# Two Corpuses of Spreadsheet Errors

Raymond R. Panko
*University of Hawaii*
*panko@hawaii.edu, Ray@panko.com*

**Abstract**

*The widespread presence of errors in spreadsheets is now well-established. Quite a few methodological and software approaches have been suggested as ways to reduce spreadsheet errors. However, these approaches are always tailored to particular types of errors. Are such errors, in fact, widespread? A tool that focuses on rare errors is not very appealing. In other fields of error analysis, especially linguistics, it has proven useful to collect corpuses (systematic samples) of errors. This paper presents two corpuses of errors seen in spreadsheet experiments. Hopefully, these corpuses will help us assess the claims of spreadsheet reduction approaches and should guide theory creation and testing.*

## 1. Introduction

Spreadsheets are widely used in business. Even in the 1980s, a large majority of all managers used spreadsheets (Panko, 1988). Today, use by managers probably is nearly universal. Although some spreadsheets are small throw-away scratch pad calculations, of these spreadsheets are very large (Floyd, Walls, & Marr, 1995; Hall, 1996) and complex (Hall, 1996). Some guide critical corporate decisions. Spreadsheeting is *not* a narrow topic.

Alexander Pope wrote that "To err is human." Although he was writing about moral error (sin), what he said was equally true of cognitive error. Today, we can even quantify his statement. Error research in a number of fields has shown that there is almost nothing that human beings can do a thousand times in a row without making an undetected error (Panko, 2000a). In fact, it is common to have undetected errors in about 0.2% to 0.5% of all simple actions, such as typing keystrokes, and in about 2% to 5% of all more complex human cognitive activities, such as writing lines of computer code (Panko, 2000a).

Fortunately for theory, but unfortunately for practice, comparable error rates have been seen in spreadsheet experiments and field audits. The Spreadsheet Research (Panko, 2000b) website has data from over a dozen experiments involving more than a thousand subjects ranging from rank novices to experienced professionals. In all of these experiments, at least 1% of all cells contained errors. In addition, three cell-by-cell field audits of real world spreadsheets have found errors in the 1% to 3% range (Hicks, 1995). Other field audits, which did not inspect all cells, found smaller error rates, but every field audit found material error rates in at least 10% of all spreadsheets audited.

One counter-argument is that despite overwhelming data, spreadsheet errors cannot be this widespread, or we would know about it. In fact, few spreadsheets are ever audited in detail after creation (Cragg & King, 1993; Hall, 1996), and while phenomenally bad results probably would be detected, damaging errors can still exist without being detected. In one case, an incorrect budgeting spreadsheet that produced substantial damage to an organization was proven to be faulty only after two years of use. In each year, the incorrect results were explained away in terms of changes in the environment. In other cases, promising investments are never made, so there is again no feedback. Quite simply, while "eyeballing" numbers for accuracy is a good idea, humans are not very good at seeing even substantial errors (Rickets, 1990).

## 2. Reducing Errors

Several ideas have been put forward for reducing spreadsheet errors. They seem to fall into three categories.

### 2.1 Development Methods.

Several methods have been created for systematic spreadsheet development. Typically, these methods borrow from classical systems analysis methods developed for software development but are modified to deal with the specifics of spreadsheet development, end user development, or both.

### 2.2 Development Tools.

Development tools are created to avoid certain errors, such as data type mismatches.

Domain-specific software tools further restrict development to a particular problem domain, such as producing pro forma income statements. This may help avoid the omission of important variables and may implement best practices in certain areas, such as the creation of cash flow analyses.

### 2.3 Inspection Tools.

Inspection tools are created to help code inspectors detect errors in spreadsheets. Code inspection has long been known as the best way to reduce errors in programming, although not all errors are discovered in inspections (Fagan, 1976). Similar results have been found in spreadsheet code inspections (Galletta, et al., 1993, 1997; Panko, forthcoming).

Some inspection tools work by trying to make the logic of the spreadsheet more visible. Others are like grammar checkers, automatically examining the spreadsheet for suspicious patterns.

### 2.4 Are Tools Safe and Effective?

Although all of these methods are promising, all would be expensive to implement, and we would like some indication that they are "both safe and effective," to borrow a catch phrase from the pharmaceuticals industry.

This is a reasonable concern because each technique tends to focus on certain types of errors. If these errors are not very frequent, then the technique will not be very effective and may even divert resources from more promising approaches. Like Yosarian in Catch 22, we will be treating a minor injury while our patient bleeds to death from more serious problems.

## 3. Error Corpuses

In error research, especially in linguistics and automobile accidents, it is common to collect corpuses (systematic samples) of actual errors. One of the most notable of these linguistic error corpuses was created by Nooteboom (1980).

Corpuses have two purposes. The first is to guide and critique error theory. Error corpuses are important in suggesting new theories because the patterns of errors in the corpus may suggest specific error mechanisms. In addition, error corpuses are important in evaluating proposed theories. If a theory cannot explain known patterns, it is not a very good theory.

The second purpose is to guide our adoption of various proposed error reduction approaches. As noted above, each approach tends to focus on the avoidance or detection of particular types of errors. If an approach focuses on errors that are not important in practice, it probably will not be very effective.

## Two Corpuses

This paper presents two error corpuses from spreadsheet development experiments. Of course, Laboratory experiments are never perfect mirrors for operational environments, but the number of errors detected and reported in code inspections of operational spreadsheets have been too few to analyze. In addition, experiments usually have known solutions. This allows *all* errors to be detected. In contrast, corpuses of errors found in inspections only reflect *detected* errors. The author (forthcoming) has shown that certain types of errors are more likely to be detected in cell-by-cell code inspections than other types of errors.

### 3.1 The Galumpke Corpus

The **Galumpke Corpus** in Appendix A comes from an experiment using 33 undergraduate students and 49 graduate students. Among the undergraduates, accounting and finance majors were excluded, so the 33 subjects for which data are reported were from other majors, especially management information systems. This experiment yielded a corpus of 68 errors.

In the Galumpke corpus, we omitted errors made in a part of the task that required subjects to compute depreciation from a capital purchase and to use depreciation as an expense. Many subjects, especially undergraduate subjects, were unable to do this subtask.

The following is the wording for the Galumpke task. The requires considerable accounting knowledge, and accounting majors were excluded from the analysis because of their specialized domain knowledge.

> The company sells galumpkes, which are small food warmers used in restaurants. The owner will draw a salary of $80,000 per year. There is also a manager of operations, who will draw a salary of $60,000 per year. The income tax rate is expected to be 25% in each of the two years. Each

> galumpke will require $40 in materials costs and $25 in labor costs in the first year. These numbers are expected to change to $35 and $29 in the second year. There will be a capital purchase of $500,000 in the first year. For depreciation, assume 10% straight-line depreciation with no scrap value. Unit sales price is expected to be $200 in the first year and $180 in the second year. There will be three sales people. Their average salary per person is expected to be $30,000 in the first year and $31,000 in the second. The rent will be $3000 per month. The company expects to sell 3,000 galumpkes in the first year. In the second, it expects to sell 3,200. Please note the finish time.

### 3.1 The Wall Corpus

The **Wall Corpus** in Appendix B comes from an experiment using 101 undergraduate students and 49 graduate students. This experiment yielded a corpus of 62 errors.

The following is the wording for the Wall Task. The task was designed to be rather simple and not to require specialized knowledge.

> You are to build a spreadsheet model to help you create a bid to build a wall. You will offer two options -- lava rock or brick. Both walls will be built by crews of two. Crews will work three 8-hour days to build either type of wall. The wall will be 20 feet long, 6 feet tall, and 2 feet thick. Wages will be $10 per hour. You will have to add 20% to wages to cover fringe benefits. Lava rock will cost $3 per cubic foot. Brick will cost $2 per cubic foot. Your bid must add a profit margin of 30% to your expected cost.

## 4. Classification of Errors

### 4.1 Quantitative versus Qualitative Errors

Following Panko and Halverson (1997), only **quantitative** errors are reported. These are errors that lead to an incorrect bottom-line value. **Qualitative** errors—design errors and other problems that may lead to quantitative errors in the future—are not reported.

In addition, quantitative errors were further divided into three types, again following Panko and Halverson (1997):

### 4.2 Omission Errors

Omission errors are facts required to be put into the model by the developer but that are omitted.

### 4.3 Logic Errors

Logic errors result from the developer having the incorrect calculation algorithm for some problem chunk or having the correct algorithm and implementing it incorrectly.

### 4.4 Mechanical errors

Mechanical errors are simple slips, such as typing the wrong number in a numerical cell, typing the wrong number or operation in a formula cell, or pointing to the wrong cell or range while entering a formula with the pointing method.

## 5. Analysis of the Corpuses

### 5.1 Purpose

The purpose of this paper is to present the two error corpuses rather than to discuss them in detail. However, we do wish to make some comments about general patterns in the corpus.

### 5.2 Diversity of Errors

These two corpuses show that errors are very diverse. First, there are many unique errors (error types). Second, the ratio of **tokens** (individual errors) to the number of unique errors is very low, indicating that error-making is extremely diverse. Although a few errors were made several times, errors made only one to three times dominated the corpuses. Catching a single errors will not produce safe spreadsheets.

The situation does not change when data are aggregated into the logic, omission, and mechanical error categories. All three types of errors are common.

Overall, the situation is similar to poisoning someone with multiple poisons. Eliminating one poison will not save the person. Specifically, eliminating all logic errors, omission errors, or mechanical errors without eliminating other errors will still produce an unacceptably large number of errors.

This suggests that error-reduction tools aimed at finding certain types of errors will at best be effective only if they are combined into a broader package of error reduction tools.

### 5.3 The Externality of Errors

Many tools examine the internal logic of the spreadsheet after it has been created. However, the majority of the errors in the two corpuses are made outside the context of the spreadsheet. We call this occurrence of errors outside the context of the spreadsheet "externality."

For instance, no internal inspection of a spreadsheet's internal logic is likely to catch most of the omission errors shown in the corpuses. If the developer leaves out the payment of income taxes, nothing in the structure of the spreadsheet is likely to indicate this to someone unfamiliar with accounting.

Most logic errors, furthermore, would be impossible to catch because the calculations are structurally correct although based on the wrong algorithm or that implement the algorithm incorrectly.

While some mechanical errors are pointing errors that might be detectable based on structural analysis of the spreadsheet, most appear to be due to misreading the facts of the problem (a trend we will return to later). Although some screening of input could detect numbers out of range, the variety of input errors suggests that this would be very difficult.

### 5.4 Translation versus Typing Errors

One surprising result was the surprisingly low number of errors that appear to be due to simple typing errors. Most people assume that typing errors will be very common, and past research on typing error frequencies would support that expectation (Panko, 2000a).

However, most errors that enter the wrong value appear to be what we will call translation errors, in which the developer transposes two numbers or misreads the number. For instance, in the Galumpke task, the unit costs for labor and materials are often interchanged.

In contrast, the number of pure typing errors, where the wrong value is entered and where the error does not look like some type of substitution error, are very small. Given the fact that the Galumpke task spreadsheets were about 40 cells and that Wall task spreadsheets were about half as large, the number of typing errors appears to be one or two orders of magnitude lower than typing research would predict (Panko, 2000a).

### 5.5 Consistency with Field Audits of Spreadsheet and Error Frequencies in Other Human Cognitive Domains

In terms of error frequency, the error rates shown in the two corpuses indicate that errors were made in about 1% to 3% of all cells. (Errors are only counted the first time they occur.) Similar error rates have been found in cell-by cell code inspections of the few operational spreadsheets that have been subjected to multiperson cell-by cell audits (Hicks, 1995) and by comparing output with that from a well-tested financial modeling program (Lukasic, 1998). More importantly, this error rate is consistent with error rates found in other human cognitive domains, including programming (Panko, 2000a). In other words, in frequency of occurrence, spreadsheet errors look like errors in other human cognitive domains. While this is depressing from the point of view of practice, it is good from a theoretical viewpoint. Quite simply, spreadsheet errors look like other errors.

### 5.6 Strong but Wrong Error

In the Galumpke task, one of the most common errors was not to multiply the monthly rent by twelve to put it on an annual basis so that it would be consistent with other variables.

Conceptually, it appears that the developers were conditioned to think of the information in annual terms because all other data were annual. This appears to be what Reason (1990) called a **strong but wrong error**. Our automatic cognitive subsystem, which does much of our daily cognitive work, saves us a great deal of cognitive energy and (literally) discomfort by working primarily through pattern matching. Just as we may mean to stop by a store on our way home from work yet drive past the store as we do every other day, when confronted by data in a slightly different format, we may ignore the specific difference.

### 5.7 Overload Errors

Baars (1992) notes that in linguistics, we can reliably produce errors in the laboratory and more specifically can reliably produce certain types of errors.

Again, our automatic cognitive subsystem appears to be the culprit. Cognitively, creating a sentence is an extremely complex process. We must balance very complex rules of grammar as well as develop a sentence that reflects what we wish to say. Research sited by Baars (1992) indicates that we appear to create many competing sentences automatically and automatically prune unwanted sentences as we move to develop the final sentence.

If we place an experimental subject under time pressure or give them distractions, the creation process frequently breaks down. In addition, when it breaks down, it tends to create a **blend error** that contains apparent elements of two competing sentences. Analysis can even give a good indicator of what stage in sentence development was underway when the blending occurred. Blend errors tend to be linguistically lawful at all stages except one.

In addition, when we write, we have a difficult task to perform. While worrying about individual sentences, we are also under pressure to make each sentence fit into the complex overall plan (Flower & Hays, 1980). This increases our overload. Presumably, in spreadsheet development, we have a similar dynamic as a developer tries to create cell contents consistent with the overall design.

Furthermore, we have very limited short-term memory. If we have to keep in mind too much information, we are likely to reach the state of overload that tends to lead to blend errors, especially if we are engaging in a cognitively complex task such as spreadsheet development, which requires us to keep a complex overall design plan in mind. This is a good recipe for omitting a fact from the final spreadsheet.

In the Wall task, there is a complex requirement that we have the following complex element in our task statement:

> Both walls will be built by crews of two. Crews will work three 8-hour days to build either type of wall.

Note, in the Wall corpus, that there were several omission errors. The most frequent was omitting the fact that each crew had two members—a fact listed in the first (separate) sentence. Subjects also omitted the fact that there would be three days per wall project. In another case, the fact that there were two team members was included twice, in different parts of the cell formula, indicating a possible blend error between two competing formulas being developed.

In the Galumpke corpus, note that there are several errors in the unit prices for labor and materials. These errors may be due to the complexity of the portion of the task statement giving this information. In addition, the wording violates the normal pattern of developing spreadsheets one row at a time.

> Each galumpke will require $40 in materials costs and $25 in labor costs in the first year. These numbers are expected to change to $35 and $29 in the second year.

In general, the injunction to "keep it simple, stupid" appears to be correct in the creation of spreadsheet models. We need to reduce developer overload as much as possible.

Keeping things simple should also aid in code inspection. The author (forthcoming) found that errors were detected much more frequently in short formulas than in long formulas.

## 6. Conclusion

The purpose of this paper is to introduce two spreadsheet error corpuses. Borrowing an approach from linguistics, these corpuses should be useful in suggesting new theories and in testing the validity of spreadsheet error theories.

In addition, when techniques and software to reduce errors are being suggested, it should be useful to compare them with the errors found in these two corpuses to see how many of these errors the techniques or software in fact are likely to be able to find.

Our limited analysis continues a basic theme of all spreadsheet research, namely that spreadsheet errors look a great deal like errors in programming and other human cognitive activities. While this is comforting from the point of view of theory, the general finding in the error literature that errors are very difficult to reduce should caution us that half-measures are not likely to avoid a substantial frequency of error in spreadsheet development. Only cell-by-cell team code inspection is likely to reduce errors by 80%, and some errors will still get through the logical sieve of cell-by-cell team code inspection.

## 8. References

Baars, B. J. (1992). A New Ideomotor Theory of Voluntary Control. In B. J. Baars (Ed.), *Experimental Slips and Human Error* (pp. 93-120). New York: Plenum.

Cragg, P. G., & King, M. (1993). Spreadsheet Modelling Abuse: An Opportunity for OR? *Journal of the Operational Research Society*, 44(8), 743-752.

Fagan, M. E. (1976). Design and Code Inspections to Reduce Errors in Program Development. *IBM Systems Journal, 15*(3), 182-211.

Flower, L. A., & Hayes, J. R. (1980). The Dynamics of Composing: Making Plans and Juggling Constraints. In L. W. Gregg & E. R. Steinberg

(Eds.), *Cognitive Processes in Writing* (pp. 31-50). Hillsdale, NJ: Lawrence Erlbaum Associates.

Floyd, B. D., Walls, J., & Marr, K. (1995). Managing Spreadsheet Model Development. *Journal of Systems Management, 46*(1), 38-43, 68.

Galletta, D. F., Abraham, D., El Louadi, M., Lekse, W., Pollailis, Y. A., & Sampler, J. L. (1993). An Empirical Study of Spreadsheet Error Performance. *Journal of Accounting, Management, and Information Technology, 3*(2), 79-95.

Galletta, D. F., Hartzel, K. S., Johnson, S., & Joseph, J. L. (1997). Spreadsheet Presentation and Error Detection: An Experimental Study. *Journal of Management Information Systems, 13*(2), 45-63.

Hall, M. J. J. (1996, January). A Risk and Control Oriented Study of the Practices of Spreadsheet Application Developers. *Proceedings of the Twenty-Ninth Hawaii International Conference on System Sciences*, Vol. II, Kihei, Maui, 364-373.

Hicks, L., NYNEX, personal communication with the author via electronic mail, June 21, 1995.

Lukasik, Todd, CPS. Personal communication with the author by e-mail, August 10, 1998.

Nooteboom, S. G. “Speaking and Unspeaking: Detection and Correction of Phonological and Lexical Errors in Spontaneous Speech,” in V. Fromkin (Ed.), *Errors in Linguistic Performance*, New York, Academic Press, New York, 1980. Cited in Reason, 1990.

Panko, Raymond R. "Applying Code Inspection to Spreadsheet Testing," *Journal of Management Information Systems*, forthcoming.

Panko, R. R. (2000a). *A Human Error Website* (http://panko.cba.hawaii.edu/humanerr/), Honolulu, HI: University of Hawaii.

Panko, R. R. (2000b). *Spreadsheet Research (SSR) Website* (http://panko.cba.hawaii.edu/ssr/). Honolulu, Hawaii: University of Hawaii.

Panko R. R. *End User Computing: Management, Applications, and Technology,* Wiley, New York, 1988.

Panko, R. R., and Halverson, R. P., Jr. “ Are Two Heads Better than One? (At Reducing Errors in Spreadsheet Modeling?” *Office Systems Research Journal* (15:1), Spring 1997, pp. 21-32.

Reason, J. T. (1990). *Human Error*. Cambridge, UK: Cambridge University Press.

Rickets, J. A. “Powers-of-Ten Information Biases,” *MIS Quarterly* (14:1) March 1990, pp. 63-77.

## Appendix A: Galumpke Error Corpus

| **Eureka Logic Errors** | **Undergrad** | **MBAs** | **Total** | **Total%** |
|---|---|---|---|---|
| Monthly rent not multiplied by 12 | 7 | 6 | 13 | 19% |
| Tax rate applied to salaries | 5 | 1 | 6 | 9% |
| Materials/ labor cost not multiplied by unit sales | 3 | 0 | 3 | 4% |
| Unit Materials times Price not Units | 0 | 1 | 1 | 1% |
| Multiplied Unit costs by Price, not Units | 0 | 1 | 1 | 1% |
| Owner salary subtract after taxes | 1 | 0 | 1 | 1% |
| Year 1 profit/ loss carried over to year 2 | 1 | 0 | 1 | 1% |
| *Subtotal* | *17* | *9* | *26* | *38%* |

| **Omission Errors** | **Undergrad** | **MBAs** | **Total** | **Total%** |
|---|---|---|---|---|
| Tax rate not applied | 4 | 2 | 6 | 9% |
| Omitted both owners and managers salaries | 2 | 1 | 3 | 4% |
| Omitted Variable Costs | 2 | 1 | 3 | 4% |
| Owners salary not included | 0 | 1 | 1 | 1% |
| No Owner, Manager Salary in Year 2 | 0 | 1 | 1 | 1% |
| *Subtotal* | *8* | *6* | *14* | *21%* |

| **Mechanical Errors** | **Undergrad** | **MBAs** | **Total** | **Total%** |
|---|---|---|---|---|
| Pointing errors | 7 | 1 | 8 | 12% |
| Years 1 & 2 sales salaries ==> salesman 1 & 2 | 1 | 0 | 1 | 1% |
| Owner salary = $60,000 | 1 | 0 | 1 | 1% |
| Materials cost = $35 in Yr1 | 0 | 1 | 1 | 1% |
| Materials cost = $25 for year 2 | 1 | 1 | 2 | 3% |
| Materials cost = $32 for year 2 | 1 | 0 | 1 | 1% |
| Materials cost = $40 for year 2 | 0 | 1 | 1 | 1% |
| Labor cost = $35 for year 2 | 1 | 0 | 1 | 1% |
| Labor cost = $39 instead of $29 | 0 | 1 | 1 | 1% |
| Cost Matrix Switch | 0 | 2 | 2 | 3% |
| Unit price = $200 for year 1 | 1 | 0 | 1 | 1% |
| Unit price = $100 for year 1 | 1 | 0 | 1 | 1% |
| Units Sold 3,100 Year 2 | 0 | 2 | 2 | 3% |
| Units Sold 32,000 | 0 | 1 | 1 | 1% |
| Taxes added - sign error | 1 | 0 | 1 | 1% |
| Manufacturing costs added - sign error | 1 | 0 | 1 | 1% |
| Parentheses error in manufacturing costs | 1 | 0 | 1 | 1% |
| Rent = $3,600 | 1 | 0 | 1 | 1% |
| *Subtotal* | *18* | *10* | *28* | *41%* |

| | | | | |
|---|---|---|---|---|
| **Total Errors** | **43** | **25** | **68** | **100%** |
| **Number of Subjects** | **33** | **49** | **82** | |

## Appendix B: Wall Error Corpus

| **Logic Errors** | **Undergrad** | **Grad** | **Total** | **Total%** |
|---|---|---|---|---|
| Profit margin on materials only | 4 | 2 | 6 | 10% |
| For profit margin, divided by .7 | 3 | 2 | 5 | 8% |
| Profit margin *.3/.7 | 1 | 1 | 2 | 3% |
| Multiplied each dimension by price | 1 | 1 | 2 | 3% |
| Used materials cost as total cost | 1 | 0 | 1 | 2% |
| Based fringe benefits on hourly rate | 0 | 1 | 1 | 2% |
| Labor w fringe - labor * .2, not 1.2 | 1 | 0 | 1 | 2% |
| Added brick and lava into a total | 1 | 0 | 1 | 2% |
| Multiplied twice by crew size of 2 | 1 | 0 | 1 | 2% |
| Multiplied price times volume by 2 | 1 | 0 | 1 | 2% |
| Doubled expected cost instead of adding profit margin | 1 | 0 | 1 | 2% |
| Added 20% to cost instead of labor | 1 | 0 | 1 | 2% |
| Figured benefit at 10% | 1 | 0 | 1 | 2% |
| Volume divided by cost per cubic foot | 0 | 1 | 1 | 2% |
| For profit margin, *1.33, not 1.3 | 0 | 1 | 1 | 2% |
| In volume, multipled by 1 ft=2/3 | 1 | 0 | 1 | 2% |
| *Total Logic Errors* | *18* | *9* | *27* | *44%* |
| | | | | |
| **Omission Errors** | **Undergrad** | **Grad** | **Total** | **Total%** |
| Omitted 2 workers in labor calculation | 15 | 5 | 20 | 32% |
| Omitted 30% profit margin | 3 | 1 | 4 | 6% |
| Omitted fringe benefits | 3 | 1 | 4 | 6% |
| Omitted 3 days in labor | 3 | 1 | 4 | 6% |
| Omitted wage rate in labor calculation | 0 | 1 | 1 | 2% |
| *Total Omission Errors* | *24* | *9* | *33* | *53%* |
| | | | | |
| **Mechanical Errors** | **Undergrad** | **Grad** | **Total** | **Total%** |
| Pointing error--added wrong 3 cells | 0 | 1 | 1 | 2% |
| Used 11% as fringe benefit--misread | 1 | 0 | 1 | 2% |
| *Total Mechanical Errors* | *1* | *1* | *2* | *3%* |
| | | | | |
| **Total Errors** | **43** | **19** | **62** | **100%** |
| **Number of Subjects** | **101** | **49** | **150** | |