

Understanding Minimalism

Peter Graham. © 2000.

Introduction

This literature review examines the origins, development, theoretical basis and practical application of minimalist instruction in technical communication. A restricted number of books and papers on the subject are analysed and summarised. A new interpretation of misconceptions of minimalism is proposed.

The purpose of this paper is to show the authors understanding of minimalism in the context of technical communication theory and research. I hope to convince the reader that the principles and heuristics of minimalism are useful for investigating technical communication.

This document is directed to readers with some knowledge of the basic concepts and framework of technical communication. Previous knowledge of minimalist instruction literature is not assumed.

What is Minimalism?

The emergence of minimalist instruction in the 1990's will be seen historically as a major milestone in the development of technical communication. 'Minimalism' is a catchy label. It is used in a variety of other fields: architecture, interior design, literature, music and linguistics. Minimalist instruction in technical communication has a specific meaning that is not readily inferred from knowledge of other fields.

The primary focus of minimalist instruction is in training manuals and training course material for computer applications and software packages. There is general debate about the degree to which these guidelines can, or should, be used for other document types. A thorough understanding of minimalism principles is a precursor to examining issues of theory and application.

Principles and Heuristics.

Van der Meij (1995) provides a comprehensive overview of the four major principles of minimalist instruction and the corresponding heuristics. See Table 1 - Principles and Heuristics of Minimalist Instruction for details. These are guides for thinking about instruction rather than prescriptions. There will always be a need to adapt the design to particular situations.

Principle	Heuristic
<p>1: Choose an Action-oriented Approach People trying to learn a skill are eager to act, to do something meaningful. To learn to do it may be psychologically necessary to act. Minimalist Instruction is always action oriented.</p>	<p>1.1: Provide an Immediate Opportunity to Act Minimalist Instruction designers must invite users to act and support their action. Give the reader less to read but more to do. Users are never convinced by talk alone.</p> <p>1.2: Encourage and Support Exploration and Innovation Users should always feel in control of their activities. These are not unguided explorations. Strive for a balance between open-ended activities and meaningful projects. Use language which invites users to explore. Consider users when offering suggestions. Focus on student evaluation more than expert evaluation</p> <p>1.3. Respect the integrity of the User's activity. The moment-to-moment goal of the user may be unsophisticated and short-term.</p>
<p>2. Anchor the Tool in the task Domain. An application is a tool to achieve an objective for which the application is designed. Non-minimalist instructions are often written as if the tool were the user's objective. Instruction tasks are selected from the core tasks of the application domain.</p>	<p>2.1: Design Instructional Tasks that are Real. Instructional activities should be instantly recognised as genuine, but they may be modest. Users must have prior experience of the task domain.</p> <p>2.2: Create Components of Instruction that reflect the task structure. Use headings to convey many procedural elements of instructional tasks. Headings also help users locate information for reference purposes.</p>
<p>3: Support Error Recognition and Recovery. Learners spend 25-50% of time making and recovering from errors. Reducing mistakes and aiding detection diagnosis and recovery will reduce frustration. Mistake categories are Semantic, Syntactic and Slip. Usability testing identifies where users encounter errors. Provision of error information takes a special place in Minimalist Instruction; more is better. User mistakes help the learning process.</p>	<p>3.1: Prevent Mistakes Whenever Possible. The best way to remedy some mistakes is to prevent them, by including hints in the manual, by rewriting sections, or by blocking.</p> <p>3.2: Provide Error Information when Actions are Error prone or Correction is Difficult. Recovery from errors can be by reconstruction or correction. Reconstruction implies recovering from a backup state. Correction implies fixing without backtracking.</p> <p>3.3: Provide Error information that supports Detection, Diagnosis and Correction. Detection can be triggered internally or externally. In diagnosis the user decides what type of error was made. In correction the user sets a new goal. Minimalist Instruction manuals support all these processes.</p> <p>3.4: Provide On-the-spot Error Information. Error information should be as close as possible to where wrongly executed actions happen. Users exploit and explore error correction out of curiosity.</p>
<p>4: Support Reading to Do, Study, and Locate. Readers do not systematically process Minimalist Instruction from start to end. Sometimes they read to study, sometimes read to locate, but mostly they read to do. A small group will read the manual from cover to cover. Others start at the beginning but abandon it for random browsing. Another group uses the manual as a last resort when stuck. Avoid giving the manual a massive appearance; minimise the content.</p>	<p>4.1: Be Brief; don't spell out everything. Users are not seeking explanations for their own sake. Create chapters of two to four pages that take a short time to work through. Omit information that is easily inferred. Don't give full screen information. Brevity communicates that the task is not difficult and stimulates users to think and use prior knowledge.</p> <p>4.2: Provide Closure for Chapters Chapter independence is not possible, but chapter closure helps. Provide a home base for starting and ending tasks. Repetitions often concern fundamental tasks.</p>

Table 1 - Principles and Heuristics of Minimalist Instruction

Origins of Minimalism

The rapid development of personal computers in the 1980's put new demands on software designers, technical writers and readers. One example of the severe criticism of software documentation came from Norman (1988). 'The word processor I use on my home computer comes with a 340-page reference manual, plus a 150-page introductory manual intended for first-time users.' He identified creeping featurism as an odious problem. 'If you set out to make something difficult to use, you could probably do no better than to copy the designers of modern computer systems.' Creeping featurism ensured that there would be a learning need for each new release of each new software product. Word processors were one of these product types.

Problems with Training Manuals

Minimalist instruction arose from empirical research at IBM. Carroll (1984) carried out extensive one-on-one observation of trainee word processor operators. He classified the problems people encountered when confronted with a training manual:

- Information overload leads to perceptions of being overwhelmed.
- Jumping the gun. 'Learners ... don't appreciate overviews, reviews, and previews; they want to do things.'
- Skipping around instead of following lessons in the designed sequence. Learners have personal agendas and seek out information to achieve personal goals.
- Drawing inferences by invoking models and constructs; ignoring material that contradicts these precepts.
- 'Nose-in-the-book syndrome'; the reader ignores feedback from the target system and concentrates on the text.
- Problems recovering from errors and unexpected system responses. Training texts assume that the exercises are followed exactly and errors will not occur. The reality is otherwise.
- Impatience; trainees want to do real work that is meaningful to them as soon as possible.

Initial Guidelines

Carroll (1984) initially devised 5 guidelines to developing more efficient training materials:

- Slash the verbiage. The learning target is not obtained by adding extra headings and explanations.
- Force coordination between the system and the training. Make the reader pay attention to the system being learnt.
- Expect readers to make mistakes and ensure that a recovery path is available.
- Focus on real tasks and activities that the user relates to.
- Allow the learner to lead. Encourage trainees to experiment and discover.

In 1990 a full volume on designing minimalist instruction was released with the intriguing title *The Nurnberg Funnel* (Carroll 1990). The title refers to a mythical device to makes people very wise quickly. *The Nurnberg Funnel* was the subject of over 60 reviews, general discussions and applications papers in following years (Carroll, 1998).

Carroll and van der Meij restated the minimalist guidelines as a set of principles and heuristics in 1995 (van der Meij, 1995).

Minimalist instruction was a radical departure from standard thinking on computer system instruction. Draper identifies three stances taken by software developers in the past.

1. Tell users nothing. Let them work it out for themselves.
2. Documentation is a designer's brain dump. Give the user as much information as possible with little consideration of how to use it.
3. Documentation is organised feature by feature. This makes it easier to maintain.

The size of training manuals for new software product releases in the 1990's reducing substantially, often down to the 60-100 pages.

Theoretical issues of Minimalism

Minimalist instruction draws upon technical communication theory, despite being founded on empirical research. Many researchers have examined the principles and heuristics to see how they interact with the current and past theories.

Rhetorical Goals

Most of the text in a minimalist instruction document consists of stepwise procedures instructing the reader what actions to perform. Farkas (1999) has examined rhetorical discourse in instruction procedures. Tutorials are examples of rich-step and paragraph format procedures. They combine the cognitive goal of retention with the rhetorical goal of generating confidence. Writers frequently include explanations, previews, reviews, feedback, questions and comments. Writers tend to use streamlined step procedures by default.

Farkas also examines the theoretical construction of procedures into a framework involving actions and states, a state transition model. Actions may be classified into human actions, system actions and external events. States are divided into prerequisite states, desired states, interim states and unwanted states. The emphasis on errors and error recovery in minimalist instruction is a proactive recognition of unwanted states. Desired states are goals of users interacting with a text and a product.

Procedures exist in a social context, implying that all procedural discourse is rhetorical. The vocabulary must fit the audience. The writer has to find the best framework for describing actions and states to the reader. Writers must establish credibility. The text must persuade the reader that solution offered is worth the effort invested. Procedures inherently dramatise an implied author (persona) and an implied reader. Procedures may be brief, or more verbose in order to engage the reader.

Learning Styles

Minimalist instruction relies heavily on reader's ability to learn from interaction with a product. The heuristics seem to assume that all readers have a learning style of action oriented learning. Redish (1998) believes minimalism's most important contribution is the finding that cognitive psychology theory and research apply well to documentation. Users try to construct meaning, rely on prior experience, follow their own hypotheses, are eager to act, and are motivated to accomplish real tasks. 'My own experiences and research support the constructivist, action-oriented principles.'

But users are not all the same. There are differences in learning mode and doing mode, different personalities and learning styles, and users work in different problem domains. Minimalist Instruction is primarily concerned with instruction of the novice to a new application or tool (tutorials and getting started guides.) Reading to learn to do. Users are not always in learning mode, not even learning to do mode. Often users want to do their own work; they want to use manuals or online help for reference. Sometimes users go to user's manuals for conceptual information.

Redish asserts that what users want in reference manuals is; action orientation, anchored in the task, error recognition and recovery, support for location and doing, and modularity. The

list is almost identical to what users want in training manuals, what minimalist instruction manuals provide.

Reading Purpose

The distinction between training manuals and reference manuals is commonly considered to be ‘reading to learn’ versus ‘reading to do’. Shriver (1997) argues that all ‘reading to learn’ is ‘reading to learn to do’.

Some texts are prototypically seen as ‘reading to learn. ‘A question arises over what is meant by “doing.”’ Doing includes making an informed decision and taking practical action ... In cognitive terms, texts that promote ‘reading to learn to do’ help readers acquire a mental model of ‘how it works’ and a procedural model of ‘how to do it.’

An analysis of the learning process shows that all ‘reading to do’ involves learning (Coe). The reader must at least use short-term memory to apply content of a text to a task. There will often be something retained in long-term memory as a result of reading. ‘Reading to do’ is also ‘reading to learn to do’.

Draper observes that all human activity is a mixture of learning and doing, drawing on internal memory and external information sources. Sometimes learning continues despite repeated actions. Some information remains unlearned despite daily exposure.

Minimalist instruction assumes that the reader ultimately wants to learn to use a tool to do a task. The best way to learn to use the tool is to start doing something with the product being learnt.

Problem solving by Heuristics

Carroll (1998b) has tried to correct the many misconceptions about minimalism that have arisen. Many of these involve a misinterpretation of the word ‘heuristic’. The heuristic statements in van der Meij (1995) appear to be simple design guidelines; many people attempt to use them as such. I don’t believe this was the authors’ intention.

‘Heuristic’ has different meanings in differing contexts and differing fields of study.

- In general parlance a heuristic is a speculative guide to solving a problem (Coe).
- A heuristic is an educational method involving learning from discoveries as a result of investigations by a student.
- In computer science ‘heuristic’ relates to a problem solving technique involving alternative methods in successive stages. (The American Heritage Dictionary of the English Language.)

Writers who attempt to apply ‘minimalism’ to technical documents should consider using the computer science understanding of the term. Each of the eleven heuristics can, and should, be applied to the document design. A limited set of these heuristics should be applied and usability tested in each iteration of the design.

Applying Minimalism

User Centred Design

One key component to minimalist instruction has ensured that both academic and non-academic technical communicators take the approach seriously. The approach is user centered; it concentrates on the needs of the reader.

Draper (1998) offers a derivative statement of minimalism in which a hierarchy of principles is established. The fundamental principle is that documents are developed with a user orientation. This commits the developer to iterative design and usability testing. At level 2 the developer decides on an action-oriented manual. The reader will interact with the product while reading the text. The next level involves organising the manual to support doing tasks

and as a resource, not a textbook. Draper places van der Meij's principles and heuristics at levels five and six respectively.

Manual Design using Heuristics

The practicing technical writer can utilise the principles and heuristics in manual development. The introduction of 'Getting Started' manuals for many mass market software packages in recent years is evidence that many writers are doing so.

Redish asks if writers using the minimalist heuristics are doing so properly. She cites examples where headings are not using the end-users terminology, but producing 'pseudo-user oriented headings'. Some manuals are concentrating on tasks at a very low level when users need help to move from a goal to performing tasks. Redish believes that overviews are required to meet this need.

There is a strong case for retaining indexes and tables of contents. Indexes help the user locate a particular process or step, as they are likely to do when revising or using the instruction manual for reference. Tables of contents provide a task overview. The reader starts building a conceptual schema of the product from the manual front matter before starting the first chapter.

Usability Testing

Some researchers have attempted to verify that following the principles and heuristics produces better, or worse, manuals than other approaches. Redish (1998) conducted an experimental study of usability of five manual variants. Some manuals included overviews and others did not. Each manual variant was used by a group of 10 subjects. The total time to carry out 11 tasks using the manual was the same for groups with and without overviews. Redish found that the latter tasks were done quicker by groups which had access to overviews. This was interpreted to mean that manuals with overviews have better long term learning outcomes than 'minimal' manuals without overviews.

I believe that Redish's analysis is flawed. There is no statistical variance or significance testing in the report. The fact that the last few tasks were done in less time does not prove that the user is better off with overviews. Redish appears to have selected a subset of the data to support a prior expectation.

Conclusions

Minimalist instruction is important to all technical writers for two reasons. Firstly, as a practical and pragmatic tool for planning, designing and developing manuals and training materials. Even in cases where the project is for a reference manual, minimalist heuristics need to be considered because learning and doing are inherently intertwined. Secondly, minimalist instruction confronts the major theoretical concerns of technical communication. Writers who are aware of the minimalist principles and heuristics will automatically take a user focussed, action oriented approach, to document development. They must confront usability testing to achieve a quality product. They must be aware of the rhetorical goals of the text and sub-text.

Bibliography and References

- Carroll, J. M., 1984, Minimalist Training. *Datamation*. Vol 30, No 18, P125-136.
- Carroll, J. M., 1990. *The Nurnberg Funnel: designing minimalist instruction for practical computer skill*. USA, MIT Press.
- Carroll, J. M., editor, 1998. *Minimalism Beyond the Nurnberg Funnel*. Cambridge, Massachusetts Institute of Technology Press.
- Carroll, J.M., 1998a. Reconstructing Minimalism. In: Carroll, J.M., editor, *Minimalism Beyond the Nurnberg Funnel*. Cambridge, Massachusetts Institute of Technology Press.
- Carroll, J.M., 1998b. Ten Misconceptions about Minimalism. In: Carroll, J.M., Editor, *Minimalism Beyond the Nurnberg Funnel*. Cambridge, Massachusetts Institute of Technology Press.
- Coe, M., 1996. *Human Factors for Technical Communicators*. USA, John Wiley & Sons.
- Hackos, J.T. 1998. *Choosing a Minimalist Approach for Expert Users*. In: Carroll, J.M., Editor, *Minimalism Beyond the Nurnberg Funnel*. Cambridge, Massachusetts Institute of Technology Press.
- Norman, D. A. 1988. *The Design of Everyday Things*. New York: Basic Books.
- Redish, J, 1998. Minimalism in Technical Communication: Some Issues to Consider. In: Carroll, J.M., Editor, *Minimalism Beyond the Nurnberg Funnel*. Cambridge, Massachusetts Institute of Technology Press.
- Shriver, K., 1997. *Dynamics in Document Design*, John Wiley & Sons, USA.
- The American Heritage Dictionary of the English Language*. 1996. Third Edition, USA, Houghton Mifflin Company.
- Van der Meij, H. 1995. "Principles and Heuristics for Designing Minimalist Instruction." *Technical Communication* 42, No 2: 243-261.

