



## Minimalism Updated 2012



*JoAnn Hackos, Comtech Services, Inc.*

The Minimalism Agenda for technical information and training was first developed in the late 1970s. John Carroll and colleagues at IBM Watson Research Center were asked to find more effective ways to help customers learn to use new desktop publishing software. Through direct observations and usability testing of new software users, they discovered what became the four principles of minimalism.

### Four Principles of Minimalism

- ◆ Principle One—Focus on an action-oriented approach
- ◆ Principle Two—Ensure you understand the users' world
- ◆ Principle Three—Recognize the importance of troubleshooting information
- ◆ Principle Four—Ensure that users can find the information they need

These descriptions are my restatements of the principles, representing my own manner of thinking and explaining what minimalism means to me today, more than thirty years after Carroll's initial work.

Minimalism, of course, has continued to advance during those 30 plus years. Carroll himself updated the original work with novice users with a study of programmers learning a programming language. The principles held true, an important data point because some benighted observers had claimed that minimalism was only useful for novices, or actually only for secretaries. At the time, that reaction represented a clear bias against clerical workers and women as learners. Obviously, the skeptics scoffed, programmers didn't need minimalism. Of course, the research proved that everyone needs minimalism when learning to use products effectively.

In the mid 1990s, we again updated the minimalism principles through the work of the consortium members sponsored by the Society for Technical Communication through collaboration with John Carroll. In fact, Carroll's second book, *Beyond the Nurnberg Funnel*, compiled the work of minimalism experts from around the world. My own research for my article in that book focused on users of scientific software used for graphical data modeling.

Since then, Hans van der Meij and his graduate students at the University of Twente in the Netherlands, continue to

update the minimalist principles with their research. Informally, I regularly receive communications from my own minimalism students about their successes in using minimalism principles to revise and restructure their information. Although anecdotal, their successes point to the continuing robust value of the principles in real world applications.

### PRINCIPLE ONE—FOCUS ON AN ACTION-ORIENTED APPROACH

John Carroll and IBM were certainly not the first to recognize that people using products are most interested in getting real work done. Even when individuals seek to understand how a product works, they intend to use that information to accomplish a task. For that reason, information developers provide procedures in documentation. The more effectively these procedures address real tasks that people want to perform, the more successful they are in meeting user needs.

But more than focusing on task- or user-goal-oriented information, Carroll reminded us that people best learn about products use by doing something rather than reading about something. As active learners, people frequently try a product out before glancing at written instructions. Or, given the extensive video available today on the web, those same people may find instructions that are presented as a recorded demonstration or even live in an interactive session.

Minimalism's first principle, then, draws information developers into a world of procedures, even if they assume that the user already knows a great deal about performing tasks. Take, for example, a skilled programmer who is trying to write software but doesn't quite know how to proceed. Although that programmer may find the answer in a reference topic that describes a command or software function, the programmer's end goal is still to perform real work—a task.

What often goes wrong with information that violates the first minimalism principles is a focus on using a product's interface rather than achieving real goals and completing real work. Although using the objects on a graphical user interface may indeed support real work, completing a form or clicking on interface objects is never the real goal. The typing and clicking is in service of reaching the goal, but not the goal itself.

To fulfill the first principle, then, means enabling real

work through instructional information. It also means, of course, understanding what users consider to be real work, those accomplishments that are meaningful in their lives or to their business. Too often, information developers, isolated from understanding their users, try to compensate by explaining the product interface, leaving the user to figure out how to get results alone.

In the past few years, I have seen a move in the industry to provide users with the information they need to solve problems. Known as scenario-based or solutions-oriented information, this content speaks to real goals and real work rather than trivial button pushing. So, for example, we might explain to a user how to take a still photo with a video camera, rather than describing a set of menus, one of which might include support for still photos.

### PRINCIPLE TWO—ENSURE YOU UNDERSTAND THE USERS' WORLD

Minimalism's second principle goes hand in hand with Principle One. Its focus on the users' real world, often referred to as the users' domain. It means that information developers must speak the users' language, rather than the language of the product developers. It means taking responsibility for understanding the users' world rather than requiring that the user learn your world.

In an early example of Principle Two, Carroll's team created a set of topics that would help their users learn word processing. The first topic was called "Typing Something." What a strange title, you might think. In fact, the original documentation, which had failed to enable the users, had called this task, "Creating a Document." Unfortunately, the users, who were typists or secretaries, didn't create documents, or at least they didn't think about their work that way. That was, in fact, what the software developers had used to label the task on the screen. The label confused the users because it wasn't part of their work, at least not in the 1970s. For example, some users considered documents as materials produced by the print shop, not the memos, letters, and reports for which they were responsible.

By renaming the task, "Typing Something," Carroll's solution communicated directly to the users in the context of their real tasks in their working world. By helping them successfully learn to type something, the instructions provided the users with an opportunity to be successful. Once someone is successful in performing an important task, they are usually willing to continue to learn a bit more.

When I teach Minimalism, I ask participants to think about their products from the users' point of view. It isn't easy to do, particularly if you have had little or no contact with the users. But once the information developers start thinking from the users' viewpoint, the results are amazing. Suddenly, obscure procedures take on a real world context.

Obviously, information developers can improve their ability to apply Principle Two by interacting with users. The more they know about how customers work and how they think and talk

about that work, the more effective and useful the information they create will be.

One important obstacle to conforming Principle Two comes from the reliance of information developers on the perspective of product developers. In many organizations, product developers may have little or no knowledge about the users. Their assignments are to create features or functions for the product, not understand why someone needs that feature or what someone will want to do using that function.

As a result of the lack of a user focus, information developers like product developers, focus on the product's features and functions rather than what users will want to do with the features and functions in their world. By contrast, entering the users' domain often means turning the information upside-down, combining features or functions or renaming them so that users' find their goals.

### PRINCIPLE THREE—RECOGNIZE THE IMPORTANCE OF TROUBLESHOOTING INFORMATION

The third Principle reminds us that troubleshooting information is always critically important to users at all stages of use, from novice to expert. We recognize that many people never consult the documentation unless they have a problem. Unfortunately, in my experience with workshop participants, most documentation fails the third Principle test. The information in general lacks troubleshooting information of any kind.

Recent research by van der Meij's team has focused on the importance of troubleshooting information that is embedded in the procedural topics themselves. By anticipating possible errors that users might make while performing tasks and documenting them where they are most likely to occur, we can help users be more successful by avoiding mistakes that interfere with learning.

Van der Meij has learned that not only do we need troubleshooting information in the context of a task but that troubleshooting information needs to be clearly labeled. That reminds me of an early practice in Apple documentation that labeled paragraphs of assistance with the italicized word, *Trouble?* It always seemed to be a perfect solution.

As a result of the troubleshooting research, the DITA Technical Communication Subcommittee has proposed three new elements for DITA 1.3. These are the addition of troubleshooting information as a note type `<note type="trouble">` in any topic, a `<steptroubleshooting>` element in the step in parallel with `<stepxmp>` and `<stepresult>`, and a `<resulttroubleshooting>` element in the task in parallel with `<result>` and `<example>`. In addition, the Subcommittee is also developing a Troubleshooting topic type in parallel with task, concept, and reference.

Of course, information developers need to incorporate more troubleshooting information into their projects. In many cases, we learn that troubleshooting information is difficult to get. It requires communication with service and support professionals and with users. However, these resources also reflect the need for

information developers to collaborate with other customer-facing organizations and to become active participants in the customer community.

One impressive program has been Microsoft's enhancement of troubleshooting topics that are identified as inadequate by customers and support team members. They have an entire group of information developers with degrees in computer science who are responsible for updating and improving the quality of the troubleshooting information.

#### **PRINCIPLE FOUR—ENSURE THAT USERS CAN FIND THE INFORMATION THEY NEED**

When Carroll's team first developed the Minimalism agenda, they were working with paper as the sole information source. For that reason, the original focus of Principle Four was on improving the quality of Tables of Content and Indexes. The problems identified with both resources remain.

Tables of Content needed to be well organized for use and reflect the users typical path through the content, rather than organized around product features. They need to reveal the structure of the information rather than be a random list of topics. And, they need to be reasonably brief so that users can scan them quickly. In the Minimalism workshop, we do a check on Tables of Content. Many of them don't meet these basic criteria and need improvement.

Indexes, as one might expect given the doleful lack of good indexing skills, are often very badly done. Many participants report that they have abandoned indexes because no one on their teams seems to know how to develop them effectively. Or, the indexes we examine using the Index Test come out quite badly. Luckily, there are several good books on how to index. We even have an indexing workshop that is actually a lot of fun to do, although we rarely get to teach it anymore because of a lack of demand. Indexing appears to be a lost art, which is unfortunate because many users report that they check indexes first to find relevant content.

Nonetheless, the major contributor to the downfall of good indexes has been the assumption that search replaces indexes. Search functions are ubiquitous on the web and with PDFs. We often assume that users will first try Google or Bing type searches

first. If the information they are seeking is behind a corporate firewall, they are quickly frustrated unless there is lots of user-generated, potentially unreliable content lurking on the web. When they get to the protected information on corporate sites, they are frequently defeated by terrible search systems. The search systems that we test during the workshops often return only PDFs of entire documents rather than individual topics. The PDFs don't have indexes, requiring that users search again on keywords, only to find that they are searching the wrong PDF.

Clearly, Principle Four continues to be updated in our practice using the work of web researchers and specialists as well as the experience of practitioners. However, we continue to support the need for good Tables of Content and indexes because so much content is still being delivered as static book-based documents. At the same time, we urge, in the context of Principle Four, that information developers begin to understand the importance of metadata associated with XML-based DITA topics. Metadata enables effective search, as do other aspects of Search Engine Optimization. Principle Four today includes advice on writing topic titles and short descriptions that contain sufficient keywords to affect content searches, even in the absence of metadata.

#### **SUMMARY**

The Minimalism Agenda has been an important contributor to best practices in information development since its origins in the work of John Carroll at IBM more than 30 years ago. It has influenced the development of many common practices among information-development professionals, including a focus on user tasks, user language, scenario-based design, comprehensive solutions content architecture, and just plain good writing. We continue to follow the important work done in the Netherlands by Hans van der Meij, the continuing contributions of Janice (Ginny) Redish, our colleague, and researchers who believe as we do that minimalism contributes significantly to high quality information best suited for users of all stages of use and in all disciplines. If you haven't yet learned about Minimalism, now is the time. It should be central to your approach to information development. 