

ISSUE 40 | JULY 2011

SOCIETY OF ACTUARIES


Technology
Section

CompAct

ELECTRONIC NEWSLETTER



CONTENTS

[Table of Contents](#)[Letter from the Chair](#)[Editor's Notes](#)[Excel Formula Rogue's
Gallery](#)[The Microsoft Excel
OFFSET Function](#)[Apps for Actuaries](#)[In Praise of
Approximations](#)[Apps, Actuaries and
Other Technology at the
SEAC and Beyond](#)[Number Puzzle](#)[Solution](#) [Articles Needed](#)[SOA News Today Has a
New Look! Improved
Navigation](#) 

QUICK LINKS

[Technology Section](#)

LinkedIn

Forward
to Colleague

Print

Search
Back Issues

EXCEL FORMULA ROGUE'S GALLERY

by Mary Pat Campbell

Enough of showing people examples of good, better or best spreadsheet practices—let's wallow in the muck of the usual mess we have to untangle! (Some of which we have caused ourselves.)

With the help of the denizens of the Actuarial Outpost, I bring to you the inaugural edition of the Excel Rogue's Gallery—with special thanks (or groans) to Outpost users Gedankenexperiment, ADoubleDot, SpaceActuary, dumplings, and Rake.

Let us start out with a relatively mild example:

EXAMPLE 1:

```
=VLOOKUP(R2,$A$2:$R$84,MATCH(S2,OFFSET($B$1:$F$1,MATCH
```

What's so bad about this?

Well, first, can you tell what this is supposed to be doing without sitting down and working it out explicitly? Would you know if it was screwing up? There are three different lookups being nested in each other, and one of the lookups isn't even looking for an exact match, and is assuming a particular ordering in the range that will be returned from the OFFSET function ... I hope that's all set up correctly.

Also, the "+ 6" at the end is a nice touch, don't you think?

[Web site](#)[Council](#)[Links of Interest](#)[Fiction Contest](#)[Howard Callif, Editor](#)**SOA Staff**[Meg Weber, Staff Partner](#)[Sue Martz,](#)[Section Specialist](#)[Sam Phillips, Staff Editor](#)

One of the main problems with the above is having a nesting of functions that goes too deep for our brain to really be able to follow well.

EXAMPLE 2:

```
=SUM( INDEX( INDIRECT( " ' Projection!G"&$D22&" :BG"&$D22
), 1, MATCH(R$1, Projection!$G$9:$BG$9, 0) ) : INDEX( INDIRECT
```

Then the author proposed his own fix to the above mess (after inserting an extra row with flags):

EXAMPLE 2':

```
=SUMIF( INDIRECT( $A$1&"
Projection!$G$8:$CG$8" ), M$18, INDIRECT( $A$1&"
Projection!$G$"&$D23&" :$cG$"&$D23) ) /M$20
```

Again, we have the generic obfuscation arising from multiply-nested functions, but I also want to put in a bit of warning against using INDIRECT if you can help it.

Consider how INDIRECT works: it takes as its argument a string that is a cell or range reference. Consider the two items—assume the value sitting in cell A1 is "B3" and the value sitting in cell B3 is 42.

Then:

```
=INDIRECT(A1)
```

Will result in 42 (it looks in cell A1, takes the string "B3" and hands it to INDIRECT, which then looks at the value sitting in cell B3).

The function call:

```
=INDIRECT("A1")
```

Will result in "B3".

Fine, nothing wrong with any of this. But what if you want to move the cells around? Say you cut-and-paste cell A1 into cell D4. The first function call =INDIRECT(A1) will get automatically updated to =INDIRECT(D4) (and will still return 42 ... as long as B3 hasn't gotten changed). The second function call, =INDIRECT("A1"), will

remain unchanged. There's nothing sitting in cell A1 after the cut-and-paste, so you will get an empty result.

Lesson: INDIRECT is dangerous in a spreadsheet where the structure may be actively changing.

Last example for this edition:

EXAMPLE 3:

```
{=SUM(($B5 = 'Loss Table'!$A$9:$A$8765) * ($Q5 =
'Loss Table'!$B$9:$B$8765) *
(IF(TRIM($K5)="", "0", TEXT($K5, "0"))) = 'Loss
Table'!$C$9:$C$8765) *
(IF(TRIM($L5)="", "0", TEXT($L5, "0"))) = 'Loss
Table'!$D$9:$D$8765) *
(IF(TRIM($M5)="", "0", TEXT($M5, "0"))) = 'Loss
Table'!$E$9:$E$8765) *
(IF(TRIM($N5)="", "0", TEXT($N5, "0"))) = 'Loss
Table'!$F$9:$F$8765) *
(IF(TRIM($O5)="", "0", TEXT($O5, "0"))) = 'Loss
Table'!$G$9:$G$8765) *
(IF(TRIM($P5)="", "0", TEXT($P5, "0"))) = 'Loss
Table'!$H$9:$H$8765) * ('Loss Table'!$I$9:$I$8765))}
```

Unfortunately, I can't get this to look as hideous as it looks in the Excel formula bar due to the way Microsoft Word auto-formats everything. But imagine this giganto-formula sitting there in a single line, difficult to read.

A word about the curly braces surrounding this, if you are unfamiliar with this, this is called an array function. I don't want to get into all the details at this moment, but the way one enters an array function is instead of hitting Enter when entering the formula, one hits CTRL-Shift-Enter.

Part of the reason I don't want to explain array formulas is that it's best not to use them if you can avoid it—most people don't understand them, and it can be difficult to edit if you have an array result (as opposed to an array formula sitting in a single cell).

Someone proposed a fix to the above, and I know I'm going to regret explaining one part because this will encourage the writing of megaformulas that I hate so much:

EXAMPLE 3':

```
=SUMPRODUCT(('Loss Table'!$I$9:$I$8765)
*($B5 = 'Loss Table'!$A$9:$A$8765)
*($Q5 = 'Loss Table'!$B$9:$B$8765)
*(IF(TRIM($K5)="", "0", TEXT($K5, "0")) = 'Loss
Table'!$C$9:$C$8765)
*(IF(TRIM($L5)="", "0", TEXT($L5, "0")) = 'Loss
Table'!$D$9:$D$8765)
*(IF(TRIM($M5)="", "0", TEXT($M5, "0")) = 'Loss
Table'!$E$9:$E$8765)
*(IF(TRIM($N5)="", "0", TEXT($N5, "0")) = 'Loss
Table'!$F$9:$F$8765)
*(IF(TRIM($O5)="", "0", TEXT($O5, "0")) = 'Loss
Table'!$G$9:$G$8765)
*(IF(TRIM($P5)="", "0", TEXT($P5, "0")) = 'Loss
Table'!$H$9:$H$8765))
```

In this case, the formula looked exactly like this, because when the author of this formula got to the end of each logical chunk, he typed Alt+Enter—which puts a carriage return in the formula. It makes the formula much easier to read, and has no effect on the calculation of the formula itself.

For future editions, please submit your own examples, or proposed cleanups to the formulas above, to marypat.campbell@gmail.com. I will credit submissions, but I also assure anonymity to those who don't want to be associated with car wrecks of bad formulas.



475 North Martingale Road, Suite 600 Schaumburg, Illinois 60173
 Phone: 847.706.3500 Fax: 847.706.3599 www.soa.org