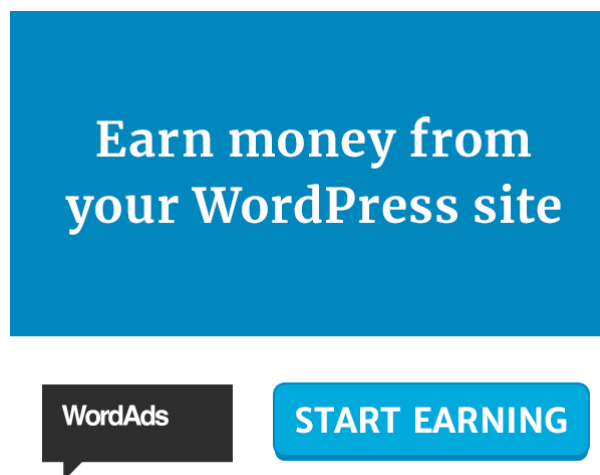


Javier Guillén

Pervasive Business Intelligence

Advertisements



Earn money from
your WordPress site

WordAds

START EARNING

REPORT THIS AD

Quartile, Percentile and Median in PowerPivot / DAX

Lets examine what the options are for calculating Quartile / Percentiles in PowerPivot. On my last [post](#), I explained one way to calculate Median using pure DAX. Though the calculation works, don't expect it to be speedy when dealing with large datasets; The reason for this is that it relies on row context iterations to rank values prior to selecting the median.

As similar technique can be used to calculate Quartile or Percentile, following this blog [entry](#). However, the result is not always optimal: not only the calculation can be slow, be the results do not match Excel's native functions. The reason for this is that Excel uses [interpolation](#) to calculate both Quartile and Percentile, whereas the DAX expression above picks one value out of the set (in other words, its just an approximation).

Can we use a technique to match Excel's results within a PowerPivot model? The answer is **yes**, though the technique discussed here does not use DAX. What I want to share is another way of using the inherent capabilities of PowerPivot: being itself a version of SQL Server Analysis Services, it is actually able to understand MDX. It is quite interesting to notice that PowerPivot can behave in ways that reveal its SSAS nature: from the ability to respond to Excel cube functions to generating a trace file which can be read in SQL Profiler.

So what does this have to do with calculating Percentile? Well, even though DAX does not include any of those functions you can leverage PowerPivot's multidimensional interface to create an MDX calculated member that will encapsulate each of those functions. Lets see how.

In order to see PowerPivot queries to the Vertipaq engine in MDX format, we can use a free codeplex add-in called 'OLAP PivotTable Extensions' (download it [here](#), and be sure to select the correct bitness version). This add-in was created with SSAS in mind, but – without any further effort from the add-in itself – PowerPivot responds to it. Once installed, create a PowerPivot table and then use the add-in to view the MDX query (select the 'MDX' tab on the window that appears when you click on the OLAP Pivot Table Extensions menu item)



Sample PowerPivot table using AdventureWorks data

The MDX query we get is:

```
SELECT
  NON EMPTY Hierarchize(
    {DrilldownLevel({[DimDate].[EnglishMonthName].[All]},,INCLUDE_CALC_MEMBERS))
  DIMENSION PROPERTIES PARENT_UNIQUE_NAME,HIERARCHY_UNIQUE_NAME ON COLUMNS
FROM [Sandbox]
WHERE (
  [DimDate].[CalendarYear].&[2004],
  [Measures].[Sum of SalesAmount]
)
```

So lets create a calculated member to compute **Median**.

Select the 'Calculations' tab on the add-in menu. This is used to submit calculated members along with the rest of the pivot table query. Type the following query:

```
Median(
  (
    [DimDate].[CalendarYear].CurrentMember *
    [DimDate].[EnglishMonthName].[All].Children
  )
  , CoalesceEmpty ( [Measures].[Sum of SalesAmount] , 0 )
)
```

This should look in the add-in window as follows:



The result matches Excel regular 'Median' function:



Lets examine the actual query. MDX has a built-in 'Median' function (see definition [here](#)). As you can see, the expression uses parameters that look like regular SSAS dimensions. This is the PowerPivot data model though, and we didn't have to use Visual Studio to create it: the PowerPivot engine automatically creates dimensions based on the tabular model loaded into Vertipaq.

The measure [Sum of SalesAmount] is the aggregate PowerPivot created based on the Sales Amount column of the in-memory fact table. Once it became an actual measure in the PowerPivot model, we can use it as the measure expression of the MDX function.

Lets try now creating Quartile and Percentiles. It gets more interesting here: Unlike the 'Median' function, MDX has no built-in Percentile or Quartile function. So here we have it: neither DAX nor MDX will help us here. However, we can leverage a another feature of SSAS: the ability to perform COM interop calls through the ExcelMDX assembly. You can see this library on a regular SSAS installation by browsing the assemblies folder:



It is quite interesting when you think about it. We are asking an Excel pivot table to communicate with the PowerPivot engine through a calculated member which in turn invokes Excel functionality not present in the MDX language. Pretty cool indeed. So lets try it: Using the following expression we can compute Quartiles:

```
Excel!Quartile(
  SetToArray (
    ( [DimDate].[CalendarYear].CurrentMember * [DimDate].[EnglishMonthName].
```

[All].Children

```
)  
, CoalesceEmpty ( [Measures].[Sum of SalesAmount] , 0 )  
)  
,1)
```

The output is the following:



As you can see, the output matches Excel's native quartile function. The calculation is now built into the PowerPivot model, and hence is dynamic and can respond to slicers and filters, as well as cell context by using MDX properties like 'CurrentMember'. And though the PowerPivot is performing a COM interop call to Excel (marshalling calls from one environment to the other), it should still perform better than a DAX calculation based heavily on row context.

If you examine the MDX that is sent to the Vertipaq engine, you will notice the calculated member is declared as a query scoped expression:

WITH

MEMBER [Measures].[Quartile] as Excel!Quartile(

SetToArray (

([DimDate].[CalendarYear].CurrentMember * [DimDate].[EnglishMonthName].

[All].Children

```
)  
, CoalesceEmpty ( [Measures].[Sum of SalesAmount] , 0 )  
)  
,1)
```

SELECT

{

[Measures].[Sum of SalesAmount],

[Measures].[Quartile]

}

DIMENSION PROPERTIES PARENT_UNIQUE_NAME,HIERARCHY_UNIQUE_NAME ON COLUMNS ,

NON EMPTY

Hierarchize(

DrilldownMember(

CrossJoin(

{

[DimDate].[CalendarYear].[All],

[DimDate].[CalendarYear].[CalendarYear].AllMembers

},

{{[DimDate].[EnglishMonthName].[All]}}

),

[DimDate].[CalendarYear].[CalendarYear].AllMembers,

[DimDate].[EnglishMonthName]))

DIMENSION PROPERTIES PARENT_UNIQUE_NAME,HIERARCHY_UNIQUE_NAME ON ROWS

FROM [Sandbox] CELL PROPERTIES VALUE, FORMAT_STRING, LANGUAGE, BACK_COLOR, FORE_COLOR,

FONT_FLAGS

The dimension used (DimDate) as well as the measure (Sum of SalesAmount) were automatically created by the PowerPivot engine. This is what provides compatibility with OLAP interfaces: for each table in the PowerPivot model, a default 'count' measure has been created as well as a dimension, in which each column is an attribute. In other words, dimensions and measures are somewhat interchangeable in PowerPivot, following a more flexible model than with the traditional SSAS engine. Also, notice the query is executed against a cube called 'Sandbox'. This is the default cube-interface name for Vertipaq PowerPivot databases. It is the same name you will see in the

cubes generated in the dedicated SSAS instance used on *PowerPivot for SharePoint* (This is an instance of SQL Server Analysis Services 2008 R2 in Vertipaq mode).

One downside of this method is that the PowerPivot field list is not aware of the calculation. However, if you open the standard pivot table field list you will see it there. In other words, and as I said before, this calculation is now part of the data model. Which means you can access it outside of the constraints of the pivot table: through Excel cube functions. As an example, using the following formula expression on a worksheet cell we can *dynamically* invoke the MDX calculated member against PowerPivot, with the ability to specify a [query context](#) to it:

```
=CUBEVALUE("PowerPivot Data",CUBESET("PowerPivot Data","Head([DimDate].[CalendarYear].[All].Children)","[Measures].[Quartile]"))
```

The result will be 925,015 again. In other words, we were able to dynamically invoke the first member of the CalendarYear attribute on the DimDate dimension and leverage the MDX calculated member, all in one cell – no need of pivot tables. Pretty powerful right?

You can verify that our Quartile calculation is now part of the intrinsic model by browsing through measures members when constructing the cube formula:



What I find pretty interesting is that the list of available measures on the PowerPivot model includes now regular DAX measures ([Measures].[Sum of Sales]) as well as MDX measures ([Measures].[Quartile]).

How about Percentile? Using the same method, it is pretty simple to calculate it. Just replace Quartile in the MDX expression... that's it!

```
Excel!Percentile(  
    SetToArray (  
        ( [DimDate].[CalendarYear].CurrentMember * [DimDate].[EnglishMonthName].  
[All].Children  
        )  
    , CoalesceEmpty ( [Measures].[Sum of SalesAmount] , 0 )  
    )  
,0.30)
```



Be careful to not define your underlying measure as 'currency' type. Though I always thought data types in the PowerPivot window were purely decorative (as they don't carry over the pivot table), the MDX calculated member will not work on currency types. Instead, try decimal type and you will be fine.

And one last thing... as you may have noticed, this technique opens the possibility to a huge range of calculated expressions. An though it is not the standard PowerPivot/DAX implementation, it still leverages PowerPivot technology by making use of its inherent data model. In other words, PowerPivot applications can not only benefit from the amazing power of DAX , but also that of MDX (which in turn opens the door to further functionality through interfaces with Excel and VBA). Things like Internal Rate of Return (IRR) are also possible using this method



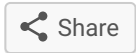
```
Excel!IRR( SetToArray ( [Table3].[b].[All].Children, [Measures].[Sum of a] ) )
```





REPORT THIS AD

Share this:



Be the first to like this.

Related

[Working with relationships in PowerPivot & SSAS BISM Tabular](#)
In "Power BI"

[Simulating an "approximate match" VLOOKUP in PowerPivot](#)
In "Power BI"

[Calculating Median in PowerPivot / DAX](#)
In "Power BI"

September 13, 2011

 **2** Replies

[« Previous](#)

[Next »](#)

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name

*

Email

*

Website

Post Comment

☐ Notify me of new comments via email.

☐ Notify me of new posts via email.



Jelle-Jeroen Lamkamp on September 13, 2011 at 12:20 pm

Javier

I'm struggling to get the percentile working in my workbook..
Its about 184KB, can you please have a look...?

regards

Jelle-Jeroen

↩ Reply



javierguillen on September 14, 2011 at 2:44 am

Hi Jelle-Jeroen

What method of percentile calculation are you trying to use? Do you need interpolated results? Is it the pure DAX or the one relying on MDX? Send me your file and I will take a look at it.

javiguillen@hotmail.com

↩ Reply

Archive

[January 2014](#) (1)
[September 2013](#) (2)
[August 2013](#) (1)
[June 2013](#) (1)
[March 2013](#) (2)
[February 2013](#) (1)
[January 2013](#) (2)
[December 2012](#) (2)
[November 2012](#) (2)
[October 2012](#) (2)
[September 2012](#) (2)
[August 2012](#) (7)
[July 2012](#) (4)
[June 2012](#) (2)
[May 2012](#) (3)
[April 2012](#) (3)
[March 2012](#) (4)
[February 2012](#) (3)
[January 2012](#) (2)
[December 2011](#) (1)
[November 2011](#) (1)
[October 2011](#) (2)
[September 2011](#) (8)

Tweets

[@CharBIGroup](#) [@SentryOne](#) [@john_welch](#) Thank you [@SentryOne](#) !! [2 weeks ago](#)

RT [@AdamDWilson](#): Shared and certified datasets are coming very soon to [#powerbi](#), enabling reuse across the organization as well as governan... [2 weeks ago](#)


[@AdamDWilson](#) Very cool - pretty much every company I have consulted on Power BI has asked me about how to build pro... [twitter.com/i/web/status/1...](#) [2 weeks ago](#)

RT [@TheBrandonMorse](#): I've never seen something more human from a robot than this. [https://t.co/bDxkBISQA1](#)
[3 weeks ago](#)

RT [@_christianWade](#): What's new for SQL Server 2019 Analysis Services CTP 3.0? Dynamic format strings on numeric measures in tabular models!... [3 weeks ago](#)

 [Follow @javiguillen](#)

Share this

 [SHARE](#)    ...

AUTOMATTIC

**We're hiring
backend developers.
Join us!**

APPLY



REPORT THIS AD

[View Full Site](#)

[Blog at WordPress.com.](#)

