AN EMPIRICAL STUDY OF SPREADSHEET ERROR-FINDING PERFORMANCE

Dennis F. Galletta*
Dolphy Abraham**
Mohamed El Louadi†
William Lekse*
Yannis A. Pollalis‡
Jeffrey L. Sampler§
*University of Pittsburgh
**Loyola Marymount University
†Université du Québec à Trois-Rivières
‡Syracuse University
§London Business School

Abstract—Several well-founded concerns exist about the integrity and validity of electronic spreadsheets. Thirty CPAs and 30 MBA students volunteered to seek up to two errors planted in each of six spreadsheets to discover if expertise in the domain of accounting and the spreadsheet program would facilitate error-finding performance. Subjects only found, on average, about 55% of the simple, conspicuous errors on the small spreadsheets. As expected, both accounting and spreadsheeting expertise contributed to the subjects’ error-finding rate, and those who performed this task with both types of expertise found the largest number of errors in the shortest time. Interestingly, while CPAs were more accurate than non-CPAs in finding accounting-related errors, spreadsheet experts did not outperform novices in finding spreadsheet formula errors. Also, while spreadsheet expertise contributed to greater speed, accounting expertise did not. Future research would further investigate the contribution of spreadsheet expertise to the error-finding task. Practitioners should be aware of the difficulties in finding even simple errors, and develop training programs to facilitate spreadsheet auditors’ performance.

Keywords: Spreadsheets, Spreadsheet errors, Errors, Auditing, Expertise, Experiment.

INTRODUCTION

Electronic spreadsheet software has altered completely the face of computing practice in the past decade. A recent study (AMA, 1988) reveals that spreadsheet software has become the most common computer application employed, used by over 91% of a sample of end users. Indeed, one can scarcely browse a trade publication published in the last decade without encountering scores of advertisements that describe software for building or enhancing spreadsheets. Also, a subscription to a computer-related publication often earns the reader frequent mailings that announce materials or training sessions for enhancing spreadsheet usage.

Concurrent with the growth in popularity of electronic spreadsheet software is a growth in concern about the high frequency and impact of errors in spreadsheet models. Although researchers have built some understanding of the conditions under which errors are made,
the possible effects of those errors, and useful tips for their avoidance, they have provided no assurance of an imminent, complete eradication of spreadsheet errors. It appears imperative to devote significant research efforts to error detection while we await technology that will eliminate errors altogether.

Error detection can be affected by several factors, not the least of which is the background of the persons attempting to find them. In this experimental study, two domains of expertise are explored for their contribution to error-finding performance of laboratory subjects; experts and novices in accounting and spreadsheet usage were asked to find and circle errors planted in spreadsheets provided to them.

ERRORS AND EXPERTISE

Difficulties in the construction and usage of spreadsheets can be considered symptoms of how end-user computing technology is managed: In concert with a general lack of planning (Alavi, Nelson, & Weiss, 1987–88; Munro, Huff, & Moore, 1987–88) and control (Galletta & Hufnagel, 1992), tasks are transferred from trained system developers to large numbers of users who are already fully employed performing their own tasks (Galletta & Heckman, 1990). Users therefore face a confusing environment (Brancheau, Vogel, & Wetherbe, 1985; Lederer & Sethi, 1986; Rockart & Flannery, 1983), employing tools that are in many ways more difficult and error-prone than are languages for developing computer programs (Xenakis, 1987).

It therefore seems likely that a staggering number of untrained end users embark on formidable software development tasks, including design, construction, and ongoing maintenance (AMA, 1988; Sumner & Klepper, 1987), and even training and support for others (Brancheau, Vogel, & Wetherbe, 1985). Unfamiliar, difficult tasks are a breeding ground for errors, and firms should exercise caution in the management of end-user computing (Davis, 1984).

There is evidence from both the laboratory (Brown & Gould, 1987) and the field (Creeth, 1985) that somewhere between one-third and one-half of all spreadsheets contain errors. In one study, only 5 of 19 spreadsheets sampled from actual practice were "totally error free" (Davies & Ikin, 1987, p. 55). Error severity appears to be quite high; errors in actual spreadsheet usage are known to have caused losses in the hundreds of thousands (Kee & Mason, 1988) to millions (Davies & Ikin, 1987; Ditlea, 1987) of dollars.

Much of the effort expended by other researchers (e.g., Ballou, Pazer, Belardo, & Klein, 1987; Olson & Nilsen, 1987–88) and practitioners (e.g., Anderson & Bernard, 1988; Pearson, 1988; Stang, 1987) has been devoted to describing how errors are made while spreadsheets are being built, estimating their effects, and prescribing techniques or software to prevent them.

Error prevention is perhaps the most desirable goal, but there is presently no "silver bullet" (Brooks, 1987) that will eradicate all errors made while constructing computer algorithms or models. Efforts in error prevention range from automating portions of the spreadsheet construction process to offering well-designed training programs. While the market potential of work-saving software constantly leads developers to chip away at the former goal, progress has been very slow due to wide variations in how spreadsheets are used. Also, each time difficult economic conditions become an imminent threat, many organizations cut back on training programs.

We should therefore turn our attention to detection of spreadsheet errors for the fore
seeable future; there appears to be a paucity of research in the empirical literature in this area. Although computer software is available that checks for the existence of several common errors, users are warned by the vendors that there are significant limitations to such software. Advances will undoubtedly be made in spreadsheet audit software, but automatic detection of all possible errors is not yet on the horizon. Therefore, there is presently a need for better understanding how humans detect errors in electronic spreadsheets.

The role of expertise

There are several approaches one can take in investigating spreadsheet error detection. As examples of the many studies that would be performed as part of a complete research program, investigators would focus on the people now performing the error-detection task, the errors themselves, the conditions under which errors are and are not found, and the impact of the spreadsheet software itself on how the task should be approached.

Attempting to investigate the latter presents a problem because there is neither coursework nor certification available to demonstrate expertise in spreadsheet error-finding; it is not obvious who should perform this task at the present time. Two territories covered by persons engaged in spreadsheet error-finding are the domain area of the task (e.g., accounting) and the device used (e.g., Lotus 1-2-3®). These areas differ markedly, and therefore must be considered separate dimensions of expertise. Such a differentiation has already been employed successfully by Mackay and her colleagues (Mackay, Barr, & Kletke, 1992; Mackay & Elam, 1992; Mackay & Lamb, 1991).

Likewise, the errors contain two possible dimensions: They are based on the domain of accounting, or they involve the use of the device (the spreadsheet software) on which the spreadsheet model is constructed. Hypotheses are presented below that suggest a correspondence between the areas of expertise and types of errors that can be found by spreadsheet auditors.

We address simultaneously pragmatic and theoretical issues. A key point is pragmatic: Who should be assigned to such tasks and how can we explain differences in their performance? Theoretical issues are also important: Is the accounting expertise implied by the CPA certificate valuable in error-finding? Is there a correspondence between spreadsheet auditors' experience and types of errors found? In this study, we seek to determine the value of both kinds of expertise in spreadsheet error-finding.

Accounting (domain) expertise

Many researchers have investigated accounting expertise, especially in the context of auditing. One of the main measures of this expertise has been years of audit experience. Although such a surrogate has proven successful in cognitive psychology research, the empirical evidence for using experience to explain differences in audit (and other judgemental) performance is mixed (Ashton, 1991; Johnson, 1988). The lack of success in using accounting experience as a surrogate for expertise has been explained by several researchers to include auditors' lack of task feedback (Ashton, 1991), researchers' neglect of the critical variables knowledge and ability (Bonner, Lewis, & Marchant, 1990; Frederick & Libby, 1986), the absence of a relationship between experimental tasks and subjects' areas of expertise (Bonner, 1990), and the unstructuredness and frequent absence of an objective answer when auditors perform such a task (Davis & Solomon, 1989; Hamilton & Wright, 1982; Johnson, 1988; Messier, 1983).

There is no conceptual disagreement that audit expertise includes audit experience
The above explanations of the lack of empirical evidence of the connection between the two constructs seem to advise us that the experience must cover the target task, auditors must have been given adequate feedback while gaining the experience, and subjects must possess not only experience, but knowledge and ability as well. If these criteria are taken into account in studying auditor expertise, it would seem likely that expert performance exceeds that of novices, in a manner similar to that generally found in the cognitive psychology literature (for example, see Chase & Simon, 1973; Chi, Glaser, & Rees, 1982; Larkin, McDermott, Simon, & Simon, 1980). That is, accounting experts will accurately identify more errors, more quickly than will novices.

Because this study employs basic accounting material found in elementary accounting courses, it is expected that CPAs have the requisite experience, knowledge, ability, and feedback to perform well on the spreadsheet error-detection task compared to novices who have only been exposed to the concepts through basic coursework. Stated more formally:

**H1:** CPAs will locate more errors than will non-CPAs.

**H2:** CPAs will locate errors more quickly than will non-CPAs.

Consistent with the semantic-syntactic differentiation in human-computer interaction presented by Shneiderman (1992), some errors can involve accounting principles (semantics), and others can involve spreadsheet commands (syntactics) to capture the accounting principles. It is reasonable to expect that semantic (accounting domain) errors will be captured more thoroughly and more quickly by CPAs than by non-CPAs because their training and experience only addressed the accounting domain.

**H3:** CPAs will locate accounting domain errors more thoroughly and quickly than will non-CPAs.

**Computer program debugging (device) expertise**

There is a great deal of research investigating differences between experts and novices in a variety of computer-related domains, including system design and programming (Adelson & Soloway, 1985; Bonar & Soloway, 1985; Soloway, Ehlich, Bonar, & Greenspan, 1982; Vitalari & Dickson, 1983) and computer program debugging (Gould, 1975; Gugerty & Olson, 1986; Oman, Curtis, & Nanja, 1989; Soloway & Ehlich, 1984; Vessey, 1985). Although there have been empirical studies that focused on spreadsheet creation in general (e.g., Brown & Gould, 1987; Olson & Nilsen, 1987-88) and errors made by beginners (e.g., Floyd & Pyun, 1987), there is virtually nothing appearing in the archival literature on spreadsheet error-finding. The similarity of spreadsheet models to computer programs is strong enough to warrant investigation of some of the other literature in computer-related expertise.

Generally, in computer-related studies, empirical support for claims of performance superiority of experts appears much stronger than empirical support for similar claims in the accounting domain. The strength of results in the computer-related area is perhaps due to the amount of feedback provided by the computer, the appropriateness of experimental tasks, and selection of subjects with experience as well as knowledge and ability. Consistent with results in the cognitive psychology literature, studies show that experts generally solve problems faster (Larkin, 1981; Simon & Simon, 1978) and commit fewer errors (Soloway et al., 1982) than novices. Expert/novice studies have been performed in the domain
of computer programming for nearly two decades. Studies have addressed program recall, comprehension, and debugging.

Program recall and comprehension are related tasks. Performance advantages for expert programmers in recalling program segments are important indicators that their greater understanding makes it easier for them to encode and decode their contents. Shneiderman (1976) provided evidence for such a process, finding that there is only a performance effect for experts when the lines of code are arranged in a sensible sequence. A nonsense sequence provides little for the experts to encode and decode. Further evidence is that experts recall program functions while novices focus on syntax (Adelson, 1981).

Several researchers have also studied expertise and program comprehension. Experts appear to comprehend programs using some of the same mechanisms they use to recall them. Expert programmers search for key lines to verify their hypotheses about a program's function (Brooks, 1977), while novices appear to study programs on a line-by-line basis (Weidenbeck, 1986). One reason for this might be that experts possess a large bank of generic, prototypical program fragments, or programming plans that relate to control flow and to variables (Ehrlich & Soloway, 1984; Soloway, Adelson, & Ehrlich, 1988; Soloway & Ehrlich, 1984), and tap this bank to infer what is occurring in a program. Another reason might be that experts know which assumptions to make about how variables are named and how code is used (Soloway et al., 1988; Soloway & Ehrlich, 1984). Indeed, novice programmers appear to spend much time reading comments and comparison statements, while experts do not seem to need them to such an extent (Crosby & Stelovsky, 1990).

Studies in program debugging have also confirmed that experts are able to locate and remove bugs faster than novices. Experts can dispense with more superficial errors quickly, then move on to more subtle logic errors (Youngs, 1974). Experts use a breadth-first approach to gain an overall understanding of a program, while novices appear to be more anxious to attend to details, resulting in more frequently changed hypotheses (Vessey, 1985). Experts also seem more flexible in shifting from breadth-intensive (comprehension) to depth-intensive (isolation) strategies than novices (Gould, 1975; Oman et al., 1989).

In summary, expertise in computer programming appears to contribute significantly to program recall, comprehension, and, most importantly, debugging. These findings are important because the experts are better able to examine sequences of computer commands, understand them, and find errors. The empirical findings in the debugging literature would lead one to hypothesize that expertise in spreadsheet construction and usage would contribute to skill in finding spreadsheet errors:

**H4:** Spreadsheet experts will locate more errors than will novices.

**H5:** Spreadsheet experts will locate errors more quickly than will novices.

Consistent with our expectations for CPAs in the accounting domain, spreadsheet experts will be likely to find more spreadsheet device (syntactic) errors, and more quickly, than will spreadsheet novices, due to the nature of their training and experience.

**H6:** Spreadsheet experts will locate spreadsheet device errors more thoroughly and quickly than will spreadsheet novices.

Finally, it would be expected that expertise in both dimensions will result in highest performance of all, considering both accuracy and speed in detecting errors, suggesting an interaction effect between accounting and spreadsheet expertise:

**H7:** The highest performers overall will possess expertise in both dimensions.
METHOD

An experiment was conducted to identify differences between individuals with varying levels of expertise in a spreadsheet error-finding setting.

Subjects

Seventy subjects volunteered to participate in the spreadsheet error-finding study. Responses to background questionnaire items resulted in subjects being categorized as CPAs versus non-CPAs, and spreadsheet experts versus spreadsheet novices. This created four cells, as illustrated in Table 1.

After 10 subjects who performed the task were disqualified,1 15 subjects remained in each of the four cells described above. Most of the subjects in cells I and II (the CPAs) were obtained by using attendees at local CPA continuing education courses. The subjects in cells III and IV (the non-CPAs) were obtained by soliciting volunteers from MBA classes at a large university in the northeastern U.S. Most of the CPA subjects were enrolled in a basic spreadsheet skills course, and others were enrolled in more advanced spreadsheet topics courses. All subjects were paid a $5 incentive for their participation and offered a chance for a $20 prize for the highest speed and accuracy in finding spreadsheet errors.

Permission was granted to approach the CPA attendees in the morning, and to solicit their participation during lunchtime. In exchange for their loss of opportunity to obtain lunch, sandwiches were offered in addition to the flat incentive and chance for the prize. On average, about 50% of the CPA attendees elected to participate; most of the others had previous lunchtime commitments. Some of them performed the task at their offices at a later date.

CPAs were considered to be accounting experts for the purposes of this task. A CPA must complete a number of accounting courses and pass a difficult exam. This exam implicitly requires basic ability and explicitly tests knowledge of accounting facts. Beyond the CPA exam, a Pennsylvania candidate must obtain 2 years of public accounting experience (1 year if he or she possesses a graduate degree). Thus, all of the components necessary for accounting expertise appear present in the CPA subjects. The non-CPAs were exposed to the basic material in an introductory accounting course, but did not enroll in any courses beyond the introductory level.

1 Five non-CPAs were disqualified because their questionnaire responses revealed that while they were not CPAs, they were accounting majors, were experienced in developing accounting software, or were Chartered Accountants from other countries. Four had spreadsheet experience above the novice threshold and below the expert threshold. One was disqualified when an error in setting up the computer prevented accurate capture of timing data.
The spreadsheet experts in cells I and III were somewhat arbitrarily identified as those who spent over 250 hours in spreadsheet creation and/or modification (but not data entry or other superficial manipulation). Subjects who reported fewer than 150 hours of such experience were classified as spreadsheet novices (cells II and IV). The mean (median) number of hours for spreadsheet experts was 1,548 (925), and for novices was 54 (44).

Spreadsheet experience was used as an indicator of expertise for two reasons. First, the subjects were assumed to have the requisite basic knowledge and ability in the level of arithmetic used in the spreadsheet tasks, because of the rigorous screening for entrance into the MBA program or requirements for CPA candidacy. Second, the key for building expertise in interactive software usage is to build on basic abilities and knowledge through experience; the feedback requirement appeared to be fulfilled by the highly experienced subjects. Because of the strikingly interactive nature of spreadsheet software, user errors are very often found immediately or as the result of further manipulation of spreadsheets. This experience will provide performance feedback because of the immediate error messages or obviously unreasonable results provided by the software. It is important for users to experience many of these errors.

Extrapolations of the results of Floyd and Pyun (1987) indicate that the expert users in this study were likely to have been given what might be considered adequate feedback for building expertise. If users indeed make about 72 errors per hour during intensive spreadsheet use, only discover a third of them, and on a long-term basis work a third as intensively as they do during short laboratory bursts, then the 250-hour threshold used to indicate spreadsheet expertise involves more than 2,000 errors involving feedback. The mean number of hours of usage (1,548) reported by spreadsheet experts implies over 12,000 feedback-related errors. In contrast, the mean number of hours of novice usage (54) implies about 450 feedback-related errors, on average.

All novice subjects had, prior to the experiment, been presented formally with at least the minimal level of material needed to perform the task. All of the non-CPAs had completed courses in introductory financial and managerial accounting about 6 months earlier. All of the spreadsheet novices received introductory training that covered all of the commands used. Only very basic accounting fundamentals and very basic spreadsheet commands were used. For example, the only Lotus commands used were addition, subtraction, multiplication, division, and summing over several cells (using the "@SUM" command). This requirement of basic skills provided a more stringent, meaningful, and realistic test of the dimensions of expertise than would be afforded by employing novices without even basic skills.

Materials

Ten spreadsheets (see Appendix for an example), adapted from examples that appeared in various introductory accounting textbooks, were created. Exactly two errors were introduced into each spreadsheet: one was an error in an accounting concept or principle (domain error), and one was an error in usage of the spreadsheet program (device error). Each error was carefully constructed to belong to only one of those two categories; formulas associated with domain error cells were otherwise correctly constructed and formulas associated with device error cells otherwise employed proper accounting principles.

Examples of the domain errors included presenting accumulated depreciation as a debit on a trial balance, classification of “expenses prepaid this year for next year” in a list of non-cash expenses, and subtraction of scrap value from the cost of an asset before com-
puting double-declining balance depreciation for book purposes. Examples of the device
errors include omission of a character in a formula for a column total (a formula that
should have been \(+G_{13}+G_{14}+G_{15}\) was entered as \(+G_{13}+G_{4}+G_{15}\) ), inclusion of an inter-
mediate subtotal in a range for calculating the sum of all items in a column, and improper
placement of parentheses in a financial ratio formula where the numerator and denomi-
nator were defined correctly in words on the spreadsheet. The impact of each error was
tightly constrained to enable scoring of the results. Had this not been done, it would have
been impossible to quantify the number of errors on any given spreadsheet. If an item was
in error, it did not carry through to further calculations on the spreadsheet.

An initial pilot test was performed on a CPA who was a spreadsheet expert using a per-
sonal computer attached to an overhead projection screen in front of the investigators. This
pilot subject was also encouraged to think aloud while performing the task so that the
experimenters could take notes of any difficulties he encountered with the task. Six final
spreadsheets were chosen from the 10, and were refined on the basis of this subject’s expe-
rience. A follow-up pilot test was then performed using PhD students and other CPAs, and
any remaining misleading or ambiguous aspects of the materials were corrected.

Experimental procedures

Subjects were asked to find and circle (but not fix) as many of the spreadsheet errors
as they could find. The spreadsheets were presented both on paper (see the Appendix for
an example) and on a personal computer running Lotus 1-2-3 (trademark of Lotus Devel-
opment Corporation), version 2.01. All subjects were free to move the cursor about the
screen to investigate the contents of any cell. Each spreadsheet was small enough to fit on
a single display screen to avoid any variability caused by differences in their screen scroll-
ing methods. Subjects were told that there were either zero, one, or two errors on each
spreadsheet, and were instructed to circle the errors they found on the paper copy, indic-
ating which they found first or second.

After they were satisfied that either no errors existed, or that they found all errors in
each spreadsheet, they were instructed to activate a Lotus macro\(^2\) that captured the com-
puter’s internal clock time, recorded the results on the diskette, and loaded the next spread-
sheet. After completing the last spreadsheet, a macro available from an exit screen
automatically summarized all of the timing information while subjects completed a post-
experiment questionnaire. Subjects began their tasks at different times to minimize any con-
 founding effects of direct competition.

Virtually all of the computers used by CPA subjects were IBM PCs, using CGA-level
13” color monitors, and virtually all of the computers used by non-CPA subjects were IBM
PS/2 model 30s with 12” monochrome VGA monitors. This difference was unavoidable
because of the differences in computer laboratories available to those subjects. Readabil-
ity was actually very similar in each setting, because both kinds of screens were used in text
mode, and any incremental sharpness of the monochrome VGA’s image (which could
enhance performance) was probably compromised by its smaller screen size (which could
degrade performance). Although studies have revealed reading and comprehension differ-
ences between users of conventional CRT screens and paper (e.g., see Gould et al., 1987a;
1987b), neither theory nor data exist to suggest that performance would be markedly dif-

\(^2\)A Lotus macro is a program embedded into a spreadsheet, using the standard spreadsheet commands.
SPREADSHEET ERROR-FINDING PERFORMANCE

Table 2. Mean number of errors found (H1, H4, H7; n = 15 per cell)

<table>
<thead>
<tr>
<th></th>
<th>Spreadsheet novices</th>
<th>Spreadsheet experts</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPAs</td>
<td>6.8 (2.2)</td>
<td>7.9 (1.8)</td>
<td>7.3 (2.1)</td>
</tr>
<tr>
<td>non-CPAs</td>
<td>6.3 (2.1)</td>
<td>5.7 (2.3)</td>
<td>6.0 (2.2)</td>
</tr>
<tr>
<td>Overall</td>
<td>6.6 (2.1)</td>
<td>6.8 (2.3)</td>
<td>6.7 (2.2)</td>
</tr>
</tbody>
</table>

Note. 12 would be a perfect score; standard deviations in parentheses.

different for either type of screen, especially with each placed in text mode. Nevertheless, generalizability of the results might be affected by screen differences, and appropriate cautions must be given for interpreting the results.

Coding of outcomes

Scoring of subjects’ correctness was performed by two judges; both assigned initial scores for each subject without knowledge of the subject’s category or the other judge’s score. Out of 720 judgments, the raters initially disagreed on only 23 occasions, but resolved their differences after discussing the results; most were simply errors in coding. Initial intercoder agreement was assessed with Cohen’s Kappa using a procedure outlined by Bishop (1975). An overall Kappa value of .94 was obtained for all initial judgments, abstracting across error types and occurrences. Kappa scores for each error type for each spreadsheet ranged from .819 to 1.000; the evidence appears to suggest that there was substantial agreement between the coders even before the differences were found and resolved.

Except where noted, hypothesis testing employed a 2 x 2 ANOVA approach, with two levels (novice and expert) in each dimension (accounting domain and spreadsheet device).

RESULTS

Overall, the subjects found only 56% of the errors (46% of the domain errors and 65% of the device errors). Because performance at a 50% level would enable maximum variability in either direction in the dependent variable, the task difficulty appeared to be nearly ideal.

Hypotheses H1 and H4 assert that accounting and spreadsheeting expertise would enable subjects to locate more errors in the spreadsheets, respectively. Further, Hypothesis H7 predicts than an interaction between the two types of expertise enables subjects with expertise in both areas to find the largest number of errors. Table 2 depicts the mean number of errors found in each cell.

The means in Table 2 were subjected to ANOVA, and only the main accounting expertise effect was significant ($F = 5.67; 1,56 df; p < .021$), supporting Hypothesis H1. Nei-

---

3One of the two judges was aware of the experimental hypotheses, and thus the subjects' expertise categories were concealed.
ther the main spreadsheet expertise effect (H4; $F = .18; 1,56 df; p < .671$) nor the interaction effect (H7: $F = 2.33; 1,56 df; p < .133$) was significant. The Bartlett-Box statistic was not significant, signifying homogeneity of variance among the cells ($F = .22; 3,5645 df; p < .881$). Therefore, an important assumption of ANOVA was not violated.

Hypotheses H2 and H5 predict that the amount of time necessary to perform the error-finding task would be lower for subjects with expertise in at least one dimension. Hypothesis 7 predicts that CPAs with spreadsheet expertise would be fastest of all. Table 3 presents the mean time for subjects in each cell.

The means in Table 3 were examined using ANOVA, and it was found that only the spreadsheet expertise main effect was significant in differentiating subjects’ time performance (H5; $F = 6.67; 1,56 df; p < .012$). Neither the domain expertise main effect (H2; $F = .01; 1,56 df; p < .937$) nor the interaction effect (H7; $F = .88; 1,56 df; p < .353$) was significant. Therefore, H5 receives support but H2 and H7 do not. One difficulty with this analysis was a lack of homogeneity of variance (Bartlett-Box = 4.5; 3, 5645 df, $p < .004$).

A natural log transformation of the time measure rectified this difficulty (Bartlett-Box = 1.7; 3, 5645 df; $p < .165$) and the statistical analysis was virtually unchanged (for H5, minutes taken, $F = 5.98; 1,56 df; p < .018$).

Because it appears that accounting expertise assists in locating errors, while spreadsheet expertise assists in speeding up performance, and because there is likely to be a trade-off between speed and accuracy, an analysis was performed taking into account both speed and accuracy. A ratio of errors per hour was computed for each subject, presented in Table 4.

ANOVA results indicate that both the domain main effect ($p = .038; F = 4.5; 1,56 df$) and device main effect ($p = .035; F = 4.7; 1,56 df$) were significant, suggesting that there is indeed a trade-off between time and number of errors found that affected the results of analysis of means in Tables 2 and 3. Furthermore, the predicted interaction effect was significant ($p = .032; F = 4.9; 1,56 df$), lending support to H7. The Bartlett-Box statistic was not significant ($F = .21; 3,5645 df; p < .886$), signifying that the ANOVA homogeneity assumption was not violated.

Hypotheses H3 and H6 predict that the performance effects can be explained by looking deeper into the types of errors found by subjects with expertise in each dimension. Tables 5 and 6 depict cell means taking into account only those six errors that match the expertise of the subjects. Table 5 compares the domain error-finding performance of CPAs

---

Note. Standard deviations shown in parentheses.
versus non-CPAs, and Table 6 compares the device error-finding performance of spreadsheet experts versus novices.

Results of t-tests performed on the data in Tables 5 and 6 are shown to the right of each table, and indicate that there is indeed some evidence that the performance of CPAs was closely related to the nature of the accounting errors (H3). The results in Table 6 do not support H6, and there is no evidence found to support the prediction that spreadsheet experts would perform better than spreadsheet novices because of their skill in finding device errors. The variances in each cell of each t-test were also compared. In each case except one, there were no differences between the variances. The variance of total time taken, in Table 6, was significantly different for spreadsheet experts and novices ($F=3.95$, $p < .001$). The separate variance estimate was used in determining the statistical significance of the two means in question, hence the calculated degrees of freedom.

The analysis above is somewhat incomplete because of the inability to isolate how much time was spent seeking each type of error. Because subjects were asked to identify which error they identified first, there is an alternative method for examining time-related results for each type of error. Table 7 presents the mean number of domain errors found first by CPAs and non-CPAs, and Table 8 presents the number of device errors found first by spreadsheet experts and novices. T-test results are again shown to the right of each table.

Once again, it is evident that accounting expertise has a specific effect on subjects' performance in finding accounting domain errors (H3), and that spreadsheet expertise has no discernable effect on subjects' performance in finding spreadsheet device errors (H6).

---

Note. A perfect score would be six accounting errors found; standard deviations are shown in parentheses.

---

Table 5. Mean accounting error-finding performance of subjects of varying accounting expertise (H3; $N = 30$ per cell)

<table>
<thead>
<tr>
<th></th>
<th>non-CPAs</th>
<th>CPAs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accounting errors found</td>
<td>2.3 (1.4)</td>
<td>3.2 (1.3)</td>
</tr>
<tr>
<td>Total time taken, minutes</td>
<td>23.1 (6.2)</td>
<td>23.3 (8.3)</td>
</tr>
<tr>
<td>Ratio, computed per-hour</td>
<td>6.3 (3.8)</td>
<td>9.1 (4.6)</td>
</tr>
</tbody>
</table>

Note. A perfect score would be six accounting errors found; standard deviations are shown in parentheses.
Table 6. Mean spreadsheet error-finding performance of subjects of varying spreadsheet expertise (H6; N = 30 per cell)

<table>
<thead>
<tr>
<th></th>
<th>Spreadsheet novices</th>
<th>Spreadsheet experts</th>
<th>ns (t = .17; 58 df; p &lt; .868)</th>
<th>* (t = 2.61; 42.81 df; p &lt; .013)</th>
<th>ns (t = -1.22; 58 df; p &lt; .228)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spreadsheet errors found</td>
<td>3.9 (1.6)</td>
<td>3.9 (1.5)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total time taken, minutes</td>
<td>25.5 (8.7)</td>
<td>20.9 (4.4)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ratio, computed per-hour</td>
<td>10.0 (4.9)</td>
<td>11.6 (4.8)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note. A perfect score would be six spreadsheet errors found; standard deviations are shown in parentheses; the separate variance estimate was used for the total time taken statistics.

DISCUSSION

While most of the hypotheses were fully or partially supported by the data, there are several limitations of this study. First, due to the requirement for choosing subjects with high and low levels of expertise in accounting and spreadsheet software usage, subjects could not be chosen at random from a single population; no single sampling frame includes subjects from all four cells. A related problem was that effects of the use of a variety of equipment were undefined. The use of a single computer for all subjects would have solved this problem, but the CPAs’ computer lab was not available to the MBAs and vice-versa. A third problem was that the errors planted in the spreadsheets had to be localized to a few cells, and thus might have not been representative of all errors found in real spreadsheets, where effects of a single error can propagate to dozens of other cells. This was necessary to enable meaningful counting of errors. A fourth problem is the employment of self-reported measures to establish spreadsheet expertise as opposed to the more objective means of establishing accounting expertise; this limitation might partially explain the different speed and accuracy results for each type of expertise. Finally, none of the CPA subjects were members of national firms, and results might have been attenuated somewhat; Messier (1983) found that CPAs from national firms outperformed CPAs from local firms. However, there were nevertheless some useful findings.

Findings

This study provided some consistent evidence of striking overall performance advantages (i.e., the number of errors found and/or the speed with which they are found) when spread-
Table 8. Mean number of spreadsheet errors found first by subjects with varying spreadsheet expertise (H6; N = 30 per cell; standard deviations are in parentheses)

<table>
<thead>
<tr>
<th>Spreadsheet novices</th>
<th>Spreadsheet experts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spreadsheet errors found first</td>
<td>3.4 (1.7)</td>
</tr>
</tbody>
</table>

*Note.* Purely by chance, subjects earning perfect scores would be expected to find 3 of the accounting errors first and 3 of the spreadsheet errors first. On average, the subjects in this sample would be expected to find 1.4 of the accounting errors first and 2.0 of the spreadsheet errors first given the overall 46% and 65% success rates of finding the 6 domain and 6 device errors, respectively.

Sheet auditors possess accounting expertise as well as spreadsheet expertise. Subjects possessing both kinds of expertise far exceeded the overall performance of subjects with expertise in only one or neither area. These overall results appear to support the notions that the CPA certificate indicates domain expertise, and that spreadsheet experience provides effective feedback for forming device expertise.

Domain-related analysis revealed that accounting expertise provides clear performance advantages in the number of accounting errors found (effectiveness). The performance advantages of accounting expertise were present when considering number of accounting errors found, number of accounting errors found per unit of time, and number of times that the accounting error was found before the spreadsheet usage error in each spreadsheet.

Device-related analysis revealed that spreadsheet expertise provides clear speed advantages in finding spreadsheet errors (efficiency). However, contrary to what might have been expected, there was no evidence to suggest that spreadsheet expertise is crucial for discovering errors strictly affecting spreadsheet formulas and structure. For these subjects and this task, one spreadsheet training session provided CPAs with enough spreadsheet background to find errors that were strictly device-related, given extra time allowed for the task.

These results appear on the surface to contradict those of previous research in two ways. (a) Accounting expertise appears to be a meaningful construct for an error-finding (i.e., auditing) task. (b) Experience appears to be an adequate surrogate for computer-related expertise in an error-finding task. The success of the accounting expertise measure might have resulted from an appropriate level of problem difficulty; the task chosen in this study probably did not involve errors that auditors seldom encounter. The computer-related expertise surrogate for expertise might have succeeded not only due to the appropriateness of the task, but also due to the amount of feedback spreadsheet builders probably receive while they work.

**Implications**

Both researchers and practitioners can benefit from several implications of this experiment. In general, this study has provided evidence that researchers should consider expertise and/or experience in several domains to study certain aspects of end-user computing performance. Besides the accounting domain, software is employed by end users of a variety of backgrounds, including engineers, architects, managers, and other professionals. Further, there is abundant opportunity for researchers to discover why even obvious, elementary errors in very simple, clearly-documented spreadsheets are so difficult to find.
Practitioners need to recognize the sheer difficulty of the spreadsheet error-finding task, while these errors, shown elsewhere to be abundant, can have wildly significant effects on decision-making. CPA firm managers might need to ascertain that any client spreadsheets are examined (or at least reviewed) for errors by CPAs with an abundance of spreadsheet experience. Perhaps teams should meet to perform spreadsheet reviews, in a manner similar to programming code walkthroughs. Training programs might be developed to increase the levels of performance of spreadsheet reviewers.

This study provides some evidence that effectiveness of the task of spreadsheet error-finding is improved via accounting domain expertise, while efficiency of that task is improved via spreadsheet device expertise. Although prevention of errors would be much more effective than attempting to find them, it would be unrealistic to expect to find permanent preventative measures. As long as spreadsheet builders remain fallible, the spectre of enormous but latent spreadsheet errors will remain. Increased understanding of the detection of spreadsheet errors is an important step in minimizing their effects.

Acknowledgements—The authors would like to thank Kirk Fiedler for his abundant assistance in material refinement and scoring validation, as well as the Pennsylvania Institute of CPAs Foundation for Education and Research and Alternative Computer Corporation for allowing CPA professional development course enrollees to serve as subjects in this study.

This study was supported by a Faculty Research Grant to the first author from the Joseph M. Katz Graduate School of Business.

REFERENCES


APPENDIX

Example of Spreadsheet #1, as seen and marked by subjects

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>#1</td>
<td>NONCASH AND OTHER EXPENSES</td>
<td>January</td>
<td>February</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Noncash expenses</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>Depreciation - delivery van</td>
<td>$757</td>
<td>$757</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>Amortization - patents</td>
<td>95</td>
<td>95</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>Amounts prepaid this year for next year</td>
<td>1,545</td>
<td>1,064</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>Total noncash expenses</td>
<td>2,397</td>
<td>1,916</td>
</tr>
<tr>
<td>9</td>
<td>Other expenses</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td>Office salaries</td>
<td>$105,579</td>
<td>$105,579</td>
</tr>
<tr>
<td>11</td>
<td></td>
<td>Officers' salaries</td>
<td>85,326</td>
<td>85,326</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td>Advertising</td>
<td>45,372</td>
<td>32,882</td>
</tr>
<tr>
<td>13</td>
<td></td>
<td>Office expenses</td>
<td>16,532</td>
<td>19,478</td>
</tr>
<tr>
<td>14</td>
<td></td>
<td>Rent</td>
<td>14,974</td>
<td>14,974</td>
</tr>
<tr>
<td>15</td>
<td></td>
<td>Electricity</td>
<td>7,732</td>
<td>6,933</td>
</tr>
<tr>
<td>16</td>
<td></td>
<td>Interest</td>
<td>3,956</td>
<td>5,489</td>
</tr>
<tr>
<td>17</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
<td>Total noncash and other expenses</td>
<td>$284,265</td>
<td>$274,493</td>
</tr>
<tr>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td>(PRESS ALT-X TO MOVE TO THE NEXT SPREADSHEET)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Error 1: 
Error 2:
Example of Spreadsheet #1, with formulas illustrated

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>#1</td>
<td>NONCASH AND OTHER EXPENSES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>January</td>
<td>February</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Noncash expenses</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Depreciation - delivery van</td>
<td>$757</td>
<td>$757</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Amortization - patents</td>
<td>95</td>
<td>95</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Amounts prepaid this year for next year</td>
<td>1,545</td>
<td>1,064</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Total noncash expenses</td>
<td></td>
<td>+C4+C5+C6</td>
<td>+D4+D5+D6</td>
</tr>
<tr>
<td>8</td>
<td>Other expenses</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Office salaries</td>
<td>$105,579</td>
<td>$105,579</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Officers' salaries</td>
<td>85,326</td>
<td>85,326</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Advertising</td>
<td>45,372</td>
<td>32,882</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Office expenses</td>
<td>16,532</td>
<td>19,478</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Rent</td>
<td>14,974</td>
<td>14,974</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Electricity</td>
<td>7,732</td>
<td>6,933</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>Interest</td>
<td>3,956</td>
<td>5,489</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Total noncash and other expenses</td>
<td>@SUM(C4..C16)</td>
<td>@SUM(D4..D16)</td>
<td></td>
</tr>
</tbody>
</table>

(PRESS ALT-X TO MOVE TO THE NEXT SPREADSHEET)