The quality of access: helping users find information in documentation

Michaël Steehouder

Reference manuals are usually consulted by experienced computer users when effective working with software is hindered by a problem. Using a general model of an information-seeking event as reference, it is argued in this contribution that the access structure of a reference manual should be related to the various types of problems that give rise to searching for information. It should also been taken into account that users give names to functions and operations which usually differ from those used in the manual and its index. Based upon the results of exploratory studies, some general strategies for designing indexes are sketched, illustrating that studying user behaviour can be a successful heuristic for improving technical documentation.

Computer documentation for users can serve two distinct functions: the tutorial function and the reference function. Roughly speaking, tutorial documentation is intended to help novices master a software system; reference documentation, although frequently used by experienced computer users as tutorial material, is primarily intended to help users to deal with complex functions of the software, to perform incidental tasks, or to solve unexpected problems.

Until now, most research into computer user documentation has been focused on the tutorial function. Important lines of research have been initiated by Carroll, resulting in the idea of the minimal manual (cf. Carroll, 1990), and by Charney and others, concerning the effects of elaborations (Charney *et al.,* 1988). In this contribution I will focus upon some aspects of the *reference* function of manuals.

A critical requirement of usable reference documentation is its accessibility. People access technical information in a very different way from the way they read essays, textbooks or newspapers. They do not just wait more or less passively for what information the text will offer, but they search for information that will help to solve specific problems. In general, readers of reference documentation are not interested in the text as such; but they regard reading as a necessary evil. Readers try to locate exactly that piece of information that is relevant to their actual situation and prefer to skip everything else. This is what Brockmann (1990, p.225) characterised as a ‘random access style' of reading.

Making documentation 'randomly accessible' therefore becomes an important issue in the field of technical documentation. A considerable number of advisory articles and textbooks on concerning devices such as indexes, tables of content, headings, cuing and layout is available. The term *Access Structure* has been introduced by Waller (1979) as a generic term. Reviewing the available literature, there seems to be a sufficient 'body of knowledge' for professional writers to establish a usable access structure. Nevertheless, the major complaints of users seem to concern just this aspect. Therefore, it seems relevant to know how exactly computer users tend to use a reference manual and which problems they may encounter.

The aim of this article is to outline some of these user problems as they came to light in exploratory studies. Although these reflections will only be explanatory, I will elaborate upon some concrete consequences for improving access structure.

Although *electronic* documentation promises a lot of new technologies to overcome problems of information seeking, this contribution will be confined to seeking in *hard-copy (paper) documentation*, because paper documentation is still common practice. Problems with paper documentation do certainly differ from those with electronic information, but perhaps less radically than some authors claim they do.

1 A model of information-seeking events

As a starting point for my analysis, I will use a task model of information seeking events (Figure 1). In this model, seven sub-tasks are distinguished which seem relevant for the overall searching task. The model is related to several models from the research literature: Guthrie (1988), Duffy et al. (1989) and Kern (1985).

Let me explain the model with a simple, but quite frustrating event from my own experience. Long ago I started using a new text processing system (which will be anonymous for now) and after a short learning period, I considered myself to be skilled enough to write an article using the system. It all went well, but after writing some pages, I decided to change the title of the article. To do so, I had to go back to the beginning of the document. But how does one go back?

One possibility is pressing the *arrow-up key* as long as needed to reach the first line of the text. Users who follow this strategy do not realise that there could be a single command which takes you there straight away. They never discover the *need for information;* step 1 in the model fails. However, probably because I had worked with an other system before, I supposed that there would be such a single command, and I wanted to know which one it was. Assuming that it would be the same as in the 'old' system, seemed to risky.

1 Detecting the need for information
2 Formulating the problem
3 Selecting the information source
4 Locating the relevant information
5 Understanding the information
6 Inferring an action plan
7 Evaluating the action plan

Figure 1: A model of information seeking events

Now item 2 in the model comes at hand: what would this command be named? Something like *begin* or *move back* or just *back* seemed a good guess, but looking in the index of the reference manual (step 4), I discovered that those entries did not exist. Only by *scanning* the index from the beginning to the end, did I discover that the correct wording should have been *cursor movement*.

Meanwhile, I passed step 3: I had decided to use the *manual* as my information source. I could have relied on a guess, but simply trying out all the function keys seemed too risky. I could have asked my colleague, but at the time he knew even less about the system than I did. And I could have used the on-line help, but I was not yet familiar with it.

Once I knew that the proper keyword was *cursor movement*, I was able to locate the section in the manual with the information I needed (step 4). But it took some more efforts to find the exact piece of information within that section.

Finally, I found the commands I had to use: *Home, home, arrow up*. That means 'Press the home key twice, and then press the arrow-up key'. Some readers may have trouble understanding this language (step 5) and may remain doubtful about exactly what they have to do to move the cursor to the beginning of the text (step 6). Anyway, I am proud to say that I understood it almost immediately.

However, that was not the end of my troubles. When I pressed the home key twice, the cursor was not moved, but the number *77* appeared on my screen. For a moment, I supposed that, after all, I had misunderstood the instructions (step 7). But fortunately I corrected this evaluation, realising that I had to press the *Num Lock* key before using the *home* key on the numerical pad of my keyboard.

From the research literature it may be concluded that in realistic situations step 2 is of paramount importance. For instance, Dreher & Guthrie (1990) presented a textbook chapter and its accompanying index, table of contents and glossary on a computer and asked 31 US high school students to search for answers to simple and complex questions. It turned out that more efficient searchers allocate more of their search time to selecting information categories for pursuing their goals than less efficient searchers.

However, in Dreher & Guthrie's experiment the starting questions of the search activities were posed by the experimenters. So they did not take into account the influence of step 1 in the model (which is actually ignored in their model). Nevertheless, this first step might be just as important a hurdle as the second. Therefore I will look somewhat more closely at both the first and the second steps in the model and at their consequences for the access structure of the documentation.

2 Detecting the need for information

Even when a manual has an exhaustive index, a convenient table of contents, useful headings and a well-structured layout, computer users still appear to experience great difficulties in searching for and locating precisely those pieces of information which they actually need. To understand this, it seems useful to examine the processes that precede the actual reading of the documentation.

When and why do people feel a need to seek information?[1] This issue seems important, because different occasions may lead to different information needs and different reading strategies. Research literature does not provide many starting points for answering these questions. However, four different situations that give rise to information needs may be distinguished.[2]

Firstly, there may be an *impasse*: while working with technical equipment or with a software system, the user cannot proceed because he does not know which functions, procedures or commands should be used. This situation typically asks for *procedural* information related to a *goal*; the typical question is a *how-to*-question.

Another type of event leading to information seeking may be an *error:* something goes wrong, something unexpected happens. An error urges the user to search for a *diagnosis* and for a *remedy*. For instance: a user of a computer program wants to insert a formula in a text but instead of opening the formula editor, he accidentally presses a wrong key which opens a text window. Once the user has detected that something has gone wrong - which may take some time - he has to understand *what* went wrong (the diagnosis) and how to recover from the text window (the remedy). It seems clear that dealing with an error requires different information from that required by an impasse. But once the user has recovered from the error, an impasse may still be left: the user has to know the right procedure.

A different situation is what I would call *dis-coordination:* a user may be confused by many disjointed details and may seek information that helps to provide an overall picture of the program. This seems to be the typical situation where users need explanatory information, such as examples or overviews.

And finally, *uncertainty* may be a motive to seek information. Sometimes users of technical equipment use a lot of guesswork in working with the system, but that may result in doubt as to whether their ideas of solving their problems are correct or not. Perhaps they simply try them out, but they may also first want to verify their guesses by consulting the documentation. Again, they need a different type of information. It seems that such *verification* goals are quite common when technical documentation is used. For example, Aaronson & Carroll (1987) analysed the conversations between computer users and help desk staff. It turned out that users, instead of asking *how-to* questions, often made assumptions about the commands or functions they had to use, and wanted the help desk staff to confirm their assumptions.

Although more reasons for information seeking may be distinguished, these four seem sufficient to illustrate my point that it is important for our understanding of information–seeking behaviour to consider not only the use of the manual, but also the problems with the software that precede the search process. More generally, it seems a prerequisite for understanding problems with instructional text to study the relationship between reading a text and performing non-textual tasks. This relationship is also reflected in a model of sub-tasks in completing forms, presented in Jansen & Steehouder (1992, p.183). Three task levels are discerned in this model: *interpretation tasks* (related to reading the text), *functional tasks* (related to completing the form) and *monitor tasks* that ‘control' the process; one of these monitor tasks is selection of relevant information in the text. The model reflects the inter-relationships between these levels.

This closer look into different motives for using technical documentation may be relevant to various issues related to the organisation of reference documentation and the access structure.

For instance, what will be the optimal *perspective* of the documentation no longer seems completely clear. In many textbooks we find the recommendation to choose the *user's perspective*, or the *action-directing* perspective: the information in the manual should be organised around user goals and actions. To put it in simple terms: the manual should not tell what happens when a specified button is pressed but it should tell which button has to be pressed in order to achieve a desired effect. Not: *F7 saves your file* but: *If you want to save your file, press F7.* In this perspective, the user's *goals* (such as saving a file) are superimposed over the actions to be performed (for instance, pressing F7). But this perspective seems to apply only to users who start reading from an *impasse* situation. But if one starts from an uncertainty situation, I would prefer just the opposite: *If you press F7, the file will be saved.* And when the user starts from an *error* situation, the action-directing perspective only tells what he *should have done.* The instruction comes too late. He needs error recovery information first, and perhaps procedural information afterwards.

A second issue pertains to the choice of entries for an index. Since each unit of information in the documentation may be approached from different information needs, it might be appropriate to index each unit with different entries, reflecting these needs. I will return to this in Section 5.

3 Formulating the problem

Moving to the next step of the model of information-seeking behaviour, the question arises, how people formulate their information needs. This again seems an important question with respect to the access structure of technical documentation. It seems obvious, for instance, that an index can be usable only if the entries cover the terms *users* choose to characterise their information needs.

Although finding the right entry seems to be a major problem for users, there is not much research indicating how people formulate their problems. A relevant study is the one by Black & Moran (1982). They studied how people who are unfamiliar with word processing naturally describe text editing operations. They showed a short text fragment to their subjects in two versions: *before* and *after* a certain operation had been carried out (see Figure 2). Then they asked their subjects to name the operation which leads from the 'before' (incorrect) version to the 'after' (correct) version. Subjects turned out to use terms that substantially differed from the technical terms commonly used.

*Before:*witout doing

*After:*without doing

Technical termsTypical Subjects' terms

insertadd

deleteomit

replacechange
moveput

Figure 2: Example from the Black & Moran (1982) study of naming of computer commands by non-users

A similar diversity, though in a totally different field, comes from an unpublished study by Vernoy (1993). She presented 43 Dutch subjects with a text containing a number of linguistic errors, and asked the subjects to name those errors. It turned out that the subjects used a wide variety of terms. Most errors were named in ten or more different ways. The official grammatical terms were scarcely used. When asked to choose an appropriate term from an index, most subjects chose a non-technical, exemplary entry. Only in four cases did more than 50% of the subjects chose the technical term.

In an exploratory study (Steehouder 1993), experienced users of the WordPerfect program were observed while they were performing several tasks. One of the tasks was putting a text in 'all capitals'. As figure 3 shows, any indication about the name of the function to be performed was carefully avoided in the formulation of the assignment.

Assignment 1 ·Load the file with the name ASSIGN.111 ·Change the text so that it will look as follows (you don't have to check the contents of the text) ·Print on screen to check whether you succeeded.
TWO ORTHOGONAL AND COMPATIBLE BASIC VIEWS OF THE ENTERPRISE OF AI SERVE OUR FIELD, BEYOND ALL THEOR-ETICAL QUIBBLES. THE FIRST IS A GEOGRAPHY OF TASK AREAS. THERE IS A PUZZLE TO SOLVING, THEOREM PROVIDING, GAME PLAYING INDUCTION, NATURAL LANGUAGE, MEDICAL DIAG-NOSIS, AND ON AND ON, WITH SUBDIVISIONS OF EACH MAJOR TERRITORY. AI, IN THIS VIEW, IS AN EXPLORATION, IN BREAD AN IN DEPTH, OF NEW TERRITORIES OF TASKS WITH THEIR NEW PATTERNS OF INTELLECTUAL DEMANDS. THE SECOND VIEW IS THE FUNCTIONAL COMPONENTS THAT COMPRISE AN INTELLIGENT SYSTEM. THERE IS A PERCEPTUAL SYSTEM, AND SO ON. IT IS THIS SECOND VIEW THAT WE NEED TO CONSIDER TO ADDRESS THE ROLE OF REPRESENTATION AND KNOWL-EDGE.

Figure 3: Experimental assignment from Steehouder (1993). The upper part was originally given in Dutch; the sample text is taken from an American publication on Artificial Intelligence.

An inventory was created of names the subjects used for this task and of entries from the index of the manual they called upon. Eight different names were used apart from the two which were included in the index (see Figure 4).[3]

Index entries:
letteromzetting(transposition of characters)
wisselen (change)

Subjects' names:
upper case (capitals) (12x!)
replace (replace)
switch (transpose)
exchange (change)
letters(types)
letter size(corps)
layout(style)
words(words)

Figure 4: Search terms mentioned by 15 subjects

More interesting than the wide variety of terms used by individuals are the strategies the participants applied to find the right term. Some participants were often unable to think of an proper search term themselves. Instead they scanned the index looking for some useful hints. One of the participants explicitly stated *I am now looking in the list of contents* [actually he was looking at the index - MFS] *to see if I can come across some words which I think have to do with the problem.* So he adopted a search term from the index instead of thinking one up himself.
In connection with this scanning strategy, some subjects applied a sort of *funnel strategy.* They formulated a *general* term for their problem, for instance *character*, looked it up in the index, and next they scanned the subordinate entries, searching for a term that might be applicable to their situation.

It seems that finding the right entry is not a linear process, but a highly recursive process, in which computer users alternately think up appropriate names and scan the index for possibilities. It may be superfluous to say that in several instances subjects looked up passages in the manual which did not contain the relevant information, resulting in loss of time.

4 Consequences for indexes

What do the results of these exploratory studies mean for the access structure of a manual, and especially for the index?

Firstly, they show that constructing a useful index, is a much more complex task than is sometimes assumed. It is not enough to incorporate all the significant terms that are used in the documentation. Users themselves come up with different terms from those chosen by the designers as standard in the software system and in the manual.

An important requirement is that an index contains sufficient and appropriate synonyms, or (to use Shneidermann's term), *aliases*. But how can such aliases be found? And how can be found out which are useful and which are superfluous? A look into some leading textbooks yields various strategies:

- Play 'word games' and think of all the synonyms with which readers may try to access your information [. . .] (Brockmann, 1990, p.231).
- Analyse your audience and find out which term is the single most appropriate pointer term (Brockmann, 1990, p.230).
- Aliases should be determined empirically by discovering what alternate names users actually try to use (Shneidermann & Kearsley, 1989, p.43).
- Look in competitors' manuals to see which single terms appear most often in their indexes (Brockmann, 1990, p.230).

Although each of these strategies seems fruitful, the list does not seem very practical. The first strategy will possibly produce a vast number of aliases, depending on the creativity of the writer. But we may ask how realistic such word games will be. The success of this strategy depends entirely on the empathy and creativity of the writer. The same reservations apply to the second strategy if 'analysing' is another term for putting oneself in the user's shoes.

When 'analysing' involves empirical research, the second and third strategies might be very successful. However, they could be very laborious and expensive too, requiring experiments such as those I outlined above, which take time and money. This guideline does not seem very practical for most situations.

The fourth strategy seems the most practical. And I would strongly recommend every technical writer to adopt it. However, there is a considerable risk of copying the competitor's deficiencies.

Considering the model sketched in this article and the exploratory research mentioned above, an additional strategy may be suggested. Since an index should provide entries that correspond with the various types of events that can lead to information-seeking actions, it might be helpful for writers to create lists of entries that would complete these four phrases:

·How to .... (for impasses)
·What if .... (for errors)
·What is .... (for dis-coordinations)
·What happens when .... (for uncertainties)

Of course, this strategy is rather general and it must probably be elaborated out further. And, in addition to the list of entries produced by this method, it will still be necessary to find more aliases. The proposed division is suggested only as a means of supplementing the strategies mentioned previously.

Another strategy for improving indexes is based on the observation that users not only come up with search terms themselves, but also *scan* the index for terms that might be appropriate. It may be expected that they could be helped by some *context* of each entry which helps them predicting whether the section the entry refers to will be relevant to their current problem or not. The context provided may prevent them from looking up irrelevant sections of the manual and instead prompt them to scan further until they come across an entry that seems more appropriate.

The context to provide may take various forms. One possibility is to use 'keyword-in–context' or 'keyword-out-of-context' indexes such as in Figure 5ab. More informative would be an index which provides a short explanation of each entry, such as in figure 6. In fact, such an index would be a combination of a traditional index and a *glossary*.

Engineering report —rustytool	1191
Companies put new zip in old salestool(tech. report)	0100
The dictionary as a writer'stool	0860
Language as an engineeringtool—preparation	1699

Figure 5a: Keyword-in-context; example from Wright (1985, p.275).

LANGUAGE	
Comments on the relevance of language	1234
A numerical notation for language systems	0987
PL360 a programming language for 360 computers	1365

Figure 5b: Keyword-out-of-context; example from Wright (1985, p.277).

Brightness control panel 35, 82
Macintosh Classic software control of the built-in video monitor's brightness. Other Macintoshes and monitors have manual brightness controls.

bulletin board (BRS) 174-175
Communication service on a computer that allows other users to leave messages and share files. Requires a modem and communications software. Some services are community-based, while a number of commercial cervices are accessible world-wide.

burn-in, video 37
Discoloration of the phosphor inside the video monitor. Long exposures of white (e.g., the menu bar) may make the image visible even when the computer is turned off. Turning down the brightness or using a screen saver can reduce burn-in.

Figure 6: Index combined with a glossary (from Danny Goodman's Macintosh Handbook)

5 Conclusion

The exploration of information-seeking behaviour sketched in the first part of this article yielded some general guidelines for improving the access structure of technical documents, in particular the index. Although these guidelines can be refined and should be tested in a formal experiment, they illustrate the value of user observations as a heuristic for improving technical documentation. But I am not sure whether developing and testing better guidelines should be our main concerns at the moment. I agree with Wright's claim, that guidelines and criteria do not always help, because many aspects of usability are situation-linked. Therefore, it makes sense to study the readers' cognitive skills and processes and to link them to the uses of documents. Understanding the information--seeking behaviour of document users might be of more help than guidelines or specifications.

Notes

1 It should be emphasised that the users under attention here are more or less *experienced* users, who do not define their position as novice or learner. They have already internalised some kind of 'mental model' of the system, although this model may be too generalised or incorrect.

2 The classification I make here is partly based upon Inagaki & Hatano's (1986) adaptation of Berlyne's (1965) theory of motivation for learning. A more elaborate version of this classification is presumably needed.

3 It is important to know that the term *capitals* was not included in the index, while all other terms were. The others referred to sections of the manual which were irrelevant or useless to the current problem.

References

Aaronson, A., & Carroll, J.M. (1987), The answer is in the question: a protocol study of intelligent help. *Behaviour and information technology* 6, p.393-402.

Berlyne, D.E. (1965), *Structure and direction in thinking*. New York.

Black, John B., & Moran, Thomas P. (1982), Learning and remembering command names. In *Proceedings Human Factors in Computer Systems*, Gaithersburg ML: Association of Computing Machinery, p.8-11.

Brockmann, R.J. (1990), *Writing better computer user documentation. From paper to hypertext. Version 2.0.* New York etc.: Wiley & Sons.

Carroll, J.M. (1990), *The Nurnberg funnel. Designing minimalist instruction for practical computer skill.* Cambridge, MA: The MIT Press.

Charney, D.H., Reder, L.M., & Wells, G.W. (1988), Studies of elaboration in instructional texts. In S. Doheney-Farina (ed.), *Effective documentation. What we have learned from research.* Cambridge MA: The MIT Press.

Dreher, M.J., & Guthrie, J.T. (1990), Cognitive processes in textbook chapter search tasks. *Reading research quarterly* 25, p.323-339.

Duffy, T.M., Mehlenbacher, B., & Palmer, J. (1989), The evaluation of on-line help systems: a conceptual model. In E. Barrett (ed.), *The society of text. Hypertext, hypermedia, and the social construction of information.* Cambridge MS: The MIT Press.

Duffy, T.M., & Waller, R. (eds) (1985), *Designing usable texts*. Orlando, etc.: Academic Press.

Goodman, D., & Wurman, R.S. (1992), *Danny Goodman's Macintosh handbook featuring system 7*. New York: Bantam Books.

Guthrie, J.T. (1988), Locating information in documents: examination of a cognitive model. *Reading research quarterly 23*, p.178-199.

Inagaki, K., & Hatano, G. (1986), *Motivation for understanding in the classroom. A cognitive Berlynean theory.* Unpublished paper, Chiba University & Dokkyo University (JP).

Jansen, C., & Steehouder, M. (1992), Forms as a source of communication problems. *Journal of technical writing and communication* 22, p.170-194.

Kern, R.P. (1985), Modeling users and their use of technical materials. In T.M. Duffy & R.H. Waller (eds), *Designing usable texts*. Orlando, etc.: Academic Press, p.341-375.

Shneidermann, B., & Kearsley, G. (1989), *Hypertext hands-on! An introduction to a new way of organizing and accessing information.* Reading MA: Addison-Wesley.

Steehouder, M.F. (1993), Informatie zoeken in computerhandleidingen. Een verkenning van problemen [Seeking information in computer manuals. An exploration of problems]. *Tijdschrift voor taalbeheersing* 15, p.3-12.

Vernoy, M. (1993), *Experimenteel onderzoek probleemherkenning en -representatie [Experimental study of problem recognition and representation].* Unpublished report, Utrecht University.

Waller, R.H.W. (1979),Typographic access structures for educational texts. In P.A. Kolers, M.E. Wrolstad & H. Bouma, (eds), *Processing of visible language.* Vol 1. New York etc.: Plenum Press, p.175-187.

Wright, K.C. (1985), Designing contents lists and indexes for access. In D.H. Jonassen (ed.), *The technology of text. Principles for structuring, designing, and displaying text.* Englewood Cliffs NJ: Educational technology publications, p.264-286.

About the author

Michaël Steehouder is an associate professor *(universitair hoofddocent)* at the University of Twente. His main research interest lies in written communication. In cooperation with his former colleague Carel Jansen, he investigated the behaviour of form fillers and developed methods for improving the usability of government forms. He has also been studying several aspects of computer manuals, currently focusing on issues of information seeking by computer users for the past eight years. He is a co-author of a leading Dutch textbook on communication *(Leren communiceren)* and a member of the editorial board of *Communicatief,* a Dutch journal on business communication. Together with Carel Jansen he is preparing a textbook on writing computer manuals.