Calculation Basics
Advanced Calculations
DAX to the MAXX()

Paul Turley
Data Platform MVP,
BI Consulting Architect
Intelligent Business LLC

- Introduction to DAX
- Essential DAX functions
- Calculated columns
- Measures
- Aggregators
- Time intelligence

- DAX function roadmap
- CALCULATE & FILTER functions
- How DAX Processes & Calculates Results
- Dependent measures
- Evolution of a measure
- Modifying Row & Filter context
- When to use iterators
- DAX Studio
Design Phases

**Get Data**
- Connect
- Import
- Refresh
- Stream

**Transform**
- Shape
- Cleanse
- Rename
- Merge
- Filter

**Model**
- Relate
- Calculate
- Hide
- Format

**Visualize & Analyze**
- Filter
- Slice
- Interact
- Drill

**Publish & Share**
- Pin
- Configure
- Slice
- Collaborate
- Embed

**DAX**
What is DAX and Where Did it Come From?

1. Expression language, used to perform calculations in:
   - Power Pivot
   - Power BI
   - SQL Server Analysis Service (SSAS) Tabular
2. Query language
3. Language elements derived from:

- Excel functions
- SQL
- MDX

DAX
Learning DAX

- Understanding essential concepts is more important than memorizing functions
- You can always lookup function syntax
- Keeping a library of working examples may be more valuable than a web search

© 2017 Intelligent Business LLC, intelligentbiz.net
DAX Learning Curve

Deceptively simple

Elegantly complex

Most calculations

#%@!
What to Learn

Examples of some important DAX functions & concepts
Beyond the basics, specific functions are useful in different business scenarios
There or than 250 DAX functions and more added with each major release
Important Concepts

• Row context
• Filter context
• Filter propagation
• Aggregators
• Iterators

• Calculated columns
• Measures
Context

Webster: “The interrelated conditions in which something exists or occurs”

Turley: “Where the heck am I?”...
...“and how did I get here?”

Row Context: “What row am I on?”

Filter Context: “What filters are applied?“
- 1st Class, Bus. Class or Economy
- Window, isle or middle seat
Implicit & Explicit Measures

No one right answer

Two schools of thought:

1. **Power BI should behave like Excel**
   - Excel uses implicit measures with workbook data
   - PivotTables & charts:
     - SUM numbers by default
     - COUNT text by default
   - Power BI Desktop: numeric columns have Summarize By property

2. **All measures should be explicitly defined**
   - Model designer maintains control
   - Default behavior may not always be right

Excel doesn’t support Power BI Implicit measures
Recommended DAX Tools

Power BI Desktop /Excel 2013+
- Power BI Desktop
- Excel
- DAX Studio

SSAS Tabular
- SSDT for Visual Studio 2015+
- SSMS
- Excel
- DAX Studio

Excel 2010/SSAS 2012, 2014
- NotePad++
- DAX Studio
- DAXFormatter.com
# DAX Function Roadmap

## Important concepts

**DAX query engine flow:**
- Get report filter context
- Apply `CALCULATE()` filters
- Apply dimension table filter context
- Filter fact table using relationships
- Perform math & calculations

**Context:**
- Evaluation...
- Filter...
- Row...
- Context transition

**Measures:**
- Majority of all calculations
- Can have implicit or explicit filter context
- Can override natural filter behavior

**Calculated Columns:**
- Good for banding
- Built-in row context
- Using measures in calc. columns will ignore row context

---

### Some examples

<table>
<thead>
<tr>
<th>Date &amp; Time</th>
<th>Time Intelligence</th>
<th>Filter</th>
<th>Information</th>
<th>Logical</th>
<th>Math</th>
<th>Statistical</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>DATE()</code></td>
<td><code>DATESMTD()</code></td>
<td><code>FILTER()</code></td>
<td><code>ISBLANK()</code></td>
<td><code>IF()</code></td>
<td><code>SUM()</code></td>
<td><code>SUMX()</code></td>
</tr>
<tr>
<td><code>TIME()</code></td>
<td><code>TOTALMTD()</code></td>
<td><code>VALUES()</code></td>
<td><code>ISERROR()</code></td>
<td><code>AND</code>, <code>&amp;&amp;</code>, <code>AND()</code></td>
<td><code>AVERAGE()</code></td>
<td><code>AVERAGEX()</code></td>
</tr>
<tr>
<td><code>TODAY()</code></td>
<td><code>PERVIOUSMONTH()</code></td>
<td><code>ALL()</code></td>
<td><code>CONTAINS()</code></td>
<td><code>OR</code>, `</td>
<td></td>
<td><code>, </code>OR()`</td>
</tr>
<tr>
<td><code>NOW()</code></td>
<td><code>SAMEPERIODLASTYEAR()</code></td>
<td><code>CALCULATE()</code></td>
<td><code>HASONEVALUE()</code></td>
<td><code>SWITCH()</code></td>
<td><code>PERCENTILEX()</code></td>
<td></td>
</tr>
<tr>
<td><code>DATEDIFF()</code></td>
<td><code>DATESINPERIOD()</code></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>EOMONTH()</code></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

© 2017 Intelligent Business LLC, intelligentbiz.net
Calculated Columns

• Typically used in the context of a single row
• May be used as intermediate calculations to support measures

Arrival Date Time =

IF( [ArrTime] < [DepTime],
    [FlightDate] + 1,
    [FlightDate] ) + [ArrTime]

This calculated column combines the flight date and arrival time values needed by measure calculations. It adds one day if the arrival time is earlier than the departure time because the flight landed the next day (after midnight).
Measure Calculations

• Understanding context
  • Row context
  • Filter context
  • Context switching

• The CALCULATE function revisited

• How DAX processes & calculates results

• Maintaining a development, testing and learning environment
CALCULATE function

CALCULATE(<expression>, <filter1>, <filter2>)

Means: “Go apply filters”
Similar in concept to **Where** in TSQL

```
Avg Weath Delay for bad weather days =
CALCULATE(
    AVERAGE([Weather Delay]),
    'Airline Performance'['WeatherDelay'] > 0
)
```

This measure calculates the average Weather Delay only for those flights where there was a delay (over 0 minutes)
How DAX Processes & Calculates Results

1. Get Row & Filter Context from report visual (or Pivot Table here)
2. Adjust the Filter Context based on CALCULATE() parameters (if CALCULATE is part of the function)
3. Filter the related lookup tables
4. Pass those filters along to the Fact table
5. Output a subset of rows
6. Do the math

Excel example: …based on filters, slicers, rows & columns
Time Intelligence

DAX functions

• NOW, DATE, TIME
• DATEADD
• DATEDIFF
• DATESMTD
• DATESBETWEEN
• TOTALMTD ...QTD, YTD
• NEXTMONTH ...DAY, QUARTER, YEAR
• PARALLELPERIOD
Time Intelligence Example

Month-To-Date Average

Avg Weather Delay MTD = \texttt{CALCULATE( [Avg Weather Delay], DATESMTD( 'Flight Date'[Flight Date]) )}

![Graph showing the relationship between Weather Delay and Weather Delay MTD by Flight Date. The graph includes a line chart with bars for different airlines and months.](image)
Iterator functions

Mixed totals: Operates on one row at a time, accumulating the result of the prior iteration

- SUMX
- AVERAGEX
- MINX
- MAXX
- COUNTX
- COUNTAX
- PRODUCTX
- CONCATENATEX
Using Iterators

These two different calculation techniques achieve the same result using different formula mechanics:

Avg Weather Delay =
AVERAGEX (FILTER ('Airline Performance', 'Airline Performance'[WeatherDelay] > 0 ), 'Airline Performance'[WeatherDelay] )

For every row in the 'Airline Performance' table, the FILTER is applied and qualifying values are added to the Average calculation. Although less efficient than the following example, complex operations can be performed within each iteration.

Avg Weather Delay =
CALCULATE([Avg Weather Delay], 'Airline Performance'[WeatherDelay] > 0 )
Dependent Measures

- Measures may be used in other measure calculations
- Measure names must be unique within the model
- Measures contained in any table can be used in any calculation

<table>
<thead>
<tr>
<th>Flights</th>
<th>COUNTROWS('Airline Performance')</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Flights</td>
<td>CALCULATE([Flights], ALL('Airline Performance'))</td>
</tr>
<tr>
<td>% of Total Flights</td>
<td>DIVIDE([Flights], [All Flights])</td>
</tr>
</tbody>
</table>
**CALCULATE() with FILTER()**

- Use FILTER() function for rich filtering

**Flights Over 15 min** = \( \text{CALCULATE(} [\text{Flights}], \) \( \text{FILTER(} \text{‘Airline Performance’}, [\text{Avg Weather Delay}] > 15 \) \)

**Comparison options:**

- Table[Column] = [Measure]
- Table[Column] = Table[Column]
- [Measure1] > [Measure2]
- \(<\text{true/false expr1}> \&\& <\text{true/false expr2}>\)
- \(<\text{true/false expr1}> || <\text{true/false expr2}>\)

**Function-based comparisons:**

- \(=\text{AND(} <\text{true/false expr1}>, <\text{true/false expr2}> \)\)
- \(=\text{OR(} <\text{true/false expr1}>, <\text{true/false expr2}> \)\)

Any expression that evaluates to true/false
Financial Calculations

common scenario

• Chart of Account dimension
• Level Rollups
• Parent-child source table
• Unary Operators
Using Disconnected Tables

May be used to:

• Pass “parameters” to a measure without filtering other model elements

• Implement conditional measure behavior

• Dynamically slice specialized measure calculations
Power BI and Excel
How is their relationship?
Use DAX Studio to Get Power BI Connection

• DAX Studio exposes the random port number Power BI Desktop uses to connect to the local SSAS server (msmdsrv.exe)
• Use the connection string localhost:XXXXX to connect with Excel, SQL Server Profiler, SSMS or any other OLAP client.
Recommended Practices

• Learn DAX concepts before function syntax
• Learn DAX essential functions... Learn these first:
  SUM, AVERAGE, MIN, MAX, COUNT, COUNTROWS, CALCULATE, FILTER, IF
• Name measure so users can find them
• Don’t try to memorize complex DAX
• Build a library of useful examples, books & articles
• Work in iterations
• Understand measure categories:
  aggregates, time & ratios, business-specific, KPI parts
Resources

• QuickStart: Learn DAX Basics in 30 Minutes

• PowerPivotPro YouTube channel
  https://www.youtube.com/powerpivotpro

• DAX Patterns, DAX Puzzle, DAX Formatter:
  http://www.daxpatterns.com, SQLBI.com

• Books:
Contact Information

Paul Turley
Intelligent Business LLC

e  Paul@IntelligentBiz.net
b  SqlServerBi.blog
l  linkedin.com/in/pturley
t  @paul_turley

Presentation and Materials:
http://tinyurl.com/MontrealPBI0517