

Re: WHY

Source: <http://www.tech-archive.net/Archive/Excel/microsoft.public.excel/2004-12/1398.html>

From: Harlan Grove (hrlngrv_at_aol.com)

Date: 12/24/04

Date: Thu, 23 Dec 2004 22:09:42 -0800

<aaron_kempf@hotmail.com> wrote...

>VAR and STDEV is also in SQL Server.. the things that you guys don't
>understand is that ANY FUNCTION that is in Excel is also in MDX-- they
share
>the same codebase..

Really? Does MDX include all the engineering functions like BETAINV and GAMMALN? Or LINEST, LOGEST, TREND and GROWTH? How about MDETERM, MINVERT, MMULT or (this be nasty) TRANSPOSE?

>It came out 5 or 6 years ago; and it's changing the world.

Here you're being inconsistent. You started off complaining that continued Excel use was stifling acceptance and growth of databases (specifically Access and MDX). Now you're claiming it's transformational. If what you're saying now were correct, you wouldn't have had to gripe about Excel use/users.

As most database processing doesn't take place on Windows systems, it may be changing the Windows world, but it's having squat all impact elsewhere.

>The difference is that you can either put your business logic in a
>spreadsheet and copy it every month..

OK, I give up. You're too damn stupid and/or prejudiced to understand how to do it right in spreadsheets.

>or you can automate and catalog the MDX and presto-chango-- you don't need
>to make the same report every week.

No, you don't need to recreate the same report each week, but if your boss expects it, you damn well better generate it each week. If you designed a spreadsheet correctly, you could just rerun it each week without new data entry. However, as previously agreed, repetitive reports would be better implemented as database apps. But again you're too damn stupid and/or prejudiced to learn that reports aren't the only thing spreadsheets can do.

>I can put together an amortization table.. all it is is a self join (in
>RDBMS) or a pretty PivotList via OLAP (much more powerful); a bunch of
>times.. not that complex. I would prefer to do it in olap

>From another response, but repetition is good for limited intellects like
yours.

So, genius, how *exactly* do you produce a query with a table joined to
itself 361 times (30 year loan with monthly payments including time periods
from month 0 through month 360)? Would these all be inner or outer joins?

What's in the table initially? Just the original loan amount and some zeros?

Why the hell would it be a crosstab? Amortization tables include columns for
month, principal portion of current payment, interest portion of current
payment and principal balance after current payment, with month 0 showing
zeros for principal and interest portions and the original loan amount as
principal balance. Seems like a table with 4 columns, and the number of
columns is invariant with respect to the number of periods over which the
loan is repaid. Perhaps you have some weird, idiosyncratic definition of
crosstab. Wouldn't come as a surprise at this point.

>OLAP makes it a lot easier than traditional database queries... We can make
>it in a pretty crosstab format so that you guys can export it from the
>database and put it in your spreadsheet; and then you can pull it out of a
>database, you make it pretty; waste a lot of time.. and then you recreate
>the same report next week.

OK, it's now clear. Since you seem to believe amortization tables could be
generated by crosstabs, you have not the slightest clue what an amortization
table is. Amortization tables aren't reports that need to be recreated ever
(well, not for fixed interest rates).

Now, since the number of periods over which loan payments could be made is
itself variable, the same query, if implemented by N+1 joins where N is the
number of loan payments, can't be used to generate amortization tables for
5-year, 10-year, 15-year and 30-year loans. You'd need a different query for
each loan period. In a spreadsheet, if the table should end when the
principal balance reaches zero, all that's needed is a little extra logic
like

A99: (month or payment period)
=IF(N(D98)>0,A98+1,"")

B99: (principal paid)
=IF(N(D98)>0,P/(1+i)^(N-A98),"")

C99: (interest paid)
=IF(N(D98)>0,P-B99,"")

D99:

=IF(N(D98)>0,D98-B99,"")

Ensure that the formulas extend down to accommodate the longest possible loan duration, and one spreadsheet handles any acceptable number of payments.

As I've mentioned before, subject matter knowledge is more important than database programming ability (and it pays a lot more, too).

>don't you guys feel hollow inside?

Nowhere near as hollow as the region between your ears.

>don't you guys see that there MIGHT be a better way?

For amortization tables? Not even close. Spreadsheets MUCH BETTER! As proven above.

>compound interest-- this kind of stuff is easy to do in a database.. this stuff is simple simple stuff.

Then provide details on how **exactly** you'd generate an amortization table. Not vague statements about arbitrary joins, but what's initially in the table that your querying, and what the query would actually look like.

>Get a clue, spreadsheets are dead.. MDX and OLAP or SQL Server can do anything you're doing.. but you guys are too egocentric to drop this obsolete program and wake up to the 21st century.

Really?! This is OT because it doesn't involve spreadsheets. I receive information in PDF files that include tables in various formats about twice a month from outside brokers (so the data in those PDF files isn't on my own company's systems except as PDF files on my own PC). In order to use that information, I run an ancient version of pdf2text to convert the PDF files to nearly unformatted plain text. Then I use awk or perl scripts to extract the information I need into CSV files.

Other than writing procedural VBA code in a module in Access, how would Access **without** VBA be able to pull anything from these PDF files?

Once again the paucity of your own experience bites you in the butt.

>I just want to make sure that every one of you understand that there is a war going on in IT departments; and this is a battle against Elitist obsolete management; and database professionals that are going to automate you out of a job.

Or to put it another way, between the experienced managers who have some idea that their companies sell goods or services other than databases against the interchangeable developer grunts who are so limited that can only understand the database they work with on a day-to-day basis.

Re: WHY

The difference between us is that I know what you do, but you don't know what I do (or any other participant in this thread does). I mean that even if we gave you our job titles and job descriptions, you still wouldn't understand.

Actually I'm giving way to much credit to your assessment of what's going on in IT departments and companies as a whole. In most companies there's movement toward recentralization. That means centralization of application development when it makes sense, and it also means greater use of centralized databases at least as storage subsystems for other applications.

Other departments will go along with this as long as they can still do what they need to do. As most IT departments have no one with a clue how the US federal corporate income tax laws work, tax departments will still generate the reports they need to (such as federal and state corporate income tax returns) without additional IT assistance. Ditto investment management departments. Ditto manufacturing production departments (aside from inventory control). Ditto engineering departments.

I have yet to see any IT department generate the important 1 to 5 page reports rather than the 1,000+ page reports that other departments have to sift through to generate the important 1 to 5 pagers (well, aside from the reports on software and hardware support and training costs, but those reports are usually generated by nondevelopers who have some clue about business). Why don't they? Because most of the IT staff has no clue what's important in the 1,000+ page reports they generate because they have no clue what their companies actually do or really understand what they sell.

>I hope that you guys diversify and expand your horizons.

Seems like you're the fool with all your eggs in one basket.