Human-Computer Interaction -- INTERACT'03
M. Rauterberg et al. (Eds.)
Published by IOS Press, (c) IFIP, 2003, pp. 415-422

Layering a Minimal Interface

Frank Vetere & Steve Howard

University of Melbourne, Department of Information Systems
111 Barry St, Carlton 3053 AUSTRALIA

{fv, showard}@unimelb.edu.au

Abstract: This paper reports on a software tutorial designed according to minimalist principles. The design extends minimalism beyond paper-based instructions to computer-based interface, and adopts a user-initiated layering technique to display relevant information. Evaluation confirms minimalist theory and shows that layering helps to alleviate some of the design tension between supporting learning and encouraging doing.

Keywords: Minimalism, minimal interface, layering, learning, doing, action-orientation

1 Minimalism

1.1 Minimalist principles

Minimalism (Carroll, 1990) is a design approach that aims to support users in their meaningful activity. The minimalist approach is a response to the "paradox of sense-making" (Carroll & Rosson, 1987). This paradox occurs when learners are required to access instructional materials in order to undertake novel activity, but they are often too busy (exploring, trying to relate what they already know and trying to recover from errors) to bother with the instructions.

Minimalist principles have evolved and been presented in various forms over the last decade (Draper, 1998). Van der Meij and Carroll (1998) have re-described minimalism as consisting of four broad design principles with corresponding heuristics for each principle (Figure 1).

The first principal recommends supporting learners' immediate desire to act. People trying to learn are usually eager to act and to do something meaningful. The second principal acknowledges that for most users a set of instructions is not an end in itself, but a tool through which to achieve broader objectives. Consequently, instructional activities should be instantly recognisable as relevant. Principal three is a response to the observation that the eagerness to learn can be a source of errors. Learners can spend a large proportion of their time correcting errors. When it is not possible to prevent errors, information should be added to support error detection, diagnosis and correction.

Principal four recommends that instructional design be brief and direct. Learners are generally not seeking explanations for their own sake, but mostly "reading-to-do" (see Van der Meij & Carroll, 1998, for a complete discussion).

1. Choose an action-oriented approach
  a. Provide an immediate opportunity to act.
  b. Encourage and support exploration and innovation.
  c. Respect the integrity of the user's actions.

2. Anchor the tool in the task domain
  a. Select or design instructional activities that are real tasks.
  b. The components of the instruction should reflect the task structure.

3. Support Error Recognition and Recovery
  a. Prevent mistakes whenever possible.
  b. Provide error information when actions are error prone or when correction is difficult.
  c. Provide error information that supports detection, diagnosis, and recovery.
  d. Provide on-the-spot error information.

4. Support reading to do, study and locate
  a. Be brief; don't spell out everything.
  b. Provide closure for chapters.

Figure 1: Four major design principles and their heuristics for designing minimalist instruction (Van der Meij & Carroll, 1998)

1.2 Minimalism at the interface

Minimalism is often discussed in terms of its implications for HCI (Carroll, 1998), but there are very few published examples where minimalism is used for interface design specifically.

Minimalism was originally investigated as an approach to documentation design. This is exemplified by the early minimalist research (i.e. the minimal manual, Carroll, Smith-Kerker, Ford, & Mazur-Rimetz, 1987). However much of the subsequent research has continued with paper-based user-manuals, rather than for designing computer interface directly.

Later examples of minimalism concerned instruction manuals (e.g. telephony, Carroll, 1990) and tutorial systems (e.g. WordPerfect, Lazonder & Van Der Meij, 1993). These examples of minimalist design were good at motivating action but they were mostly paper-based and tended to suffer from coordinative problems or split attention effects (Sweller, Van Merrienboer, & Paas, 1998). The paper-based manual is intrinsically separate to the computer system that it is trying to support. Even though the minimal-manuals are grounded in real tasks, the degree to which they can be task-oriented has natural limitations because the relevant sections of the manual are not immediately present or integrated with the task.

Examples of minimalism in interface design are given by Farkas (1998). His examples are computer-based, but they exemplify only part of the minimalist approach. Help manuals alone (be they electronic or paper) tend to miss the critical action-oriented nature of minimalism. Help systems assist users to recover from errors and to clarify misconceptions (a key principle of minimalism), but minimalism is more than a help system.

Not surprisingly, the best examples of minimalism at the human-computer interface are found in Carroll's own work. Carroll used a minimalist approach to create 'View Matcher' and 'Molehill' (Carroll, 1990). Both products provide integrated computer-based instruction for Smalltalk. These applications are action-oriented and task-based, and don't suffer from coordination and integration problems common in paper-based manuals.

1.3 Layering

Minimalism is not without its critics. Some of the criticisms suggest that minimalism is inappropriate for knowledgeable learners (Hallgren, 1992), that it is unsuitable for complex domains (Horn, 1992), and that it tends to support doing at the cost of real learning (Draper, 1998). Most of these criticisms (or 'misconceptions') have been addressed directly in Carroll & Van der Meij (1998).

Other criticism is given by Farkas (1998), who sees minimalism as a risky approach to design. He is concerned that by reducing the amount of information, (i) there may not be enough information to complete the task; or (ii) an unreasonable amount of effort may be needed to complete the task; or (iii) the user may complete the task but form an incorrect mental model of the system. Even though minimalism is more than simply reducing the information and "slashing the verbiage" (Draper, 1998), Farkas highlights a key dilemma. How much is *just* enough information? Farkas proposes a layering technique to help deal with this dilemma.

Layering is fundamentally about choice. It involves providing information that is optional yet easily accessible. Layered information is displayed to those who choose it and so can accommodate varying demands. The information can be simple for novices or advanced for experts, or configured for a special need (e.g. technical information for IT managers). The layering technique creates a type of configurable display comparable to popup windows (Bétrancourt & Bisseret, 1998). Layering is not easily incorporated into paper-based manuals, however it is uniquely suited to computer-based instructions.

Two examples of layering given by Farkas are Apple Macintosh Balloon help, and Windows 'What's This?' help topic, which activated by a right click. In both examples, explanations are only displayed on the user's command. Layering helps deal with user diversity because explanations are only called upon when needed. Layering may also be useful for dealing with changing information needs during an activity.

Farkas proposes layering as a way of addressing an inadequacy in minimalism. He suggests that layering provides a 'safety net' to those for whom minimalist design is not suited. The implication is that a layered design cannot be a minimalist one. However we believe layering is not contrary to minimalist ideas. The superordinate minimalist principle is task orientation (Carroll & Van der Meij, 1998). If layering genuinely supports task success, then it too can be included within the broad minimalist rubric. We view layering as a useful technique within the minimalist toolkit, not an addendum.

1.4 Aim of the study

The aim of this paper is to report on an example of computer software whose development has been directly motivated by minimalism. The software is computer-based (unlike earlier minimalist research), action-oriented (unlike many of the help systems) and integrates task and instruction (unlike paper manuals). The minimalist design incorporates layered information to help deal with audience diversity and changing information needs.

We then report on an evaluation of the design. Instead of comparing a minimalist design with a non-minimalist one (as in Lazonder & Van Der Meij, 1993), our analysis evaluates the degree to which learners participate in action behaviour and make use of the layered information. The evaluation is diagnostic rather than comparative..

2 Designing the Minimalist interface

2.1 Starter Circuit

The software used in our study was an instructional tutorial (Figure 2). The instruction concerned the function and operation of the 'starter circuit' (first presented as a different design in Kalyuga, Chandler, & Sweller, 1998). The starter consists of a pair of push buttons. The Start button energises a coil, which then closes a switch and creates a holding circuit to keep the current flowing. Learners confronting this circuit for the first time often find it challenging (Vetere & Howard, 2000). Understanding the current flow under various configurations and the relationship between the components is a non-trivial exercise.

Furthermore, the starter can be easily represented graphically, and easily lends itself to action-oriented representations because the physical device has push buttons for operation. Hence the starter circuit provides a useful vehicle for investigating minimalist design.

The software tutorial was consistent with minimalist principles and heuristics. However principles alone do not make a minimalist design "Given a set of minimalist principles, we cannot just crank out a training manual" (Carroll, 1990, p.91). A minimalist design must be also consistent with an overall philosophy that aims to continually support learner-directed activity, and minimise the degree to which instructional materials may get in the way of accomplishment.

There are no formal methods for establishing and verifying minimalist design. There is no dedicated task-analysis to help realise minimalist principles (Horn, 1992). However like other design practices, minimalism relies on an iterative process of observation and reflection (Carroll, 1998).

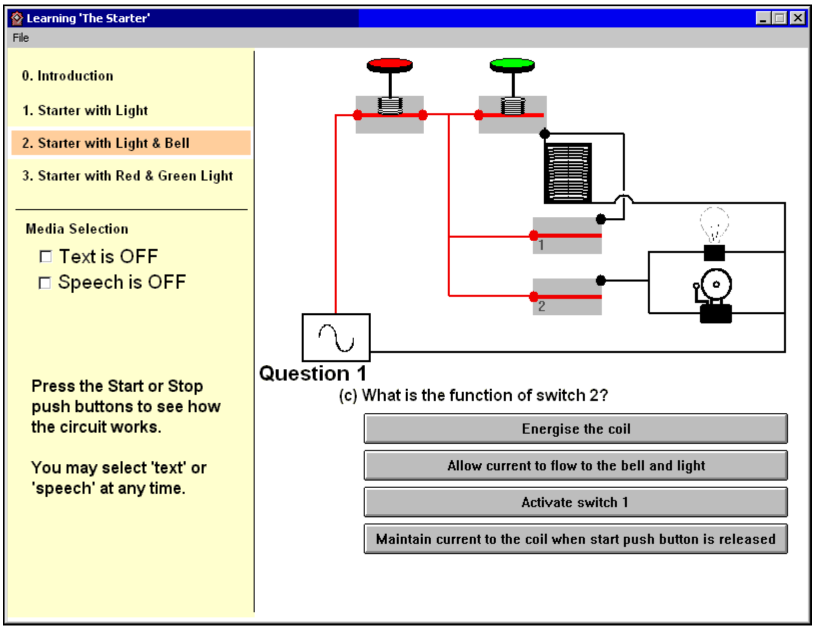

Figure 2: A screen shot of the Starter Circuit tutorial

The starter-circuit design was the result of extensive observation and analysis in prior studies (Vetere & Howard, 1999, 2000). Conceptual difficulties and common sources of errors were observed and identified over several iterations. Difficulties were clarified and errors were supported in the new design.

Moreover, through the process of iterative design, it was possible to remedy basic usability problems such as navigation, feedback and visual clarity. The layout of the circuit was simplified and the functions of the components their relationships were made visually explicit (Vetere, 2002). Improving the usability of the software before undertaking the study was important, though uncommon in minimalist studies. The validity of our outcomes is improved because results are more likely due to minimalist factors rather than confounded by general usability problems.

The final design was refined through use and analysis, with minimalism as the engine for redesign.

2.2 Action oriented approach

The instructional design adopted an action-oriented approach. Learners were invited to act on the circuit and encouraged to explore early in the tutorial (e.g. "Press the Start or Stop push buttons to see how the circuit works"). Learners could press the push buttons at any time and see an immediate response. There was no need for 'background information' explaining the underlying concepts. When learners clicked the start push button, an animation sequence would begin that showed the current energising the coil and closing the switch. User action was respected because it was integral to the instruction.

2.3 Anchor tool in the task domain

The tasks demanded by the tutorial were not abstract or hypothetical. Users had clear problem descriptions (e.g. "What is the purpose of the coil?"). They had to interact with the circuit in order to find the answers to specific questions. The explanations and the task were integrated into one display. The cognitive burden of dividing attention between 'what-to-do' and 'how-to-it' was greatly reduced. There was no need to refer to a separate instruction set on paper or another screen to achieve task success. The instructional tool was clearly anchored in the task at hand.

2.4 Brevity

An important aspect of minimalism is brevity (though brevity is often mistakenly considered the totality of the minimalist approach, see Carroll, 1998). We aimed to achieve brevity by removing all explanations and then layering minimalist instructions when initiated by user action.

Previous studies showed that it is was possible to facilitate learning new concepts without detailed explanations (Vetere & Howard, 2000), provided: (i) the design encourage opportunities to act (by creating an interactive system); (ii) the interactive system is integrated with relevant tasks; and (iii) user error is well supported.

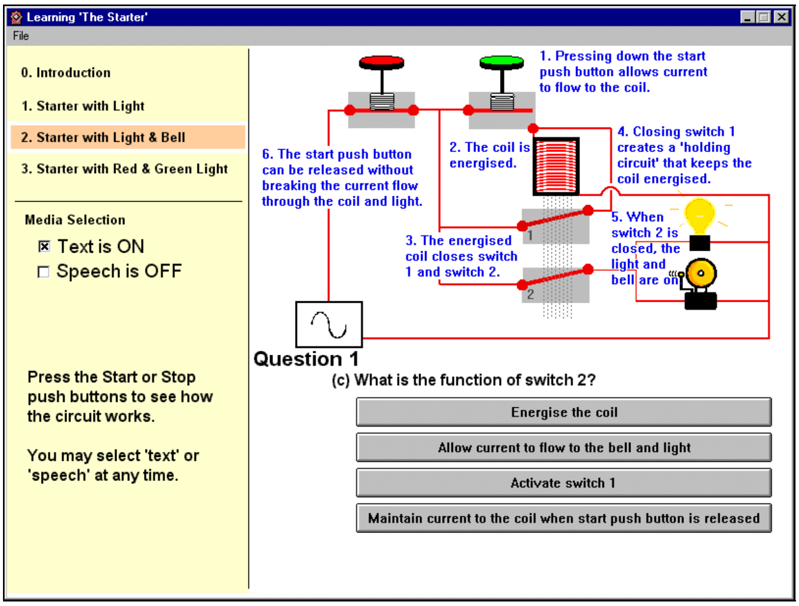

Figure 3: A screen shot of the Starter Circuit tutorial with layered explanations

In other words brevity is not the source of minimalism, but rather the result of an action-oriented approach.

In order to deal with a varied audience and allay the risks of insufficient information, the interface design was layered with relevant explanations (Figure 3). The layered explanations were given only when the relevant check box was selected.

The explanations were divided into small compact sentences. The textual explanations were integrated with the appropriate animated sequence. The layered information explained impact of the current flowing through the starter circuit. The explanations were available either text or speech or both, depending on the user choice. When the speech check box was selected, a spoken explanation was heard as the circuit was animated. The animation and the speech were temporally coordinated, so that the relevant explanations were when the corresponding animations occurred. The text was a literal transcript of the spoken words.

2.5 Error recognition and recovery

In adopting the action-oriented approach, users were now able to act in 'inappropriate' ways. Users could click the start push button while the coil was already energised, and click the stop push button when the coil was not energised. In both cases, there was no result on the status of the circuit. If such a circuit existed in real-life, these actions would be possible but would have no discernable effect. Rather than excluding such behaviour, or issuing warnings against it, our software made this action possible. The ability to press the button and see no change was in itself an opportunity to better understand the system. This is an important design consequence arising directly from minimalism.

Users could of course seek an explanation of this non-event by revealing the layered explanations. It was necessary therefore, to introduce two statements; "The coil is already energised, so pressing the start push button has no effect." and "The coil is not energised, so pressing the stop push button has no effect." The statements correspond to error information rather than content information. The inclusion of these additional statements is consistent with minimalist design principals promoting the support of error recognition and recovery. The minimalist approach does not simply involve the removal of expository material; it often requires addition of error information (Carroll, 1990).

Descriptions of this type would not normally appear in traditional instructional materials. There would to be little point in describing actions that do not change the state of the system (especially since the actions are not initiated by the learner). The ability to act upon the system in whatever way is meaningful to the learner, even if this seems illogical or irrational to the designer, is an inevitable consequence of an action-oriented approach.

3 The Study

3.1 Method

We adopted a within-subject design in a type of quasi-experiment (Kirk, 1995) with no randomly assigned independent variable.

Nine participants were paid $15 each to be part of the experiment that lasted approximately one hour. The six males and three female participants were all studying postgraduate degrees (mostly PhD). The average age was 27.6 years (std dev 4.8). All participants were subjected to the same experimental condition.

The behaviour of each participant was individually observed and recorded. Data collection consisted of written notes (articulations and behaviour), video recording in usability lab (quad-display showing screen and three room views) and the electronic data logging. Three sets of data were electronically logged and time-stamped; selection of layered explanations, pressing of action buttons and answers to questions. The use of layered explanations is reported in this paper.

3.2 Materials

The software was written in Macromedia Authorware and provided instruction into the operation and function of the starter circuit. The Starter circuit contained stop and start push buttons and a coil that triggered a switch, which created a holding circuit for the current. The starter circuit was incorporated into three configurations: Circuit 1 - Starter with Light; Circuit 2 - Starter with light and bell; Circuit 3 - Starter with red and green light. These three circuits provided the basis of the tutorial.

Participants were asked to answer 4 questions for each circuit. These were (i) multiple-choice questions; (ii) extended-answer question; (iii) design question; (iv) fault-finding question. Each question was scored out of three, giving an overall score out of 12.

Participants could choose to display or hide the explanatory text or speech at any stage of the experiment. The instructional circuit was available for participants to explore throughout the experiment, including while answering the questions.

4 Results

4.1 Data

A summary of the results is shown in Table 1. The table shows the use of layered information (text and/or speech) for the three circuits.

For each circuit, the column is divided into ‘exploration’ and ‘task’ stage. In order to encourage exploration, participants were able to investigate the circuit before beginning to answer the questions. Many participants used this opportunity to become familiar with the operation of the starter circuit. Once they completed this exploration the participants began answering the questions (i.e. the task stage).

The rows show the use of the four types of layered explanations. The un-layered row includes both ‘true un-layered’, (where explanations were not selected) and ‘effective un-layered’ (where layered explanations were selected, but no information was displayed). Several participants quietly studied the circuit, thinking about the question without pressing any buttons. Since the participants did not interact with the circuit no information was displayed. Data from this ‘effective un-layered’ interaction is incorporated in the un-layered row.

Each row is subdivided into the number of users of the layered formats (e.g. 3 users explored circuit 1 without any layered explanations, whereas 5 users had text/speech layers) and the average time per user for that format (e.g. each of the 3 users spend on average 1 min 29 sec exploring circuit 1 without explanations). Where participants used more than one exploratory format, the format to which the participant devoted the most amount of time was the one counted.

During the task stage, the number of users per format was averaged across the four questions (e.g. on average, 5.8 participants used the un-layered format for completing the questions in the first circuit). The time refers to the average time per format on the task stage (e.g. each participant took on average 7 min 16 sec to complete the tasks with the un-layered format).

The total score for each format is shown as a percentage

4.2 Discussion

The data suggest important trends, even though low subject numbers prevented statistical comparisons. The results indicate that layering helped users configure the minimalist display to better support their changing needs during the learning activity.

During the exploratory stage, most of the nine participants chose to layer their circuit with some explanations. Six, nine and nine participants (for each circuit respectively) felt that they needed text and/or spoken explanations in order to better understand the circuits.

		Circuit 1		**Circuit 2**		**Circuit 3**	
		Exploration	Task stage	Exploration	Task stage	Exploration	Task stage
un-layered	number of users	3	5.8	0	4.8	0	6.5
	time (m:ss)	01:29	07:16	00:00	10:03	00:00	12:04
	task score		76%		74%		76%
text-layer	users	1	0.5	1	2.0	2	1.7
	time per user (m:ss)	03:29	02:19	01:07	15:59	01:25	10:43
	task score		42%		78%		67%
speech-layer	users	0	0	1	0	0	0
	time per user (m:ss)	00:00	00:00	02:00	00:00	00:00	00:00
	task score						
text & speech layer	users	5	2.7	7	2.2	7	0.8
	time per user (m:ss)	02:12	15:24	03:26	23:57	02:27	04:38
	task score		47%		69%		33%

Table 1: Results showing the use of layered information for the three circuits.

Even in the last circuit, when learners had gained considerable knowledge about the starter, all participants chose to explore the circuit with layered explanations.

However once the participants felt sufficiently confident with their understanding to begin answering questions, most of them did not want layered information. 5.8, 4.8 and 6.5 participants (for each circuit respectively) chose an un-layered display during the task stage. This behaviour suggests that users preferred to avoid explanations when they were task focussed. Furthermore those that had un-layered displays while undertaking tasks obtained the highest scores (76%, 74%, 76%). However, is should be noted that it is unclear whether the results are due to the display directly, or because those choosing layered explanations understood the circuit less and so obtained lower scores.

5 General Discussion

Minimalism recommends removing material that does not directly support the immediate task. Our experiment provides additional empirical evidence for this recommendation. Just like Carroll, we observed task-oriented learners being too preoccupied to use explanations. Users wanted to be left alone to devote their mental energies to completing the task.

In contrast, explanations became important when participants believed they were learning rather than doing. This behaviour was most noticeable when participants' knowledge or ability was challenged. When the learning the concepts became difficult (or when the users simply expected the material to be difficult), the participants understandably called upon explanatory information.

This behaviour is not surprising. What is unique however is a computer system that supports the changing information needs throughout the interaction. By creating a minimalist design with user initiated layering there was support for both learning and doing.

The tension between learning and doing has been identified as a major practical problem in minimalist design. "Are we trying to get users to succeed at the immediate tasks or to learn the maximum?" (Draper, 1998, p.352). This tension is exemplified by software wizards that are used to automate tasks. Wizards are reported to be very successful at achieving task success, but not at helping users to learn to do the task on their own (Redish, 1998). In an educational context, instructions intending to help users complete certain learning tasks quickly and accurately should be ideally designed to support task success; whereas instructions intending to help learners understand the basic concepts and principles (for retention or generalisation) should be designed to maximise pedagogical goals. Draper (1998) acknowledges that this dichotomy between learning and doing is somewhat spurious and that nearly all human activity involves both learning and doing. However, he also says that simply acknowledging an indelible link between learning and doing does not mitigate the genuine design dilemma.

The results of our study suggest that user controlled layering of information could be used to support learning, and then withdrawn when cognitive resources are directed towards the task. The layering technique helped to deal with the problem of deciding upon just the right amount information at the right time. Users were able to reveal the information they wanted, when they wanted it. The ability to control the display was crucial in managing their changing needs.

6 Conclusion

There is very little research on computer systems that use minimalist principles for integrated software designs. This paper provides a useful contribution to facilitating the progression from minimal-manual to minimal-interface. Furthermore we show that layering a minimalist design can alleviate some of the design tension between learning and doing.

Our study raises several questions for future work. For example:

- What type of task analysis is best suited for collecting knowledge about doing and learning?
- What specific interface design techniques are appropriate for supporting action-orientation and error recognition?
- What are the appropriate interface dialogues for layering that involves multiples layers of complex representations?

Further research would also need to broaden the scope of this study to undertake longitudinal studies to investigate the changing information needs as user expertise improves, and explore how these changing needs can be incorporated into a layered minimalist design.

References

Bétrancourt, M., & Bisseret, A. (1998). Integrating Textual and Pictorial Information Via Pop-Up

Windows : An Experimental Study. *Behaviour and Information Technology, 17*(5), 263-273.

Carroll, J. M. (1990). *The Nurnberg Funnel: Designing Minimalist Instruction for Practical Computer Skill*. Cambridge, MA: MIT Press.

Carroll, J. M. (Ed.). (1998). *Minimalism beyond the Nurnberg Funnel*. Cambridge, MA: MIT Press.

Carroll, J. M., & Rosson, M. B. (1987). The paradox of the active user. In J. M. Carroll (Ed.), *Interfacing thought: Cognitive Aspects of Human-Computer Interaction* (pp. 80-111). Cambridge, MA: MIT Press.

Carroll, J. M., Smith-Kerker, P. L., Ford, J. R., & Mazur-Rimetz, S. A. (1987). The Minimal Manual. *Human Computer Interaction, 3*, 123-153.

Carroll, J. M., & Van der Meij, H. (1998). Ten Misconceptions about Minimalism. In J. M. Carroll (Ed.), *Minimalism beyond the Nurnberg Funnel* (pp. 55-90). Cambridge, MA: MIT Press.

Draper, S. W. (1998). Practical Problems and Proposed Solutions in Designing Action-Centered Documentation. In J. M. Carroll (Ed.), *Minimalism beyond the Nurnberg Funnel* (pp. 349-374). Cambridge, MA: MIT Press.

Farkas, D. K. (1998). Layering as a Safety Net for Minimalist Documentation. In J. M. Carroll (Ed.), *Minimalism beyond the Nurnberg Funnel* (pp. 247-274). Cambridge, MA: MIT Press.

Hallgren, C. (1992). The Nurnberg Funnel: A minimal collection. *Journal of Computer Documentation, 16*(1), 11-17.

Horn, R. E. (1992). Review of The Nurnberg Funnel. *Journal of Computer Documentation, 16*(1), 3-10.

Kalyuga, S., Chandler, P., & Sweller, J. (1998). Levels of Expertise and Instructional Design. *Human Factors, 40*(1), 1-17.

Kirk, R. E. (1995). *Experimental Design: Procedures for the Behavioral Sciences* (third ed.). Pacific Grove CA: Brooks/Cole.

Lazonder, A. W., & Van Der Meij, H. (1993). The Minimal Manual: Is Less Really More? *International Journal of Man-Machine Studies, 39*, 729-752.

Redish, J. C. (1998). Minimalism in Technical Communication: Some Issues to Consider. In J. M. Carroll (Ed.), *Minimalism beyond the Nurnberg Funnel* (pp. 219-245). Cambridge, MA: MIT Press.

Sweller, J., Van Merrienboer, J. J. G., & Paas, F. G. W. C. (1998). Cognitive Architecture and Instructional Design. *Educational Psychology Review, 10*(3), 251-296.

Van der Meij, H., & Carroll, J. M. (1998). Principles and Heuristics for Designing Minimalist Instruction. In J. M. Carroll (Ed.), *Minimalism beyond the Nurnberg Funnel* (pp. 19-53). Cambridge, MA: MIT Press.

Vetere, F. (2002). *Redundant Multimedia at the Human-Computer Interface.* Unpublished Doctoral thesis, University of Melbourne, Australia, Melbourne.

Vetere, F., & Howard, S. (1999). *Redundancy Effects in Instructional Multimedia Systems.* Paper presented at the INTERACT '99: IFIP TC.13 International Conference on Human-Computer Interaction, 30th August-3rd September 1999, Edinburgh, SCOTLAND.

Vetere, F., & Howard, S. (2000). *Prior Knowledge and Redundant Multimedia.* Paper presented at the IEEE International Conference on Multimedia and Expo (ICME 2000), 30 July - 2 August 2000, New York, NY USA.