

The Baseline Scenario

What happened to the global economy and what we can do about it

The Importance of Excel

Posted on [February 9, 2013](#) by [James Kwak](#) | [164 Comments](#)

By James Kwak

I spent the past two days at a [financial regulation conference](#) in Washington (where I saw more BlackBerries than I have seen in years—can't lawyers and lobbyists afford decent phones?). In his remarks on the final panel, Frank Partnoy mentioned something I missed when it came out a few weeks ago: the role of Microsoft Excel in the "London Whale" trading debacle.

The issue is described in the appendix to JPMorgan's internal investigative [task force's report](#). To summarize: JPMorgan's Chief Investment Office needed a new value-at-risk (VaR) model for the synthetic credit portfolio (the one that blew up) and assigned a quantitative whiz ("a London-based quantitative expert, mathematician and model developer" who previously worked at a [company](#) that built analytical models) to create it. The new model "operated through a series of Excel spreadsheets, which had to be completed manually, by a process of copying and pasting data from one spreadsheet to another." The internal Model Review Group identified this problem as well as a few others, but approved the model, while saying that it should be automated and another significant flaw should be fixed.** After the London Whale trade blew up, the Model Review Group discovered that the model had not been automated and found several other errors. Most spectacularly,

"After subtracting the old rate from the new rate, the spreadsheet divided by their sum instead of their average, as the modeler had intended. This error likely had the effect of muting volatility by a factor of two and of lowering the VaR . . ."

I [write periodically about](#) the perils of bad software in the business world in general and the financial industry in particular, by which I usually mean back-end enterprise software that is poorly designed, insufficiently tested, and dangerously error-prone. But this is something different.

Microsoft Excel is one of the greatest, most powerful, most important software applications of all time.** Many in the industry will no doubt object. But it provides enormous capacity to do quantitative analysis, letting you do anything from statistical analyses of databases with hundreds of thousands of records to complex estimation tools with user-friendly front ends. And unlike traditional statistical programs, it provides an intuitive interface that lets you see what happens to the data as you manipulate them.

As a consequence, Excel is everywhere you look in the business world—especially in areas where people are adding up numbers a lot, like marketing, business development, sales, and, yes, finance. For all the talk about end-to-end financial suites like SAP, Oracle, and Peoplesoft, at the end of the day people do financial analysis by extracting data from those back-end systems and shoving it around in Excel spreadsheets. I have seen internal accountants calculate revenue from deals in Excel. I have a probably untestable hypothesis that, were you to come up with some measure of units of software output, Excel would be the most-used program in the business world.

But while Excel the program is reasonably robust, the spreadsheets that people create with Excel are incredibly fragile. There is no way to trace where your data come from, there's no audit trail (so you can overwrite numbers and not know it), and there's no easy way to test spreadsheets, for starters. The biggest problem is that anyone

can create Excel spreadsheets—badly. Because it's so easy to use, the creation of even important spreadsheets is not restricted to people who understand programming and do it in a methodical, well-documented way.***

This is why the JPMorgan VaR model is the rule, not the exception: manual data entry, manual copy-and-paste, and formula errors. This is another important reason why you should pause whenever you hear that banks' quantitative experts are smarter than Einstein, or that sophisticated risk management technology can protect banks from blowing up. At the end of the day, it's all software. While all software breaks occasionally, Excel spreadsheets break all the time. But they don't tell you when they break: they just give you the wrong number.

There's another factor at work here. What if the error had gone the wrong way, and the model had incorrectly doubled its estimate of volatility? Then VaR would have been higher, the CIO wouldn't have been allowed to place such large bets, and the quants would have inspected the model to see what was going on. That kind of error would have been caught. Errors that lower VaR, allowing traders to increase their bets, are the ones that slip through the cracks. That one-sided incentive structure means that we should expect VaR to be systematically underestimated—but since we don't know the frequency or the size of the errors, we have no idea of how much.

Is this any way to run a bank—let alone a global financial system?

* The flaw was that illiquid tranches were given the same price from day to day rather than being priced based on similar, more liquid tranches, which lowered estimates of volatility (since prices were remaining the same artificially).

** But, like many other Microsoft products, it was not particularly innovative: it was a rip-off of Lotus 1-2-3, which was a major improvement on VisiCalc.

*** PowerPoint has an oft-noted, parallel problem: It's so easy to use that people with no sense of narrative, visual design, or proportion are out there creating presentations and inflicting them on all of us.

Update 2/10: There is an interesting follow-on discussion that includes a lot of highly-informed technical people, including some who work in finance, over at [Hacker News](#).

SHARE THIS:



7 bloggers like this.

RELATED

[More Bad Excel](#)
In "Commentary"

[What Is the Trans-Pacific
Partnership \(TPP\) Really All
About?](#)
In "Commentary"

[The PR War](#)
In "Commentary"

This entry was posted in [Commentary](#) and tagged [JPMorgan](#), [software](#), [technology](#). Bookmark the [permalink](#).