Gender and End-User Computing

Laura Beckwith
Oregon State University, USA

Margaret Burnett
Oregon State University, USA

Shraddha Sorte
Oregon State University, USA

INTRODUCTION

Although gender differences in a technological world are receiving significant research attention, much of the research and practice has aimed at how society and education can impact the successes and retention of female computer science professionals. The possibility of gender issues *within software*, however, has received almost no attention, nor has the population of female end users. However, there is relevant foundational research suggesting that gender-related factors within a software environment that supports end-user computing may have a strong impact on how effective male and female end users can be in that environment. Thus, in this article, we summarize *theory-establishing* results from other domains that point toward the formation of grounded hypotheses for studying gender differences in end-user computing.

There has been much background research relevant to human issues of *end-user computing*, which we define here as problem-solving using computer software, also termed *end-user programming* in some of the literature (e.g., Blackwell, 2002; Green & Petre, 1996; Nardi, 1993). (See the glossary for definitions of these and related terms.) Despite this, few researchers have considered potential *gender HCI issues* and gender differences that may need to be accounted for in designing end-user computing environments. The most notable exception is Czerwinski's pioneering research on the support of both genders in navigating through 3-D environments (Czerwinski, Tan, & Robertson, 2002; Tan, Czerwinski, & Robertson, 2003). Although individual differences, such as experience, cognitive style, and spatial ability, are likely to vary more than differences between gender groups, evidence from Czerwinski's work as well as work in other domains, such as psychology and marketing, has found gender differences relevant to computer usage. In fact, some research has shown that some software is (unintentionally) designed for males (Huff, 2002).

One reason gender HCI issues in end-user computing are important is that ignorance of gender issues has already proven to be dangerous: today's low percentage of computer science females (Camp, 1997) has been directly attributed to the past unawareness of gender issues in computer science education and in the workforce. There is a risk that if gender HCI issues in end-user computing environments are ignored, a similar phenomenon could occur with female end users.

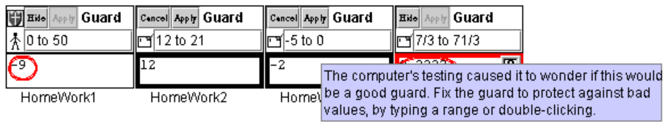
WHAT COULD GO WRONG?

What gender differences might matter in the design of end-user computing environments? Consider the following scenario in one particular end-user computing environment.

Imagine a female teacher engaged in preparing a spreadsheet to track her students' scores and to calculate ways of providing students with the best grades. Part of her process of preparing her spreadsheet is to test the spreadsheet. While she is engaged in testing, the system surprises her by decorating some of the spreadsheet cells, as in Figure 1.

The surprises were intentionally placed into the software by the designers relying on a strategy for

Figure 1. A spreadsheet calculating the average of three homework scores. Assertions about the ranges and values are shown above each cells' value. For example, on HomeWork1 there is a user-entered assertion (noted by the stick figure) of 0 to 50. The other three cells have assertions "guessed" by the Surprise-Explain-Reward strategy. Since the value in HomeWork1 is outside of the range of the assertion, a red circle notifies the user of the violation. A tooltip (lower right) shows the explanation for one of the guessed assertions.



end-user computing environments called *Surprise-Explain-Reward* (Wilson et al., 2003). The surprise, which was intended to capture the teacher's attention and arouse her curiosity, reveals the presence of an "information gap" (Lowenstein, 1994). In this case the system is using the surprise to interest her in assertions (Burnett et al., 2003), which she can use to guard against future errors by specifying, for example, that the value of a cell calculating a grade average should always fall between 0 and 100.

What could go wrong in surprising the user? According to Lowenstein's information gap theory, a user needs to have a certain level of confidence in order to reach a useful level of curiosity (Lowenstein, 1994). However, given documented gender differences in computer confidence (Busch, 1995; Huff, 2002), the teacher's level of computer confidence could interfere with the surprise's ability to capture her interest.

Returning to our scenario, suppose for this particular user, the surprise is effective at arousing her curiosity, she looks to the object that surprised her (the assertion) for an explanation. The explanation, viewed through a tooltip, includes the semantics, possible actions she can take (regarding the assertion), and the future reward(s) of taking the action. See Figure 1.

What could go wrong with the explanation? According to one theory, males and females process information differently (Meyers & Sternthal, 1991; O'Donnell & Johnson, 2001), and thus both the presentation and the content of the explanation may impact its effectiveness for males versus females. If the information needed by the user is not effectively communicated, the user's ability to problem solve is likely to be reduced.

Another role of the explanation is to help users make a reasonably accurate assessment of the risk in taking some action—but since males and females differ in their perceptions of risk (Byrnes, Miller, & Schafer, 1999), the explanation may need to serve these two populations differently in this respect as well. (An example of risk may be the fear that the user will lose their work if they try a certain feature.) If one gender perceives an explanation of a feature as communicating higher levels of risk than another, the users with higher risk perceptions may avoid supposedly "risky" features that may be important to overall effectiveness.

Perhaps the most important role of explanations is to make clear the rewards of using particular features of the software. Providing information about rewards in the explanation is consistent with the implications of the Model of Attention Investment (Blackwell, 2002), an analytic model of user problem-solving behavior that models the costs, benefits, and risks users weigh in deciding how to complete a task. An implication of this model is that if the system provides the user an idea of future benefits, users can better assess if the cost of using a feature (here assertions) is worth their time. The reward aspect of the strategy refers to rewards such as the automatic detection of errors, which is depicted by the red circle around HomeWork1's erroneous value in Figure 1.

What could go wrong with rewards? Since males and females are often motivated by different factors, there may be gender differences in what actually is a perceived "reward." If the rewards are only tailored to one gender's perceptions of rewards, the other gender may not be motivated to use the devices that will help them be effective.