THE USE OF GUIDELINES IN INTERFACE DESIGN

Linda Tetzlaff

IBM, T.J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598
lst@ibm.com

David R. Schwartz

IBM, Programming Systems
11000 Regency Park
Cary, NC 27511

ABSTRACT

We studied the use of an evolving interface style book to evaluate the role of such guidelines in the development of style-conforming interface designs. Although the designs were judged to be generally conforming, study participants had significant difficulty in interpreting the guidelines. Our designers were manifestly task oriented and impatient with extraneous material. They depended heavily on the pictorial examples, often to the exclusion of the accompanying text. We conclude that dependency on guidelines should be minimized, and that guidelines should be developed primarily to *complement* toolkits and interactive examples, focussing on information intrinsically unavailable through those vehicles.

KEYWORDS: Guidelines, user interface design

INTRODUCTION

The purpose of user interface guidelines is to enable the development of usable, consistent applications, that conform to designated conventions. It is the experience across the industry that a wide variety of techniques must be enlisted to achieve this end [3, 4, 8, 9]. Among them is the creation of an interface style guide.

Guidelines vary in content: Some enumerate general principles; others explicitly specify interface detail; some focus on the presentation and behavior of the interface, while others address performance characteristics of the interface in use. Some are general recommendations for usability [6]; others, including those discussed here, define a style to be used in the development of applications on particular platform [1, 5] or across platforms [7]. These latter guides generally provide a centralized, explicit statement of the style to which developers are expected to conform. They provide a reference point for both development and evaluation.

Guidelines are almost never used in isolation. They may be accompanied by itemized checklists, a compliance process, and company-wide computer conferencing. Other techniques include press and user feedback, a set of highly functional, conforming applications, and supporting toolkits for application development.

Many of the techniques for assuring compliance are motivational (e.g., incentives) or focus on recovering from failures in the development process (e.g., consulting, and reviewing organizations). In an ideal world, developers would produce conforming applications on the first try (probably not fully usable and consistent, but at least conforming). While there are social vehicles that help achieve this (e.g., educational classes, informal information exchange), it is ultimately a cognitive task for the individual developer. In this study we explore the developer's use of guidelines by means of a practice design task.

When we began, we were skeptical that guidelines by themselves would be sufficient for development of conforming designs. Our objectives were the following:

- to determine what role guidelines actually play in the design of conforming applications,
- to determine how style guidelines are used in practice,
- to study the interface design process empirically.

METHODOLOGY

Participants were asked to develop a paper and pencil interface design conforming to guidelines. The problem concerned on-line management of a company sponsored, recreational activity (see Figure 1), and took about a day to complete. The problem covered the critical design issues developed in the guidelines, and was expected to elicit a range of interface detail.

Participants worked at their own pace, and were asked to verbalize all aspects of their work, including references to the book. Sessions were video taped, and the verbal protocols were logged by the experimenters in real time. Following the session, participants were debriefed and completed a satisfaction questionnaire.

Materials

In addition to the problem statement, participants were provided a copy of a document describing how to build interfaces in conformance with a prescribed style that covered both the appearance and the behavior of the

interface. The document was divided into three sections: The first introduced the interface style and the principles on which it was based; the second described the interface design process; and the third (Reference Section) provided an alphabetically organized description of each element (button, window, dialog mechanism ...) in the design. Each item in the Reference Section included a textual definition, a graphic illustration, and a list of details covering constraints and conditions of use. The book was roughly 300 pages long, of which about 250 pages were devoted to the itemized Reference Section.

The book was evolving as we worked, and participants received the most recent version of the document. Detailed results were fed back to designers and writers as they became available, resulting in upgrades to the document. The results reported here transcend individual versions of the document.

Participants

Nine people participated in the study. All were experienced users of the system they were designing for, and all were experienced interface designers. We tested people with varying degrees of experience with the guidelines themselves and with different levels of general programming experience, including backgrounds in both procedural and object oriented programming.

Your company recreation club would like to streamline its procedures for managing its various team sports. The general situation is as follows. Individual employees or their spouses submit either a paper or on-line application to play a particular sport. For each sport, someone is responsible for making up teams, scheduling games and keeping players informed about what's happening.

Coordinators routinely create a set of team rosters, a schedule for the season, and an address list, so that team members can contact each other. In addition, they need to be able to respond to requests for detailed information, and for summary reports about the league.

Coordinators are volunteers, and are hard to come by. Furthermore this activity only happens once a year, so they never get a chance to become experts with the system. As a result, the club is very concerned that the system be efficient and easy to use. Most of the coordinators work for the company and have personal computers available to them for their regular work, so naturally, the club would like these facilities to conform to company guidelines.

The system is also made available to the teams for tracking individual events.

Your Task

Your job is to design a guideline conforming interface which will meet these objectives. You should design the interface to support the range of sports sponsored by the club (e.g. soccer, baseball, tennis, etc.) although, for purposes of the exercise, you should illustrate your design for the softball season.

Your design should be detailed enough so that a user could figure out how to use it and a compliance coordinator could determine if it is guideline conforming.

Describe in detail how each of the following scenarios is to be performed. For each scenario please include:

1. A sketch of all relevant windows, including their contents and any controls, icons, title bars, etc.
2. A description of all available menus.
3. A description of any dialogs used in the scenario.
4. A detailed description of the anticipated user-interactions to accomplish the scenario.

The Scenarios

1. The user has just installed your software. Describe what the user now sees.
2. The coordinator needs to reschedule a game because four players will be out of town. It will not be possible to swap times with another team, and the game can't be scheduled for a night when another activity has the fields. Describe how the coordinator changes the team's schedule and the master schedule for the field.
3. One of the team captains calls up and asks the coordinator for all the information about his/her team, including all the information from the registration forms, the team lineup, the game schedule and the name and network id of the captains of opposing teams. S/he would like to get the information via the network. Describe how the coordinator makes this information available.
4. The club makes portable PCs available for recording events during the game (e.g., score, strikes, walks).

   The situation is as follows: Bases are loaded. The batter hits the ball and makes it successfully to first base; the player on second gets safely to third; the runner on third makes it home; but the player on first is tagged out before getting to second base.

   Describe in detail how the event is recorded.
5. How would a club staff member make an on-line application form for a new sport - for example, tennis?

Figure 1: The interface design problem

Two expert participants (designers of the guidelines themselves) were used to provide a baseline for comparing the resulting designs and the interface design process. We will refer to the other seven participants as 'novices,' though this is merely to distinguish them from the guideline authors.

RESULTS

Guidelines in practice

The guidelines covered many new concepts as well as intricate details of the proposed interface style. Many of the new concepts proved problematic for participants:

- Some were only partially understood.
- Some distinctions that seemed critical to guideline designers were not appreciated at all.
- Some proved difficult to think through and implement.
- Others struck them as fraught with potential usability problems.

The perception of usability problems left them with the dilemma of whether to conform in the face of their better judgement. Guidelines that result in unusable applications, may contaminate not only developers feelings about the particular recommendation, but may affect their view of the guidelines as a whole. For example, the guidelines we studied encouraged developers to create logical and hierarchical structures of objects, often resulting in long interaction paths to get to even commonly used functions. Some participants reluctantly preserved the long paths, believing it was necessary for conformance. Others pushed the guidelines aside and created more direct routes.

Despite a decidedly imperfect understanding, as evidenced by both the protocols and their own testimony in the debriefing, participants were able to produce design solutions that our evaluators (including a guideline author) judged largely conforming. However, conforming did not mean that the designs were identical. So, for example, some participants developed a direct manipulation solution to the event recording scenario (4), designing a picture of a baseball diamond around which players could be dragged to indicate the course of the game. Others developed variants on a score board which could be updated by means of controls or direct entry. The presentation and behavior of these solutions were quite different, but were mostly within the range of conforming designs.

In the discussion that follows, we will focus first on the use of examples, which proved key in conveying interface style, and then move on to a more general discussion of how participants worked with the book.

Examples

The graphic illustrations of the user interface components and style were central to conveying the sense of the guidelines both in general and in detail. They were cited spontaneously and almost universally as the preferred vehicle for learning. The first example in the first chapter took on unusual importance, with participants returning to it disproportionately — probably because it was easy to find. Examples were inspected carefully, were frequently referenced, and were copied directly in constructing the design. Participants all felt there should be more of them.

Early versions of the book included examples that illustrated the use of a single control (e.g., a particular menu type). Participants wanted examples to integrate interface elements, to show how they could be put together. Some were frustrated by the inability to explore the examples interactively.

Participants used examples beyond the scope of their explicit context. So, for example, if scroll bars happened to appear in an illustration intended to exemplify the use of radio buttons, if it was timely, the participant might take the example as representative of both without further reference to the entry for the scroll bar. In some cases, these incidental details had been provided primarily to make the examples more concrete. If these incidental items appeared generic, such as an icon representing the workstation itself, they were sometimes taken to be part of the system and covered by the guidelines. On the other hand, things that were clearly application specific, such as a pie chart, were not.

In summary, though the examples were no doubt constructed to *illustrate* the associated text, in practice (particularly in the Reference Section) the text was used more as if it *annotated* the examples.

Book usage

There were two general strategies for approaching the book. The canonical strategy of reading the introductory material first was spontaneously initiated by only two of the seven guideline novices. The others tried to embark on the task directly, looking only for the information they believed relevant to their work. However, these participants found they were unable to begin working on the task without getting a more general feel for the overall sense of the guidelines, and reluctantly returned to the introductory material. They read until they found what they thought they needed to know — and often no farther, missing details that designers no doubt expected them to absorb. Thus the prevailing *preferred* pattern of work was similar to the task-oriented behavior of adult, novice computer users, described by Carroll [2].

To the extent that the introductory material was germane to their objectives, participants read it with interest. Experienced participants found the general human factors guidelines contained in the book (e.g., providing useful feedback to the user, friendly error messages, aesthetically pleasing screens ...) uninformative. However, specific guidance on strategies for approaching the design were welcomed, and were consistently and carefully considered. Participants were impatient with needlessly long introductory material and with terminology that seemed peripheral to the task at hand. Carroll's minimalist approach [2] seems as relevant here as it does to the development of user manuals.

The alphabetical arrangement of the Reference Section encouraged browsing and facilitated serendipitous contact with other entries. For participants with previous experience, the alphabetical arrangement of the individual terms and interface controls proved effective, as there was considerable evidence of transfer of termi-

nology and concepts from previous versions of the book. However, participants often did not read sections they believed they recognized, leaving them vulnerable to more subtle changes in the design.

Design process

There were two discernible stages in the design. In the first stage, participants attempted to identify the major user interface objects. The second stage consisted of iterations through the scenarios, each iteration achieving a new level of detail in the design:

Object decomposition: The guidelines mandated interfaces that were populated by the "objects" that users need to perform tasks, and object decomposition was recommended by the book as the first step in the design process. All participants began with this task, roughing out a set of interface objects, largely based on the noun entities present in the problem statement. These were revised as the design progressed.

It was not obvious to most participants what a number of the "objects" should be. Participants considered the objects in the user's world, interface objects, and underlying application data, and it was not clear how these different "objects" were to relate. This problem was most clearly manifested in their design of the schedule (scenario 2). Should there be an independent schedule object, or should schedule be an action on some other object such as a field or a team? If schedule were an independent object, how should they implement scenario 3 which asks for the schedule for a particular team? If maintained for each team, how would they coordinate team schedules within the league?

Functional design: Whether working generally in the task domain or following the story line of the scenarios, participants began by considering in some detail the domain and task requirements of the design. What kinds of functions needed to be provided? How would the application be used? For novices, this stage of the design was completely anchored in the problem and made little reference to the book. In contrast, experts designed immediately in terms of the guideline detail.

This stage was characterized by frequent use of short what-if scenarios used to elaborate understanding of the total function. For example: What if I want the system to accumulate long-term statistics ... What if I can't get the mouse to work on a park bench ... The scenarios were spontaneous (not a systematic elaboration of alternatives), and transient (only some were encoded as notes for the design), though they seemed to make an important contribution to participants' internal model.

Coarse interface design: Functional design was followed by working through the scenarios to consider the kinds of windows and interaction techniques to be used, but with little specification of the explicit detail that would appear on the screen. This involved generally frustrating attempts to get guidance from the book, which provided little integrative information on how objects should combine to form applications.

At this point, most participants felt that by working out the *logical* design, they had completed the task. Some indicated that at this point, in real life, they would take their sketch of the design to users for feedback. They regarded the elaboration of the design into explicit controls and visual layout as a mundane and uninformative exercise. It was only at the insistence of the experimenter that they sketched out screen designs that could be reviewed for compliance.

Detailed interface design: In the final pass through the scenarios, participants worked out the details of the design. This stage made the heaviest and most successful use of the Reference Section.

Guideline sociology

We asked participants if the process we had observed matched their normal way of working. While some indicated that it did, others made clear that the experiment artificially induced guideline usage. One participant claimed he never used guidelines in real life, preferring to submerse himself in interactive examples of the interface environment. Another said that she and her colleagues rarely referred to guidelines. Instead they designed as best they could in terms of the task and usability requirements of their users, and negotiated for compliance status only at the end of the process, making adjustments as necessary to obtain it. A similar sentiment was echoed by another, impatient with the high level principles and introductory material: He wanted the book to explicitly specify the criteria against which his application would be evaluated. Both in conversation with the participants and in informal discussions with others involved in the guideline compliance process, it became clear that guideline compliance was a subjective and interpretive process that used the book merely as a starting point.

CONCLUSIONS

We had expected that guidelines would prove difficult source material to effect conforming designs. Our protocols indeed indicated that participants missed many critical concepts and details. That the resulting designs were judged to be largely conforming raises some interesting issues.

For one thing, we believe that the design per se may not be a highly sensitive test of the effectiveness of the guidelines to elicit conforming design. The protocols revealed many problems not apparent in the paper and pencil design that could be expected to prove problematic had the process been pursued through implementation. Problems were finessed by designers and evaluators alike. Designers copied by rote or invented where they did not understand, and evaluators compensated for designer error from their own understanding, for example, treating serious conceptual flaws as minor labeling errors.

Second, there is much yet to be understood about the process of judging design conformance. Reviewers were not consistent among themselves in their qualitative evaluation of the designs. Personal assessment of usability sometimes intruded into their judgement of conformity. Further, there is no reason to suppose that conformance issues which seem critical to guideline authors will also be salient to reviewers in the press or to end users. Users, at least, are more likely to think in terms of usability. Conformity will mean little if these are not achieved.

Nonetheless, the guidelines were more effective than we anticipated. They were valuable in providing:

1. Examples of conforming interface design.

2. Explicit strategies for tackling interface design.
3. Identification of the major interface concepts and components.

Examples appear to be the single most powerful vehicle for the communication of interface design style. Examples in this study were static and limited in scope. It seems clear that carefully crafted, rich, interactive examples would be more powerful still. Indeed, Tognazzini [8] credits the simultaneous release of MacWrite and MacPaint with the Macintosh for much of the consistency across Macintosh applications.

Important considerations in the crafting of examples include compelling coverage of critical material, attention to implications of incidental material, and integration of components. In those cases in which it is anticipated that the designer will need to extrapolate beyond the scope of the documentation, we would recommend multiple examples, so that the direction and intent of the guidelines are clear. Thus a model of mouse selection is better illustrated by drawing on text, spreadsheet and graphics than simply by illustrating with text alone.

To assure conformity, guidelines should be developed in conjunction with the supporting environment and toolkits. They should address novel and integrative conceptual issues intrinsically unavailable in toolkits or examples. They should document their scope and limitations, addressing conditions of use for individual components. Thus toolkits may dictate the form and parameters of a control, but something on paper needs to relate the variety of controls to the designers task, and to the conditions under which each alternative is appropriate. An application may illustrate effective assignment of choices to menus; nonetheless, the underlying organizational principle may be hard for designers to infer.

But it must be understood that much written material may go unread, misunderstood, and consequently unheeded. The longer and more complex the document, the greater the risk. To the extent that conformance to the interface is critical, the interface style needs to be carried in the examples, facilitated by toolkits, illustrated by interactive demos or well-formed applications, and supported by iterative usability testing. Guidelines will be most effective as a succinct *complement* to these more compelling techniques.

ACKNOWLEDGEMENTS

We are grateful to John Bennett, Tom Dayton, John Karat and Bob Mack for comments on early drafts of this paper.

REFERENCES

1. Apple. *Human interface guidelines: The Apple desktop interface.* Addison-Wesley Publishing Company, Inc., Reading, MA, 1987.
2. Carroll, J.M. *The Nurnberg funnel: Designing minimalist instruction for particular computer skill.* MIT Press / Bradford Books, Cambridge, MA, 1990.
3. Good, M. Developing the XUI style. In Nielsen, J., ed. *Coordinating user interfaces for consistency.* Academic Press, Inc., Boston, MA, 1989, 75-88.
4. Happ, A.J. & Cohen, K.C. Consistency in the user interface: Common User Access impact study. (TR54.513). IBM, ESD, User Interface Technology Department, Boca Raton, FL, 1989.
5. IBM. *Common User Access: Advanced interface design guide.* (SC26-4582). International Business Machines Corporation, Boca Raton, FL, 1989.
6. Shneiderman, B. *Designing the user interface.* Addison-Wesley, Reading, MA, 1987.
7. Sun Microsystems, Inc. *OPEN LOOK™: Graphical user interface application style guidelines.* Addison-Wesley, Reading, MA, 1990.
8. Tognazzini, B. Achieving consistency for the Macintosh. In Nielsen, J., ed. *Coordinating user interfaces for consistency.* Academic Press, Inc., Boston, MA, 1989. 57-74.
9. Zevin, R. Systems Application Architecture Common User Access compliance process. IBM, ESD, User Interface Technology Planning Department. Boca Raton, FL, 1990.