

# Reader-centered Documentation

Provides  
the Necessary  
**Context**



ImageZoo/Punchstock

The large group of users who want to learn how to do something requires considerably more context than a feature list provides. The most inexperienced readers in this group are a particular challenge since they need a broad overview of what they can accomplish with the product, the main components that permit them to accomplish these goals, and how all the components fit together. More experienced readers already know something of this overall context, and instead need more intermediate-level information, as well as a comprehensive view of the goals they can accomplish with the product. Lastly, all readers need to know the specific tasks that must be performed in order to reach each goal. Below this comes the finest level of detail: the steps required to complete each individual task.

In this article, I'll discuss each level of information, or context, with an emphasis on how each meets specific reader needs.

### Provide Broad-scale Context

Consider a topic near and dear to our hearts: desktop publishing, or, in more modern terms, document design. In this topic, the broad-scale context is the construction of documents. A simplified list of the high-level goals that users might wish to accomplish includes the following:

- Constructing complex documents ("books" with multiple chapters)
- Designing master pages that control how individual pages will appear
- Designing paragraphs to fill the page
- Inserting graphics
- Manipulating paragraphs and graphics
- Generating outputs (paper, PDF, HTML, and so on)

Clearly, there are many more possibilities; this short list is designed solely to illustrate what constitutes broad-scale context. Each example in the list represents a large group of related topics that define overall user goals. Although most documentation does an adequate job of describing the software features used in each of these topics, the broad overview that shows how they fit together is often lacking; there is no discussion of when it

BY GEOFFREY J. S. HART, *Fellow*

**F**ar too many documentation projects start out with nothing more than a comprehensive inventory of the "widgets" a product uses—the menus, icons, and dialog boxes of software, or the switches, buttons, and levers of hardware. Although it's certainly true that each of these items will eventually need to be documented, the problem with this approach is that it focuses more on the tool than on the reader. As a result, the reader's needs are subordinate to the idiosyncrasies of the product, resulting in documentation that works only for a specific class of user: the relatively rare character who already knows how everything works and is interested only in finding out a few details. A features-based approach is certainly appropriate for reference manuals, where the goal is to provide information on something the reader already knows. But it works poorly in user manuals, where the goal is to teach the reader how to *do* something.



might be necessary to break documents into separate chapters rather than keeping everything in a single document, there is little discussion of which elements should repeat from chapter to chapter and page to page (and which should not), and so on. The documentation focuses too narrowly on tasks and procedures and, in so doing, fails to provide the larger goals that would help readers understand why they should perform the tasks and how each task fits within a broader understanding.

nents of white space

- Recurring elements (such as headers and footers)
- Columns of type

The textual characteristics of the page include the following:

- Space above and below paragraphs
- Indents and tab settings
- Line spacing (leading)
- Typography (typefaces, type sizes, kerning, special characters, and so on)



### Provide Intermediate Context

Having defined the broader structure of the document, the next step is to define the intermediate level of context for each of these groups: the tasks that, taken together, allow the user to accomplish a larger goal. Let's pick just one of the broad-scale topics from the previous section to explore how we can meet the reader's goal: designing an effective page. This goal has a range of components, which can be divided into two broad groups: (1) the visual or graphical properties of a page and (2) the textual properties of the page. The visual properties include the following:

- Page size (such as North American vs. European) and the constraints imposed by this choice
- Margins, gutters, and other compo-

All these characteristics of page design are generally covered in adequate technical detail in most documentation; for example, generally each item in the **Font** dialog box is thoroughly described. What is often missing is an explanation of why each choice is relevant to the reader's needs. In the discussion of columns and gutters, for example, it would be helpful to explain how the choice of the number of columns affects line width, which in turn affects and is affected by type size and the required space between columns (the gutter). This would clearly lead us to include cross-references to type size, word spacing, line spacing, and other factors that affect legibility once a given line length has been chosen.

### Provide Low-level Context

At the lowest level of our hierarchy of context, we must provide information that guides readers in their use of tools and procedures. In other words, we are not just helping readers use the product's tools, but helping them use these tools *effectively*. Again, let's explore just one example from the list in the previous section to illustrate necessary context. Since typography is near and dear to the hearts of most technical communicators (witness the frequent, often-heated debates over the aesthetics—or lack thereof—of Times New Roman), let's consider that topic. A simplified list of typographical topics for which context is necessary includes the following:

- Typeface families (serif, sans serif, scripts)
- Typeface function (display type or headings, body text, ornamental type)
- Justification and line length
- Character and word spacing
- Line spacing (leading)

Again, most documentation provides complete information on the available settings for each of these characteristics of text, but many authors fail to explain when each setting might be preferable. For most readers who lack formal training in typography, it's useful to explain concepts such as legibility. Characters must be spaced sufficiently far apart that they don't merge, words must be sufficiently far apart that each cluster of letters is perceived as a single unit (a word) and not so close that two words overlap, and so on. This level of context should not become a treatise in cognitive psychology, laden with jargon such as *saccades* (eye movements from one fixed position, such as the end of a word, to the next fixed position); instead, it should provide gentle guidance on effective use of the tools. For example, some third-party books on desktop publishing software provide rules of thumb for effective leading as a function of font size and column width, without ever delving into the cognitive psychology of how readers' eyes scan to the end of the line and return to the start of the next line. The trick is to choose an appropriate level of detail.

## Cover Everything

The approach discussed thus far ensures that your documentation meets the primary goal of any documentation project: providing enough understanding of a product that users possess the necessary intellectual tools to plan an overall approach to using the product and can determine all the tasks necessary to attain each of their goals. But, ironically, the problem with this approach is that by focusing so strongly on the reader, it may neglect the product itself. As a result, certain key features of the product may be missed. This poses an obvious problem when the user encounters a feature, asks how to use it, and is unable to find any information on how to do so.

The solution is straightforward: prepare a comprehensive list of product features and match it to the broad-scale, intermediate, and low-level contexts that you've prepared to ensure through your consideration of context that you have covered each feature. You'll undoubtedly find some aspects of the product that you missed by focusing on the reader, and some of these may become new topics at one or more of the three levels of context.

Other features won't seem to fit in anywhere—an unfortunately common occurrence when software is designed by marketing committees rather than through a careful analysis of user needs. Once you've verified that a feature doesn't really fit well anywhere, you can create a separate place in your documentation (such as an appendix) that provides the necessary information about this feature in a manner that doesn't interfere with your presentation of the more important material.

## Don't Forget the Procedures

Note that by focusing on context in this article, I have entirely omitted any discussion of actual procedural information. Even the most reluctant readers will eventually learn enough context to want to get on with the steps required to accomplish a task. To prevent readers from being overwhelmed by a sea of text that conceals the procedural information, clearly separate the context from

It's simply  
not possible  
to provide the  
full context for  
any product.

the procedures. In an online document, this might be accomplished by means of hyperlinks. For example, a help topic on typography might begin with a series of related topics, such as a link entitled "[Learn more about classical typography](#)" or a series of links ("Learn more about: [typefaces](#), [space between letters](#), [space between lines](#), [etc.]").

In print, which is inherently more linear, this might be accomplished by structuring every topic to contain a main heading (the task) followed by two main subheadings: the context (for example, "Basic elements of typography"), and the content ("Applying typographic formats to text"). A short introduction beneath the main heading, perhaps as short as a single sentence and certainly no more than two or three sentences, provides the necessary context for how to use this part of the documentation. For example, "This topic explains how to format text. It begins with a discussion of the elements of typography, then concludes (on page 217) with detailed instructions on how to apply these principles." Conversely, you might choose to completely separate the contextual information from the procedural information. In that case, the topic might begin as follows: "This chapter discusses how to format text. For a discussion of classical typography, see Appendix 1 on page 340."

## Providing Tools for Understanding

In this article, I've focused on an often-neglected aspect of documentation, namely the context that readers

must understand (or be taught) before they can successfully apply a product's features. For many of us, there is no clear mandate to teach readers everything they must know about the environment in which a product will be used. Furthermore, it's simply not possible to provide the full context for any product. For example, graphic design is an entire college-level course comprising many topics, each with its own textbook and hours of classroom instruction. It's clearly impossible for us to write all that course material ourselves and still accomplish our primary task: documenting a product that graphic designers will use. This is particularly true given that much of the audience for desktop publishing and other graphical software will have already taken that course and may know far more than we do about the topic.

In such cases, we instead must perform triage: Our goal is to identify the essential context that readers must know before they can use the product successfully, and it's this context we must provide. A seldom-used approach that makes this kind of triage more effective is to include a bibliography of related works, and refer readers to it. For example, in the documentation for graphing tools such as the data presentation tools offered by spreadsheets, we might refer readers to the works of Edward Tufte, a professor and statistician known for his work in the areas of information design and visual literacy.

As in so much of our work, we must choose an appropriate compromise between too much and too little detail. In this article, I've provided some guidelines that should help, but the onus is on you, the communicator, to apply those guidelines based on an understanding of your actual audience's needs. **1**

*Geoffrey J. S. Hart can be reached at [ghart@videotron.ca](mailto:ghart@videotron.ca). Visit him online at [www.geoffhart.com](http://www.geoffhart.com).*

Discuss this article  
online at [stcforum.org/  
viewtopic.php?id=1113](http://stcforum.org/viewtopic.php?id=1113)

