INTERNATIONAL JOURNAL OF HUMAN–COMPUTER INTERACTION, *13*(4), 411–419

Task-Selection Bias: A Case for User-Defined Tasks

Richard E. Cordes
IBM Corporation

Usability evaluations typically occur throughout the life cycle of a product. A number of decisions and practical biases concerning the tasks selected for usability evaluations can influence the results. A pervasive bias is to select only tasks that are possible to perform with the product under evaluation, introducing a subtle bias for the participants. One way to avoid this problem is to employ user-defined tasks (UDTs) in usability evaluations. In addition, having participants define tasks to perform in a product evaluation allows a more accurate assessment of product usability. This is because UDTs based on users' requirements and expectations should be relatively independent of the functional capabilities of a product. However, there are a number of methodological and practical issues that result from the introduction of UDTs in a usability evaluation. The best approach is to design hybrid evaluations using both UDTs and product-supported tasks.

1. INTRODUCTION

Within the computer industry, a common practice is to have products, before their public release, go through a usability evaluation (Nielsen, 1993). A usability evaluation typically consists of having representative users of a product perform typical product tasks within a controlled laboratory environment (Rubin, 1994). The fundamental goal of the evaluation is to identify usability problems with a product so developers can improve the product before real users encounter these problems (Dumas & Redish, 1993). Sometimes, developers reach conclusions about a product's usability by determining whether a product can meet some predefined usability criteria, for example, successful completion of 98% of the tasks. If the participants in the usability evaluation achieve or exceed the criteria, have a positive impression of the product, and no major usability problems remain in the product, then the typical assumption is that actual users will not have serious usability problems with the product.

Requests for reprints should be sent to Richard E. Cordes, IBM Corporation, 4205 South Miami Boulevard, CX4A/503, Research Triangle Park, NC 27709. E-mail: cordes@us.ibm.com

However, as shown by Molich et al. (1998), there is very little agreement among usability laboratories on the number and nature of problems uncovered by usability testing that follows this methodology. Part of the disagreement involves selecting the proper mix of tasks that is representative of the tasks the intended users will perform with the product. The degree to which usability practitioners achieve this goal can help determine how well they can consistently apply and generalize their results to real-world perceptions of usability (and see more consistent results between laboratories). Although there are numerous biases that can affect the results of usability evaluations (e.g., Cordes, 1992), task selection is a major and often overlooked one. In this article, I focus on task-selection biases, including ways to select tasks that better match users' expectations.

2. TASK-SELECTION BIAS

There are a number of potential biases that can occur when choosing which product tasks to evaluate. Tasks selected for evaluation should be representative of what users will do with the product and must be manageable and suitable for a laboratory evaluation. Such tasks are typically:

- Tasks that are short and fit within a test session. Long tasks reduce the number of participants for a given test schedule. There are also problems of participant availability and attrition for sessions that take longer than 8 hr.
- Tasks the evaluator knows how to do. Some product tasks are subtle, quite complex, and require a higher level of expertise than the practitioner has. For obvious reasons, practitioners do not evaluate tasks that they themselves do not know how to perform. The tasks they select typically represent the domain of product tasks with which they are familiar and know how to do. These are not necessarily the key tasks from the user's perspective.
- Tasks that are consistent with a laboratory environment. Some important tasks (e.g., product migration from a competitor's product) are difficult to replicate in a laboratory environment. Some tasks might be so user specific that it is not possible to construct typical scenarios. Also, there may not be adequate resources available to replicate a complex multiproduct customer environment.
- Tasks the evaluator finds interesting. The evaluator may have a predisposition to focus on tasks addressing product areas that were important or controversial during design or may simply feel more comfortable testing familiar areas of the user interface such as a graphical user interface and pop-up dialogs instead of documentation and messages.
- Tasks with available participants. The availability of participants will affect the makeup of the tasks chosen to evaluate. For example, tasks that require participants from the international community are less likely to undergo evaluation than tasks requiring local participants.

3. "I KNOW IT CAN BE DONE OR YOU WOULDN'T HAVE ASKED ME TO DO IT" BIAS

3.1. Background

The previous list of task-selection biases stems primarily from the practical logistics of conducting usability evaluations. Some are apparent and, once acknowledged, are easy to fix. Others equally apparent are more difficult or impossible to fix (so practitioners must simply accept them). There is, however, another task-selection bias that is less obvious and has far-reaching implications in how practitioners conduct and interpret usability evaluations. In most usability tests, practitioners only select tasks that the product supports. They do not ask participants to perform domain-relevant tasks that the product does not support. If participants know this, it can have a profound effect on their performance and attitude about a product. Part of the implicit demand characteristics (Orne, 1962) of usability studies is that all tasks that participants perform with a product are possible to do with that product. In contrast, users interacting with a new product are learning about a product's capabilities and limitations. Indeed, determining what a product can and cannot do is a fundamental aspect of learning how to use a product. Therefore, in a nonlaboratory situation, users may not be so sure that they can do the tasks they want to do with a given product. If users are not certain that they can perform their tasks with a product, this belief can bias the amount of time they are willing to spend learning to use a product to perform a specific task. Consequently, they might be much more likely to give up in times of difficulty and to feel that the product is much more difficult to use.

3.2. Magnitude of the Effect

Cordes (1989) evaluated whether a simple addition to task instructions, one that questions the ability of a product to support all tasks, would have an effect on users' thresholds for giving up when learning to use a product. In the experiment, I investigated whether this change in task instruction would affect the number of "I give up" phone calls made by the participants. In the double-blind study, two groups of 8 people participated in a usability evaluation of a software product under development. The participants received random assignment to one of two groups: control or experimental. Both groups received identical task instructions, except that the experimental group heard on their private videotaped instructions, "As in the real world, don't assume that the product can perform each task that we are going to ask you to do." The control group received identical instructions without this statement. The objective measure of a person giving up on a task was a telephone call to the experimenter.

An analysis of variance was conducted to evaluate the effects of Group, Task, and the Group × Task interaction using the dependent measure of the number of "I give up" phone calls. The two main effects and their interaction were statistically

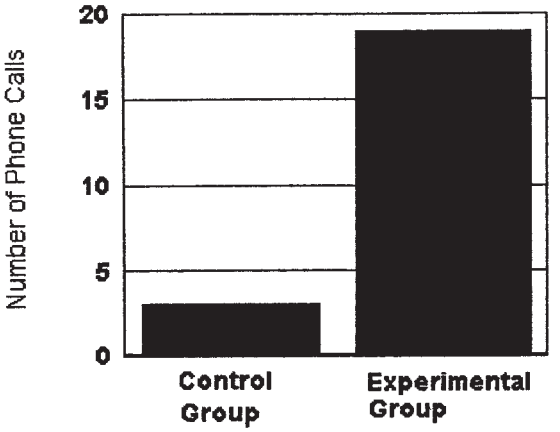

FIGURE 1 Number of "I give up" phone calls for each group.

significant, $F(1, 14) = 7.56$, $p < .016$ (Group); $F(10, 113) = 5.02$, $p < .001$ (Task); and $F(10, 113) = 2.86$, $p < .003$ (Group × Task interaction). The experimental group made a total of 19 phone calls (2.37 per person) compared to 3 (0.37 per person) for the control group (see Figure 1). This is an increase of over 6 times more phone calls due to the instructional change. The significant Group × Task interaction (see Figure 2) indicated that the difference in phone calls between the two groups was related to the task performed. The control group did not give up on any task that the experimental group did not also give up on. However, besides having a higher rate of giving up on these tasks, the experimental group gave up on two additional tasks. This suggests that the experimental group had an overall lower threshold for

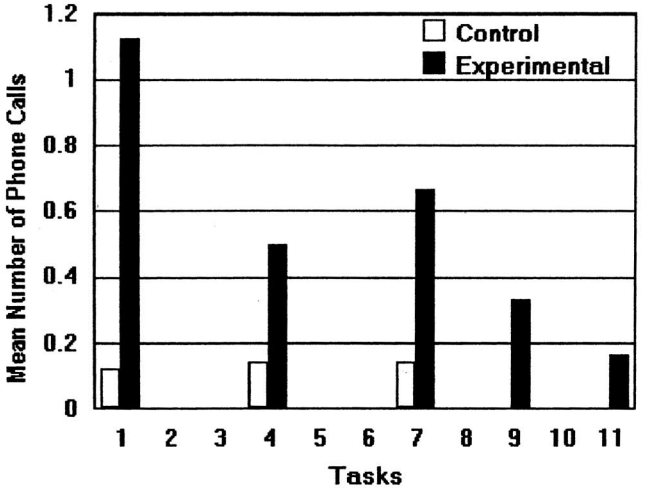

FIGURE 2 Mean number of phone calls by group and task.

giving up. The difference between the groups was attributed to what the participants judged to be the more difficult tasks. As would be expected, tasks judged easy were not sensitive to this manipulation (i.e., the participants gave up only on tasks they perceived to be difficult). Looking at only the "gave up" tasks, the participants' task ratings (using magnitude estimation; Cordes, 1984) showed the experimental group rated these tasks to be over 14 times less difficult than the control group. In addition, the experimental group took a little more than half the time to give up on these tasks as compared to the control group (33 min vs. 59 min). This suggests that not only were the participants in the experimental group willing to give up more frequently, but they also were willing to do so sooner while experiencing less difficulty than the control group.

4. USER-DEFINED TASKS (UDTs)

4.1. Rationale

Cordes's (1989) results demonstrate that task instructions that bring into doubt one's ability to use a product to perform assigned tasks can directly affect a person's threshold for giving up. Therefore, usability evaluations that do not control for this effect have a strong bias in favor of a product achieving a higher successful task completion rate. More people would fail to complete tasks in usability studies if they were less certain that the tasks they attempted were possible to do. Although a change in task instructions appeared to successfully reduce this certainty, this manipulation was artificial. After all, how many products come with a warning label stating "Don't assume you can do everything you want with this product"? Also, hearing an instruction of this nature is unusual and may serve to arouse suspicions and cause an overreaction. For example, participants may believe "Some of these tasks must not be doable with this product or I wouldn't have been given this instruction." Therefore a better method for controlling this bias needs to be explored. Typically, the lack of certainty that a product can perform users' tasks arises from an incomplete match between user requirements and product functions. The Venn diagram in Figure 3 summarizes the situation.

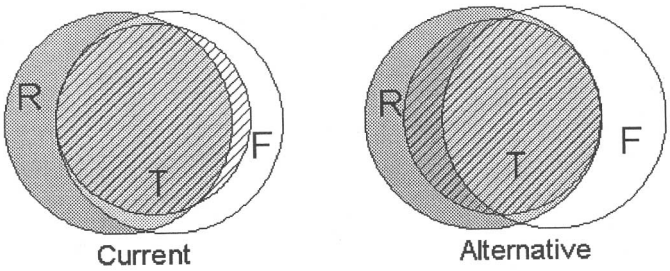

FIGURE 3 Venn diagram of product requirements (R), product function (F), and human factors testing (T).

Figure 3 shows user requirements (R) and product functions (F). The overlap between these areas represents the degree to which a product has met the user requirements. Typically, usability testing (T in the figure) rests entirely in the F domain, based on tasks that are mostly a subset of R. The usability that users experience, however, depends on the R rather than the F domain. Because users are never completely sure that a product will meet their requirements, they are very likely to try to perform tasks that the product cannot do. Alternatively, if the evaluation included tasks sampled from the R domain, the results would provide a more accurate picture of product usability than sampling strictly from the F domain.

Of course, if product tasks better matched user requirements, then selecting tasks from the F domain would be less of a problem. However, the functions and features that end up in a product represent a compromise between users' requirements and what developers can achieve in a given time, resulting in a reduced set of product functions.

When selecting only the tasks that products can support, practitioners are not evaluating usability from the user's perspective but are only evaluating usability from the perspective of the product functions. When users spend hours trying to perform a task with a product that cannot do it, this experience must have an impact on their perception of product usability. This "learning the capabilities" of a product and how they match user needs is an important component of usability that rarely receives evaluation in laboratory-based usability studies. Field studies that focus on users performing tasks in their real environment do tap into the users' perspective but are typically time consuming, expensive, logistically difficult, and, therefore, less often performed. One way to design laboratory-based usability evaluations that better reflect users' perception of usability and control for the "I know it can be done or you wouldn't have asked me to do it" bias is to include UDTs.

4.2. Definition

UDTs are simply tasks that participants bring into the usability evaluations as opposed to the product-supported tasks (PSTs) that make up most usability evaluations. By having users bring into the laboratory tasks they want to perform and believe they should be able to perform with a product, a truer picture of usability can emerge.

From experimental and practical points of view, however, UDTs can present problems:

1. Because the choice of tasks is up to the participant, practitioners lose control over the content and, potentially, the duration of the evaluation. This can be a problem if participants will perform multiple UDTs, particularly if any of them are not possible to do.
2. The quality of tasks that users bring into the evaluation can vary considerably. For example, in an attempt to test the capabilities of a product, some participants may choose tasks that are currently a challenge for them to perform, but others may choose routine tasks to make sure they are still being supported.

3. Experience with prior or similar versions of the product may influence the tasks the participants will bring into the evaluation. Their expectations may have previously been set, and this will affect their choice of tasks to perform. For example, participants may have a requirement for spell checking on a simple text-editor product, but because the version of this product they are familiar with does not have this feature, the tasks they choose will not include it either.
4. You may not know who the participants will be until they show up, and unless you know this, you cannot collect the tasks needed for the study. Also, if participants are being reimbursed, it is not apparent how to reimburse them for the time spent working on a task to provide for the study.
5. The tasks chosen by the participants will be unique. This introduces the possibility that the tasks chosen by each participant will only be performed by that participant, making it difficult to provide summary analyses (such as means).

Setting time limits for accomplishing tasks can give the practitioner control over the duration of an evaluation. Of course, this control will be at the expense of knowing whether the participants who exceeded the time limit would have eventually given up if they had more time. However, as shown earlier, when task "doability" was uncertain, participants who gave up did so in about half the time, so this issue may not be faced that often. Regarding the second point, although UDTs may vary in quality, it is better to accept this range than to screen the tasks that are submitted by the participants because these tasks come from the user requirements domain. Restricting or selecting the UDTs based on quality implies a more complete knowledge of user requirements than is typically the case. As mentioned previously, based on previous experience, the participants themselves might restrict or otherwise alter their choice of tasks. However, if participants do self-limit their choice of tasks, or even choose tasks that have been dropped from the product, then this is really not a bias problem at all because it accurately reflects their product requirements view. Regarding the difficulty in obtaining UDTs from unknown participants (reimbursement, etc.), a solution would be to specifically hire participants to create the UDTs prior to starting the evaluation then use these UDTs in the study. Overcoming the last problem requires that a study employing UDTs have each participant (or at least more than one participant) perform the same task. A possible solution is to collect a single UDT from each participant before the evaluation and then have each participant perform all of the collected UDTs. For example, if there were 8 participants planned for a study, the evaluator could collect one UDT from each participant and then have each participant perform all eight UDTs. That would permit the use of basic descriptive statistics when summarizing the results. However, this approach also presents some problems:

1. Equating the number of UDTs to the number of participants in a study will only be practical for small numbers of evaluations if the individual tasks take long to complete. For example, it may be too time consuming to run a study with a larger number of participants if each task takes on average longer than one-half hr.
2. For each participant, one of the tasks to perform will be the task that the participant brought to the evaluation. It is likely that the participant will perceive this task

differently than the others (e.g., the participant may have a greater interest in or more motivation to perform this task), introducing some extraneous variability into the study.

A way to solve the first problem is to randomly select a manageable number of tasks from the group of participants. For example, if 12 tasks solicited from 12 participants would incur too long of a session, then the evaluator might randomly select three of the UDTs for inclusion in the study. Regarding the second problem, this might not be a liability because in the real world users will perform some tasks of their own choosing and others that they receive an assignment to perform. A bigger concern is that in the time interval between submitting the task to the practitioner and actually performing it, a participant may plan the effort and, in some cases, perform the task ahead of time. One way to address this problem is for the practitioner to collect tasks from at least t more participants than he or she plans to observe ($n^* = n + t$). The practitioner then randomly selects the t tasks and excludes from the study the t participants who provided these tasks. For example, in a study with only time for 12 participants performing three tasks, you collect tasks from 15 participants, randomly select three tasks, and then exclude the 3 participants who provided the tasks.

Another way to address the second problem is to make the a priori decision not to collect data from participants on the task that they contributed to the study and treat these data points as missing values.

Regardless of how UDTs are employed, their introduction can only serve as a method for controlling the "I know it can be done or you wouldn't have asked me to do it" bias if the participants know that at least some of the tasks came from other participants and that the tasks were not selected or tested for doability by the practitioner. Rather than identifying the origin of each task, this information can best be conveyed by simply telling them that these tasks come from their fellow participants and that they are all going through them for the first time. This is an honest and realistic statement that implicitly brings into question the doability of all the tasks without sounding artificial or manipulative.

4.3. Recommendations and Discussion

Although practitioners should adopt UDTs in their usability evaluations, they should not stop using PSTs. PSTs serve a very useful purpose by

1. Assuring the evaluation of most of the functions of a product.
2. Allowing the evaluator to include tasks that users might perform rarely and, therefore, would be unlikely to be UDTs.
3. Evaluating tasks that are not currently part of the user requirements.
4. Enabling the experimenter to focus on specific tasks suspected to include usability problems.

Usability practitioners should design usability studies that incorporate UDTs with their PSTs. This hybrid approach allows a more realistic assessment of product usability through the introduction of tasks that are based on user requirements and are independent of product capabilities. This approach can also help practitioners avoid the "I know it can be done or you wouldn't have asked me to do it" task-selection bias by implicitly bringing into doubt the doability of all the tasks that participants are asked to perform. By incorporating UDTs into usability evaluations, practitioners will be in a position to assess product usability from the users' perspective with greater accuracy.

Although my colleagues and I have successfully followed the recommendations presented in this article, more systematic work needs to be done to validate the benefit of UDTs and their effect on the "I know it can be done or you wouldn't have asked me to do it" bias. No study comparable to the one highlighted in this article has been performed to evaluate the effectiveness of adopting UDTs in controlling this bias. For example, what effect do actual impossible tasks have on this bias? Participants' willingness to give up on a task might be quite different if they find that all prior tasks were doable versus only 10%. Also, is there an optimum mix of UDTs and PDTs that produces the best results in terms of incorporating user requirements and evaluating product functionality? Perhaps one UDT per participant is sufficient to be able to generalize to real-world product usage, but more likely more are necessary. Finally, if we hope to improve the practice of usability, additional studies need to be performed to investigate the ramifications of the task-selection biases and remedies discussed in this article.

REFERENCES

Cordes, R. E. (1984). Application of magnitude estimation for evaluating software ease-of-use. In G. Salvendy (Ed.), *First USA–Japan Conference on Human Computer Interaction* (pp. 199–202). Amsterdam: Elsevier.

Cordes, R. E. (1989). Are software usability tests biased in favor of your product? *Proceedings of the 28th ACM Annual Technical Symposium* (pp. 1–4). Gaithersburg, MD: ACM.

Cordes, R. E. (1992). Bias in usability studies: Is this stuff science? *HFS Test and Evaluation Newsletter, 7*(2), 2–5.

Dumas, J. S., & Redish, J. C. (1993). *A practical guide to usability testing.* Norwood, NJ: Ablex.

Molich, R., Bevan, N., Butler, S., Curson, I., Kindlund, E., Kirakowski, J., & Miller, D. (1998). Comparative evaluation of usability test. In *Proceedings of the Usability Professionals Association 1998 Conference* (pp. 189–200). Washington, DC: UPA.

Nielsen, J. (1993). *Usability engineering.* Boston: Academic.

Orne, M. T. (1962). On the social psychology of the psychological experiment: With particular reference to demand characteristics and their implications. *American Psychologist, 17,* 776–783.

Rubin, J. (1994). *Handbook of usability testing: How to plan, design, and conduct effective tests.* New York: Wiley.