



# A critical review of the literature on spreadsheet errors <sup>☆</sup>

Stephen G. Powell <sup>\*</sup>, Kenneth R. Baker, Barry Lawson

*Tuck School of Business, Dartmouth College, Hanover, NH 03755, USA*

## ARTICLE INFO

### Article history:

Received 8 December 2006

Received in revised form 20 May 2008

Accepted 2 June 2008

Available online 7 June 2008

### Keywords:

Spreadsheets

Error classification

Decision support

End-user computing

## ABSTRACT

Among those who study spreadsheet use, it is widely accepted that errors are prevalent in operational spreadsheets and that errors can lead to poor decisions and cost millions of dollars. However, relatively little is known about what types of errors actually occur, how they were created, how they can be detected, and how they can be avoided or minimized. This paper summarizes and critiques the research literature on spreadsheet errors from the viewpoint of a manager who wishes to improve operational spreadsheet quality. We also offer suggestions for future research directions that can improve the state of knowledge about spreadsheet errors and mitigate spreadsheet risks.

© 2008 Elsevier B.V. All rights reserved.

## 1. Introduction

The problem of eliminating errors from software has been around since the beginning of the computer era. The discipline of software engineering [38] arose out of a need for error-free software code. With the advent of the personal computer in the 1980s and the rapid rise of end-user computing, control of software development passed out of the hands of professionals and into the hands of millions of spreadsheet users, few of whom had any formal training for the task.

As spreadsheets have diffused throughout business, evidence has accumulated that many spreadsheets contain errors [16,24,25,29] and that errors can be costly to the organizations that use them (European Spreadsheet Risks Interest Group: <http://www.eusprig.org/stories.htm>). Nevertheless, end users and organizations that rely on spreadsheets have generally not recognized the risks of spreadsheet errors [21]. In fact, spreadsheets are somewhat ignored, both as corporate assets and as sources of risk.

Although research has suggested that errors are prevalent in spreadsheets, there is much we don't know about the types of

errors that occur, why they occur, and how to avoid them. We believe that a critical review of the relevant literature can inform future research on this topic. This paper provides such a review.

Rather than give a chronological account of the literature on spreadsheet errors, we organize the discussion around the following topics:

- Classification: what types of errors occur?
- Impact: what are the consequences of errors?
- Frequency: how common are errors?
- Creation and prevention: how can we build trustworthy spreadsheets?
- Detection: how can we audit spreadsheets to correct errors when they occur?

We devote a section of the paper to each of these topics in turn and conclude by suggesting guidelines for future research.

## 2. Types of errors

Any classification system allows us to understand the commonalities among individual instances. The Linnaean system, for example, classifies living things into species, genera, families, and so on. This hierarchy allows us to infer that individuals in the same species are more alike than those in different species, and species in the same genera are more alike than those in different genera. An effective classification of spreadsheet errors would allow us to compare errors across

<sup>☆</sup> This work was performed under the sponsorship of the U.S. Department of Commerce, National Institute of Standards and Technology. Reproduction of this article, with the customary credit to the source, is permitted.

<sup>\*</sup> Corresponding author.

E-mail address: [sgp@dartmouth.edu](mailto:sgp@dartmouth.edu) (S.G. Powell).

studies and application areas. It would also allow us to diagnose the causes and prescribe the cures for errors.

Spreadsheet errors can be placed into categories in a variety of ways. For example, errors could be distinguished by *cause*, *effect*, *form*, *stage*, or *risk*. Some examples of causes of errors are typing mistakes, copy/paste errors, and lack of knowledge. A second type of classification would be based on the effects of errors. Some errors lead to erroneous numerical results, while others may simply make the spreadsheet hard to use correctly. Errors could also be distinguished by their form. For example, one form of error is a mistaken formula; another is an input parameter that appears in multiple cells; a third (perhaps) is poor documentation. A fourth classification scheme would be based on the stage at which the error was created. Some errors first appear during the conceptual design of a spreadsheet, while others appear during the building process. Some do not appear until a spreadsheet is actually used. Finally, errors could be categorized by the risks they pose. In a multi-user spreadsheet, for example, failing to use cell protection to ensure that formulas are not overwritten would be a significant error.

There is no single accepted classification scheme for spreadsheet errors. Perhaps that is unavoidable, because errors can be classified in different ways for different purposes. Any well-developed classification should have three characteristics.

- First, it should specify the purpose for which it was created and/or the context in which it is meant to be used.
- Second, each category should be clearly defined and examples provided – that is, some errors that fall into the given category and some that do not.
- Third, the classification should be tested in the relevant context and evidence provided that different people classify errors consistently.

Classifications are created for a purpose, and a suitable classification for one purpose may not be suitable for another. In the spreadsheet realm, we can imagine that the classification a lab experimenter might find helpful would not be suitable for field audits. For example, in a lab environment, it might be possible to observe an incorrect formula being entered with a typing error. However, when discovered during a field audit, that same error might be indistinguishable from an error in logic.

Second, categories should be clearly defined and illustrated with examples. Broad categories such as “qualitative errors” are difficult to define precisely, and subcategories such as “constants in formulas” are not well defined without specific examples. For example, the following formula includes the constant 1, but few spreadsheet users would consider it erroneous:

$$\text{Sales\_06} = \text{Sales\_05} * (1 + \text{growth\_rate})$$

Third, the classification should be tested and proven in its area of application. Thus, if a classification is intended for use in auditing operational spreadsheets, it should be tested on spreadsheets from the field. To pass this test, researchers must show that individuals can categorize actual errors successfully and that multiple researchers place the same error in the same category a high percentage of the time.

### 2.1. Early classifications

The earliest studies concerned with errors either gave examples or listed overlapping types of errors. Brown and

Gould [2], for example, observed 17 errors, of which 15 led to wrong numbers. Of these, 11 involved errors in formulas, three involved mistyping, and one involved a rounding error. The remaining two errors involved misuse of cell protection and an error in logic.

Ronen et al. [36] cited eight error types mentioned in the practitioner literature:

- mistakes in logic
- incorrect ranges in formulas
- incorrect cell references
- confused range names
- incorrectly copied formulas
- incorrect use of formats and column widths
- accidentally overwritten formulae
- misuse of built-in functions

Cragg and King [7] offered a different list of error types:

- erroneous formulae
- incorrect ranges
- omitted factors
- data input errors
- incorrect use of functions
- duplication of effort

Note that most of these sources focus on errors in formulas, although other types of errors (such as data input errors and duplication of effort) are mentioned.

Galletta et al. [11] were the first to offer conceptual distinctions between classes of errors. They distinguished *domain* errors and *device* errors. Domain refers to the spreadsheet's application area (e.g., accounting), while device refers to the spreadsheet technology itself. For example, a mistake in logic due to a misunderstanding of depreciation is a domain error, but entering the wrong reference in the depreciation function SLN is a device error.

Saarilouma and Sajaniemi [37] distinguished between *location* errors and *formula* errors. Location errors occur in formulas that are conceptually correct but in which one or more of the cell references are incorrect. Formula errors include misuse of operators or the wrong number of operators.

Panko and Halverson [26] offered two more distinctions. One distinction is between *eureka* and *Cassandra* errors. Eureka errors are logic errors that are easily proven, while Cassandra errors are difficult to prove (even if detected). The other distinction is between *pure* and *domain* errors. Pure errors arise from a lapse in general logic while domain errors arise from a lack of specific domain knowledge.

### 2.2. Development of taxonomies

The first serious attempt to offer a complete classification of errors came from Panko and Halverson [26]. They distinguished between *quantitative* and *qualitative* errors and further categorized quantitative errors. Quantitative errors lead to wrong numbers in the current version of the spreadsheet. An example is the wrong range in a SUM calculation, resulting in an incorrect total. Qualitative errors refer to practices that don't lead to wrong numbers in the current version of the spreadsheet but are risky practices that could lead to errors in subsequent uses. An example is hard-coding input parameters into a formula.

Quantitative errors are thus well defined and unambiguous, although not necessarily easy to find. Qualitative errors are not always so clear. For example, hard-coding numbers into formulas is not a quantitative error if the current version of the spreadsheet is correct, but it could lead to quantitative errors in subsequent uses if the user does not change inputs appropriately. Qualitative errors can also include complex formulas, confusing layout, or insufficient documentation.

Panko and Halverson further divided quantitative errors into three subcategories:

- *mechanical* errors, due to mistakes in typing or pointing
- *logic* errors, due to choosing the wrong function or creating the wrong formula
- *omission* errors, due to misinterpreting the situation to be modeled

This classification, which is based on the causes of errors, raises a number of questions. The first is: what is the context in which errors are being studied? If the context is laboratory experiments, we may be able to observe the spreadsheet developer in action and detect how each error was committed. On the other hand, if the context is evaluating operational spreadsheets in the field, we can only observe cells in the spreadsheet, not the process that created them. An appropriate classification of errors for one context may not be appropriate for the other. A second question is: what is the purpose of the classification? If the purpose is theoretical understanding, then we can usefully list all possible causes of errors. But if the purpose is a practical one, then an appropriate classification should include only errors we can observe and distinguish from one another. In field audits, we must concentrate not on causes but on outcomes. For example, the ultimate cause of an erroneous financial formula could be that the inputs were not typed correctly, the developer did not understand the necessary financial theory, the developer misunderstood the function, or the developer omitted an important factor from the model itself. In theory, these causes can be distinguished, but in practice we can observe only the error itself, so the finer categories may not be relevant.

One of the shortcomings of existing classifications is the lack of examples. However, Panko [28] provided a detailed listing of the errors made by subjects in several spreadsheet-design experiments. A total of 130 errors were briefly described and categorized into logic errors, omission errors, or mechanical errors.

Teo and Tan [40] built on the classification scheme of Panko and Halverson, adding two additional types of qualitative errors: *jamming* errors and *duplication* errors. A jamming error occurs when more than one parameter is placed in a single cell, as when a volume is calculated by multiplying parameters for length, height and width, but the individual parameters are not isolated elsewhere. A duplication error occurs when the same parameter appears in two or more cells.

Rajalingham et al. [33] and Purser and Chadwick [32] developed the most elaborate taxonomy of errors available to date. The first distinction in this hierarchy is between *application-identified* errors and *developer/user-identified* errors. Excel has eight categories of errors that it displays in a cell when it cannot resolve a formula. Some examples are #DIV/0! for dividing by zero and #VALUE! for the wrong type of argument in a cell reference. These are application-identified

errors. Most such errors are likely to be caught by spreadsheet developers as they build a spreadsheet. However, in our own error research [30,31] we have occasionally observed these errors even in operational spreadsheets.

The hierarchy for developer/user-identified errors can be presented in an outline form, as in Table 1. At the highest level, quantitative errors are distinguished from qualitative errors. Qualitative errors are divided into *structural* errors, which arise from flaws in the design, and *temporal* errors, which arise from the use of non-current data. Structural errors are further divided into *visible* and *hidden* errors. Presumably, a visible structural error is one that can be observed in the current spreadsheet while a hidden one arises only during use.

Note that these authors defined qualitative errors somewhat differently from Panko and Halverson. As best we can tell, structural errors in Rajalingham et al. correspond to Panko and Halverson's category of qualitative errors. But the former also classify as a qualitative error the use of non-current data, which presumably leads to quantitative errors in the current spreadsheet.

Quantitative errors are divided first into *reasoning* and *accidental* errors. Reasoning errors arise from a lack of knowledge, whereas accidental errors are created while entering a formula. Reasoning errors arise from lack of *domain knowledge* or from *implementation* errors. Lack of domain knowledge itself can involve lack of *real world knowledge* or lack of *mathematical knowledge*. Implementation errors can involve problems with *syntax* (the rules for creating proper formulas in Excel) or with *logic*. Accidental errors arise either during *insertion* or *updating*. Updating itself can involve *modification* or *deletion*.

The taxonomy developed by Rajalingham et al. [33] uses at least three different criteria to distinguish categories of errors. The first distinction, between application-identified errors and developer/user-identified errors, is based on the *detection method*. The distinction between errors due to lack of real world knowledge and lack of mathematical knowledge is based on the *cause* of the error. Finally, the distinction between an error occurring during insertion or updating is based on the *stage* at which it was caused. Although we acknowledge that a theoretical taxonomy may be based on

**Table 1**  
Developer/user errors (after Rajalingham [33] and Purser and Chadwick [32])

A. Qualitative errors
1. Structural
a. Visible
b. Hidden
2. Temporal
B. Quantitative errors
1. Reasoning
a. Domain knowledge
1. Real world knowledge
2. Mathematical representation
b. Implementation
1. Syntax
2. Logic
2. Accidental
a. Insertion
b. Update
1. Modification
2. Deletion

such disparate criteria, we question the usefulness of such a taxonomy in practice.

Although this hierarchy is the most detailed categorization of errors in the literature, as far as we know, it has not been tested on spreadsheets from the field. Thus, we have no empirical evidence on how well the classification categorizes errors found in practice. Common sense suggests that because this hierarchy is focused in large part on the *process* by which an error occurred, it may not be possible for an auditor (or even the original developer) to reconstruct the cause and thereby categorize it when only the completed artifact is available. For example, if a complex formula involving nested IF, VLOOKUP, and NPV functions gives a wrong result, we cannot determine whether the cause was incorrect specification of the problem, lack of domain knowledge, or an accident in typing.

A more serious concern with this taxonomy is that its categories overlap. For example, an error caused by lack of real world knowledge (category B1a1) could occur during modification (category B2b1). Likewise, an error due to lack of mathematical knowledge (B1a2) could surface as an application-identified error (e.g., #VALUE!). In the end, this taxonomy may therefore be of more theoretical than practical value.

### 2.3. Other approaches

Most of the work cited to this point concerns itself with errors in *completed* spreadsheets. Panko [24,25] correctly pointed out, however, that different kinds of errors can occur at different stages in the life cycle of a spreadsheet, and the incidence of errors also may vary by stage. Errors can occur in the formulation of a model, before an actual spreadsheet is constructed. They can also occur during cell entry, during testing, and during use. Studies of errors committed during cell entry [20,22] found, not surprisingly, that developers commit a large number of errors while actually building a spreadsheet, but most are corrected immediately.

Ayalew et al. [1] took a somewhat different approach to categorizing errors. They avoided categorizing based on the cause of the error but based their three-level schema on the spreadsheet modeling concept involved. Their three categories are *physical-area related* errors, *logical-area related* errors, and *general* errors. Physical-area related errors involve formulas that refer to blank cells, cells with values of the wrong type, or cells in the wrong range. Logical-area related errors involve overwriting a formula with a constant or copying a formula with incorrect references. Finally, the category of general errors includes all other possible errors in formulas, including typographical errors and domain or mathematical errors.

Finally, we mention a study by Caulkins et al. [5] based on field interviews of 45 managers whose organizations rely on spreadsheets. These subjects mentioned observing the following errors, in descending order of frequency:

- inaccurate data
- errors inherited from reusing spreadsheets
- model error
- error in use of functions
- misinterpretation of output/report
- link broken/failed to update
- copy/paste
- lost file/saved over file

Note that some of these categories, such as “link broken/failed to update” are not explicitly represented in any of the existing classifications.

### 2.4. Summary

Spreadsheet errors can be categorized in many different ways, for different contexts and purposes. No generally-accepted taxonomy of spreadsheet errors has yet been established. In our survey we have identified several shortcomings of the existing classifications:

- Classifications are offered without specifying the context or purpose for which the classification is intended.
- The existing classifications do not include sufficient examples of specific errors that satisfy each category.
- Classifications are not rigorously tested to demonstrate that multiple users can consistently classify actual errors into the proper categories.
- The boundary between quantitative errors and qualitative errors remains vague.
- No studies are available that compare the types of errors by development stage.

Our own research into errors in operational spreadsheets [30,31] suggests that a wide variety of types of errors occur. Furthermore, there are numerous borderline cases, where careful researchers can disagree as to whether a particular cell is an error. If classification of actual errors is as difficult as our experience suggests, any useful classification must be based on extensive analysis of actual spreadsheets, and must provide many examples to help researchers use the classification effectively.

Future research on error classifications should clearly state the purpose for which the classification is designed. Then a set of examples should be provided, both of errors that fall in a given category and those that do not. Finally, any proposed classification should be tested to determine if it fits its purpose. Such tests should at a minimum establish three things:

- Errors can be identified in the subject population that fall into each category.
- No errors are encountered that cannot be classified.
- Independent examiners agree on the proper classification of the majority of errors.

## 3. Impact of errors

Ironically, the *impact* of errors on spreadsheet results is the least studied of all the topics we address. Impact can be measured in several ways. An obvious measure is the percentage error in the outputs of the spreadsheet. But a 1% error in one spreadsheet could be devastating, while a 10% error in another could be inconsequential. A more telling measure of impact would be the actual dollar losses from erroneous or poor *decisions* resulting from spreadsheet errors. To estimate this impact would require tracking a problem within an organization from recognition through to implementation of a solution, and determining to what extent a suboptimal outcome was due to spreadsheet errors. This would be an ambitious undertaking, to say the least.

The evidence we do have on the impact of errors in audited spreadsheets is largely anecdotal. In a detailed audit of a single operational spreadsheet, Hicks (cited in Panko [25]) reported that the errors found in the audit caused the results to be off by 1.2%. Clermont et al. [6] found errors in an average of 3.03% of cells in three large spreadsheets but reported that “we did not find any tremendous erroneous result values that might have had severe negative effects on the company.” Lukasic (1998, personal communication cited in Panko [25]) found a 16% error in the results of one of two spreadsheets audited. Panko’s interviewees (Panko [25]) suggested that 5% of spreadsheets had “serious” errors.

Although we have little data on the impact of errors in audited spreadsheets, many stories have been published documenting losses due to mistakes involving spreadsheets. The European Spreadsheet Risks Interest Group (EUSPRIG) maintains a web page (<http://www.eusprig.org/stories.htm>) that documents dozens of these cases. Here is a small selection.

- Some candidates for police officer jobs are told they passed the test when in fact they had failed. Reason: the spreadsheet was sorted improperly.
- A school loses £30,000 because its budget is underestimated. Reason: numbers entered as text in a spreadsheet.
- Benefits of unbundling telecommunication services are understated by \$50 million. Reason: incorrect references in a spreadsheet formula.

A wide variety of errors has been documented by EUSPRIG, including formula errors, data entry errors, sorting errors, copy and paste errors, errors due to inadequate internal controls, formatting errors, and so on. The great majority of these reports, perhaps 80%, suggest that the error was actually in a formula. But there are also cases where the error was due to poor naming conventions, interaction with other software, and even fraud. Such examples remind us that the existing literature on errors focuses on what must be only a fraction of the actual errors surrounding the use of spreadsheets.

Caulkins et al. [5] also asked their survey subjects to evaluate the risk of spreadsheets in decision making. Of the 44 who responded, 25 (57%) agreed with the statement “Spreadsheet errors are a significant threat to decisions.” Ten respondents (23%) said they used spreadsheets to inform decisions but were not particularly concerned about errors. The remainder did not use spreadsheets directly to inform decisions.

Essentially no research has been conducted on the impact of errors in spreadsheets. Research in this area should be based on an explicit taxonomy of errors and a well-specified auditing procedure. In addition, it will be necessary to define carefully what is meant by “impact.” An error in a single formula can affect thousands of other cells in a spreadsheet. The impact can be small on some cells and large on others. To understand the overall impact will require a deep understanding of the purpose and uses of the spreadsheet, and probably the close collaboration of the developer.

#### 4. Frequency of errors

How common are errors in spreadsheets? Not surprisingly, the answer to this question depends on the definition of errors, the lifecycle stage, and the setting (operational or laboratory). About the only general conclusion we can draw

from the literature is that no studies have suggested that errors are *not* a problem in spreadsheets, with the exception of Nardi and Miller [22] who concluded that “users devote considerable effort to debugging their spreadsheet models – they are very self-conscious about the probability of error and routinely track down errors before they can do any real harm.” That conclusion, however, was based on self-reports by spreadsheet developers, not on an actual audit of the spreadsheets in question.

Panko and Halverson [26] and Panko [24,25] summarized several dozen studies on the frequency of errors. We focus here on the latest of these studies, Panko [25]. Panko’s summary is divided into four categories that correspond roughly to the spreadsheet development life cycle:

- cell entry experiments
- development experiments
- code inspection experiments
- field audits

The first three of these categories involve experiments in which subjects develop or debug spreadsheets. Only the last category directly addresses the frequency of errors in operational spreadsheets.

##### 4.1. Cell entry experiments

Panko reported three cell entry experiments, in which subjects were observed as they created formulas. The percentage of cells with errors ranged from 11.3% to 21%. These errors were counted as they occurred, whether or not they were subsequently corrected. Thus, some of these errors were essentially typing errors, not errors in finished spreadsheets. This research may be of interest to those who wish to improve the efficiency with which spreadsheets are created, but it should not be taken as indicative of the error rate in finished spreadsheets. (After all, the error rate for keystrokes in word processing might be 5–10%, depending on the typist, but the error rate in finished documents would typically be much less than 1% of characters entered.)

##### 4.2. Development experiments

In development experiments, subjects are typically given a word problem and asked to create an appropriate spreadsheet model. Panko cited eleven of these studies. Cell error rates ranged from a low of 2% to a high of 17%. (In one study, individuals had a cell error rate of 4.6% while groups of three had a cell error rate of 1.0% on the same task.) The percentage with at least one error ranged from 24% to 86%.

In a later study, Reinhardt and Pillay [35] reported on the errors made by computer literacy students. In their experiment, 82% made errors when using absolute addressing, 87% made errors using an IF function, and 93% made errors when using a financial function.

Laboratory experiments on spreadsheet errors may be useful for certain purposes, but we should be extremely careful in making inferences about the error rate in operational spreadsheets from errors rates in laboratory experiments. The two contexts differ in a host of ways. Operational spreadsheets are typically large, developed by a team over a significant period of time, and subjected to repeated use.

Spreadsheets in laboratory experiments, on the other hand, are typically small, built over a short period of time by one person, and rarely used.

Reason [34] provides two warnings that apply here about investigating errors in the laboratory:

First, the need to establish precise control over the possible determinants of the error often forces investigators to focus upon rather trivial phenomena.... Second, it is usually the case that the greater the measure of control achieved by the experimenter, the more artificial and unnatural are the conditions under which the error is elicited.

Certainly laboratory experiments on spreadsheets are to some extent artificial and unnatural, when compared to spreadsheet use in organizations. In a later section, we provide some perspective on the appropriate use of laboratory experiments in spreadsheet error research.

#### 4.3. Code inspection experiments

Code inspection experiments measure the ability of subjects to find errors in spreadsheets. We discuss these results in a later section on detecting errors.

#### 4.4. Field audits

Panko reported on thirteen field audits in which operational spreadsheets were examined (typically by an outsider to the organization). Since this class of studies is most directly relevant to the subject of this paper we review these studies in more detail. Panko summarized this literature by reporting that 94% of spreadsheets have errors, with an average cell error rate of 5.2%. We will revisit these estimates after reviewing the studies individually.

- Davies and Ikin [8] examined 19 Lotus 1–2–3 spreadsheets from a variety of firms. Five of the 19 had no quantitative or qualitative errors, while four had “major errors.” Presumably the remaining 10 had minor errors.
- Ditlea [9] reported that one Price-Waterhouse consultant found 128 errors in four large spreadsheets.
- Butler (personal communication 1992, cited in Panko [25]) reported errors requiring additional tax payments in 10.7% of 273 small-business tax returns audited by HM Customs and Excise (the tax authority in the UK).
- Cragg and King [7] reviewed 20 spreadsheets from 10 companies. A single auditor spent an average of 2 h examining each spreadsheet. Five of the 20 contained errors.
- Dent (1995, personal communication with Panko [25]), described an audit in a mining company that found errors in about 30% of spreadsheets.
- Hicks (1995, cited in Panko [25]) reported on an extensive code inspection process for one large spreadsheet at a major firm. Errors were found in five of 19 modules. In all, 45 errors were identified, with a cell error rate of 1.2%.
- Freeman [10] reported on the experience of auditors at Coopers and Lybrand, who investigated 23 spreadsheets of which 91% had errors. KPMG (1997, cited in Panko [25]) investigated 22 spreadsheets and again, 91% had “major”

errors. Butler [3,4] reported on audits of 7 tax spreadsheets using the same procedure as in Butler (1992, cited in Panko [25]) and 86% had errors.

- Lukasik (1998, personal communication with Panko [25]) audited two spreadsheets and found cell errors rates of 2.2 and 2.5%.
- Clermont et al. [6] audited three large workbooks used by a single industrial firm. These workbooks involved 78 worksheets and 60,446 cells. The average cell error rate was 3.03% (0.08% in one spreadsheet, 1.3% in a second, and 6.7% in the third). Of the total of 1,832 error cells identified, 241 (15%) were quantitative errors.
- In two sets of interviews in 2003 with experienced consultants who perform spreadsheet audits professionally, Panko was told that approximately 5% of spreadsheets had serious errors [25].
- Lawrence and Lee [19] reported on 30 audits conducted by Mercer Finance and Risk Consulting on project financing models. These models involved an average of 2182 unique formulas. On average, the auditor raised issues with the developer concerning 6.9% of all unique formulas in the spreadsheet; however, the range was from 3.1% to 22.5% of unique formulas.

These sources range from personal communications to well-documented research studies. Perhaps the most important conclusion to draw from them is that *no* source has maintained that operational spreadsheets are error-free. But what do they suggest is the average error rate?

Panko attempted to answer this question by averaging over the seven most recent studies, on the grounds that auditing procedures improved after 1995. These studies are summarized in Table 2. In total, 88 spreadsheets are represented in the table. For all 88, the weighted average percentage of spreadsheets with errors is 94%. Data on cell error rates were available on 43 of these spreadsheets, and the weighted average for this sample is 5.2%.

There are several reasons to question the reliability of these estimates. First, three of the seven sources are unpublished. Second, the majority of the sources gave little or no information on their definition of errors or on the methods used to find errors. Third, Lawrence and Lee [19], whose observations account for 70% of the sample used to estimate the cell error rate, did not actually report a cell error

**Table 2**  
Spreadsheet error rates (after Panko [25])

Source	Number	Percent audited with errors	Cell error rate (% of cells)
1. Coopers and Lybrand (1997)	23	91	N/A
2. KPMG (1998)	22	91	N/A
3. Hicks (1995)	1	100	1.2
4. Lukasik (1998)	2	100	2.2
5. Butler (2000)	7	86	0.4
6. Clermont (2002)	3	100	3.0
7. Lawrence and Lee (2004)	30	100	6.9
Average		94 <sup>a</sup>	5.2 <sup>b</sup>

<sup>a</sup>Weighted average of 88 spreadsheets in sources 1–7.

<sup>b</sup>Weighted average of 43 spreadsheets in sources 3–7.

rate of 6.9% on *completed* spreadsheets. Rather, they reported that auditors had “issues” on the *initial review* of a model that they subsequently discussed with the developers concerning 6.9% of the cells. The authors provided no details on the definition of an “issue” or its relation to an actual error. Moreover, the initial model version that was reviewed was revised an average of six times before the model was complete. Thus the estimate of a 6.9% error rate applies only to issues raised by auditors on initial model versions, not to errors in completed spreadsheets.

To be clear, we do not mean to suggest that the “true” error rates in operational spreadsheets are negligible. Our own studies [30,31] suggest the opposite. However, the evidence behind the estimates in the literature does not provide sufficient grounds for a reliable, quantitative estimate.

Managers who understand the importance of spreadsheet analysis to their business want to know how prevalent errors are in spreadsheets. Unfortunately, this is simply not a question we can answer accurately at this time, despite a number of studies in the area. The problem is a lack of standardization regarding:

- the definition of errors
- the methods used to detect errors
- the sample of spreadsheets studied

Previous studies have used different definitions of errors, when they have provided a definition at all. Many studies have not revealed details about the detection methods used; those that provided details used different methods. Finally, the characteristics of the spreadsheets used in the experimental sample have rarely been reported, so we cannot know whether the results apply to any particular area of interest, such as stage of development, application domain, or size of spreadsheet.

What we do know from this literature can be summarized as follows:

- Laboratory experiments have shown high error rates, whether measured by cell error rates or percent of spreadsheets with errors.
- Field audits have generally shown high cell and spreadsheet error rates, but methods and results vary widely.
- No studies of spreadsheets themselves (rather than interviews of developers) have shown errors to be rare or inconsequential.

To be effective, future research on the prevalence of errors must satisfy three criteria:

- The procedures used to select spreadsheets and the characteristics of those spreadsheets (such as the number of formulas) must be described.
- A well-tested taxonomy of errors must be used.
- The procedure used to audit the spreadsheets for errors must be clearly described.

## 5. Creation and prevention of errors

We know remarkably little about how errors are created by end users. Not surprisingly, perhaps, no one has attempted to study spreadsheet development in the field at a level of detail that would permit observation of developers making

errors. What little we do know comes from laboratory experiments, yet as previously stated, the relevance of laboratory results to the field is questionable.

In a study mentioned earlier, Brown and Gould [2] performed experiments in which their subjects created three spreadsheets from written descriptions. Subjects were videotaped during this process and their keystrokes recorded, thus allowing the causes of errors to be determined. The researchers observed typing mistakes, misuse of cell protection, errors in logic, erroneous cell references, copying mistakes, and misplacement of data.

Olson and Nilson [23] and Lerch [20] tracked actual keystrokes while subjects built spreadsheets. In 56 formulas written by these subjects, 12 had at least one extra keystroke compared to the optimal sequence. (Extra keystrokes are those beyond the minimum required for the purpose.) Although these studies reported on errors made during formula creation, not errors in finished spreadsheets, they suggest that the methods used by spreadsheet developers may have a significant impact on the efficiency with which formulas are created. Lerch also observed that mechanical errors increased when the cells being referenced were in different columns and rows, which suggests that a logical layout may reduce the incidence of errors.

Many papers have offered *advice* on how to build error-free spreadsheets, but very few studies provide concrete *evidence*. Brown and Gould [2] observed that participants in their study “did not spend a lot of time planning before launching into creating a spreadsheet.” That observation raises the question whether planning might reduce the incidence of errors.

Nardi and Miller [22] noted that in all the cases they studied, more than one person was involved in the creation or debugging of a spreadsheet. That observation raises the question of whether groups can develop models more successfully than individuals and what group mechanisms might contribute to success.

Janvrin and Morrison [15] carried out two experiments in which subjects were taught to use a structured design method (Data Flow Diagrams) when developing a spreadsheet. Subjects were given 1 h of instruction on spreadsheet development and then asked to develop a workbook with ten worksheets and 51 formulas requiring links among the worksheets. In the first study, subjects who used the structured design approach had a 7% error rate in linking formulas to inputs, whereas subjects who used an *ad hoc* approach had a 10% error rate. That work suggests that structured methods reduce errors, at least modestly. In a second study, conceptual and oversight errors were also observed, but although the structured approach still reduced the incidence of linking errors, it had no impact on the other types.

Kruck and Sheetz [18] and Kruck [17] first developed and then tested a theory of spreadsheet accuracy. In the first paper, they identified three beliefs or hypotheses about practices that reduce errors: reduce problem complexity, test for errors, and use an explicit design process. In the second paper, they tested whether providing information on these practices to student subjects reduces the number of errors committed. The information that was provided to subjects is summarized in Table 3 below. The experimental results show that all three aids improve performance: a control group committed 6.4

**Table 3**  
Aids for spreadsheet accuracy (after Kruck [17])

Planning and design aids
1. Use a planned layout so that movement of the data after entering will be minimized.
2. Use an organized layout to isolate data and computation areas.
3. Use descriptive labels.
4. Repeat the input data near the output.
Formula complexity aids
1. Formulas should contain only cell references.
2. Formulas can be used to repeat data in other locations.
3. Split complex formulas.
4. Relative and absolute cell addressing should be used where appropriate.
Test and debugging aids
1. Use Excel's auditing tools to verify formulas.
2. Review worksheet for error messages.
3. Determine if the numbers look reasonable.
4. Verify all calculations with a calculator or test with known models.

errors on average, but those given one of the aids committed only 4.2 to 4.6 errors on average. Finally, an expert was asked to evaluate each of these spreadsheets for ease of use and proper segmentation of inputs and outputs. Spreadsheets that rated best on these criteria had an average of 4 errors, while those rated worst had an average of 24 errors.

The literature does not tell us a great deal about how errors occur. A few laboratory experiments have observed developers making errors, but these experiments may not provide much insight into the causes of errors in *operational* spreadsheets. Most spreadsheet users create large numbers of errors as they enter data and formulas, but many of these are corrected immediately, and some are eventually corrected during testing or use. We still know very little about why the remaining errors escape detection.

Preventing errors in spreadsheets is a complex undertaking. Designing a serious research program to determine ways to improve the process would be equally complex. Research in this area can focus either on individual or organizational practices. At the individual level, laboratory research could establish whether practices such as use of explicit design processes or use of auditing software reduces errors. The same question could be studied in the field by surveying or observing developers, but the difficulty in deriving clean results would be substantial. At the organizational level, it would be helpful to know whether background education, in-house training, or spreadsheet standards have a measurable effect on spreadsheet errors.

## 6. Detection of errors

Two types of studies have examined detection of errors in completed spreadsheets. In laboratory experiments, subjects are asked to find errors placed in spreadsheets by the researcher; in field audits, experts try to find errors in operational spreadsheets.

In the great majority of laboratory studies, the subjects have been given no specific training or instruction on how to identify errors. Thus, these studies may have limited implications for detecting errors. One exception is Teo and Tan [40], who performed a two-part experiment in which subjects first built a spreadsheet from a written problem description and

then performed “what-if” analysis by changing some parameters. This experiment provided insight into the possibility that some errors can be detected during the use of a spreadsheet. Their results showed that only 13 of 70 subjects actually corrected errors in their spreadsheets during the second exercise. Of all mechanical errors committed in the first exercise, 39% were corrected in the second exercise. A somewhat smaller percentage, 30%, of logic and omission errors was corrected at the second stage. However, offsetting this improvement was an increase in the overall number of errors during the second exercise: 49% new mechanical errors, 30% new logic errors and 65% new omission errors. Therefore, it appears that during use, errors made during development can be corrected, but new errors can also be introduced in the process of modifying the spreadsheet.

Teo and Lee-Partridge [39] conducted a related set of experiments to test the dependence of error detection on the nature of the error, individual factors (such as expertise), and prior practice. Their subjects were not taught how to look for errors in spreadsheets. The results suggest that success in locating errors increases with practice, although more for logic and omission errors than for mechanical, jamming or duplication errors.

Panko and Halverson [27] conducted an experiment in which both individuals and groups examined a spreadsheet for errors. The groups found about two-thirds of all errors, while individuals found only one-third, suggesting that groups may be more effective than individuals in certain tasks related to spreadsheets.

Galletta et al. [11,12] performed two studies in which subjects searched for errors that had previously been seeded in spreadsheets. In the first study, about 55% of the errors were identified by subjects, and expertise in both accounting and spreadsheets contributed to success. In the second study, the subjects were shown either a printed copy or an electronic copy of the spreadsheet, with or without formulas. Again, about 50% of the errors were identified and subjects who could refer to the formulas did not find more errors than those who could only check numbers. Howe and Simkin [14] similarly found that experimental subjects found 67% of errors seeded in a spreadsheet. Their regression analysis failed to identify any significant determinants of error detection rates by error type.

Purser and Chadwick [32] administered web-based surveys to professionals and students asking them to identify errors in spreadsheets. Both groups were given a first survey in which no information was provided on the types of errors to look for and a second survey in which the error classification scheme discussed above was provided. The results showed experienced users detecting a much higher percentage of errors than novices, but knowledge of error types did not always improve performance. In fact, the number of qualitative errors identified went *up* as expected, but the number of quantitative errors identified went *down*.

In addition to laboratory experiments, a few studies have reported the results of auditing operational spreadsheets. Panko [24,25] cited seven reports on field audits. Most of these disclose no details on the procedure for identifying errors. One exception is Hicks (cited in Panko [25]), who reported on the audit of a large-scale capital budgeting spreadsheet at NYNEX. In this audit, each of the six main



modules of the spreadsheet was audited by a three-person team. The audit began with an explanation from the developer covering the module and its relation to the model as a whole. The team then verified formulas and checked cell references. One cell in each column was studied in detail and the others in the same row were checked for consistency. Test data were used to audit some portions of the module. Finally, Excel's formula auditing tool was used.

The auditing procedure used by HM Customs and Excise was described by Butler [3]. This procedure involves the use of a software tool (SpACE) created for government auditing of small-business tax returns. This auditing procedure has been documented in [13], which is the most explicit published auditing protocol available. The Customs and Excise procedure works as follows. First, the auditor identifies the chain of cells from inputs to end result and uses the software to follow the chain of dependent cells so that the key formulas can be checked. Then the auditor checks the original formulas that were used to copy related formulas, and checks that the copies are correct. Again the software is used to speed up this task. Finally, fourteen types of high-risk cells are checked for arithmetic and logical correctness. These include, for example, cells that contain constants, have no dependents, or involve complex functions such as NPV.

Clermont et al. [6] used specially-developed auditing software in a field audit of three large spreadsheets. The auditor first discussed each workbook with its developer and collected summary data about the workbook (number of cells, number of formulas, and so on). The second step was to spot-check the spreadsheet for errors. Finally, the software tool was run and highlighted cells were investigated.

Ideally, spreadsheets would be created free of errors, but it seems more practical to develop efficient methods and tools for detecting errors. Unfortunately, the literature provides little guidance for this effort. Few laboratory experiments have been performed that test the effectiveness of different approaches to error detection. Although several field audits have been done to identify errors, most have not reported on the methods used, and none has compared different approaches to the task.

Finally, we should keep in mind the complementary roles of prevention and detection. In terms of the end product, it may not matter whether a developer creates an error-free result by making a lot of mistakes and correcting them or by doing careful planning and making relatively few mistakes. Nevertheless, there is an unavoidable economic trade-off between prevention efforts and detection efforts. Perhaps further research into these topics might help identify the kinds of initiatives that could reduce errors at least cost.

Research that establishes effective methods for detecting errors obviously requires a well-tested taxonomy of errors. Laboratory experiments that compare the performance of subjects using different auditing procedures would contribute significantly to our understanding. Ideally, such experiments would use business analysts (not students) as subjects and operational spreadsheets as the objects of study. Field experiments would be more challenging but could offer deeper results. A natural experiment in which an organization or workgroup introduces new procedures for error detection would be an ideal research platform.

## 7. Research directions

Research on spreadsheet errors can be conducted either in the laboratory or in the field. Each type of research offers its own insights and has its own limitations.

In some ways, laboratory research is easier to conduct than field research, but its limitations are significant. In particular, error rates in laboratory experiments should not be used uncritically to infer error rates in operational spreadsheets because the underlying conditions differ. We don't yet know the impact of those differences on error rates. More research needs to be carried out in both domains before we can judge how the two error rates compare.

Laboratory experiments can be particularly useful in identifying the types of errors that occur, the stage of the modeling process at which they occur, and their causes. Laboratory experiments would also be helpful in comparing auditing procedures and auditing software.

It is perhaps remarkable that more laboratory experiments have not been performed on subjects creating spreadsheets. The ideal experiment would give a subject a written description of a problem to solve using a spreadsheet and then track the process through from start to finish. The method of verbal protocols [41] would seem to be ideal for this purpose, especially when used in conjunction with software that tracks keystrokes. A series of such experiments could determine the stage at which errors of different types are made (and discovered). They could also allow us to identify specific approaches to spreadsheet design, building, and testing that work best.

The great advantage of field research on operational spreadsheets is that we are dealing with the real thing, not an experimental substitute. Unfortunately, field research is generally difficult and it has its own limitations. Certainly, more work needs to be done in identifying the types and frequency of errors that occur in the field. To be most useful, such research must precisely specify error types and detection methods. It must also describe the spreadsheets that were audited: how they were selected, their size, complexity, and so on.

Field research is also needed on methods for detecting errors. Here are some of the questions research in this area could answer:

- Which procedures for error detection work best?
- How do teams and individuals compare at detecting errors?
- Can we identify high-risk cells before auditing them?
- Does auditing software improve error detection rates?
- To what extent do the results of laboratory experiments on error detection apply to error detection in the field?

Field research can also uncover other useful information about spreadsheet use. For example, what best practices are used by developers whose spreadsheets are relatively error-free? What organizational policies and norms exist in organizations with good spreadsheets? And what can we learn from the sociology of spreadsheet use; that is, how do individuals and groups come together to solve problems using spreadsheet technology?

In summary, errors in spreadsheets appear to be frequent and are potentially costly. We have reviewed the research

literature on spreadsheet errors, focusing on five topics, and drawn the following conclusions:

- Classification: Taxonomies exist, but in most cases they are context-dependent, imprecisely defined, and untested.
- Impact: Little is known about the quantitative (economic) consequences of errors in spreadsheets.
- Frequency: If we want to estimate cell error rates accurately, we must standardize the procedures for doing so.
- Creation and prevention: Laboratory experiments have been inconclusive or suggestive, but broad progress has been limited.
- Detection: Serious research, detailing the methods used and comparing alternative approaches, is lacking on this subject.

The state of the art allows us to outline several promising directions for future research. Laboratory research can better identify the types of errors that occur, when and how. It can also be used to compare auditing procedures and auditing software. Two kinds of field research are needed: narrow research into errors and their frequency, and broad research into best practices and the sociology of spreadsheet use.

## References

- [1] Y. Ayalew, M. Clermont, R. Mittermier, Detecting errors in spreadsheets, Proceedings of the European Spreadsheet Risks Interest Group Annual Conference University of Greenwich, London, 2000, pp. 51–62.
- [2] P. Brown, J. Gould, An experimental study of people creating spreadsheets, *ACM Transactions on Office Information Systems* 5 (1987) 258–272.
- [3] R. Butler, Is this spreadsheet a tax evader? Proceedings of the 33rd Hawaii International Conference on System Sciences, 2000, pp. 1–6.
- [4] R. Butler, Risk assessment for spreadsheet developments: choosing which models to audit, H. M. Customs and Excise UK (2000).
- [5] J. Caulkins, E. Morrison, T. Weidemann, Spreadsheet errors and decision making: evidence from field interviews, *Journal of End User Computing* (to appear) (2006).
- [6] M. Clermont, A spreadsheet auditing tool evaluated in an industrial context, Proceedings of the European Spreadsheet Risks Interest Group Conference Cardiff Wales, 2002, pp. 35–46.
- [7] P. Cragg, M. King, Spreadsheet modelling abuse: an opportunity for OR? *Journal of Operational Research Society* 44 (8) (1993) 743–752.
- [8] N. Davies, C. Ikin, Auditing spreadsheets, *Australian Accountant* 57 (11) (1987) 54–56.
- [9] S. Ditlea, Spreadsheets can be hazardous to your health, *Personal Computing* (1987) 60–69.
- [10] D. Freeman, How to make spreadsheets error-proof, *Journal of Accountancy* 181 (5) (1996) 75–77.
- [11] F. Galletta, D. Abraham, M. El Louadi, W. Leske, Y. Pollalis, J. Sampler, An empirical study of spreadsheet error-finding performance, *Accounting, Management & Information Technology* 3 (2) (1993) 79–95.
- [12] F. Galletta, K. Hartzel, S. Johnson, J. Joseph, S. Rustagi, Spreadsheet presentation and error detection: an experimental study, *Journal of Management Information Systems* 13 (3) (1997) 45–63.
- [13] H. M., Customs and excise computer audit service, *Methodology for the audit of spreadsheet models*, 2001.
- [14] H. Howe, M.G. Simkin, Factors affecting the ability to detect spreadsheet errors, *Decision Sciences Journal of Innovative Education* 4 (1) (2006) 101–122.
- [15] D. Janvrin, J. Morrison, Factors influencing risks and outcomes in end-user development, Proceedings of the 29th Annual Hawaii International Conference on System Sciences, 1996, pp. 349–355.
- [16] J. Kreie, T. Cronin, J. Pendley, J. Renwick, Applications development by end-users: can quality be improved? *Decision Support Systems*, 29 (2000) 143–152.
- [17] S. Kruck, Testing spreadsheet accuracy theory, *Information and Software Technology* 48 (2006) 204–213.
- [18] S. Kruck, S. Sheetz, Spreadsheet accuracy theory, *Journal of Information Systems Education* 12 (2001) 93–108.
- [19] R. Lawrence, J. Lee, Financial modelling of project financing transactions, *Institute of Actuaries of Australia Financial Services* 19 (2004).
- [20] F. Lerch, Computerized financial planning: discovering cognitive difficulties in knowledge building, Unpublished Ph.D. dissertation, University of Michigan, Ann Arbor, MI (1988).
- [21] T. McGill, J. Klobas, The role of spreadsheet knowledge in user-developed application success, *Decision Support Systems* 39 (2005) 355–369.
- [22] B. Nardi, J. Miller, Twinkling lights and nested loops: distributed problem solving and spreadsheet development, *International Journal of Man-Machine Studies* 34 (1991) 161–184.
- [23] J. Olson, J.E. Nilsen, Analysis of the cognition involved in spreadsheet software interaction, *Human-Computer Interaction* 3 (4) (1987–88) 309–349.
- [24] R. Panko, What we know about spreadsheet errors, *Journal of End-User Computing* 10 (1998) 15–21.
- [25] R. Panko, What we know about spreadsheet errors, 2006 <http://panko.cba.hawaii.edu/ssr/Mypapers/whatknow.htm>, accessed September 2.
- [26] R. Panko, R. Halverson, Spreadsheets on trial: a survey of research on spreadsheet risks, Proceedings of the 29th Annual Hawaii International Conference on Systems Sciences, 1996, pp. 326–335.
- [27] R. Panko, R. Halverson, Are two heads better than one (at reducing errors in spreadsheet modeling?), *Office Systems Research Journal* 15 (1) (1997) 21–32.
- [28] R. Panko, R. Halverson, Two corpuses of spreadsheet errors, Proceedings of the 33rd Annual Hawaii International Conference on Systems Sciences, 2000, pp. 1–8.
- [29] R. Panko, R. Sprague, Hitting the wall: errors in developing and code inspecting a 'simple' spreadsheet model, *Decision Support Systems* 22 (1998) 337–353.
- [30] S. Powell, K. Baker, B. Lawson, Errors in operational spreadsheets, *Spreadsheet Engineering Research Project working paper*, 2006.
- [31] S. Powell, K. Baker, B. Lawson, An auditing protocol for spreadsheet models, *Spreadsheet Engineering Research Project working paper*, 2006.
- [32] M. Purser, D. Chadwick, Does an awareness of differing types of spreadsheet errors aid end-users in identifying spreadsheet errors? Proceedings of the European Spreadsheet Risk Interest Group Annual Conference, Cambridge, UK, 2006, pp. 185–204.
- [33] K. Rajalingham, D. Chadwick, B. Knight, Classification of spreadsheet errors, Proceedings of the European Spreadsheet Risks Interest Group Annual Conference, Greenwich, England, 2000, pp. 23–34.
- [34] J. Reason, *Human Error*, Cambridge University Press, Cambridge, UK, 1990.
- [35] T. Reinhardt, N. Pillay, Analysis of spreadsheet errors made by computer literacy students, Proceedings of the IEEE International Conference on Advanced Learning Technologies, 2004, pp. 852–853.
- [36] B. Ronen, M. Palley, H. Lucas, Spreadsheet analysis and design, *Communications of the ACM* 32 (1) (1989) 84–93.
- [37] P. Saariluoma, J. Sajaniemi, Transforming verbal descriptions into mathematical formulas in spreadsheet calculation, *International Journal of Human-Computer Studies* 41 (1994) 915–948.
- [38] K. Sommerville, *Software Engineering*, 7th Edition Addison Wesley, 2004.
- [39] T. Teo, J. Lee-Partridge, Effects of error factors and prior incremental practice on spreadsheet error detection: an experimental study, *Omega – The International Journal of Management Science* 29 (2001) 445–456.
- [40] T. Teo, M. Tan, Quantitative and qualitative errors in spreadsheet development, Proceedings of the 30th Hawaii International Conference on Systems Sciences, 1997, pp. 149–155.
- [41] M. Van Someren, Y. Barnard, J. Sandberg, *The Think Aloud Method: A Practical Guide to Modeling Cognitive Processes*, Academic Press, New York, 1994.

**Steve Powell** is a Professor at the Tuck School of Business at Dartmouth. His primary research interest lies in modeling production and services processes, but he has also been active in research in energy economics, marketing, and operations. At Tuck, he has developed a variety of courses in management science, including the core Decision Science course and electives in the Art of Modeling, Business Process Redesign, and Applications of Simulation. He originated the Teacher's Forum column in *Interfaces*, and has written a number of articles on teaching modeling to practitioners. He is the academic director of the INFORMS Annual Teaching of Management Science Workshop. In 2001 he was awarded the INFORMS Prize for the Teaching of Operations Research/Management Science Practice. He is the co-author with Kenneth Baker of *The Art of Modeling with Spreadsheets* (Wiley, 2004).

**Ken Baker** is a faculty member at Dartmouth College. He is currently Nathaniel Leverone Professor of Management at the Tuck School of Business and also adjunct professor at the Thayer School of Engineering. At Dartmouth, he has taught courses relating to decision science, manufacturing management, and environmental management. Over the years, much of his teaching and research has dealt with production planning and control, and he is widely known for his textbook *Elements of Sequencing and Scheduling*, in addition to a variety of technical articles. He has served as the Tuck School's associate dean and directed the Tuck School's management development programs in the manufacturing area. In 2001 he was named a Fellow of INFORMS' Manufacturing and Service Operations Management (MSOM) Society, and in 2004 a Fellow of INFORMS. He is the co-author with Stephen Powell of *The Art of Modeling with Spreadsheets* (Wiley, 2004).

**Barry Lawson** is a research associate at the Tuck School of Business at Dartmouth and is also a visiting scholar in the geography department of the college. He founded and has served as president of Barry Lawson Associates, a consulting firm, since 1978. As visiting scholar, he coordinates the development of an atlas of the upper Connecticut River Watershed in New Hampshire and Vermont. As research associate at Tuck he serves as the program manager for the Tuck Spreadsheet Engineering Research Project. Lawson has taught in graduate programs at Boston University and Wayne State University as well as in short courses at Bentley College. He has moderated a host of public hearings for local, state and federal governments on controversial environmental and energy-and waste-related projects, and has considerable experience in group facilitation, conflict resolution and simulation design.