Gender Differences and Programming Environments: Across Programming Populations

Margaret Burnett[1], Scott D. Fleming[1], Shamsi Iqbal[2], Gina Venolia[2], Vidya Rajaram[2], Umer Farooq[2], Valentina Grigoreanu[2], Mary Czerwinski[2]

[1]Oregon State University
Corvallis, Oregon, USA
{burnett,sdf}@eecs.oregonstate.edu

[2]Microsoft
Redmond, Washington, USA
{shamsi,ginav,vidyaraj,umfarooq,valeng,marycz}@microsoft.com

ABSTRACT

Although there has been significant research into gender regarding educational and workplace practices, there has been little investigation of gender differences pertaining to problem solving with programming tools and environments. As a result, there is little evidence as to what role gender plays in programming tools—and what little evidence there is has involved mainly novice and end-user programmers in academic studies. This paper therefore investigates how widespread such phenomena are in industrial programming situations, considering three disparate programming populations involving almost 3000 people and three different programming platforms in industry. To accomplish this, we analyzed four industry "legacy" studies from a gender perspective, triangulating results against each other and against a new fifth study, also in industry. We investigated gender differences in software feature usage and in tinkering/exploring software features. Furthermore, we examined how such differences tied to confidence. Our results showed interesting, significant gender differences in all three factors—across all populations and platforms.

Author Keywords

Gender, programming, programming tools

ACM Classification Keywords

H.1.2 [Information Systems]: User/Machine Systems—Human Factors.

INTRODUCTION

A modern concept in software design is *pluralism*—that is, the design of artifacts that "resist any single, totalizing, or universal point of view" [2]. Pluralism implies that designers can produce more inclusive designs through sensitivity to marginal or marginalized users. The concept of pluralism is in fact a central tenet of feminist HCI [2]. To inform the design of gender-pluralist software, we are investigating differences in the ways males and females perceive and use programming tools. Most prior work on gender differences in technology has emphasized practices in society and education. The possibility of gender issues within programming tools has received almost no attention.

However, a few researchers have recently begun to investigate such gender differences among populations of spreadsheet users and relatively novice programmers. Their results have shown differences in the features males and females used when performing programming tasks [4, 5, 6, 16, 24, 25]. For example, laboratory studies of spreadsheet debugging showed gender differences in feature usage, feature-related confidence, and tinkering (playful exploration) with features [4, 5, 6]. However, these academic studies involved populations with little programming experience. In this paper, we investigate whether these findings generalize—to industrial programming tools used across a wide spectrum of programming populations.

Investigating gender differences with programming tools matters for a number of reasons. First, if males and females work differently with programming tools, such differences would reveal a *need* to change programming environments to take this new understanding into account. Second, if such differences do exist, there is a solid scientific base that can tell us *how* to make such changes: extensive theoretical ammunition exists (discussed in the next section) that can provide a theory-driven foundation upon which to base design decisions and improvements. Finally, making such changes is likely to help *both* genders. Indeed, research has shown that studying the needs of one subpopulation can provide benefits that extend beyond that subpopulation [18]. For example, phenomena that affect large numbers of females may affect some males as well.

However, to investigate the generality of gender differences with programming tools across populations requires data about multiple populations and tools—a potentially prohibitively expensive undertaking. To address this problem, we leveraged data from multiple studies that had been conducted on industry-based programming populations ranging from end-user programmers, such as technical administrators, to professional software developers. We analyzed data from four of these studies, then ran a new study as a validity check. Finally, we triangulated across all five studies to

ESEM'10, September 16–17, 2010, Bolzano-Bozen, Italy.

further ensure validity. This paper presents the results, covering nearly 3000 participants. Conducting an industry-based investigation in this fashion allowed broad coverage of thousands—not tens—of people who program.

THEORETICAL BACKGROUND

We have pointed out the recent research in the realms of end-user programming and novice programming reporting gender differences (e.g., [4, 5, 16, 24, 25]). Underlying connections among these studies become more apparent when considered in the context of applicable theories, several of which help to explain such differences.

For example, gender differences related to problem solving approaches have been reported in psychology, marketing, and mathematics [3]. These differences have been linked to factors that, although individual in nature, tend to statistically cluster by gender. These factors include confidence, problem-solving style, and information-processing style.

One specific form of *confidence* is *self-efficacy*: a person's confidence about succeeding given a specific task [1]. Self-efficacy is important in problem solving because it influences the use of cognitive strategies, amount of effort put forth, level of persistence, and strategies for coping with obstacles. Findings in the domain of computing report that lower self-efficacy impacts attitudes toward a new software package prior to its use [14], that females have lower self-efficacy than males in their ability to succeed at tasks in word processing, spreadsheets, file manipulation, and software management tasks [8, 30], and that females have low self-confidence in their ability to succeed at technical problem-solving and programming [19, 20]. In the area of end-user debugging of spreadsheets, females displayed unwarrantedly low levels of self-efficacy [4]. This played out in their debugging behavior: self-efficacy predicted females' willingness to try unfamiliar debugging features and females' effectiveness in adopting such features [4].

Many gender differences have also been reported in *problem-solving strategies*—findings that seem highly pertinent to how males and females work with programming tools. For example, gender differences have been found in males' and females' preferred spreadsheet debugging strategies, and the debugging strategies with which males and females were most effective [28]; in spatial way-finding strategies [17]; in financial decision-making strategies [23]; and in mathematical problem-solving strategies [11, 12].

To solve problems, people often need to process new information. Gender differences in *information processing style* have been studied extensively in the field of marketing. In particular, the Selectivity Hypothesis [21] predicts that females are more likely to use comprehensive information processing, attending to details and how they fit together in the overall picture. In contrast, the theory predicts that males will follow up on information selectively, focusing on one related cue found, and pursuing it in depth before trying to attend to the other details. Empirical studies support the theory (e.g., [21]); particularly relevant are those tying gender differences in information processing style to software-based tasks, such as with e-commerce web sites [27] and software-based auditing [22].

These theories suggest that gender differences are likely in technical problem solving *approaches* in programming settings. Such differences in turn suggest gender differences with programming tools' *features*.

METHODOLOGY

This theoretical basis suggests that the previous academic findings about novice and end-user programmers may generalize. But what is an economically feasible approach for generalizing across multiple populations in industry?

To solve the problem of economic feasibility, we gathered data from over 20 studies previously conducted by a large software company. Our methodology for secondary analysis of these data combined (1) principled selection of data (as in case study methodology), (2) from only selected portions of multiple independent studies that used different instruments to collect the data (as in meta-analysis methodology), with (3) statistical techniques to account for potential covariance (as per statistical experiment methodology) and (4) a fresh study and triangulation across all studies for validity (as in mixed-method study methodology).

Principled Study Selection

The company provided access to over 20 previous studies. These candidates used multiple designs (some qualitative, some quantitative) and involved participants from multiple programming populations. As in case study methodology [32], we selected four of these studies using principles that followed naturally from our research questions:

RQ1: Are there gender differences *across* programming environments and populations as to *which features* males and females use?

RQ2: Are there gender differences *across* programming environments and populations as to willingness to *tinker and explore*?

RQ3: Are any such differences related to males' or females' technical problem-solving *confidence*?

These research questions served as the foundational principles we used to select studies. Thus, to be selected, a study must have collected gender data and also collected feature, tinkering, or confidence data. Additional principles were that a study needed to contribute populations or platforms beyond those previously collected (for generalization), needed to have been designed by empirical researchers or professionals (for validity), and, in the case of statistical studies, needed to collect sufficient gender data for statistical analysis (for viability). All selected studies had been completed within the 18 months prior to our investigation, which began in spring 2009. Table 1 summarizes the four studies that we selected according to these principles (#1–#4) as well as the new study we devised and conducted for validity (#5). In the discussion section, we also describe four studies that we did not select.

Principled Data Selection

Note that meta-analysis in the statistical sense was not a possibility, because there were no analyses to "meta"—the original studies did not analyze gender differences. However, we borrowed from meta-analysis methodology the method of selecting only the portions of these studies that related to our research questions. Specifically, we used the following portions of each study: background data on participants' gender, technical background, job, experience, and age; and all data related to feature preference and usage (RQ1), tinkering or exploring technology (RQ2), and confidence in technical problem-solving tasks (RQ3). To reduce the risk of bias, we used only the study instruments—not the results—to decide which portions of the study met these criteria. Furthermore, we included *any and all* data that met our above criteria.

Analysis Methodology

Using previous studies that were not targeted at gender differences introduced statistical challenges. Therefore, our statistical procedure was to first identify potential confounds in background data by testing for significant differences in males' versus females' background attributes of age, experience levels, and the technical degree of job titles. Age did not turn out to be statistically significant in any of the studies, but in several the females were significantly less experienced or had significantly less technical job titles than males. We therefore used ANCOVA to factor in the effects of these covarying factors.

To handle the unequal sample sizes for males and females, we rank transformed the data before the ANCOVA analyses. Rank transformations add robustness to non-normality and resistance to outliers and unequal variance to ANOVA and ANCOVA [10]. These statistical methods amount to nonparametric tests, a common approach when parametric techniques are not robust to unequal sample sizes [26]. Some statistical researchers advise more sophisticated techniques [26], so as a double-check, we verified main results using unequal variance testing (Levene test) followed by *t*-tests-given-unequal-variance when the Levene test so indicated, as we explain in the results sections.

Central to our methodology's validity is triangulation: whether the same results manifest themselves multiple times from multiple sources of evidence, including with the triangulation a fresh study as a further validity check. Our within-study triangulation is presented with each study, and between-study triangulation presented in a separate section.

Study	Type	Programming Population	Females	Males	Total
#1	Survey	IT-Support Users	32	72	104
#2	Survey	Hobbyists	112	2367	2479
#3	Field Interviews	Hobbyists	2	4	6
#4	Survey	Professional Developers	23	134	157
#5	Survey	Beta-Testing Professional Developers	96	144	240
		Total:	265	2726	2991

Table 1: The populations and studies.

STUDY #1: USERS OF THE IT-SUPPORT SITE

Study #1 emphasized technical problem-solving. It was an in-house survey of company employees who were users of a particular web-based IT-support site. Conducted by a product group to understand the needs of their target audience, the study gathered data about users' technical problem-solving habits, attitudes toward technology, and specific techniques for technical problem solving. The survey, primarily consisting of Likert-style questions (5-point), proved ideal for addressing all three research questions, providing data regarding respondents' feature preferences (RQ1), willingness to tinker (RQ2), and technical problem-solving confidence (RQ3).

104 people responded to the survey (72 male, 32 female). In analyzing the background data, we identified three potential confounds with gender: technical level of job, years of professional experience, and age. Participants' jobs ranged from administrative users of technology ("Admin") to user experience professionals ("UX") to software project managers ("PM") to professional software developers ("Dev"). To measure technical level of job, we coded each participant's job title on an ordinal scale from 1 (least technical) to 4 (most technical). To validate our coding scheme, two researchers independently coded one-third of the data set. Their agreement rate exceeded 96%, differing on only 1 of the participants, so a single researcher finished the coding.

As Table 2 suggests, the study had potential confounds: a higher percentage of the females held less technical jobs (one-way ANOVA: $F(1,102)=14.586$, $p<0.001$) and had fewer years of experience (one-way ANOVA: $F(1,102)=6.027$, $p=0.016$). Age, however, did not differ significantly. Thus, we use one-way ANCOVAs, with gender as the fixed factor and job and experience as the covariates, in the remainder of this section.

IT-Support Users RQ1 and RQ2: Features and Tinkering

Two questions on the survey related to feature preferences (RQ1). One question asked participants to check off from a list all the applications installed on their work computer. One way to view an application is as a set of features, so we used that question as one indication of participants' interest in employing a variety of features. We counted up the number of applications participants had installed, then binned these into three groups: 1–4, 5–8, 9–12 applications installed. (No participant indicated more than 12.) Even ac-

	Technical Level*	Years of Experience	Age
Females (N=32)	1.75 (.984) Median: 1	6.88 (5.912)	36.00 (10.084)
Males (N=72)	2.62 (1.092) Median: 3	9.92 (5.774)	33.75 (8.717)

* Ordinal scale from 1 (least technical) to 4 (most technical)

Table 2: Means (SDs) of IT-Support participant backgrounds in Study #1. Medians are also shown for categorical data.

counting for differences in experience and job title, the results indicated a significant difference between genders. Table 3's top row summarizes the analysis results, and Fig. 1a shows the distribution. Thus, accounting for differences in job and experience, the females used fewer applications (feature sets) than males did.

The other feature-related question asked participants about wizards. The results showed that females favored wizards significantly more than males. The analysis is given in Table 3, and Fig. 1b shows the distribution. As with the other results presented here, this gender difference was not due to job or experience. For example, Fig. 2 shows its persistence across job titles.

The survey included questions about willingness to tinker and explore (RQ2), and ANCOVA analyses showed significant gender differences on *all* such questions. For example, females agreed significantly more with feature-conservative questions (e.g., "I only learn the technology I have to know to perform my duties") and significantly less with questions about enjoying feature exploration (e.g., "I enjoy piloting/dogfooding next-generation technology"). Table 3 shows the detailed statistical results. Figs. 1c and 1d show two of the distributions. Experience was also significant for one question, "prefer established technology" ($F(1,98)=4.06$, $p=0.047$), but neither job nor experience were significant for any of the other questions.

In summary, the females in this survey used significantly fewer features than the males and were significantly more enthusiastic about wizard features. Females also had less interest in tinkering and exploring new features than males did. Although experience was significant in 1 of the 7 outcomes, gender was significant in all 7.

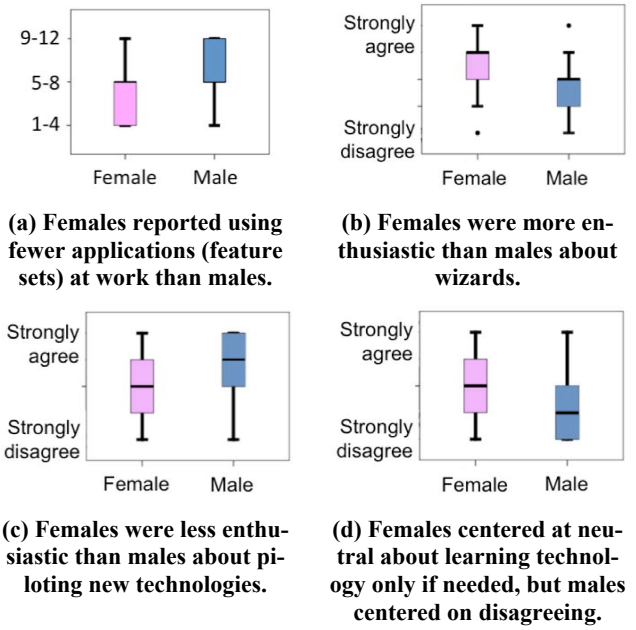
IT-Support Users RQ3: Confidence

The survey asked two Likert questions relating to confidence: whether respondents perceived themselves to be experts, and whether they thought others perceived them as experts. Taking covariates into account, one-way ANCOVAs with gender as the fixed factor and job and experience as covariates showed significant differences for gender. (Self perception: $M_{male}=3.13$, $M_{female}=1.97$; $F(1,97)=15.326$, $p<0.0010$. Others' perception: $M_{male}=3.31$, $M_{female}=2.25$; $F(1,97)=9.265$, $p=0.003$.) Not surprisingly, differences were also significant by job, with less technical job holders' confidence lower (self perception: $F(1,97)=4.829$, $p=0.03$; others' perception: $F(1,97)=12.983$, $p<0.001$).

We then correlated males' and females' confidence responses with their expressed interest in features and tinkering (Table 4). In some cases, confidence alone seemed at least partially tied to attitudes toward feature usage and tinkering for both males and females. For example, participants' perceptions of their own expertise were significantly correlated with their propensity to explore technology, to avoid learning technology not needed for their duties, and to pilot next-generation technology.

Question (RQ)	Male		Female		*df*	Gender	
	M	*SD*	*M*	*SD*		*F*	*p*
Apps installed (1)	7.32	2.58	6.31	2.25	1,99	6.30	0.014
Prefer wizards (1)	2.69	0.96	3.53	1.05	1,99	9.33	0.003
Exploring technology (2)	4.13	0.93	3.44	1.13	1,99	7.03	0.009
Only learn required tech (2)	1.99	1.17	2.75	1.39	1,99	5.90	0.017
Piloting new tech (2)	3.94	1.11	3.00	1.37	1,99	7.37	0.008
Using workarounds (2)	4.14	0.97	3.65	1.08	1,98	4.74	0.032
Prefer established tech (2)	3.23	0.91	3.78	1.04	1,98	6.63	0.012

Table 3: IT-Support users (Study #1) ANCOVAs for RQ1 and RQ2. Gender was the independent variable, technical level of job and experience were covariates. Significant differences are highlighted with darker backgrounds.



(a) Females reported using fewer applications (feature sets) at work than males.

(b) Females were more enthusiastic than males about wizards.

(c) Females were less enthusiastic than males about piloting new technologies.

(d) Females centered at neutral about learning technology only if needed, but males centered on disagreeing.

Figure 1: Male and female IT-Support participants' attitudes toward features and tinkering. Females: light, males: dark. *Boxplot formats*: The thick middle lines represent medians. The boxes contain half of the responses. The whiskers show the range of the rest, except for outliers, which are shown as dots and stars beyond the whiskers.

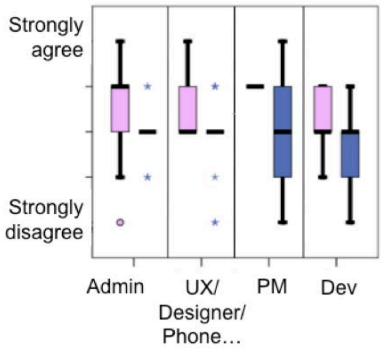


Figure 2: The gender-pairs of boxplots by job show that female IT-Support participants in every job type had greater interest in wizards than their male peers in the same job type.

Feature/Tinkering Question	Confidence			
	Male		Female	
	r_s	p	r_s	p
Prefer wizards	0.389	0.001		0.227
Exploring tech.	0.352	0.002	0.355	0.046
Only use required tech.	-0.268	0.023	-0.388	0.028
Piloting new tech.	0.501	<0.001	0.584	<0.001
Using workarounds	0.302	0.010		0.143
Prefer proven tech.		0.444		0.444

Table 4: Correlations tested with the non-parametric Spearman's rho test from the IT-Support study. (r_s values shown for significant results only.)

Interestingly, some of the outcomes that one might expect to be due to confidence were not tied to confidence. For example, Spearman's rho showed no significant relationship, for either males or females, between confidence and preference for established technology. Likewise, it showed no correlation between confidence and females' propensity to devise workarounds for problem solving. Most strikingly, interest in wizards, although related to confidence for males, was not significantly confidence-related for females.

In summary, the females were less confident than the males, and some feature/tinkering preferences did seem tied to this confidence difference. However, confidence was not implicated in some of their preferences.

STUDIES #2 AND #3: HOBBYIST DEVELOPERS

Two of our selected studies focused on *hobbyist developers*, who program in Visual Studio Express. We first present the hobbyist survey, followed by the field-interview study.

Hobbyist Users of Visual Studio Express

Visual Studio Express publicity targets "first-time or casual" users. The company surveyed 2517 such Visual Express users outside the company (112 female, 38 did not reveal their gender), who responded to questions about Visual Studio Express. In this study, we did not include covariates in our statistical model because a one-way ANOVA revealed no significant gender difference in any of the relevant background factors.

Although the survey did not relate to RQ2 or RQ3, the survey elicited a feature "wish list" directly pertinent to RQ1. That question asked respondents to choose their top three wish list picks from a set of 13 possible choices. We analyzed the eight features that were selected by at least 20% of males or 20% of females. We partitioned these eight features into three categories: wizards (1 feature), expertise-related (4 features: starter kits, beginner, intermediate, advanced), and task-related (3 features: controls and APIs, web development, Windows development).

Table 5 summarizes one-way ANOVA results for each feature, grouped by category. The feature's presence in the top three served as dependent variable, and gender served as independent variable.

As Fig. 3 and Table 5 show, similar to the IT-support females, the hobbyist females wanted wizards, choosing the feature significantly more than the males (p=0.003).

Females also favored beginner-level features more than males did. In fact, 45% of the females picked beginner-level, 21% picked intermediate features, and only 13% picked advanced-level features. As Table 5 shows, significantly more females than males ranked beginner-level features in the top three (p<0.001), and significantly fewer females than males ranked advanced-level features in the top three (p=0.008). Both genders expressed interest in starter kits (48% of females and 39% of males), but females' interest bordered on being significantly stronger (p=0.054).

Table 5 also shows that males wanted features for different programming tasks than females. 30% of males wanted Windows-development features as opposed to 19% of females—a significant difference (p=0.022). Also, 34% of males picked controls and programming APIs as opposed to 17% of females—again, statistically significant (p=0.001).

Recall, these gender differences were between male and female programming hobbyists—a population that seems by definition to be interested in exploring new technologies. The fact that this population of self-identified programming enthusiasts still showed significant gender differences in

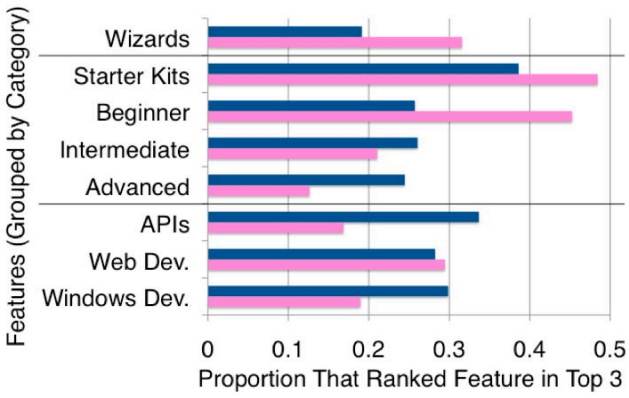


Figure 3: Each bar shows the proportion of Study #2 hobbyist participants who ranked the associated feature in their top three picks. (Females: light, males: dark).

Feature Type (RQ1)	Proportion		F	df	p
	Male	Female			
Wizards	0.19	0.32	8.98	1, 2238	0.003
Starter kits	0.39	0.48	3.69	1, 2238	0.054
Beginner	0.26	0.45	17.87	1, 2238	<0.001
Intermediate	0.26	0.21	1.19	1, 2238	0.275
Advanced	0.24	0.13	7.02	1, 2238	0.008
APIs	0.34	0.17	11.70	1, 2238	0.001
Web dev.	0.28	0.29	0.07	1, 2238	0.796
Windows dev.	0.30	0.19	5.21	1, 2238	0.022

Table 5: Study #2 ANOVA analysis of differences in male and female hobbyists' feature picks.

their feature picks makes a particularly strong statement.

Hobbyist Field Interviews

In tandem with the survey, a company researcher interviewed six hobbyist developers. Three were beginners (1 female), and three were intermediate-level programmers (1 female). None were professional programmers. We qualitatively analyzed the interview transcripts as well as the original analyst's notes. The interviews related to two of our research questions, RQ2 and RQ3.

With regard to tinkering and exploring (RQ2), the interviews with the males were remarkably consistent with the other studies' statistical results. All four males, regardless of expertise, described themselves as geeks, tech enthusiasts, or tinkerers. Even Lew (a fictional name), who wanted "technology to be a tool, not an end in itself" was a tinkerer: "I don't understand every single knob, but I have a good idea what it would do… You learn so much more by screwing up." In contrast, even among these hobbyists, the females were divided about tinkering. Alison (the female intermediate) was a tinkerer, but Lisa, the female beginner was not eager to tinker or explore. Her reasoning came down to cost/benefit: "once you get used to one product, you don't really want to go learn a whole new product."

Regarding RQ3, both females mentioned low confidence. Lisa did so repeatedly. Even Alison, who expressed keen interest in programming ("you have to care about the code"), described Visual Studio as "really intimidating…like fifty million things you can click on the first page." In contrast, none of the males hinted at any lack of confidence in their technical abilities.

STUDY #4: PROFESSIONAL DEVELOPERS

A survey (mostly 7-point Likert-style questions) of professional developers' work habits and their programming tools was relevant to RQ1 (feature usage). The 157 respondents (23 female) were all employees of the company.

Males and females reported no significant difference in their time allocations to various programming activities (e.g., writing vs. understanding vs. testing code). However, the females had significantly fewer years programming experience (one-way ANOVA: $F(1,155)=15.693$, $p<0.001$). The females were also somewhat younger than the males (one-way ANOVA: $F(1,155)=3.13$, $p=0.071$). Because age was marginal, we performed two separate analyses: one that took both experience and age into account as covariates, and another with only experience as a covariate. The age (marginal) covariate neither contributed to any of the models' fits nor significantly impacted the results; thus, our analyses used experience as the sole covariate.

The survey contained four questions about which programming environment the developers had been using over the previous week for four types of development tasks: writing new code, understanding existing code, editing existing code, and unit testing code. There were also corresponding questions about the most effective programming

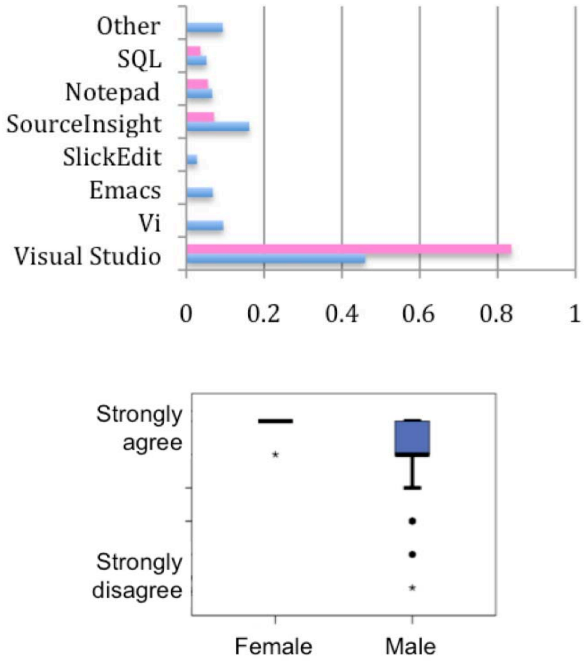


Figure 4. Top: In Study #4, female professionals allocated most of their time to one environment when editing existing code, whereas males spread their attention over several. Bottom: Females had significantly higher ratings of the effectiveness of Visual Studio when writing new code. (Graphs of the other questions are almost identical to these.)

Question (RQ)	Male		Female		*df*	Gender	
	M	*SD*	*M*	*SD*		*F*	*p*
Writing new code: % time spent using VS (1)	0.46	0.46	0.78	0.36	1,119	5.709	0.018
Writing new code: VS is effective (1)	2.01	1.36	1.17	0.39	1,89	6.915	0.010
Understanding new code: VS is effective (1)	2.22	1.31	1.60	0.63	1,99	1.711	0.194
Editing existing code: % time spent using VS (1)	0.46	0.46	0.84	0.30	1,125	8.291	0.005
Editing existing code: VS is effective (1)	2.01	1.17	1.23	0.44	1,92	7.204	0.009
Unit testing: VS debugger vs. other products (1)	Count: VS 228 Other 44		Count: VS 28 Other 1		N/A	N/A	0.096 (two-tailed) 0.048 (one-tailed)

Table 6: Results from Study #4 of ANCOVA (top rows) and Fisher's Exact Test (bottom row) for survey questions pertaining to RQ1 regarding attitudes toward company's usual environment (Visual Studio) versus other products.

environment for these tasks.

As in the other studies, the analysis used one-way ANCOVAs on rank-transformed data to account for unequal samples and variances, with gender as the independent variable and experience as covariate. However, for unit testing, in which only 1 female used other products, the analysis required Fisher's exact test, which is suitable for such small sample sizes. The results showed that females used Visual Studio (a "standard" environment in that company's practices) for a greater proportion of these tasks than males. In addition, females rated that programming environment higher in terms of its effectiveness for these tasks. Table 6 shows the statistics for each of these questions, and Fig. 4 (bottom) graphs a sample of these results.

The result that females in this study, who were engaged in the same sorts of tasks as the males, liked and used Visual Studio significantly more than the males suggests that the females had different feature usage patterns and attitudes toward features. However, this survey addressed RQ1 only. To gather more data about professional developers, we developed a new survey, which we describe next.

STUDY #5: PROFESSIONAL DEVELOPERS BETA-TESTING VISUAL STUDIO

To expand coverage of professional developers and to test the generality of our findings with a study designed especially for that purpose, we conducted a new survey of professional developers, with Likert-style questions (5-point) on all of RQ1, RQ2, and RQ3: feature preferences, willingness to tinker, and confidence.

We recruited survey participants from a list of developers employed by the company who were pilot-testing a beta version of Visual Studio. To encourage a balance of male and females in our sample, we estimated candidate participants' genders with the Genderyzer tool [15], which probabilistically derives gender from first names. We then recruited the 444 participants estimated to be females and also a randomly selected equal number of male pilot testers. As an incentive, we offered participants entries in a drawing for a $500 gift certificate.

242 people responded to the survey (96 female, 2 declined to state and were discarded). Among the respondents, 52 were program managers (23 female), 83 were software-development engineers (18 female), 76 were software-test engineers (36 female), and 29 fell into the "other" category (19 female). We coded the technical level of each participant's job using the same technique as in the IT-Support study, ranking jobs from 1 (least technical) to 4 (most technical). As in the other studies, participants explicitly stated their gender in the survey.

Professionals RQ1 and RQ2: Features and Tinkering

To test for differences between males and females in feature choice (RQ1) and tinkering (RQ2), we ran a one-way ANCOVA for each relevant question. The answer to a question served as the dependent variable, gender served as the independent variable, and job and experience were covariates. Table 7 summarizes the results of our analyses.

Regarding RQ1, in contrast to the other studies, males and females showed no significant difference in their feelings about wizards. The raw data trended toward females rating wizards higher than the males (Table 7), but the difference did not reach statistical significance.

Regarding RQ2, the females were less interested in exploring and piloting new technology than males—a somewhat surprising result (but not given our previous findings) given that the participants were all piloting new software, an activity that inherently entails some degree of exploration. Females also did not like to learn technology not required for their jobs or to upgrade software frequently. Table 7 shows the details. Thus, males in this population clearly preferred tinkering and exploring more than their female counterparts.

Professionals Who Beta-Test RQ3: Confidence

We used two independent measures of technical problem-solving confidence to check whether confidence related to feature preferences or willingness to tinker (RQ3). The first measure generally aimed at confidence in technical expertise (*general confidence* for short). To measure this form of confidence, we asked each participant if she considered herself a tech expert and if others perceived her as a tech expert. Participants provided answers on a 5-point Likert scale. We summed the two answers into a composite confi-

Question (RQ)	Male		Female		*df*	Gender	
	M	*SD*	*M*	*SD*		*F*	*p*
Prefer wizards (1)	2.93	1.06	3.15	0.92	1,231	2.29	0.13
Exploring tech (2)	4.13	0.82	3.49	0.83	1,231	35.83	<0.001
Only learn required (2)	2.13	1.01	2.54	0.91	1,231	12.84	<0.001
Piloting new tech (2)	3.97	0.93	3.42	0.99	1,231	18.65	<0.001
Using workarounds (2)	4.40	0.59	4.19	0.49	1,231	10.11	0.002
Upgrading software (2)	4.04	0.90	3.69	0.94	1,231	9.18	0.003
Prefer established (2)	3.24	0.86	3.31	0.82	1,231	0.53	0.47

Table 7: Study #5 (beta-testing professionals) results of ANCOVA for survey questions pertaining to RQ1 and RQ2.

Question	Confidence				Self-Efficacy			
	Male		Female		Male		Female	
	r	*p*	*r*	*p*	*r*	*p*	*r*	*p*
Prefer wizards		0.19	0.17	0.09		0.79		0.21
Exploring tech	0.41	<0.01		0.14	0.20	0.03		0.51
Only learn required tech	-0.40	<0.01	-0.22	0.02	-0.24	<0.01		0.84
Piloting new tech	0.20	0.03	0.20	0.04		0.24		0.89
Using workarounds	0.31	<0.01	0.28	0.008	0.30	<0.01		0.36
Upgrading software	0.24	<0.01		0.58		0.18		0.98
Prefer proven tech		0.56		0.87		0.14	0.22	0.04

Table 8: Study #5's RQ3 correlation coefficients for general confidence and self-efficacy in using a new programming tool.

dence score (possible score: min=2, max=10).

The second measure of confidence was self-efficacy in using a new programming tool to meet a software-development deadline (*self-efficacy* for short). To measure self-efficacy, we used a slightly modified version of Compeau and Higgins' validated scale [9]; the modifications made the questions task-specific to using a new programming tool to meet a deadline. We then computed a self-efficacy score by summing the participant's Likert answers to the ten questions (min. possible score = 10, max. possible = 50).

Regarding gender differences in confidence, a one-way ANCOVA with confidence score as dependent variable, gender as independent variable, and technical level of job and experience as covariates revealed that females had significantly lower general confidence than males ($F(1,226)=72.38$, $p<0.001$), a result consistent the other studies' findings. In this case, the males' mean general confidence score was 8.18, whereas the females' mean was 6.42. The ANCOVA did not, however, reveal a significant difference in self-efficacy ($F(1,225)=2.21$, $p=0.14$); the male mean self-efficacy score was 38.94, and the female mean was 38.01.

To investigate whether either form of confidence explained the differences in attitudes toward tinkering and exploring, we performed linear regression modeling with confidence scores and feature/tinkering answers as independent variable. We analyzed each gender individually. Table 8 summarizes the results.

As the results show, for this population, confidence and/or self-efficacy were significant factors that may help to explain several of the gender differences in tinkering and exploring. Even so, not all of the differences could be attributed to either confidence factor and in many cases the correlation amount (r) was fairly low. Furthermore, the results show several feature/tinkering preferences that were unrelated to any form of confidence (e.g., preference for wizards). Thus, although confidence was implicated, it falls short of explaining all of the differences. This is consistent with the other studies in this paper, as well as with earlier findings on spreadsheet users, in which males' and females' self-efficacy predicted their tinkering and exploring behaviors differently [5].

RESULTS TRIANGULATION

Table 9 shows the triangulation of results across the five studies. Each research question occupies a major row of the table. The subrows report particularly interesting specializations of a research question. For example, the studies reported numerous instances of differences in feature usage (RQ1 major row), but a particularly interesting finding within that category was wizards (subrow under RQ1).

In the table, each "√" denotes a study that provided evidence relevant to a particular research question (row) and a particular programming population (column). If more than one study was run on a population, a number in parentheses clarifies the study that produced that evidence. The "√?" notation for the hobbyists' confidence reflects a design issue in that study as it pertains to our research question. The question asked participants to rate themselves as absolute beginners, beginners, intermediates, or experts, which could either reflect their number of years experience or their confidence in their own capabilities. Therefore, we cannot count on its ability to isolate confidence.

As the table shows, evidence from multiple populations and platforms pointed to the same results for all three research questions.

DISCUSSION

Changing the Tools: Disadvantaging to Males?

Our findings underscore the importance of taking gender differences into account when designing programming tools. But such changes need not trade off one gender against the other: researchers have shown that taking gender differences into account in designing software features can benefit *both* genders. For example, Tan et al. showed that displaying optical flow cues benefited both females and males in virtual world navigation [29], and Grigoreanu et al. showed changes to spreadsheet features relating to confidence and feature support that reduced gender gaps while improving both genders' attitudes and feature usage [13]. These findings are consistent with similar findings in changing educational practices to take gender differences into account. For example, in education, researchers found that pair programming, which was expected to help female computer science students, not only reduced the gender gap but also increased success and reduced attrition among both male and female students [7, 20].

	IT-support users: (#1) survey	Hobbyists: (#2) survey, (#3) field interview	Professional developers:(#4) needs survey, (#5) beta-testers survey
RQ1: Which features.	√ (#1)	√ (#2)	√ (#4)
Interesting example: Wizards.	√ (#1)	√ (#2)	
RQ2: Tinkering, exploring.	√ (#1)	√ (#3)	√ (#5)
One aspect: Attitude re: new technology.	√ (#1)	√ (#3)	√ (#5)
RQ3: Confidence differences.	√ (#1)	√? (#2), √ (#3)	√ (#5)
One aspect: Evidence of ties with only *some* differences.	√ (#1)	√ (#2)	√ (#5)

Table 9: Triangulation of results, with (#study) denoting which study produced each result. √ denotes statistically significant differences, except where (#study) refers to a qualitative analysis. (See text regarding the "√?" for the hobbyists' confidence results.)

Four Studies Not Included

In our investigation, we looked closely at four additional studies that we ultimately did not include, for reasons we explain here.

Two of these studies focused on team-oriented behavior. We rejected them because they did not address our research questions, but we mention them here because they raise an open question. One study investigated developers' information needs in team projects, and the other investigated team practices, problems, and norms. It should be noted that these studies' data did not appear to contain gender differences. The open question is whether, in matters of team behaviors, individual preferences may be suppressed in the interest of team cooperation.

The other two rejected studies were log studies that seemed pertinent to RQ1. The data from these studies comprised logs of the actions users took while working with programming environments. One problem with the log files was the pervasiveness of trivially obvious features, or features offering little alternative than to use them, neither of which were useful to our research questions. The other problem was the sparseness of log data. Even logs with hundreds of users, when distributed across complex feature sets containing hundreds of feature choices, led to very sparse data matrices with too few points in any applicable feature type for statistical power. In one log file we did find a small but statistically significant gender difference confirming the wizard finding for professional developers, but the finding was too isolated to be trustworthy, so we chose not to include it.

Reflections on the Methodology

Our methodology was aimed at leveraging use of a company's store of industrial data collected in other studies as an economically feasible approach to external validity, which provides more capability for generalization than is possible in single studies. Because its results accomplished these goals, we offer reflections to other researchers who may wish to use the same methodology.

First, we caution that using this methodology needs to emphasize conservative decision-making, and that using just part of the methodology may sacrifice the validity of the results. For example, principled selection of studies, and selection of study portions rather than result portions, are a critical aspect, and together distinguish this methodology from "cherry picking" of results. Likewise, it depends on careful use of statistical techniques to account for covariance, and triangulation to validate the results.

Second, we noticed a side-benefit from the methodology: enhanced interest and awareness in the software company regarding the possibility of gender differences relating to software features. Our hope is that this work will advance both scientific knowledge *and* awareness inside the industrial software teams who actually build products.

THREATS TO VALIDITY

No empirical study is perfect. One reason is the inherent trade-off among different types of validity [31]. In this section, we describe how our study balanced three types of validity: *external*, *internal*, and *conclusion* [31].

External validity refers to the ability to generalize the findings of a study. The primary goal of this study was to fill the external validity gap of prior studies and to generalize across a range of programming populations in industry. We addressed this goal while minimizing the risk of introducing new threats to external validity by analyzing multiple industrial studies spanning a large and diverse set of samples. Even so, our use of employee and customer data from only one, albeit large, software company may limit our ability to generalize the results to programmers outside these groups.

Internal validity refers to causality between independent and dependent variables. An unmeasured or uncontrolled variable threatens internal validity if it influences the dependent variable. In leveraging existing industrial study data to increase external validity, we sacrificed control over the studies' designs. Thus, one threat was that others implemented the original studies. Our safeguards were requiring that the original researchers had to be empirical professionals, and confirming all results via triangulation. Another threat was that the backgrounds of the female participants sometimes differed from those of the males. As discussed earlier, we addressed this threat by including background covariates in our ANCOVA analyses. A final threat to internal validity arose from use of similar questions with different wordings to measure the same theoretical constructs, a threat we addressed conservatively by using triangulation *instead of* statistical aggregation.

Conclusion validity is concerned with whether a statistically significant relationship exists between treatment and outcome. Violated assumptions of statistical tests commonly threaten conclusion validity. For many statistical tests, our low proportion of females would seem a problem. However, we used an appropriate test for such data and performed supplementary checks for agreement using additional tests (discussed in the Methodology section). Moreover, the statistical studies' raw counts of females were reasonable, ranging from 23 to 96, with 265 in total, a huge number given females' low presence in programming populations [19]. Finally, we used triangulation, which emphasizes evaluating agreement among results to reduce the risk that a relationship occurred by chance.

CONCLUSION

This paper presents the first investigation of its kind to research gender differences. Our analysis was based on data from almost 3000 participants from a variety of programming *populations* and *real-world platforms* in *industry*. Triangulating the five studies' results showed:

RQ1: There were significant gender differences across programming environments and populations as to *which features* males and females elected to use.

RQ2: There were significant gender differences across programming environments and populations as to males'

and females' willingness to *tinker and explore*.

RQ3: Although there were significant differences between males' or females' technical problem-solving *confidence*, these differences clearly were not the sole factor in the differences in feature usage and tinkering.

Note that these gender differences do not suggest that males are somehow "better" software users than females. For example, using more features is not always better than using fewer, and tinkering/exploring is not always productive. Furthermore, although gender differences in confidence, attitudes, problem-solving styles, and information processing styles are all implicated in our results, no single female has every trait statistically associated with females, nor does any male have every trait statistically associated with males. As the discussion of recent work shows, informing the design of programming tools based on the differences revealed by our investigation need not penalize either gender—doing so can help everyone.

ACKNOWLEDGMENTS

We thank Jofish Kaye for extending his Genderyzer tool for our use. This work was supported in part by the EUSES Consortium under NSF 0325273 and by NSF 0917366.

REFERENCES

1. Bandura, A. *Social Foundations of Thought and Action*. Prentice Hall, Englewood Cliffs, NJ, USA, 1986.
2. Bardzell, S. Feminist HCI: Taking stock and outlining an agenda for design, *In Proc. CHI 2010*, ACM (2010), 1301–1310.
3. Beckwith, L. and Burnett, M., Gender: An important factor in end-user programming environments? In *Proc. VLHCC 2004*, IEEE (2004), 107–114.
4. Beckwith, L., Burnett, M., Wiedenbeck, S., Cook, C., Sorte, S., and Hastings, M. Effectiveness of end-user debugging software features: Are there gender issues? In *Proc. CHI 2005,* ACM (2005). 869–878.
5. Beckwith, L., Kissinger, C., Burnett, M., Wiedenbeck, S., Lawrance, J., Blackwell, A., and Cook, C., Tinkering and gender in end-user programmers' debugging, In *Proc. CHI 2006,* ACM (2006), 231–240.
6. Beckwith, L., Inman, D., Rector, K., and Burnett, M. On to the real world: Gender and self-efficacy in Excel, In *Proc. VL/HCC 2007*, IEEE (2007), 119–126.
7. Berenson, S., Slaten, K., Williams, L., and Ho, C.-W. Voices of women in a software engineering course: Reflections on collaboration. *J. Educ. Resour. Comput. 4*, 1 (2004).
8. Busch, T. Gender differences in self-efficacy and attitudes toward computers. *J. Educ. Comput. Research 12*, (1995).
9. Compeau, D., Higgins, C. Computer self-efficacy: Development of a measure and initial test. *MIS Quarterly 19*, 2 (1995).
10. Conover, W. J. and Iman, R. L. Rank transformations as a bridge between parametric and nonparametric statistics. *The American Statistician 35*, 3 (1981), 124–129.
11. Fennema, E., Carpenter, T., Jacobs, V., Franke, L., Levi. L. A longitudinal study of gender differences in young children's mathematical thinking. *Educational Researcher 27*, 5 (1998).
12. Gallagher, A. M., De Lisi, R. Gender differences in scholastic aptitude-test mathematics problem-solving among high-ability students. *J. Educational Psychology 8,* 2 (1994), 204–211.
13. Grigoreanu, V., Cao, J., Kulesza, T., Bogart, C., Rector, K., Burnett, M., and Wiedenbeck, S. Can feature design reduce the gender gap in end-user software development environments? In *Proc. VL/HCC 2008,* IEEE (2008), 149–156.
14. Hartzel, K. How self-efficacy and gender issues affect software adoption and use. *Comm. ACM 46*, 9 (2003), 167–171.
15. Kaye, J. Some statistical analyses of CHI. In *Proc. CHI 2009 Extended Abstracts*, ACM (2009), 2585–2594.
16. Kelleher, C., Pausch. R., and Kiesler, S. Storytelling Alice motivates middle school girls to learn computer programming, In *Proc. CHI 2007*, ACM (2007), 1455–1464.
17. Lawton, C. Gender differences in way-finding strategies: Relationship to spatial ability and spatial anxiety. *Sex Roles Journal 30*, 11-12 (1994), 765–779.
18. Ljungblad, S. and Holmquist, L. Transfer scenarios: Grounding innovation with marginal practices, In *Proc. CHI 2007*, ACM (2007), 737–746.
19. Margolis, J. and Fisher, A. *Unlocking the Clubhouse*, MIT Press, Cambridge, MA, USA, 2003.
20. McDowell, C., Werner, L., Bullock, H. E., Fernald, J. The impact of pair programming on student performance, perception and persistence, In *Proc. ICSE 2003*, ACM (2003).
21. Meyers-Levy, J. Gender differences in information processing: A selectivity interpretation. In P. Cafferata & A. Tybout (Eds) *Cognitive and Affective Responses to Advertising.* Lexington Books, Lexington, MA, USA, 1989.
22. O'Donnell, E., Johnson, E. Gender effects on processing effort during analytical procedures. *Int. J. Auditing 5*, 2001, 91–105.
23. Powell M. and Ansic D. Gender differences in risk behavior in financial decision-making: An experimental analysis, *J. Economic Psychology 18,* 6 (1997), 605–628.
24. Rode, J. An ethnographic examination of the relationship of gender & end-user programming, *Ph.D. Thesis*, Univ. Calif., 2008.
25. Rosson, M. B., Sinha, H., Bhattacharya, M., and Zhao, D. Design planning in end-user web development, In *Proc. VL/HCC 2007*, IEEE (2007), 189–196.
26. Sheskin, D. J. *Handbook of Parametric and Nonparametric Statistical Procedures*. Chapman & Hall/CRC, 2007.
27. Simon, S. The impact of culture and gender on web sites: An empirical study. *The Data Base for Advances in Information Systems 32*, 1 (2001), 18–37.
28. Subrahmaniyan, N., Beckwith, L., Grigoreanu, V., Burnett, M., Wiedenbeck, S., Narayanan, V., Bucht, K., Drummond, R., and Fern, X. Testing vs. code inspection vs. what else? Male and female end users' debugging strategies. In *Proc. CHI 2008*. ACM (2008), 617–626.
29. Tan, D., Czerwinski, M. and Robertson, G. Women go with the (optical) flow, In *Proc. CHI 2003*, ACM (2003), 209–215.
30. Torkzadeh, G. and X. Koufteros. Factorial validity of a computer self-efficacy scale and the impact of computer training. *Educational and Psychological Measurement 54,* 3 (1994).
31. Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., and Wesslén, A. *Experimentation in Software Engineering: An Introduction*. Kluwer, 2000.
32. Yin, R. *Case Study Methodology* (3rd edition), Sage, 2003.