Chapter 1: Introduction

1. Aims

The aim of this thesis is to explore the use of cognitive and other principles in interface evaluation, by investigating the use made of evaluation material which can be shown to derive from such principles. Specifically, in contrast to Nielsen's heuristics, expanded evaluation principles will be compared with Nielsen's Heuristics, and general principles used in heuristic evaluation (Nielsen 1993, 1994d). The focus of these analyses was to be the usability problems identified by both novice and expert evaluators. The thesis is therefore an attempt to encompass a wider range of sources than in particular the application of cognitive psychology to interface design.

Thus the aims from the viewpoint of usability are as follows:

1. To investigate whether expanded evaluation principles can be used by both experienced and novice evaluators to identify a common set of problems with a given interface.
2. To investigate the claims made for Nielsen's heuristics that only a small number of evaluators may be required to uncover most of the problems with a Nielsen's heuristics alone.

This thesis differs from the viewpoint of interface evaluation in the following ways:

1. The focus of these analyses was to be the usability problems identified by both novice and expert evaluators.
2. The focus of these analyses was to be the usability problems identified by both novice and expert evaluators.

Chapter 1: Introduction
The fact that this proved to be quite difficult, and then only for restricted evaluation scenarios, makes up the first of the two themes to be explored in the thesis. The second theme concerns the predictive power of heuristics. The proponent of one widely used set, Jakob Nielsen (citations below), has claimed that using such a set of heuristics, suitably experienced evaluators can between them identify most of the usability problems with a given interface. The thesis will also explore the validity of this claim in respect of both novice and experienced evaluators. It will investigate the ability of both heuristics and principles to enable not only the identification but also the prediction of usability problems of both novice and experienced evaluators. If not, this thesis will also explore the validity of this claim in respect of both novice and experienced evaluators. If this is the case, most of the usability problems are explained by Jakob Nielsen's (citations below) heuristics. The second theme concerns the predictive power of heuristics. The proponent of one of these evaluators, Jakob Nielsen (citations below), has claimed that using such a set of heuristics is the best way of identifying most of the usability problems with a given interface. The thesis will also explore the validity of this claim in respect of both novice and experienced evaluators. If this is the case, most of the usability problems are explained by Jakob Nielsen's (citations below) heuristics. The second theme concerns the predictive power of heuristics. The proponent of one of these evaluators, Jakob Nielsen (citations below), has claimed that using such a set of heuristics is the best way of identifying most of the usability problems with a given interface.
Introduction

The deficiencies of guidelines, derived from cognitive psychology principles, are addressed in a study by Smith and Mosier (1986) on "Design Guidelines". The problem of navigating large guideline sets has been addressed by search and retrieval tools such as hyperSAM (1995) and Sherlock (2000). More recently, guideline management systems such as Guide and GUIDE aim to allow both the retrieval and application of online guidelines.

3.2 Principles

One approach to the deficiencies of guidelines is to back up each application or domain-specific piece of design advice with an underlying rationale from which it can be seen to derive. That is, to combine what Smith (1986) and Dix et al. (1998) called 'design rules' with a more general 'design principle' (Hammond et al. 1987, Dumas & Redish 1993, Preece et al. 1994, Sutcliffe 1995). Thus, each design specification, applied to a particular interface domain or component, would be accompanied by its appropriate principle. Each principle would be couched in sufficiently general terms to have application to a wide range of design problems, enabling designers to generalise from one problem to another.

The author, then, agrees with Hammond et al. (1987) that the intent of guidelines is to capture some psychological principle which is sufficiently invariant across a range of contexts for the guideline to remain useful. If they are to be much more than a basis for checklists (Thimbleby 1995), guidelines need to address the novel and integrative conceptual issues (Tetzlaff et al. 1991) which guidelines need to address the novel and integrative conceptual issues. The problem with guidelines is that they often address the longer-term issues of design, whereas more general psychological principles might be more directly relevant across a range of design problems. The principle set will be described below.

The format and content of the principles set will be described below. The formal and content of the principles set will be described below. The formal and content of the principles set will be described below. The formal and content of the principles set will be described below. The formal and content of the principles set will be described below. The formal and content of the principles set will be described below. The formal and content of the principles set will be described below.

3.2 Principles

Guidelines, such as Guide and GUIDE (1999), aim to allow both the retrieval and application of online guidelines, derived from cognitive psychology principles. These are arranged into a "category framework of relevant concepts" (ibid., p222).
om Consistency guideline (or principle) derives from such a set.

Thus making a set of all in this category. It will be clear in the next section that the author's
om Consistency of the situations where it holds. --
- If an action or product has to be used inconsistently, the user should be warned of the
- Domain (for instance, ‘ask application’, etc.).
- When consistency is not possible throughout the interface, consistency within the immediate
- Interactions with other elements: and those from consistency throughout the dialogue
- Interactions are not an issue.
- All procedures and communicative elements in a dialogue should have consistent properties, names

to 122 (Marshall et al. 1987 p267):

of the seven). However, the author would appear to the above pair guidelines 116 and 117
where these authors separately derived (the above sample is intended to illustrate just two
without most of the seven (of the 156 in Gardiner & Christie 1987) attitudinal guidelines
in the author's view, the content of the above example from Marshall et al. (1987) is sufficient

Unlike most guidelines, which would at best include only the last two parts of this example,

different contexts. The guidelines themselves are listed separately, referenced to the
more clear definition of the collected guidance(s) in compact form. Finally, some comments are then
statement of the collected guidance(s) in which the guidance(s) might apply in practice (usually office-related communicative tasks). Then a
principle(s) might apply in practice (usually office-related communicative tasks). Then a

Most of the Marshall et al. (1987) sensitive dimensions (guideline categories) have the
more on the content and formal of such guidelines than their organisation.

have implications for the usability problem identification process. The focus of this Section is
management: loops of control. This thesis will demonstrate that such categorisations can

test match; feedback, selection, terms, working and display, consistency, screen

name: design of procedures and tasks, analogy and metaphor, training and practice.
The guidelines of the Smith & Mosier (1986) framework is illustrated by the example in Figure 1.2. This is an adapted version of that created by Smith & Mosier (1986). There are clear differences in format. This is an adapted version of that created by Smith & Mosier (1986), but new material has been added.

The full format of the Smith & Mosier (1986) guidelines is illustrated by the example in Figure 1.2. It is similar to that used by Marshall et al. (1987), but includes exceptions and a more detailed explanation of the rationale behind each guideline. The guidelines are designed to be used in a variety of contexts, from software development to educational settings.

Figure 1.2. The guidelines of Consistency (Marshall et al. 1987 pp238-239).

Consistency

Consistency is fundamental to effective interface design. It contributes to usability in a number of ways. For example, it facilitates learning, lessens the number of errors made and helps the user to develop an accurate system model. It is rarely possible to be completely consistent within and between the applications of a system. So the designer will often have to determine the priorities of the system and make tradeoffs accordingly. For example, should the designer be rigidly consistent if the result is a slow and inappropriate procedure when an inconsistent procedure would achieve the goal more efficiently? The decision has to be made based on the function of the system and users for whom it is intended.

The guidelines in this category emphasise the importance of consistency in certain domains, and identify areas where tradeoffs should be made. For example, it is suggested that consistency in certain domains, such as user input, is more important than consistency in other domains, such as output. This is reflected in the different formats used in the different guidelines provided.
principles have more than one attribute (in such cases the attribute name is the same as the
and, where, and even sometimes, such as cognitive psychology, but which includes,
formal is does to what Stolfo (1995, pp-9) called an, configuring, principle - one which
of any guideline involves consideration of exceptions and trade-offs. The Smith & Mosier
encompasses a range of dialogue types including graphic interaction,
enumerous points (such as flexibility, menu-based, the scope of the collection
interfaces (thus mainly command-line and menu-based), the scope of the collection
1995). Through written for 1980s generation
data projection (from Hyperscan - Lunarline 1995), thought written for 1990s generation
numbers (data entry, data display, sequence control, user guidance, data transmission and
MIL-STD-1472C 1983 § 5.15.3.2.1 5.15.3.3.4
Foley Van Dam 1982
Stewart 1980
See also: 4.0/6 Consistent Display Format
Figure 1.2 Consistency 2.5/1 Consistent Display Format.

2 DATA DISPLAY
2.5 Format
2.5/1 Consistent Format
Statement
Adopt a consistent organization for the location of various display features on one display to
consistent. However, once a suitable format has been devised, it should be maintained as a
consistent. However, once a suitable format has been devised, it should be maintained as a
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
consistent, accommodating the different displays, and other areas dedicated to display of control options, instructions, etc.
Introduction

### User - System Principles

<table>
<thead>
<tr>
<th>Principle</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements Match</td>
<td>Functional Needs [1]</td>
</tr>
<tr>
<td></td>
<td>Requirements Needs [2]</td>
</tr>
<tr>
<td>Functional Utility</td>
<td>Functional Organisation [3]</td>
</tr>
<tr>
<td></td>
<td>Functional Provision [4]</td>
</tr>
<tr>
<td>Navigational Effort</td>
<td>Navigation [5]</td>
</tr>
<tr>
<td>Minimum Steps</td>
<td>Minimum Load [6]</td>
</tr>
<tr>
<td>Feedback</td>
<td>Feedback [7]</td>
</tr>
<tr>
<td>Location and Navigation</td>
<td>Location [8]</td>
</tr>
<tr>
<td>Module Choice</td>
<td>Module Availability [9]</td>
</tr>
<tr>
<td>Length and Language Style</td>
<td>Length [10]</td>
</tr>
<tr>
<td>Choice Terminology and Language</td>
<td>Choice Terminology [12]</td>
</tr>
<tr>
<td>Consistency</td>
<td>Consistency [13]</td>
</tr>
<tr>
<td>Minimum Steps</td>
<td>Minimum Steps [14]</td>
</tr>
<tr>
<td>Retraction</td>
<td>Retraction [15]</td>
</tr>
<tr>
<td>Accuracy of Content</td>
<td>Accuracy [16]</td>
</tr>
<tr>
<td>Location and Navigation</td>
<td>Location and Navigation [17]</td>
</tr>
<tr>
<td>Multiple Initiation</td>
<td>Multiple Initiation [18]</td>
</tr>
<tr>
<td>Multiple Inputs</td>
<td>Multiple Inputs [19]</td>
</tr>
<tr>
<td>Flexibility</td>
<td>Flexibility [20]</td>
</tr>
<tr>
<td>Motivational and Aesthetic</td>
<td>Motivational [21]</td>
</tr>
<tr>
<td>Multiple Inputs</td>
<td>Multiple Inputs [22]</td>
</tr>
<tr>
<td>Visual Metaphor</td>
<td>Visual Metaphor [23]</td>
</tr>
<tr>
<td>Terminology and Language Style</td>
<td>Terminology and Language Style [24]</td>
</tr>
<tr>
<td>Consistency</td>
<td>Consistency [25]</td>
</tr>
<tr>
<td>Multiple Input</td>
<td>Multiple Input [26]</td>
</tr>
<tr>
<td>Inter-changeability</td>
<td>Inter-changeability [27]</td>
</tr>
<tr>
<td>Consistency</td>
<td>Consistency [28]</td>
</tr>
<tr>
<td>Context-Sensitive Help</td>
<td>Context-Sensitive Help [29]</td>
</tr>
</tbody>
</table>

### System Performance Principles

<table>
<thead>
<tr>
<th>Principle</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manipulability</td>
<td>Manipulability [30]</td>
</tr>
<tr>
<td>Responsiveness</td>
<td>Responsiveness [31]</td>
</tr>
<tr>
<td>Audio-Perceptual Load</td>
<td>Audio-Perceptual Load [32]</td>
</tr>
<tr>
<td>Motor-Load</td>
<td>Motor-Load [33]</td>
</tr>
<tr>
<td>Clarity</td>
<td>Clarity [34]</td>
</tr>
<tr>
<td>Contrast</td>
<td>Contrast [35]</td>
</tr>
</tbody>
</table>

### User Support Principles

<table>
<thead>
<tr>
<th>Principle</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Help</td>
<td>General Help [36]</td>
</tr>
<tr>
<td>Context-Sensitive Help</td>
<td>Context-Sensitive Help [37]</td>
</tr>
</tbody>
</table>

The scope of the full set is similar to but wider than that developed by Bastien and Scapin, discussed in Section 4.3. The structure (originally based on earlier work on a usability audit) has and will continue to evolve as new content is incorporated. It is only one of many possible organisations, of which Bastien's & Scapin's is an alternative.
The three main experiments reported in Chapters 2 to 5 make use of different versions and/or subsets of the full principles set. (See Appendixes B to D for transcripts.)

Experiment 1 (Chapter 2) uses a three-page version (as then developed) covering 26 principles but the same seven categories as shown in Figure 1.3. In this version each principle is stated in a single paragraph on one or two pages of computer text. The version is available in the Hypertext Experiment 1 directory, and the version used by the participants is also available in the Experiment 1 directory. This version is used as the basis for the experiment and is referred to in the rest of this chapter. Any deviations from this version are noted in the text. The version used by the participants was identical to the version used in the experiment, except for the addition of a few extra principles.

### Attribute

**CONSISTENCY**

**Principle**

Consistency

**Set**

Comparative principles

**Explanation**

The steps required to complete any one operation should be consistent. Movement between components should also be consistent, such that the user should be able to predict what the result of a particular movement will be. The layout any one component or state should not differ according to the type of operation being performed. Thus the range of options available from any one state should not change, nor should the relationship between different components. Navigation within directories is also more difficult to define precisely. The current limiting factor is the reach of the mouse, which is not as extensive as it is within a general-purpose, full-screen editor. Consistency is the most commonly found, and easiest to agree upon, of all usability criteria.

**Example(s)**

- The user actions necessary to perform a particular operation (e.g. save file with a new name, delete file) should not differ according to the stage reached in interaction, or under arbitrary conditions. Once the user has learned how to move between components, such as between worksheet and help information, the nature of that movement should not change without reason. Though it may be necessary to enable or disable certain options within a component under particular conditions, the layout of each component should not change. The relationship between components (i.e. the functional organisation of the system) should not change either. Navigation within on-line help should be consistent with that used elsewhere.

**Exception(s)**

- In larger systems it may be possible to initiate operations from more than one state or component. It may also be possible to navigate around the system in more than one way. It may be necessary to enable or disable certain options under certain circumstances, and when the reach of a particular component should not be consistent. The reach of a particular component may also be different.

**Related to** or affects

- Multiple initiation
- Salience
- Step modification
- Choice availability
- Error management
- Terminology & language style

**Comment(s)**

Consistency is the most commonly found, and easiest to agree upon, of all usability criteria. Unfortunately it is also one of the most difficult to define precisely. The above represents an attempt at describing the broad limits of the problem, with qualifications where appropriate. While there are sometimes good reasons to break this principle, in general consistency is to be aimed for, at least within modes.

**Source(s)**


**Figure 1.4. Attribute (principle) no. 21, Consistency, of the author's principles set.**
The principle comprised just one paragraph, generally a statement of the principle followed by examples. By Experiment 2 (Chapter 3) this had been expanded into the full version, covering 23 principles and 30 attributes, described above (some of the original 26 principles having been merged and others split apart). In Experiment 2 the size of this full version proved unwieldy for evaluators to navigate (like some other guideline collections), so an additional summary feature (consisting of just the Explanation part of each attribute) was provided. It proved immediately useful and often superior to the full version. In Experiment 2 (Chapter 3) this had been expanded into the full version.

The contents of the principles set draw on many sources including Marshall et al. (1987). For the most part these represent principles collections (whether or not actually called "principles") which can be said to be domain-independent and which offer some underlying rationale. Acknowledged as such, this series will have to be illustrated by the examples offered. It is also evident that much could be included by way of illustrations of the examples offered. It will be apparent that this version of a principles set is by no means complete (for example, it lacks user control). It will be apparent that this version of a principles set is by no means complete (for example, it lacks user control). It will be apparent that this version of a principles set is by no means complete (for example, it lacks user control). It will be apparent that this version of a principles set is by no means complete (for example, it lacks user control). It will be apparent that this version of a principles set is by no means complete (for example, it lacks user control). It will be apparent that this version of a principles...
Chapter 1

3.4 Heuristics

Interface design heuristics attempt to encapsulate the principles which underpin guidelines or other material into shorter sets of 'rules of thumb'. They are meant to accompany, rather than replace, that guidance material, with which, it is assumed, designers are already familiar. In that sense, heuristics can be considered as principles shorn of their background rationale.

Heuristics thus rely to a greater or lesser extent on the ability of designers to relate their contents to the larger material on which they are based. Many of the guidelines collections cited in Figure 1.5 include such a summary, which could therefore be extracted as heuristics. The most well-known is Shneiderman’s (1998, Foley & Van Dam 1982 pp217-242 (seven “design principles” for “interactive computer graphics”: give feedback, help the user learn the system, allow backup and accommodate errors, control response time, design for consistency, structure the display, minimize memorisation).

Williges & Williges 1984 (six “fundamental design principles” for “interactive computer systems” - compatibility, consistency, brevity, flexibility, immediate feedback - plus extensive low-level guidelines).

Galitz 1985 (11 brief “desirable qualities of a system”, followed by a mixed collection of general and interface component-specific guidelines, including: design tradeoffs, flexibility, information load, response time).

Murphy & Mitchell 1986 (18 “cognitive attributes” for “highly automated control systems” and “real-time decision-making environments”, organised into three groups: knowledge structures in memory (schemas), active processing, problem solving).


Brown 1988 (12 “general concepts” underlying the interface component-specific guidelines which follow, including: allocation of functions, consistency, physical analogies, stimulus-response compatibility, providing multiple paths).

Norman 1988 pp8-18 (the “fundamental principles for designing for people” - providing good conceptual models and visibility - plus affordance, mapping, feedback).

Thimbleby 1991 (24 “design choices and principles” for “interactive systems”, including: consistency, display inertia, modes, minimising the user’s memory load).

Denley et al. 1993 (12 classes of “generic principles” for integrated broadband communication systems, including: compatibility, coherence, simplicity, salience, reversibility, controllability, flexibility, feedback, support orientation).

Hix & Hartson 1993 pp27-55 (26 interface “guidelines to ensure usability”, including: keep the locus of control with the user, use modes cautiously, make user actions easily reversible).

Nielsen 1993 pp115-163 (the 10 “usability heuristics”, described in full below, which are compared in this thesis with the author’s principles set).

Sutcliffe 1995 (seven “basic principles” for HCI design - consistency, compatibility, predictability, adaptability, error prevention, user control, structure, plus three “basic concerns of design quality” and eight “principles of dialogue design”.


Scapin & Bastien 1997 (eight “ergonomic criteria for evaluating the ergonomic quality of interactive systems” - guidance, workload, explicit control, adaptability, error management, consistency, significance of codes, compatibility - and 13 sub-criteria).

Cox & Walker 1998 pp174-187 (20 “general principles” for user interface design, including: consistency, closure, easy reversal).

Dix et al. 1998 pp162-175 (three “principles to support usability”: learnability, flexibility and robustness, with 14 sub-principles; plus a later chapter on user support (help) systems).

Jordan 1998 pp25-38 (10 “principles of usable design” - consistency, compatibility, consideration of user resources, feedback, error prevention and recovery, user control, visual clarity, prioritisation of functionality and information, appropriate transfer of technology, explicitness).

Shneiderman 1998 pp67-90 (three overall principles - recognise the diversity, use the eight golden rules of interface design (outlined below), prevent errors - and sub-principles; plus later guidelines for data display and data entry).
“Eight golden rules” for interface design (listed as “Principle 2” in the 1987 version), namely: 1. Strive for consistency, 2. Enable frequent users to use shortcuts, 3. Offer informative feedback, 4. Design dialogs to be easy to start and finish, 5. Reduce short-term memory load, 6. Even style guides such as Microsoft’s (1995) Windows®, IBM’s (1992) CUA and Apple’s (1989) Hypercard® guidelines include, respectively, a “guidelines summary,” “goals and design principles,” and a “ten general principles” which could serve the same purpose. However, Shneiderman (1987) appeared to have intended his “golden rules” to stand relatively alone, since only short sections of that book are devoted to further guideline exposition. Shneiderman (1998, p74) later describes the rules as “derived heuristically from experience”. This origins of Nielsen’s heuristics, in a paper by Rolf Molich and Nielsen (1990), appear to support the view that they are derived mainly from experience. The then nine heuristics are described as “a short checklist of usability considerations in a good dialogue” (Molich & Nielsen 1990, p339), and, through corresponding to similar principles described by others such as the Apple 1987 style guide, the checklist “reflects our personal experience” (ibid., p339). The heuristics are first described in short paragraphs of no more than 50 words, comparable in format to Shneiderman’s ten. Only later, in Nielsen’s (1994) book, are they fleshed out with examples and their origins made more explicit. By 1994 (in Nielsen 1994d and several other papers) they and the heuristic evaluation method had become fully formed.

It is not clear from Nielsen’s descriptions whether the heuristics (as used in heuristic evaluation) are “rules” or “principles”:

“For the discount [usability engineering] method I advocate cutting the complexity [of guidelines containing thousands of rules] by two orders of magnitude, to just 10 rules. Relying on a small set of broader heuristics such as the basic usability principles listed in Table 2 of Nielsen 1993 and discussed in Chapter 5 of [Nielsen 1993], Usability Heuristics: Nielsen’s (1993, p9, thesis author’s italics.)

“During the heuristic evaluation session, the evaluator goes through the interface several times and inspects the various dialogue elements and compared them with a list of recognised usability principles. These heuristics are general rules that seem to describe common properties of usable interfaces...” (Nielsen 1994d p28, thesis author’s italics.)

Such conflation lends support to the view that the heuristics are, after all, meant to stand on their own, as “rules,” even “checklists,” without supporting principles or other background material. The role of this other material, and the extent to which evaluator expertise should compensate for its absence, remains unclear.

However, the status of heuristics as evaluative criteria is clear, and that is the form in which they (meaning both Nielsen’s heuristics and the author’s principles set) will be used in this...
In the event, Experiments 1 to 3 (Chapters 2 to 5) differed in both the size and scope of the materials compared and the degree of training offered in their use. In that sense, these experiments did vary the amount of 'background material' which accompanied the 'heuristics'. However, the method of use was held the same within each experiment, thus representing comparisons of methods rather than measures. The intention of these experiments was not to compare alternative evaluation methods (though such studies will be compared below). The aim was to attempt some closure on the degree of training offered in each experiment. These experiments did vary the amount of background material which accompanied the theses, in the event. Experiments 1 to 3 (Chapters 2 to 5) differed in both the size and scope of the 'heuristics' which might be used in a heuristic evaluation.

3.5 Nielsen’s Heuristics

Experiments 1 to 3 will compare the author’s principles set with Nielsen’s (and Molich’s) ten usability heuristics, as described in Nielsen (1993) and Nielsen et al. (1994). The primary source of the Nielsen & Molich (1990) version of the heuristics (plus Help and Documentation) was Nielsen (1990). Appendix A shows the numbered transcripts used in the experiments.

Nielsen’s heuristics were used for two reasons. First, heuristic evaluation has been a focus of much research (just some of which is cited above), including Nielsen’s own. The second reason is the claims made for the heuristics by their proponents. The weaker claims include:

"These [basic usability] principles can be used to explain a very large proportion of the problems one observes in user interface designs" (Nielsen 1993, p. 27).

and

"The problems one observes in user interface designs [...] can be used to explain a very large proportion of the problems one observes in user interface designs" (Nielsen 1993, p. 27).

The following reasons is the claim made for the heuristics by their proponents. The weaker claims include:

The original set (slightly shortened but not substantially changed in the 1993 and 1994 versions) is reproduced in Figure 1.6. Appendix A shows the numbered transcript used in the experiments.


The stronger claims include:

"These [basic usability] principles can be used to explain a very large proportion of the problems one observes in user interface designs" (Nielsen 1989, p. 397, Nielsen 1993, p. 19).

and

"These [basic usability] principles can be used to explain a very large proportion of the problems one observes in user interface designs" (Nielsen 1989, p. 397, Nielsen 1993, p. 19).
Stronger claims include the assertions that “Almost all usability problems fit well into one of the categories.” (Molich & Nielsen 1990 p339) and “These nine usability principles should be followed by all user interfaces.” (Molich & Nielsen 1990).

The authors of these statements elsewhere qualify them. The original nine (or ten) heuristics are merely “similar to” or “correspond to” other usability principles or guidelines (Molich & Nielsen 1990, Nielsen 1990e, Nielsen & Molich 1990, Nielsen 1993, 1994d). Additional “relevant” problems, not elicited by the heuristics themselves, may obviously be (Molich & Nielsen 1990, Nielsen 1990e, Nielsen 1993, Nielsen 1994d).

Additional relevant problems not elicited by the heuristics themselves, may obviously be.

Figure 1.6: Nielsen’s and Molich’s heuristics, as appeared in Nielsen (1990) and Molich & Nielsen (1990). (with the addition of ‘Help and Documentation’, in Nielsen 1993 (1961)), and listed (with the addition of ‘Help and Documentation’) in Nielsen (1993) and Nielsen (1994d).

5. Error Prevention

Meaningful suggestions to the user about what to do next.

6. Error Messages

Informative and experienced users.

7. Provide Feedback

For systems with common user populations.

8. Provide Documentation

For systems with common user populations.

9. Help and Documentation

For systems with common user populations.

10. Speak the User’s Language

Speak the User’s Language.

11. Simple and Natural Dialogue

Simple and Natural Dialogue.

12. Error Correction

Error Correction.

13. Consistency

Consistency.

14. Help

Help.

15. Searchable Help

Searchable Help.

16. Help Using the User’s Memory Load

Help Using the User’s Memory Load.

17. Searchable Help

Searchable Help.

18. Consistency

Consistency.

19. Help

Help.

20. Searchable Help

Searchable Help.

21. Consistency

Consistency.

22. Help

Help.

23. Searchable Help

Searchable Help.

24. Consistency

Consistency.

25. Help

Help.

26. Searchable Help

Searchable Help.

27. Consistency

Consistency.

28. Help

Help.

29. Searchable Help

Searchable Help.

30. Consistency

Consistency.

31. Help

Help.

32. Searchable Help

Searchable Help.

33. Consistency

Consistency.

34. Help

Help.

35. Searchable Help

Searchable Help.

36. Consistency

Consistency.

37. Help

Help.

38. Searchable Help

Searchable Help.

39. Consistency

Consistency.

40. Help

Help.

41. Searchable Help

Searchable Help.

42. Consistency

Consistency.

43. Help

Help.

44. Searchable Help

Searchable Help.

45. Consistency

Consistency.

46. Help

Help.

47. Searchable Help

Searchable Help.

48. Consistency

Consistency.

49. Help

Help.

50. Searchable Help

Searchable Help.

51. Consistency

Consistency.

52. Help

Help.

53. Searchable Help

Searchable Help.

54. Consistency

Consistency.

55. Help

Help.

56. Searchable Help

Searchable Help.

57. Consistency

Consistency.

58. Help

Help.

59. Searchable Help

Searchable Help.

60. Consistency

Consistency.

61. Help

Help.

62. Searchable Help

Searchable Help.

63. Consistency

Consistency.

64. Help

Help.

65. Searchable Help

Searchable Help.

66. Consistency

Consistency.

67. Help

Help.

68. Searchable Help

Searchable Help.

69. Consistency

Consistency.

70. Help

Help.

71. Searchable Help

Searchable Help.

72. Consistency

Consistency.

73. Help

Help.

74. Searchable Help

Searchable Help.

75. Consistency

Consistency.

76. Help

Help.

77. Searchable Help

Searchable Help.

78. Consistency

Consistency.

79. Help

Help.

80. Searchable Help

Searchable Help.

81. Consistency

Consistency.

82. Help

Help.

83. Searchable Help

Searchable Help.

84. Consistency

Consistency.

85. Help

Help.

86. Searchable Help

Searchable Help.

87. Consistency

Consistency.

88. Help

Help.

89. Searchable Help

Searchable Help.

90. Consistency

Consistency.
The few attempts to develop and test such versions will be described below.

## 4. Comparison of Guidelines, Principles and Heuristics

### 4.1 Guidelines versus Heuristics

Jeffries et al. (1991), Miller & Jeffries (1992) compared heuristic evaluation with guidelines as part of a study of four methods (the other two being cognitive walkthrough and usability testing). The heuristics used were probably the original Molich & Nielsen (1990) set. Heuristic evaluation found more serious problems than any of the four methods. However, it is not clear how the guidelines were used (nor, indeed, how the heuristic evaluation was carried out). It is difficult to draw firm conclusions from this study regarding guidelines versus heuristics.

In a later study focused on direct manipulation, Cuomo & Bowen (1994) compared heuristic evaluation, guideline evaluation and cognitive walkthrough against the results of a heuristic evaluation. The heuristics were the Smith & Mosier (1986) collection described above. An analysis based on Norman's (1986) user activity model (Norman's 'theory of action') found the guidelines on average unsurpassed, but only by the method of Nielsen (1990). The results confirm that guidelines are more effective in identifying a wider range of problems than heuristics. However, in this study the two heuristic evaluators were allowed to proceed in any manner they wished, whereas the single guideline evaluator was forced to use the guidelines in a specific order. This may explain why heuristic evaluation found more serious problems than any of the four methods.

### 4.2 Principles versus Heuristics

Most recently reported in Bastien et al. (1999).

In addition, several studies have been conducted to compare the effectiveness of different methods of evaluation. For example, Nielsen (1994d) has proposed a new set of nine heuristics that are more reliable than the original Nielsen & Molich (1990) set. The results of these studies indicate that heuristic evaluation is more effective than guideline evaluation, but less effective than cognitive walkthrough and usability testing.
4.2 Comparing Heuristics

Nielsen (1994a) created a new “candidate set” of nine heuristics by matching each of the 101 heuristics from seven published sets including Molich & Nielsen (1990) against the 249 usability problems collected from 11 earlier (unspecified) studies by the same author. A factor analysis revealed a list of seven most important (“summarising”) factors. Two additional heuristics with the “widest coverage” (i.e., most important) factors were added to the new set of 14 heuristics. These plus Help and Documentation are listed in Figure 1.7.

Visibility of system status
Match between system and the real world
User control and freedom
Consistency and standards
Error prevention
Recognition rather than recall
Flexibility and efficiency of use
Aesthetic and minimalist design
Help users recognise, diagnose, and recover from errors
Help and Documentation

Figure 1.7. Nielsen’s (1994a) “candidate set” of nine new heuristics, plus Help and Documentation, listed in Nielsen (1994d) p30.

Through the 1990s, Bastien and Scapin (primarily and in English, Bastien et al. 1999, Scapin & Bastien 1995, Bastien et al. 1997, Bastien et al. 1996) have proposed an additional four heuristics to encompass what they call “navigation-based walkthroughs and user testing.” The experienced evaluators who used the heuristic-based walkthroughs and user testing “were the most critical of their own evaluations, and earmarked for the evaluation of hypertext.” Elsewhere, Karat et al. (1992) used a set based on Nielsen (1993) (minus error messages and error prevention) in their comparison of (team and individual) navigational dimensions and elements for the evaluation of hypertext. Elsewhere, Karat et al. proposed additional or domain-specific heuristics, Nielsen (1994d) has identified further “navigational dimensions and elements for the evaluation of hypertext.”

4.3 Comparing Principles

During the 1990s, Bastien and Scapin (primarily and in English, Bastien et al. 1999, Scapin & Bastien 1995, Bastien et al. 1997, Bastien et al. 1996) have developed and validated a set of ergonomic criteria for evaluating the ergonomic characteristics of interfaces. Their set of 14 ergonomic criteria includes the following:

Help users recognise, diagnose, and recover from errors
Aesthetic and minimalist design
Consistency and standards
User control and freedom
Error prevention
Recognition rather than recall
Flexibility and efficiency of use
Visibility of system status
Help and Documentation

Figure 1.7. Nielsen’s (1994a) “candidate set” of nine new heuristics, plus Help and Documentation, listed in Nielsen (1994d) p30.
Chapter 1

38

quality of interactive systems'. Their eight ergonomic criteria and sub-criteria (making 18 'elementary criteria') are based on the set originally devised by Scapin (1990). This was drawn from a range of 'existing human factors guides', namely an earlier collection by Scapin (1987), Smith & Mosier (1986), Williges & Williges (1984), Ergonomics Abstracts and "previous unpublished work" (Scapin 1990, p.208 and p.209). See Figure 1.8.

The eight ergonomic criteria developed by Bastien and Scapin (1992) in that study, two groups were tested in an evaluation exercise (Bastien & Scapin 1995). In the study, two groups were tested: one which could be further subdivided (Scapin & Bastien 1997, p.222). The 18 elementary criteria are the ones which cannot be further subdivided (Scapin & Bastien 1997, p.222). The 18 elementary criteria were compared with Part 10 of the 17-part international standard for Ergonomic Requirements for Office Work with VDTs (ISO 9241-10) (1996). No difference was found between a control group and the same elementary criteria were compared with Part 10 of the 17-part international standard for Ergonomic Requirements for Office Work with VDTs (ISO 9241-10) (1996). The elementary criteria (control group) were significantly higher for the criteria group than the control group.

The eight ergonomic criteria were compared with Part 10 of the 17-part international standard for Ergonomic Requirements for Office Work with VDTs (ISO 9241-10) (1996). The elementary criteria (control group) were significantly higher for the criteria group than the control group.

The difference between usability problems counts taken when using the control materials and those using the different database system using either the criteria or control materials was evaluated. A "modified" set (though with the identical structure) of the elementary criteria (control group) were relatively robust and might remain unchanged, while the other criteria (control group) were relatively improved. A "modified" set (though with the identical structure) of the elementary criteria (control group) were relatively robust and might remain unchanged, while the other criteria (control group) were relatively improved.

The results showed that half of the criteria, among them Feedback, Error protection, and distinguishing factors by format, were relatively robust and might remain unchanged, while the other criteria, among them Feedback, Error protection, and distinguishing factors by format, were relatively improved. In a later study (Bastien & Scapin 1995), the results showed that half of the criteria, among them Feedback, Error protection, and distinguishing factors by format, were relatively robust and might remain unchanged, while the other criteria, among them Feedback, Error protection, and distinguishing factors by format, were relatively improved.

In a later study (Bastien & Scapin 1995), the results showed that half of the criteria, among them Feedback, Error protection, and distinguishing factors by format, were relatively robust and might remain unchanged, while the other criteria, among them Feedback, Error protection, and distinguishing factors by format, were relatively improved.

In a later study (Bastien & Scapin 1995), the results showed that half of the criteria, among them Feedback, Error protection, and distinguishing factors by format, were relatively robust and might remain unchanged, while the other criteria, among them Feedback, Error protection, and distinguishing factors by format, were relatively improved.

In a later study (Bastien & Scapin 1995), the results showed that half of the criteria, among them Feedback, Error protection, and distinguishing factors by format, were relatively robust and might remain unchanged, while the other criteria, among them Feedback, Error protection, and distinguishing factors by format, were relatively improved.

In a later study (Bastien & Scapin 1995), the results showed that half of the criteria, among them Feedback, Error protection, and distinguishing factors by format, were relatively robust and might remain unchanged, while the other criteria, among them Feedback, Error protection, and distinguishing factors by format, were relatively improved.
introductions - towards previously identified problems.

individual evaluations will also be a major consideration of this thesis (Chapter 4 onwards).

whether ISO 9241 Part 10 is a principles set or a heuristics set is probably not an issue worth

whether the ISO 9241 Part 10 is a principles set or a heuristics set is probably not an issue worth
More importantly, this thesis does not compare principles sets, nor does it set out to validate the author’s own set. Even if some superiority (such as problem count) could be found for one large and wide-ranging set over another, it would be very difficult to determine what it was about the winning set which made it superior, and there would be no guarantee of repeating the trick with a different system. For that reason, this thesis attempts only the “soft option” of comparing principles with heuristics. However, even this reduced objective suffers from the same limitation, namely that without clear indications of the strengths and weaknesses of both materials, any success for one over the other will be difficult to interpret. While there are strong indications of a common set of core issues (e.g., consistency, feedback, error management), exactly what makes principles better than heuristics (if they are) will need more work than is attempted here. This important issue will be returned to in Chapter 8.

5. Usability and Usability Evaluation Methods

The remainder of this chapter will be taken up with a necessarily brief discussion of usability and usability evaluation methods, including heuristic evaluation. This aim is to place the principles versus heuristics theme in a wider context, and lead up to some more specific remarks regarding comparisons of evaluation methods. The chapter concludes with a discussion of the role of cognitive psychology in usability research.

5.1 Usability

As late as 1992, Adler & Winograd wrote that “... the typical human factors effort is given low priority among a design team’s objectives. Usability issues are often left to the latest possible date, by which time modifications are too expensive to make.” (Adler & Winograd 1992 p4). In 1994, the mean staff utilization for usability laboratories in 13 companies was 17 per company, with IBM and Microsoft having the largest lab count (Nielsen 1994). By 1997, however, Microsoft had 76 usability engineers and 18 labs (Williams & McElrath 1997), and this author felt in early 1998 more were being built. In the same year, John Carroll observed that “in industry, HCI practitioners have become widely integrated in system development.” (Carroll 1997 p506). Even by 1994, according to Rubin (1994), the demand for usable products was outpacing the supply of trained usability professionals. In 2000, this demand produces around 10 jobs per week world-wide for usability engineers and related personnel (HCI RN Jobs Index).

This “explosion of usability” (Rubin 1994 p78) has been accompanied by a dramatic growth in research and other publications. A search of the Social Science Citation Index from 1981 to 1990 inclusive (via the BIDS/ISI Data Service) on keyword “usability” (in title, 6

6 In theory. In practice, most are based in the US.

7 BIDS has since changed to Web of Science (WoS).
Authors & journal) returned just 19 entries. The same search for years 1991 to 1994 inclusive returned 91 citations, and for 1995 to July 2000 inclusive there were 246. Books published between 1993 and 2000 include practical guides (Dumas & Redish 1993, Rubin 1994), a ‘usability in practice’ collection (Wiklund 1994), handbooks (e.g. George 1995) and collections on the politics of usability (Bias & Mayhew 1994, Trenberth & Kaya 1995). The approach adopted in these two experiments was to have subjects freely explore the whole of the software to be evaluated, rather than perform specific tasks. As we shall see, the enabled subjects performed significantly better than those in Experiment 1 and 2 in experiments 1 and 2 incorporated requirements and functionality principles set used in Experiments 1 and 2 that were not present in many sets including that of Bastien & Scapin. The approaches to usability and utility of software have usually not been subjected to the same level of scrutiny as other fields of technology. ["... the usefulness and usability of software have usually not been subjected to the same level of scrutiny as other fields of technology. 

This thesis will encompass aspects of both usability and utility. The versions of the author's principles set used in Experiments 1 and 2 incorporate ‘requirements and functionality’ principles set used in Experiments 1 and 2 that were not present in many sets including that of Bastien & Scapin. The approaches to usability and utility of software have usually not been subjected to the same level of scrutiny as other fields of technology.

What seems to have happened is that usability (the match between what a system does and how well it can be used) has taken over from utility (the match between what a system does and what users might want to do with it). The early and continual focus on users seems to have been superseded by the focus on what the systems do, which has become the focus of usability. The many and continual focus on the context determines the focus of utility, which has been superseded by the focus on what the context determines the focus of usability. The end result is that we have lost the match between what a system does and what users might want to do with it.

According to Landauer (1995), the solution is to make computers do less, or rather, to match the many separate tasks that we perform by producing not general-purpose devices but separate “information appliances” special for particular activities. The approach adopted in these two experiments was to have subjects freely explore the whole of the software to be evaluated, rather than perform specific tasks. As we shall see, the enabled subjects performed significantly better than those in Experiment 1 and 2 that were not present in many sets including that of Bastien & Scapin. The approaches to usability and utility of software have usually not been subjected to the same level of scrutiny as other fields of technology.

According to Landauer (1995), the solution is to make computers do less, or rather, to match the many separate tasks that we perform by producing not general-purpose devices but separate “information appliances” special for particular activities. The approach adopted in these two experiments was to have subjects freely explore the whole of the software to be evaluated, rather than perform specific tasks. As we shall see, the enabled subjects performed significantly better than those in Experiment 1 and 2 that were not present in many sets including that of Bastien & Scapin. The approaches to usability and utility of software have usually not been subjected to the same level of scrutiny as other fields of technology.
Chapter 1

42

Figure 1.9 shows a 2x2 taxonomy of usability evaluation methods (UEMs) adapted from Whitefield et al. (1991). The classification scheme is simpler than most others, but involves minimal overlap between categories. The discussion which follows draws on both Whitefield et al. (1991) and Macleod's (1992) elaboration.

5.2 Usability Evaluation Methods

Figure 1.9. A 2x2 taxonomy of usability evaluation methods, adapted from Whitefield et al. (1991). Groupings are "broadly indicative", only. In Whitefield et al. the term 'system' is used to refer collectively to the user and computer; 'inspection methods' and 'survey methods' were dubbed 'specialist reports' and 'user reports', respectively.

On this "broadly indicative" classification, both inspection methods and observational methods take place in the absence of the system. The difference between inspection and observational methods is the presence or absence of users. Normally by observation is meant that observations are carried out without the presence of users (normally by specialists or designers), whereas methods are performed using the system under evaluation, where judges and survey methods are performed on a system prototype (including mock-ups of systems). Observation of designers' (e.g., with users, without users) performance includes end users.

Other inspection methods include cognitive walkthrough, pluralistic walkthrough, and error analysis. In cognitive walkthrough (Polson et al. 1992), evaluators work through the interaction process with the user's task in mind. At every step, the interface is assessed in terms of its ability to support the user's goals (using a goal structure). The evaluators judge the interface against some criteria (Mack & Nielsen 1994). Judgment and predictions are made about the system, and predictions are made about the users (Mack & Nielsen 1994). Jurgens and Anderson's 'judgment and prediction model' of user models is used to judge the system.

On this "broadly indicative" classification, both inspection methods and observational methods are performed on a system prototype (including mock-ups of systems), whereas observational methods are performed without the presence of users (normally by specialists or designers).
The two main themes of this thesis are concerned with the distinction between inspection methods and observational methods, as depicted above. The first theme (Chapters 2 to 5) involves assessment of the range of possible errors which might be associated with each step of interaction sequence.

Survey methods include questionnaires, interviews and focus groups.

Analytic Methods
without system, without users] are those in which experts or specialists, derive predictions about system use from formal or semi-formal models of the system, its interface, or its users. Targets of such predictions include user errors, task time, user goals and task models.

Analytic methods include GOMS (card et al. 1980), KEYS (Kieras 1985), and the Keystroke Level Model (KLM). Survey methods include GOMS (card et al. 1983), and the Keystroke Level Model (KLM). Setting for analytic methods include ‘usability engineering’ towards measurable criteria, and CBT (computer-based training).

Survey methods include questionnaires, interviews and focus groups.

Chapter 5 features an outline of cooperative evaluation and discusses one of the studies on participatory design (Schuler & Namioka 1993). Participants are involved in the development of the system with the system designer(s) and users.

Observational methods are those in which data is collected during user-system interaction. Methods range from interview and observation to experimental testing. Performance data, error logs, and user feedback are used to evaluate the system.

Chapter 6 will make use of data from an earlier study performed by the author.
explores the use of one form of inspection method, namely guideline review, to identify usability (and utility) problems. The reviews used in this thesis make use of principles and heuristics, and most of the procedures prescribed for heuristic evaluation. The second theme of the thesis (Chapter 4 onwards) concerns the number of evaluators required to identify the majority of usability problems. In Chapter 5 onwards, this will be further explored in relation to problems occurring under one form of observational method, namely 'think-aloud' user testing.

This contrast between predicted (as opposed to merely reported) and observed problems will be characterised in terms of Gray & Salzman's (1998) broad distinction between analytic and empirical methods. Analytic methods attempt to make predictions about user-system interaction from intrinsic system features (including mockups or prototypes), while empirical methods measure some aspect(s) of interaction while a (full or partial) system is in use. On this classification, both inspection methods and (for example) GOMS are analytic, while observational methods are empirical. However, unlike observational methods, inspection methods can be carried out in the absence of the system being evaluated. In Chapter 5, the term 'analytic' will be used in a wider sense, that is, to encompass the predictive nature of inspection methods such as heuristic evaluation, particularly those concerning evaluators' expertise and their use of heuristics. In this thesis, some of Nielsen's recommendations are relevant to the evaluation of any other method (or combination of methods) in order to compare evaluation materials. It is recommended that the findings from about 3 to 5 evaluators be aggregated together, the exact number depending on cost and time available. The second theme of the thesis (Chapter 4 onwards) concerns the number of evaluators required to identify the majority of usability problems, and a single evaluator will usually be used in user tests (Nielsen 1994d).

5.3 Heuristic Evaluation

In heuristic evaluation (principally Nielsen & Molich 1990, Nielsen 1993, Nielsen 1994d) an evaluator systematically inspects an interface to assess compliance with a set of heuristics. This involves identifying the degree to which an interface adheres to the principles proposed by Nielsen & Molich (1990), plus the evaluator's "general knowledge of usability principles" (Nielsen 1994d p61). It is recommended that the evaluator go through the interface twice, focusing first on its "flow" (Nielsen 1993 p158) and then on its "individual dialogue elements" (Nielsen 1994d p61). Evaluators will usually evaluate problems through the interface twice, focusing first on its "flow" (Nielsen 1993 p158) and then on its "individual dialogue elements" (Nielsen 1994d p61). It is recommended that the evaluator "general knowledge of usability principles" (Nielsen 1994d p61). In addition, some of Nielsen's recommendations are relevant to the evaluation of any other method (or combination of methods) in order to compare evaluation materials. It is recommended that the findings from about 3 to 5 evaluators be aggregated together, the exact number depending on cost and time available. The second theme of the thesis (Chapter 4 onwards) concerns the number of evaluators required to identify the majority of usability problems, and a single evaluator will usually be used in user tests (Nielsen 1994d).

Heuristic evaluation is a method used to identify usability problems in a user interface. This method involves an evaluator systematically inspecting the interface to assess compliance with a set of heuristics. The evaluator usually goes through the interface twice, focusing first on its "flow" and then on its "individual dialogue elements." Evaluators will usually evaluate problems through the interface twice, focusing first on its "flow" and then on its "individual dialogue elements." It is recommended that the evaluator use general knowledge of usability principles in addition to Nielsen's recommendations. The number of evaluators recommended for this method is about 3 to 5, depending on cost and time available. The second theme of the thesis (Chapter 4 onwards) concerns the number of evaluators required to identify the majority of usability problems, and a single evaluator will usually be used in user tests.
Heuristic evaluation is explicitly intended as a "discount usability engineering" method (Nielsen 1994d). Heuristic evaluation is explicitly intended as a "discount usability engineering" method because it is faster and cheaper than traditional usability methods. Nielsen (1994d) argues that heuristic evaluation is a useful tool for identifying usability problems, and that it can be used as a complement to other usability methods. Heuristic evaluation is based on a combination of qualitative and quantitative methods, and it can be used to evaluate both new and existing systems.

One drawback with heuristic evaluation is that it tends to find more minor usability problems than major problems (Nielsen 1994d). For this reason, some means of prioritising is necessary. One solution is to have evaluators assign severity ratings. According to Nielsen, the severity of a usability problem is based on a combination of user interface, usability, and user experience factors (Nielsen 1994d).

Experiments 1 of 3 of this thesis made use of most of the above recommendations for heuristic evaluation (Nielsen 1994d).
5.4 Comparative Usability Evaluation

5.4.1 Summary of Related Research

There have been a large number of studies comparing different usability evaluation methods (including psychology teaching packages). In Experiment 1 the software evaluated was depicted as an "early prototype." In each experiment subjects were instructed to first run through the whole interface and then to proceed in more detail. Subjects were to 'think aloud,' using the evaluation material as a guide, but also to identify any issues of their own (unlike in Bastien & Scapin's studies). In Experiment 2, subjects were specifically asked to consider remedies and other recommendations related to the problems identified. In Experiments 1 and 3 the experimenter acted as observer and recorder, assisting and prompting only if necessary and taking great care not to influence the subjects' comments. In Experiment 2, subjects were asked to consider the problems identified in the interface and then to proceed in more detail. Subjects were to think aloud, using the evaluation material as a guide, but also to identify any issues of their own (unlike in Bastien & Scapin's studies). In Experiment 2, subjects were specifically asked to consider remedies and other recommendations related to the problems identified. In Experiment 1, the software evaluated was depicted as an...
Studies which have compared heuristic evaluation with other methods are in general agreement that heuristic evaluation is good at finding a wide spread of general usability problems (Virzi et al. 1993, Cuomo & Bowen 1994) at comparatively low cost (Jeffries et al. 1991, Nielsen & Phillips 1993). Heuristic evaluation and a 'think-aloud' test were roughly equivalent in finding core problems (Virzi et al. 1993), but heuristic evaluation identified many more minor problems and problems not observed. Cuomo & Bowen (1997) also found that heuristic evaluation correctly predicted a considerably higher proportion (mean 47%) of observed problems than cognitive walkthrough (24%), but that heuristic evaluation, while 55% of non-experts' predictions could not occur, Sears (1997) also found that 44% of experts' predictions by analytic methods, Desurvire et al. (1992) found that 56%, found that 45% of the two methods (and a third from Karat et al. 1992) detected most real uses.

There is less agreement on the proportion of empirical problems that can be successfully detected in the usability test. In a case study of cognitive walkthrough and five other methods, Desurvire et al. (1994) found that the methods' abilities to detect problems were not equivalent in parallel experiments, and that the methods' abilities to detect problems were not equivalent in parallel experiments. However, the results were consistent with the hypothesis that the methods' abilities to detect problems were not equivalent in parallel experiments. However, the results were consistent with the hypothesis that the methods' abilities to detect problems were not equivalent in parallel experiments. However, the results were consistent with the hypothesis that the methods' abilities to detect problems were not equivalent in parallel experiments. However, the results were consistent with the hypothesis that the methods' abilities to detect problems were not equivalent in parallel experiments. However, the results were consistent with the hypothesis that the methods' abilities to detect problems were not equivalent in parallel experiments. However, the results were consistent with the hypothesis that the methods' abilities to detect problems were not equivalent in parallel experiments.
John & Mashyna (1997) found that just 5% of observed problems had been precisely predicted and another 5% vaguely predicted, with 27% false alarms, by one evaluator using cognitive walkthrough. Of the total predicted problems from this study, John & Marks (1997) reported that 52% were not observed, and the changes made as a result of those which were observed resulted in just 23% fewer problems (and 11% new problems). In another study, Bailey et al. (1992) estimated the ratio of “potential” to “real” problems found by heuristic evaluation at 10:1. The predictive power of heuristic evaluation seems to be higher than that of cognitive walkthrough, though only two of the studies agree on this. The second main theme of the thesis examines the validity of the claim for heuristic evaluation's role as a discount technique. Even if most of the false alarms are taken up by a proportion of real problems (such as those found in user testing) to be accounted for by a mixture of real issues, the claim for a discount approach is still a high one. The role of evaluator expertise in heuristic evaluation, in particular Nielsen’s (1992) use of “double specialists” (those with both domain and evaluation expertise), appears not to have been confirmed. Of the above studies, only two (Desurvire et al. 1991, Desurvire et al. 1992) set out to compare the effects of experience level on problem identification. As has been mentioned, the two studies agree on this. Though many aspects of heuristic evaluation have been accounted for, the central procedural issues - precisely how the heuristics are to be used to elicit usability problems, how problems are recorded and counted as separate issues - remain open to interpretation.
Introduction

Having pointed out, Experiment 1 of this thesis uses both experienced evaluators (HCI researchers) and novices (mainly psychology undergraduates). However, Experiments 2 and 3 used only (different) novices from the same pool. Evaluator expertise is an issue for this thesis and will be returned to in Chapter 8.

Gray & Salzman (1998) have criticised the comparative UEM literature in general, and five studies (Jeffries et al. 1991, Karat et al. 1992, Nielsen 1992, Desurvire et al. 1992, Nielsen & Phillips 1993) in particular, for lacking both internal and external validity. According to these authors, the five studies lack internal validity in relying on small sample sizes and one-off experiments and their claims to determine the scope of problem categorisation. Gray & Salzman specifically criticise the use of problem count as a measure of the effectiveness of UEM, recommending that researchers limit both their expectations and claims for UEM studies. One way would be to focus on the predictive power of analytic methods by distinguishing between correct predictions ('hits'), over-predictions ('false alarms' or false positives), misses, and 'correct rejections'. Another way would be to distinguish clearly between problem tokens (separate instances of usability issues) and problem categories. A third would be to employ multiple (convergent) means of usability assessment.

The approach taken in this thesis addresses many of Gray & Salzman's criticisms and recommendations. Subject sample sizes vary from 7 to 9 per cell, with full reporting of statistical analyses. Care has been taken to be explicit about procedures used and the scope of evaluation materials. Experiments 2 and 3 offer (albeit limited) inter-rater reliability. Subject sample sizes vary from 7 to 9 per cell, with full reporting of inter-rater reliability. Chapter 4 introduces the model of the problem reduction process to be introduced in Chapter 4. Experiments 2 and 3 offer (albeit limited) inter-rater reliability. Subject sample sizes vary from 7 to 9 per cell, with full reporting of inter-rater reliability. Chapter 4 introduces the model of the problem reduction process to be introduced in Chapter 4. Experiments 2 and 3 offer (albeit limited) inter-rater reliability. Subject sample sizes vary from 7 to 9 per cell, with full reporting of inter-rater reliability.

This author agrees with Gray & Salzman that problem count is an impoverished measure of the effectiveness of a UEM. Given that it is the main dependent variable in these experiments, the author believes that the establishment of some internal validity for this measure is more important than whether or not evaluator expertise is an issue for this thesis.

Usability assessment

Usability assessment

This thesis and will be returned to in Chapter 8, and 3 used only (different) novices from the same pool. Evaluator expertise is an issue for these studies (Jeffries et al. 1991, Karat et al. 1992, Nielsen 1992, Desurvire et al. 1992, Nielsen & Phillips 1993). According to these authors, the five studies lack internal validity in relying on small sample sizes and one-off experiments and their claims to determine the scope of problem categorisation. Gray & Salzman specifically criticise the use of problem count as a measure of the effectiveness of UEM, recommending that researchers limit both their expectations and claims for UEM studies. One way would be to focus on the predictive power of analytic methods by distinguishing between correct predictions ('hits'), over-predictions ('false alarms' or false positives), misses, and 'correct rejections'. Another way would be to distinguish clearly between problem tokens (separate instances of usability issues) and problem categories. A third would be to employ multiple (convergent) means of usability assessment.

Gray & Salzman (1998) have criticised the comparative UEM literature in general, and five studies (Jeffries et al. 1991, Karat et al. 1992, Nielsen 1992, Desurvire et al. 1992, Nielsen & Phillips 1993) in particular, for lacking both internal and external validity. According to these authors, the five studies lack internal validity in relying on small sample sizes and one-off experiments and their claims to determine the scope of problem categorisation. Gray & Salzman specifically criticise the use of problem count as a measure of the effectiveness of UEM, recommending that researchers limit both their expectations and claims for UEM studies. One way would be to focus on the predictive power of analytic methods by distinguishing between correct predictions ('hits'), over-predictions ('false alarms' or false positives), misses, and 'correct rejections'. Another way would be to distinguish clearly between problem tokens (separate instances of usability issues) and problem categories. A third would be to employ multiple (convergent) means of usability assessment.
Chapter 1

5.5 Cognitive Psychology and Usability

As stated earlier, this thesis starts from the view that it is possible to identify a common set of principles which underlie a range of interface families, and that the rationale behind these principles can most readily be understood by cognitive psychology. The author's view is that by structuring each principle or sub-principle in the manner illustrated in Figure 1.5, it is possible to avoid the pitfalls of many guideline collections. It is also possible to make explicit the rationale for each principle by including the source(s) from which it is derived.

Figure 1.5 illustrates a method for developing principles of the type described in the previous chapter. The method was developed by the author and is based on the principles of cognitive psychology. The method consists of identifying a range of interface families, and then identifying the principles which underlie each family. These principles are then structured in a way that makes them easy to understand and to apply.

Some practitioners do not think so. Views range from the 'strong' position - that there is nothing in theories of human behaviour with sufficient generality and detail to guide the design and evaluation of computer systems - to the 'weak' - that the role of psychology in HCI has not yet been fully understood or utilized in sufficient detail.

The issue, then, is whether HCI research in general, and cognitive psychology in particular, is up to the job.

Though the project began by these authors remains incomplete (not all of the 14 "sensitive dimensions" are developed as fully as the example in Figure 1.1), and (as the authors say) the guidelines may require updating in the light of new research, the ideal - of generating design and evaluation principles from empirical findings - remains valid.

Some practitioners do not think so. Views range from the 'strong' position - that there is nothing in theories of human behaviour with sufficient generality and detail to guide the design and evaluation of computer systems - to the 'weak' - that the role of psychology in HCI has not yet been fully understood or utilized in sufficient detail.

The discipline of cognitive ergonomics (cognitive engineering in the US) exists to articulate just such a role for psychology:

"Cognitive ergonomics is concerned with the mental aspects of the [human-computer] interaction and so with developing specifications of the knowledge required by the human to interact with the computer to perform work efficiently. [...] To support the development of specifications of the cognitive ergonomics of the user-work-environment, the discipline of cognitive ergonomics needs to acquire and apply principles concerning the cognitive and affective aspects of human-computer interaction. [...] To support the acquisition of knowledge required by humans to use computers, the discipline of cognitive ergonomics needs to acquire and apply principles concerning the cognitive and affective aspects of human-computer interaction.

The discipline of cognitive ergonomics is concerned with the mental aspects of the [human-computer] interaction and so with developing specifications of the knowledge required by the human to interact with the computer to perform work efficiently. [...] To support the development of specifications of the cognitive ergonomics of the user-work-environment, the discipline of cognitive ergonomics needs to acquire and apply principles concerning the cognitive and affective aspects of human-computer interaction. [...] To support the acquisition of knowledge required by humans to use computers, the discipline of cognitive ergonomics needs to acquire and apply principles concerning the cognitive and affective aspects of human-computer interaction.

"What is different about the guidelines presented here is that they are built directly, in a structured manner, from the theoretical bases which had been largely untapped until now. Very few of the design guidelines which are currently available have been derived directly from research on high-level cognitive processes, along the lines described here. In previous chapters [of Gardiner & Christie 1987], psychological theory was distilled into broad 'principles' or summaries. These principles are now taken through a process of simplification and turned into guidelines which are as jargon-free and explicit as possible. [...] The aim is to demonstrate an approach which may prove more acceptable to designers and evaluators, and which is based on research which is more rigorous and detailed than that of previous guideline collections. Other researchers and writers have taken up the challenge of developing guidelines from research on high-level cognitive processes, and some have even developed guidelines which are more rigorous and detailed than those presented here.

"But what is different about the guidelines presented here is that they are built directly, in a structured manner, from the theoretical bases which had been largely untapped until now. Very few of the design guidelines which are currently available have been derived directly from research on high-level cognitive processes, along the lines described here. In previous chapters [of Gardiner & Christie 1987], psychological theory was distilled into broad 'principles' or summaries. These principles are now taken through a process of simplification and turned into guidelines which are as jargon-free and explicit as possible. [...] The aim is to demonstrate an approach which may prove more acceptable to designers and evaluators, and which is based on research which is more rigorous and detailed than that of previous guideline collections. Other researchers and writers have taken up the challenge of developing guidelines from research on high-level cognitive processes, and some have even developed guidelines which are more rigorous and detailed than those presented here.

What is different about the guidelines presented here is that they are built directly, in a structured manner, from the theoretical bases which had been largely untapped until now. Very few of the design guidelines which are currently available have been derived directly from research on high-level cognitive processes, along the lines described here. In previous chapters [of Gardiner & Christie 1987], psychological theory was distilled into broad 'principles' or summaries. These principles are now taken through a process of simplification and turned into guidelines which are as jargon-free and explicit as possible. [...] The aim is to demonstrate an approach which may prove more acceptable to designers and evaluators, and which is based on research which is more rigorous and detailed than that of previous guideline collections. Other researchers and writers have taken up the challenge of developing guidelines from research on high-level cognitive processes, and some have even developed guidelines which are more rigorous and detailed than those presented here.
Introduction

Psychology in usability.

Summaries were offered of both usability evaluation methods and the role of cognitive versus heuristics.

1. Heuristic evaluation was summarized and set in the context of the theme of principles and methods.

2. The author’s principles set was introduced and contrasted with Nielsen’s heuristics.

3. Heuristic evaluation was summarized and set in the context of the theme of principles and methods.

4. Summaries were offered of both usability evaluation methods and the role of cognitive versus heuristics.

Summary of Chapter 1

1. This Chapter introduced the two main themes of the thesis and attempted to place them in the context of research comparing both usability guidelines and usability evaluation methods.

2. The author’s principles set was introduced and contrasted with Nielsen’s heuristics.

3. Heuristic evaluation was summarized and set in the context of the theme of principles and methods.

4. Summaries were offered of both usability evaluation methods and the role of cognitive versus heuristics.

It appears, then, that at least some explanatory power for models based on principles concerning “the knowledge required by humans to use computers” has been and is being established. It does not, therefore, seem too much to expect that the cognitive principles in such collections as Marsh (1987) may prove to have similar potential when expressed explicitly.

1.97. See A. Moss (1969) and empirical basis of action theory (e.g. 5.9). E.g. A. Moss (1969), elaborations of cognitive walkthrough (e.g. 5.9).

2.5.2 (GOMS, KLM, CT and TAG) predates 1987. Since then, these have been extended to a comparable extent (see Section 5.3 for an analysis of these methods). The author’s interpretation of Barnard’s view is that in 1987 such design issues of formal experimental, even complex models such as GOMS (can el. 2.983) design issue of formal experimental, even complex models such as GOMS (can el. 2.983) (as opposed to informal experimental, even complex models such as GOMS (can el. 2.983) design issue of formal experimental, even complex models such as GOMS (can el. 2.983) design issue of formal experimental, even complex models such as GOMS (can el. 2.983) design issue of formal experimental, even complex models such as GOMS (can el. 2.983) design issue of formal experimental, even complex models such as GOMS (can el. 2.983)
Several issues were flagged and positioned for later Chapters. They include:

- comparison of principles and heuristics rather than principles sets;
- usability vs. utility;
- the role of evaluator expertise in heuristic evaluation;
- ethnographic reduction; the role of evaluator expertise in heuristic evaluation;
- inspection; usability problem severity judgment; usability problem identification and reduction; the role of evaluator expertise in heuristic evaluation;
- free exploration vs. set tasks; usability testing vs. inspection;
- usability problem identification and reduction; the role of evaluator expertise in heuristic evaluation;
- ethnographic approaches and usability vs. utility.