Experience Factors and Tool Use in End-User Web Development

Sue E. Kase
The Pennsylvania State University, USA

Mary Beth Rosson
The Pennsylvania State University, USA

INTRODUCTION

In 1995, based on an earlier survey by the U.S. Bureau of Labor Statistics (USBLS), Boehm predicted that the number of end-users performing programming-like tasks would reach 55 million by 2005 (Boehm, Clark, Horowitz, Madachy, Selby & Westland, 1995). Adjusting this information for the accelerated rate of computer usage and other factors, Schaffidi, Shaw, and Myers (2005b) now predict the end-user population at American workplaces will increase to 90 million by 2012, and that these workers will probably execute some type of programming-like task. In a 2004 report, USBLS published projections of occupational growth patterns to 2012 and reported slightly over 3 million professionals in computer-programming occupations in 2002. To summarize, the probability is that 90 million end-users are engaged in programming-like tasks at work compared to only 3 million professionally trained programmers. Thus, the pool of end-user programmers will substantially exceed the small population who view themselves as programmers for the foreseeable future.

Programming systems employed by end-users include spreadsheets, Web authoring tools, business authoring tools, graphical languages, and scripting and programming languages (Myers, Ko & Burnett, 2006). Myers et al. (2006) estimates that 50 million people in American workplaces currently use spreadsheets or databases (and therefore may do programming). More specifically, Myers et al. (2006) estimates that over 12 million people in the workplace would say that they actually do programming at work. This diverse and growing population of end-user developers performing programming-like tasks is researched with respect to the emerging subpopulations forming around application specific activities (e.g., spreadsheets, database, Web development). Each of these subpopulations or communities of end-users has characteristic needs and abilities requiring specialized attention.

There are even more end-users participating in Internet-based tasks related to programming. During 2003, the Pew Internet and American Life Project found that more than 53 million American adults used the Internet to publish their thoughts, repond to others, post pictures, share files and otherwise contribute to the explosion of content available online. At least 13% (nearly 7 million) of those Internet users claimed they maintained their own Web sites (Pew Internet and American Life Project, 2003). We characterize this nonprofessional population as *end-user web developers*, in that they have not been trained to develop software as part of their work responsibilities, but nevertheless have found themselves developing and maintaining Web content more and more as part of their daily activities. This review targets this large and growing population, one that presents both opportunities and challenges for information systems researchers studying Web development tools, resources, and education.

BACKGROUND

Over 20 years ago, surveys of management information systems (MIS) executives, researchers, and consultants ranked end-user computing (EUC) among the 10 most important MIS issues (e.g., Brancheau & Wetherbe, 1987). Rockart and Flannery (1983) declared that EUC was booming and spreading throughout entire organizations. "Users are becoming more agressive and more knowledgable" and they "require significant managerial attention." Cotterman and Kumar (1989) attempted to understand and classify the widely differing conceptualization of the end-user into a graphical

taxonomy—the "User Cube." Davis (1985), while discussing the need for a typology of end-users, stated "In the absence of a proper classification scheme for end-users, the results of empirical investigations are likely to remain inconclusive, contradictory, and at worst, erroneous" (p. 158).

Today, the quest to understand and categorize the end-user continues. Through a survey of programming practices, Scaffidi, Shaw, and Myers (2005a) characterized end-users according to the way they represent abstractions. The use of abstraction can facilitate or impede achieving key software engineering goals (such as improving reusability and maintainability). Scaffidi et al. (2005a) believe this categorization improves the ability to highlight niches of end-users and support them with special software engineering capabilities.

In addition to typologies, a growing number of researchers and developers are defining methods to make the software produced by end-user computing more reliable (e.g., Elbaum, Karre, & Rothermel, 2003; McGill, 2002). Errors are pervasive in software created by end-users, and the resulting impact is sometimes enormous. In most cases, end-users are not striving to create the best software they can; rather, they have "real goals" to achieve: accounting, teaching, managing safety and financial data, search engine queries, or simply managing e-mail. While some software development and dependability problems have been addressed by existing methods and tools for professional programmers, such methods are usually not suitable for end-user programmers (Myers & Burnett, 2004). End-users have very different training and background than professional programmers. They face different motivations and work constraints, and are not likely to know about quality control menchanisms, formal development processes, or test adequacy criteria.

END-USER WEB DEVELOPMENT SURVEY

Empirical investigations of Web developers have received only minimal attention in past information systems research. A survey conducted by Vora (1998) is an exception. Vora queried Web developers about the methods and tools they utilized and the problems that they typically encountered. Some of the key issues developers reported back then—ensuring Web browser interoperability and usability, standards compliance of WYSIWIG editors, integrating multimedia components—are the same obsticales reported by today's end-user Web developers.

We devised an end-user Web development survey that built on an earlier qualitative study of community Webmasters (Rosson, Ballin, & Nash, 2004), as well as the Vora survey mentioned above. Guided by themes from this earlier work, we chose to take a broad-based approach to survey design and recruiting. The survey included 39 questions (10 of these were multipart) and required approximately 20-30 minutes to complete. A complete set of survey questions is at http://cscl.ist.psu.edu/public/users/mrosson/websurvey. We aimed to attract a sample population with widely varying backgrounds and development contexts. We conducted an online survey and recruited participants in two phases. We wanted to reach individuals who might not think of themselves as software developers, but nevertheless could identify with our concept of end-user Web development.

The two rounds of data collection yielded a total of 544 responses: 336 from the first round, and 208 from the second. Across the two samples, 37% of the respondents self-identified themselves as programmers (a yes/no question in the survey), and 42% of the respondents were women.

Two earlier papers reported preliminary findings from the first recruitment phase of the survey, analyzing a subset of the data and focusing on respondents' general approach to development and testing (Rosson, Ballin, & Rode, 2005), and their use of Web development tools (Rosson, Ballin, Rode, & Toward, 2005).

WEB DEVELOPMENT MEASURES

The survey was composed of six sections: Web development activities, tools, and issues; technology background; personal working style; and general background. Our exploratory analysis began by identifying several constructs that might characterize the nature of respondent's Web development practices. Because the survey consisted of a large number of items, many with multiple subitems, exploratory factor analysis was used to identify items that were intercorrelated and that had a logical interpretation as a single construct. In some cases, the final measure was a combination of several contributing subscales; in other cases, it was a more straightforward combination of responses to