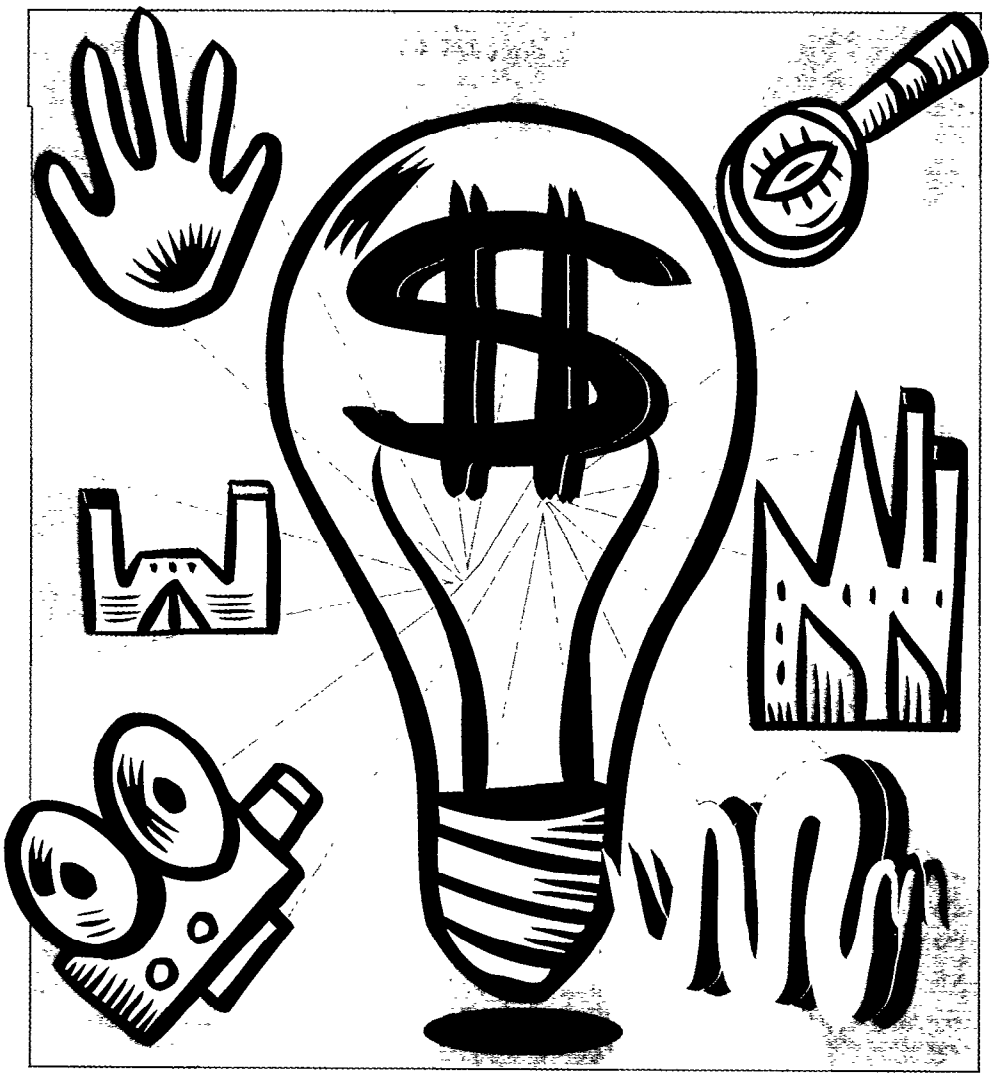
JAKOB NIELSEN

Why GUI Panic is Good Panic

One of the oldest jokes in computer science goes as follows:

Q: How many programmers does it take to change a light bulb?
A: None; it is a hardware problem!



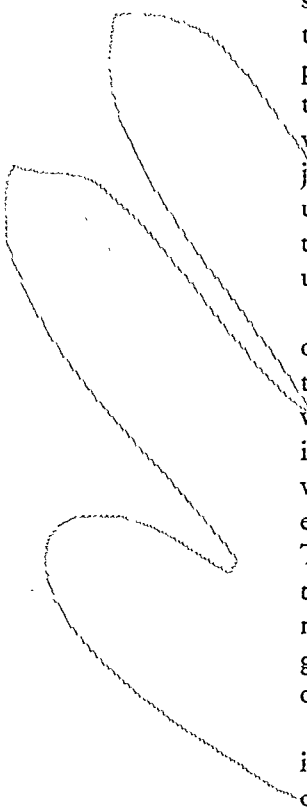
Pamela Hobbs

When asking how many usability specialists it takes to change a light bulb, the answer might well be four: Two to conduct the field study and task analysis to determine whether people really need light, one to actually screw in the light bulb, and one to operate the video camera filming the event. Unfortunately, the perception that anybody touching usability will come down with a bad case of budget overruns is keeping many software projects from achieving the level of usability their users deserve.

As an example of the cost of usability engineering, consider the usability laboratory, which is the most visible aspect of usability engineering in many companies. According to a survey I did of thirteen usability laboratories, the median usability lab takes up 670 square feet (64 m^2). When assessing the cost of this lab space, one needs to consider not just the rent but also the costs of lighting, heating and air conditioning, cleaning, building security, and many other overhead elements. Adding in these cost factors results in an estimated cost of lab space of about $70 per square foot per year in many of those fairly expensive parts of the United States where a large proportion of the world's information technology industry is located. These estimates lead to an annual cost of about $47,000 for the lab space alone. The equipment costs for usability laboratories vary significantly (in my survey, from $55,000 for a small lab without any rebuilding to $650,000 for a large lab with seven test rooms), but a representative number seems to be about $100,000, or about $25,000 per year given that the equipment has to be depreciated very fast whereas the remodeling costs can be depreciated over many years. Finally, the median usability lab required support staff equivalent to one

full-time person for maintaining and scheduling the lab and bringing in test subjects. Add maybe $78,000 for the loaded salary and other personnel costs of the support staff, and the total annual cost of a usability lab comes to $150,000.

Faced with a cost of $150,000 (which does not even include the cost of the usability specialists who will be running the tests), it is not surprising that many development managers are tempted to forget about usability or at least postpone it to next year's budget. In fact, even the most expensive usability laboratory is well worth its cost, and experience shows that project managers in companies with an established usability effort normally end up overbooking the usability lab with requests for many more user tests than the facility can support.



The problem with usability laboratories and other advanced usability methods is not that they cost too much in the sense of not being worth the money. The problem is that they intimidate developers and project managers who have not yet experienced how usability engineering can help their products succeed. Therefore, I have been advocating an alternative approach called "discount usability engineering" to lower this intimidation barrier and get more people started using usability methods.

The discount usability engineering approach is based on cutting the costs of usability methods as much as possible while reducing their benefits as little as possible. Typically, a kind of 80-20 rule applies, whereby about 80% of the results can be achieved with 20% of the effort. Specifically, discount usability engineering uses the following four techniques to make usability engineering methods easier to use:

• Simplify personnel requirements by making usability methods easier to learn. For example, heuristic evaluation, which is one of the prime discount usability methods, is based on only ten general principles for good user interface design as opposed to guidelines-based inspection which often uses fat books of a thousand rules or so.

• Simplify the equipment requirements. Cheap user tests can be conducted before a usability laboratory is built by having the experimenter take notes in real time on a notepad instead of relying on videotape to record the experiment. Of course, important information will get lost since it is impossible to notice everything in real time, but there will still be much more information gained from the handwritten notes of a simple user test than from the vacuum of knowledge that would be the result of doing no testing at all.

• Simplify the communication channels. Every time information has to be communicated by written reports, some of it will be lost in transit, and the time needed to write and read the reports will increase the expense of the study. Alternatively, members of the development team can be invited to observe user tests or other usability engineering activities directly. Of course, it is still preferable to have usability specialists analyze the results of the studies, and it is normally not feasible to have all developers observe all phases of all studies. Therefore, some kinds of reports will still have to be produced, but they can be made much simpler when the developers know what happened in the study by virtue of having participated in part of it.

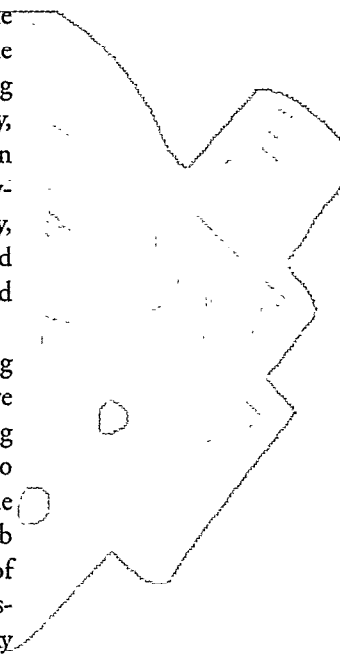
•Reduce repetition. Almost always, one learns the most the first few times one conducts a certain form of study, and diminishing returns rapidly set in. For example, user testing should normally be done with about 3-5 users. Across six studies, the proportion of usability problems found with three test users was 61% and the proportion found with five users was 78%. Certainly, these statistics mean that 22% of the problems had still not been found after five test users, but the point is that it would be very expensive to find these remaining usability problems. For example, in these studies, 11 users would have been needed to find 95% of the usability problems and 18 users would have been needed to find 99%. If the interface is being developed using iterative design, there will be no reason to test it exhaustively at this much higher cost since it will be changed anyway. It will be much better to "save" some of the test users for later iterations and make sure to test them too. For many development projects, the relative benefits of many small tests are even greater than indicated by the averages presented here since the most severe usability problems have a tendency to be somewhat easier to find

than the less severe ones. Thus, true interface catastrophes are likely to be revealed in a simplified usability test, even though some more subtle problems may remain hidden.

As always in the user interface field, there are potential tensions and trade-offs between these guidelines for making usability engineering cheaper. Consider, for example, ways of doing discount field studies without the travel cost of having usability engineers actually go to see the customers. Two approaches that have been tried are diary studies where users log events they think may be of interest to the usability team and self-directed videotapes where users are given a hand-held 8mm video camera and asked to tape themselves while they narrate their work in its natural setting. Both methods reduce cost and simplify personnel requirements, but they complicate the communication channels since the usability engineers will have to interpret the users' diaries and videos rather than gaining the opportunity for direct observation of the users and their work environment. Furthermore, the logistics with respect to equipment are also made more complicated if video cameras have to be shipped to the users rather than having a usability specialist bring a camera as part of a field trip. As a final comment, serendipity is a major element of field studies since one always observes new and unexpected phenomena when visiting users. These added findings would probably be lost if the usability specialists stayed home and relied completely on user-controlled discount field studies. Even so, it may very well be possible to combine results from a few actual field trips with additional data from cheaper, user-controlled studies to achieve approximately the same results at a lower cost than needed to visit a complete set of customer sites.

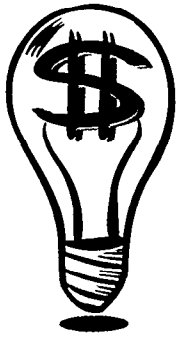
It is very common for usability engineering to be able to double the measured usability of a product, corresponding, for example, to cutting a company's training budget in half. The difference between using usability specialists and regular developers to carry out usability activities is around 130% (for example, they are able to find more than twice as many usability problems). This number is based on only two studies and should therefore be considered as a preliminary estimate. The added benefits from good usability as opposed to only average usability engineering skills are about 34% when measured as the ratio between the top quartile and the median results of usability engineering. This number is based on nine studies and is thus somewhat more reliable, though it would still be preferable to have more data. The conclusion from these results is that the two most important aspects of usability are to use usability engineering methods and to employ usability specialists to do so. Exactly what methods are used and exactly how they are carried out are of less importance, meaning that there is no reason to hesitate and wait for the perfect method to be developed or the perfect lab to be built. Does this mean that one should not strive for better methods or for better facilities or for better staff? Not at all! The last 34% improvements in the user interface are certainly worth taking.

As an example, consider an insurance company with 2,000 agents who use custom-built computer systems for about 5% of their working day on the average. If the usability of the systems could be improved by the 34% we expect from world-class usability as opposed to average usability efforts, the company would save about five million dollars per year, assuming that the loaded cost of an insurance agent is $150,000. There are two main ways for the company to realize these benefits given that the agents will need to spend less time on training and at the keyboard. In an expanding company, the agents might be able to spend more time on the phone or do other customer-oriented activities to generate more business. Alternatively, the company may follow the downsizing trend and lay off staff corresponding to the saved time.



What would be the added cost of improving the insurance company's usability efforts? If we assume that it has a small usability engineering group of five people, it might be possible to move the group from average to great by the following steps: Give them a top-of-the-line lab with extensive support staff, at a cost of $250,000 per year; attract top talent by increasing annual salaries by $20,000 per usability engineer and $40,000 for the group manager (corresponding to an increased expense of maybe $150,000 per year since loaded salaries only increase slightly more than take-home

pay); and improve methodology by allocating $100,000 per year for conferences, consultants, and similar activities (remember that the cost of a conference trip is not just the out-of-pocket expenses but also the cost of not doing other work for a week). The total expense of upgrading usability would be about $500,000 or about 10% of the expected benefits.



Jakob Nielsen is a distinguished engineer in the Human Factors Engineering organization at SunSoft in Mountain View, California. email: jakob.nielsen@eng.sun.com

In the beginning of this column I suggested that many managers would balk at paying $150,000 for an average usability laboratory, and they would surely balk even more at paying half a million for upgraded usability engineering unless they hear very convincing arguments. As we have seen, there might be about $4.5 million to be made from the investment, but these savings will only be fuzzy promises of future success at the time the investment is made. In my example, the deluxe usability engineering approach had a ratio of benefits to costs of about 10, but I normally observe ratios of between 50 and 100 in the discount usability engineering projects I am involved with, corresponding to a cost of maybe $5,000 or $10,000 to save $500,000. Of course, it is better to save $4.5 million than "only" $0.5 million, but the cheap projects are easier to get approved on a trial basis. Once management has seen the immense value of usability engineering, additional resources are almost always allocated and the organization can begin evolving towards recommended usability engineering practices, and maybe even get world-class usability engineering facilities.

In the new book *Cost-Justifying Usability* (edited by Randolph Bias and Deborah Mayhew and published this month by Academic Press), Kate Ehrlich and Janice Rohn have an interesting chapter about the development of organizational attitudes towards usability engineering from skepticism over curiosity to acceptance, ending with a true partnership between an established usability engineering group and the development groups. I think it is inevitable that a development organization will pass through most of these early stages before it evolves to the point of world-class usability engineering. Therefore, usability specialists should not push too hard on an organization that is still in the skepticism stage: doing so will not instantly transform the company to the partnership stage but will likely result in even deeper skepticism. Instead, we should take advantage of every opportunity to move the company to the curiosity stage and the first, simple attempts at usability engineering.

In my experience, many companies enter the curiosity stage when market pressures force them to move to a new interface technology with which they are unfamiliar. Having to design completely new interfaces often serves as a powerful motivator to learn more about usability and to bring in a small amount of usability expertise. In the last few years, this effect has most often been observed due to the shift from character-based interfaces to graphical user interfaces, and I therefore refer to the phenomenon as "GUI panic strikes" though very similar panics are likely to occur in connection with the coming shifts to strongly object-oriented interfaces and nomadic interfaces. As we have seen, from the perspective of introducing usability engineering methods into reluctant development organizations, GUI Panic Is Good Panic.