Gender HCI Issues in End-User Software Engineering

Laura Beckwith
beckwith@cs.orst.edu

Introduction

There has been considerable work in empowering end users to problem solve by writing their own programs, and as a result, end users are indeed doing so. In fact, based on U.S. Department of Labor and U.S. Census figures, the number of end-user programmers is projected to reach 55 million by 2005 in the U.S. alone. Unfortunately, evidence from the spreadsheet paradigm, the most widely used end-user problem-solving environment, abounds that end-user programmers are extremely prone to errors. This problem is serious in IT, because the programs created by end users, who are IT workers can be quite important to their business livelihood.

We have been working to help reduce the error rate in end-user programs, a vision we call *end-user software engineering*. It is a holistic approach to the facets of software development in which end users engage. Its goal is to bring some of the gains from software engineering research to end-user problem-solving environments, without requiring training or even interest in traditional software engineering techniques.

Despite the considerable HCI research relevant to end-user problem solving and end-user software engineering, researchers have not focused on potential gender HCI issues. In particular we will focus on *IT workers*, a majority of whom do not have background in computer/information science. (In this document, we contrast these IT workers with females who are professional programmers, whom we term *computer science females*.) We believe that research on end-user problem-solving environments must delve into how these systems can support *both* genders for the following critical reasons:

Women working in IT-dependent fields face a glass ceiling: Of the women in the IT workforce, many are in low-level positions, occupying 86% of data-entry positions, while the number of females at the higher end of the IT industry is estimated at 5%. The goal of our research is to work towards gender-conscious problem-solving environments that do not add to the glass ceiling.

Ignorance of gender HCI issues is risky: The number of women graduating with degrees in computer science has been steadily declining. Today's low percentage of computer science females has been directly attributed to the past unawareness of gender issues in computer science education and the workforce. There is a risk that if gender HCI issues of problem-solving environments are not investigated, female IT workers too will be driven from IT-dependent fields, just as has occurred with computer science females.

If gender HCI issues in end-user software engineering continue to be neglected, software supporting the growing millions of end-user programmers may be making the same mistakes as did our academic ancestors: excluding women from full participation.

Research Questions

To shed some light on this problem, we propose to begin our investigation with the following research questions:

1. Confidence: We currently know aspects of how gender differences, such as lack of confidence, affect computer science students; do men and women, while engaged in end-user software engineering activities, perceive risk differently? If we change the system according to our findings from this question, does this reduce the gender difference in perceived risks?
2. Support: When engaged in software engineering activities what type of built-in help support do women need versus the support that men need? Once we provide help support to suit both genders, do men and women perform similarly?
3. Motivation: Are the issues that typically motivate women supported for those engaged in software engineering activities? If we make changes to the system do both genders benefit equally?
4. Collaboration: Given collaborative settings, what attributes lead to failure/success for women in end-user software engineering?

Although our research questions focus on ways to include females, our studies will be designed to also measure whether our changes inadvertently exclude males.

The Research Setting

In our end-user software engineering research prototype, assertions are a new feature that provide a context for investigating HCI issues. Users can protect

their programs from invalid values by placing assertions on objects to specify, for example, that the value for an object should fall within a specific range. The user is notified of assertion inconsistencies via a visual marking on the program, which can be attended to as the user chooses. Despite assertions' usefulness in finding errors (when already placed on a program), users are unlikely to create their own assertions without support. To encourage users to create assertions we devised a strategy we term *surprise-explain-reward*. The system generates assertions, which are intended to *surprise* the user and generate their interest (in assertions). The user can then look to the assertion for an *explanation*, which includes the semantics, possible actions the user can take (regarding the assertion), and the future reward of taking the action. Providing the information about the reward in the explanation is tied to Attention Investment, an analytic model of user problem-solving behavior that allows a designer to consider the costs, benefits, and risks users weigh in deciding how to complete a task. By providing users with an idea of future benefits, they can better assess if the cost of using assertions is worth their time.

Research Questions 1 and 2: Confidence and Support

"I'm actually kind of discouraged now. Like I said before, there's so many people who know so much more than me, and they're not even in computer science. Like I was talking to this one kid, and …oh my God! He knew more than I do. It was so… humiliating kind of, you know?… But I feel like I'll always be behind, and it's discouraging."

Low confidence levels, as computer science females compare themselves to the males, are well documented. But low confidence relating to technology is not confined to computer science females. One study investigated how males and females viewed their ability to complete complex tasks in spreadsheets and word processing programs and found that females were significantly less confident than the males.

One definition of confidence is belief in one's abilities. We hypothesize that low confidence and fear of failure affects how a user decides to explore unfamiliar aspects of software. If females have a fear of failure when exploring unfamiliar features (such as assertions), and if our surprise-explain-reward attempt to decrease the perceived risk is not gender-conscious, females may still view the risk as being too high.

Research Question 2 asks whether built-in support itself has gender implications. For example, studies show that males and females require different kinds of built-in support when navigating around 3D worlds. We plan to investigate gender differences in built-in explanations (as part of the surprise-explain-reward strategy).

Research Paradigm

Our research paradigm is to investigate each research question with a pair of studies: a qualitative one and then a quantitative one. The qualitative studies follow verbal protocols to elicit subjects' reactions and rationale during as series of tasks. The results from the first qualitative study are used to inform design changes in our original research prototype. The quantitative empirical studies are then used to evaluate if the changes were successful. The results of each study pair will be used to refine the design of the next pair of research questions.

Research Question 3: Motivation

Research has shown that computer science females are motivated by how technology can help other people, whereas males tend to enjoy technology for its own sake. As an example, the following quote is from a computer science female at CMU describing why she chose to major in computer science:

> "I think with all this newest technology there is so much we can do with it to connect it with the science field, and that's kind of what I want to do (study diseases) … Like use all this technology and use it to solve the problems of science, the mysteries."

Miller, a leader in women's psychology issues, highlights that, in general, women often make it their life work to serve others, while men are specifically discouraged from doing so by society.

Drawing upon women's orientation toward serving others, we will research the ways motivation can influence HCI techniques for end-user software engineering. For example, if two women are sharing a program, does one engage in software engineering more thoroughly so she can give the "gift" of more reliable program to her co-worker? Following the research paradigm in the same way as for Research Questions 1 and 2 above, we will investigate gender differences in how motivation relates to rewards of using end-user software engineering features.

Research Question 4: Collaboration

Relying on collaboration to problem-solve is common among IT workers. One study found differences in how same-gendered teams best divided team work when problem-solving on a computer with two mice; the female teams performed better by passing-off control of the mouse to their partner, whereas the males we just the opposite, performing better by being able to take control of the mouse. Using the same research paradigm, we will research how gender HCI issues relate to collaboration in end-user software engineering.