solid document design principles to

create our design, or structure. Development comes next, followed by a documentation review (and perhaps a Beta test if we're lucky). When release time comes, it's back to the drawing board to document the next set of features. Follow these steps, and your bear will dance. Incorporate usability methods into these activities, however, and your bear may become the new Nureyev.

Audience Analysis

Audience analysis is the most critical part of any information design project. Without an intimate understanding of our users and their needs, how can we design information intended to assist them, or help them do their jobs more efficiently? If you are lucky enough to work at a company with an established human factors or usability group, you may have enough information about your users to get started. But many writers are not that lucky. In that case, where do you begin?

The best way to start is by digging through your usability toolbox for some tried-and-true methods. If you have access to your users, take advantage of it by using the following methods.

Field Observation

Sometimes referred to as ethnographic study, field observation consists of actually traveling to your customers' job sites and observing them as they do their job. Take note of your users' work environment, daily tasks, and interaction with other users. Job shadowing is an effective technique that involves sitting with your users for a length of time—not talking to them, but merely watching how they do their job. By walking in your users' shoes, you will gain a personal understanding of the challenges they face when they use your product in the context of their workday. This understanding will inevitably impact many of your design choices down the road.

Contextual Inquiry

Contextual inquiry consists of interviewing your users within the context of their work environment. Instead of just observing your users do their job, you ask questions as they perform their tasks. While formal interviews may seem stilted at times, positioning contextual inquiry as a "master-apprentice" activity can be extremely enlightening. Have your users show you how they do their job. Have them instruct you in how they use your product. Extending your site visits to include this type of inquiry will give your audience analysis a wonderful depth that you can use not only when designing your content, but in your navigation schemes and information access methods. You will understand the tasks your users are performing when searching for content, and how they go about looking

for that content. Putting the right information in the right place for users to access quickly and efficiently is a home run for any technical communicator.

Survey

While field observation and contextual inquiry will provide you with depth of analysis, sometimes it is impractical for writers to travel or to interact with their users on-site. The next best method is to obtain survey data directly from your users. I have seen extremely beneficial surveys, and others that are a waste of time. The key to conducting a valuable survey is avoiding the temptation to throw together a quick list of questions. Survey design is an art unto itself, and warrants careful research. Begin with a list of goals that specify what you want to learn from the survey. Next, determine whom you want to survey and how you want to conduct the survey (e-mail? mail? telephone?). Carefully craft your questions, and then test those questions with a pilot group to determine whether you are getting responses that tell you what you want to learn. Revise if necessary; then conduct the survey and analyze the results.

In some cases, writers have no access to their users whatsoever. This doesn't mean that you should skip the audience analysis phase, or even skimp. In fact, this situation can be beneficial in that it will force you to use your internal resources to conduct research, which can lead to increased buy-in from internal stakeholders. Begin by taking a close look at your organization and finding the individuals with any level of customer contact. In one company, training, customer support, and professional services may have the highest level of customer contact. In another company, the technical marketing and engineering test group may have the most contact.

In any case, assemble focus groups consisting of at least one representative from each functional area that has contact with customers. Bring everyone together in a room, and lead an activity to generate detailed user profiles. You'll be glad you did. Not only will you have assimilated the knowledge of

each of these individuals, you will have garnered their respect and shown the value that you can provide as a user advocate within your organization. For more information on this method, see my article "Teaming Up to Define Your Users," in the November 2002 issue of *Intercom*.

Design

Now that you have a solid, in-depth knowledge of your audience, it's time to go back to your usability toolbox. Dive in and use these methods to ensure that you have a design that works for you, your audience, and your organization.

Affinity Diagramming

Writing a new document or help system from scratch? Start here. Gather the same focus group that you used during your audience analysis, then explain the goals that you are trying to accomplish with the information. Allow everyone to brainstorm a list of topics, scenarios, tasks to document, or any other level of information that you seek. Record these ideas on sticky notes, then have the group come to a consensus on how it wants to sort those sticky notes on the wall. This is the fun part—and can turn into a corporate game of *Twister*. Once the cards are grouped, brainstorm headers for the groups. Record the header labels on sticky notes and place them at the top of each group. Last, draw lines to connect the groups, thus creating relationships among the concepts. Use this diagram as a basis for your document design.

You can also use this method to conduct task analysis. Start by brainstorming your users' goals, and then use the sticky notes to record tasks that fall within those goals. Categorize the tasks by user profile to help you determine what level of information you need to provide for each task. For example, if only beginners need details on task A, then mark it so that you will know where to embed that information when it comes to development. Within your document, beginners will be able to click a link for more detailed instructions, while the rest of your users will be unaffected by the extra content.

Card Sorting

This can be especially effective for an existing document that needs to be reorganized and revised. A recent project of mine, for example, involved revising a long, rambling document to match our users' actual workflow. To attack this type of project, assemble a group of stakeholders who have an intimate knowledge of the new workflow, and make redesign a group activity. Record the content headers and an overview of the content on cards—one card for each section or subsection. Have your team members divide the cards into logical groups. Ask them to remove the cards that are not necessary and create new cards for missing content. When you are done, you will have a new, detailed outline, complete with gap analysis.

Development

Now that you have performed a thorough audience analysis and obtained team consensus on design decisions, you are ready for development. Simple, right? Use your specialized skills to integrate your product knowledge and team input into an effective information product, complete and ready for review—right? Wrong! Part of fighting the lumbering bear syndrome is involving both stakeholders and subject matter experts (SMEs) throughout the entire development process. Your technical SMEs can tell you if your content is accurate, but it is that initial group of stakeholders with heavy customer contact, as well as your customers themselves, who can tell you if you are moving toward a useful information product. Dig back into your usability toolbox, and use the following methods to help you toward this goal.

Horizontal Prototyping

This is a useful activity throughout the entire development process, and it is great for new help systems. Create empty topics with just headers and navigation, and then have actual users test that navigation to see if they can easily find the topics they are looking for. If you are designing a paper document, you can test placement of sections, sub-

sections, and procedural content without actually having developed content. Watching frustrated users scour the document will quickly flush out discrepancies between your mental model and theirs.

Vertical Prototyping

If your navigation is solid but you want to test the level of detail you are providing in your content, consider vertical prototyping. Write a complete chapter or topic group, then test that information to see if you are delivering the type and level of content that your users require. A great example of the value of this step is in procedural documentation. At one company where I worked, half of our SMEs insisted that our users had the basic knowledge required for installation. The other half disagreed. Writing one section without the basic knowledge and testing it with all levels of users will end that argument quickly, one way or another.

Standards Inspections

In traditional standards inspections, usability professionals evaluate products against industry standards. For our purposes, we can use this method to evaluate our information products against corporate and industry standards for good information design. Create a step in your development process to assess your information product against interface design best practices, corporate style guides, and established document design principles. Use research to back up your design decisions and to fight the “I don’t like that color or font” argument from reviewers.

Feature Inspections

Be proactive during your development. Constantly evaluate your information against the product and the tasks you are documenting. Are you providing efficient, audience-appropriate instruction for each task that your users will be performing with your product? For example, will your users be able to increase their productivity with your software by using your information? Are there any holes? If so, address them now—don’t wait for reviewers to catch

**By the review stage,
we should have a solid
information product
that meets the needs
of our users.**

them. I once heard a writer say that draft reviews are traditionally only 50 percent complete, and my jaw dropped to the floor. If we are to prove our value, then we have to do our homework *before* we ask reviewers to approve our work. In my opinion, 50 percent is not good enough. By the review stage, we should have a solid information product that we believe meets the needs of our users.

Review

Ah ... review time. We have done our homework, assessed our users, created a thoughtful design, and then sweated it out during development. Now we are ready for a productive review. In addition to technical accuracy and feature creep, reviewers need to be assessing quality and usability. Here are some usability methods that we can adapt to help you and your review team meet those goals.

Heuristic Evaluation

This typically involves a usability expert evaluating whether a user interface (UI) design aligns with an established set of usability principles. To ensure that the heuristic evaluation goes well, establish a set of document design principles to which all of your documents should adhere, and then evaluate each deliverable on the basis of those principles. Borrow from established heuristics for

Web or UI design if you are creating online or Web-based help. Consult texts for best practices in document design. Having an established set of criteria against which you and other writers in your organization can measure yourselves will raise not only the quality of your publications but will increase their consistency as well.

Consistency Inspection

This sort of inspection is most helpful when performed by a documentation project manager or team lead who is evaluating all of the information products involved in a given release. In interface design, consistency inspections ensure that separate areas of the software work in a similar manner or share the same look and feel. Users shouldn’t feel as if they are using different products when they enter different areas of



the program. The same rule goes for information products. Users should feel as if various products' help was written by the same person (even though it was not). Help and documents should share the same look and feel, voice and style.

Informal Usability Testing

Steve Krug, author of *Don't Make Me Think: A Common Sense Approach to Web Usability*, refers to this as "lost our lease" usability testing. In contrast to traditional usability testing, informal testing involves a small group of users (internal users are fine) who evaluate your product against a set of features or tasks. It can be as simple as having a few new customer service representatives use your help or documentation to perform a set of tasks.

The important thing about this type of testing is where it occurs in your development process. Flushing out gaps in information or usability problems

Users should feel as if various products' help was written by the same person (even though it was not).

during the review stage enables you to incorporate this input prior to release. This is where the concept of iterative development comes in. Flush out the problems, fix them, and then retest. You'll be glad you did.

Release

You made it. Release day. It's a strong temptation to wipe the slate clean at this point. The best of organizations will conduct postmortem reviews of the project, while the most spry and aggressive will jump feet-first into the next release. No matter which kind you work in, you should make time to perform formal usability testing of your information product. I am not saying that you have to spend thousands of dollars on a usability lab and create a 200-page report. But it is a useful exercise to revisit your initial goals and verify that your information product has met them. Close the loop.

Consult one of the many books or Web sites on usability testing and design a test to verify that you accomplished what you set out to do. If you uncover a gap in information or a significant design flaw, incorporate that fix into your next project. This is a sure-fire way to be certain that your information products are continually useful and valuable. You took great pains to teach that bear to dance—now it's time to make sure he can perform with the Royal Ballet.

Fighting the "No One Reads the Documentation" Syndrome

I once went to a seminar given by Edward Tufte, an expert in informational graphics. At the end of his talk, he said something that has stuck with me throughout my career—to the effect of "before you make any decision, gather as much data as you can." That is the key

to the dancing bear. It is not enough for us to sit in our cubicles and do our best. We have to start pounding the pavement and involving our users and internal stakeholders in our work. **1**

SUGGESTED READINGS

Barnum, Carol M. 2002. *Usability Testing and Research*. New York, NY: Longman Publishers.

Cooper, Alan. 2004. *The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity*. Upper Saddle River, NJ: Pearson Education.

Filippo, Elizabeth. 2002. "Teaming Up to Define Your Users." *Intercom* (November).

Hackos, JoAnn T., and Janice C. Redish. 1998. *User and Task Analysis for Interface Design*. New York, NY: Wiley and Sons.

Krug, Steve. 2005. *Don't Make Me Think: A Common Sense Approach to Web Usability*. Second Edition. Berkeley, CA: New Riders Press.

Rubin, Jeffrey. 1994. *Handbook of Usability Testing*. New York, NY: John Wiley and Sons, Inc.

STC Usability and User Experience Community. www.stcsig.org/usability

Usability Professionals Association. www.upassoc.org

Useit.com: Jakob Nielsen's Web site. www.useit.com

Elizabeth Filippo is a senior technical writer at Configuresoft, Inc., in Colorado Springs, Colorado. She is a senior member of the Rocky Mountain Chapter STC and a graduate student in Mercer University's MSTCO program. Contact Elizabeth at Elizabeth.filippo@configuresoft.com.

Documenting APIs and SDKs Workshop-on-a-DVD



Learn everything you need to know to begin documenting APIs now.



Featuring
Manuel Gordon
Award-winning author,
speaker, and consultant

800-481-6728 or (514) 733-0354

www.thirdwavestudios.com

Discuss this article
online at [stcforum.org/
viewtopic.php?id=1109](http://stcforum.org/viewtopic.php?id=1109)

