www.sqlbi.com

sqlbi

Loader

Microsoft Partner
Gold Business Intelligence
Gold Data Platform

SSAS MAESTRO by Microsoft

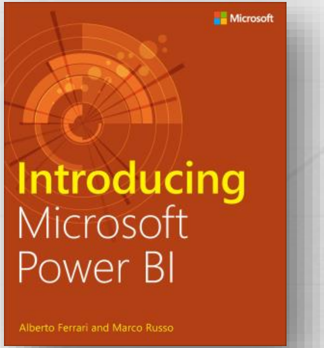
MVP Microsoft Most Valuable Professional

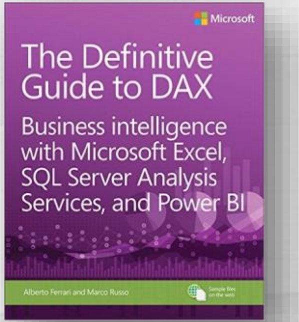
Time intelligence in Power BI

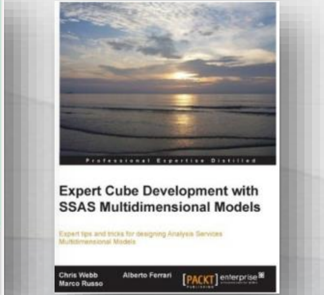
Alberto Ferrari
Mail: alberto.ferrari@sqlbi.com
Twitter: @FerrariAlberto

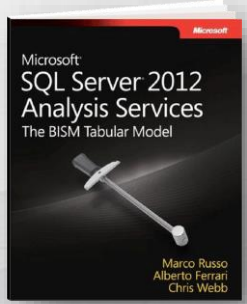
Alberto Ferrari

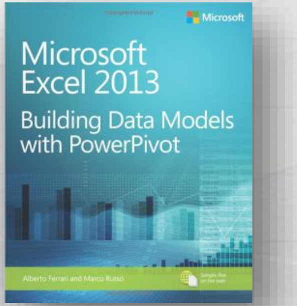
- BI expert and consultant
- Founder of **www.sqlbi.com**
- Consulting, mentoring, training
- Book writer

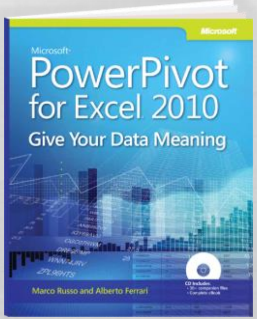



Microsoft
The Definitive Guide to DAX
Business intelligence with Microsoft Excel, SQL Server Analysis Services, and Power BI
Alberto Ferrari and Marco Russo


Professional Expertise Distilled
Expert Cube Development with SSAS Multidimensional Models
Chris Webb
Marco Russo
Alberto Ferrari
[PACKT] enterprise




Microsoft
Microsoft
Excel 2013
Building Data Models with PowerPivot
Alberto Ferrari and Marco Russo



What is Time Intelligence?

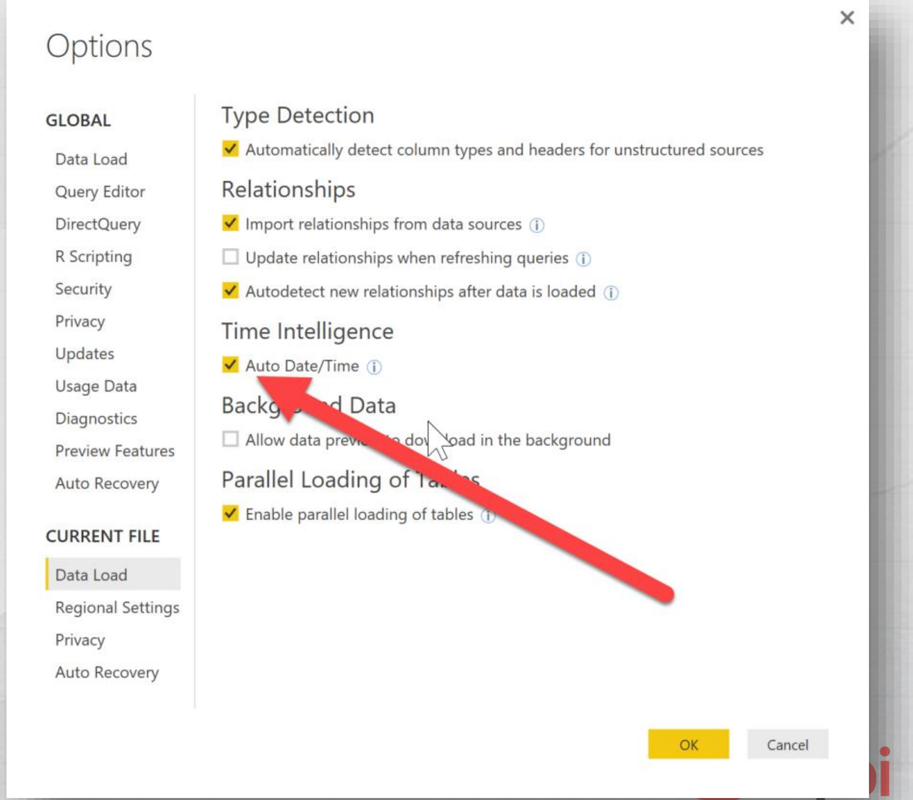
- Many different topics in one name
  - Year To Date, Quarter To Date, Running Total
  - Same period previous year, Working days computation
- In short: anything related with time
- Power BI does some time intelligence for you
  - Unfortunately, the wrong way
- If you want something done… do it yourself!
- As you might expect… DAX is the key

Power BI knows how to slice by date… well, almost.

Power BI and dates

Slicing by date in Power BI

- Auto Date/Time
- Enabled by default
- Builds a date hierarchy for you
- Lets you slice by
  - Year, Quarter, Month, Day
- Looks nice…



Type Detection

☑ Automatically detect column types and headers for unstructured sources

Relationships

☑ Import relationships from data sources

☐ Update relationships when refreshing queries

☑ Autodetect new relationships after data is loaded

Time Intelligence

☑ Auto Date/Time

Background Data

☐ Allow data preview to download in the background

Parallel Loading of Tables

☑ Enable parallel loading of tables

Problems of auto date/time

- Works only with standard calendars (01/01 to 31/12)
- Needs a datetime column in the fact table
- Does not work if you have multiple fact tables
- No time intelligence calculations
  - YTD, MTD, same period last year…
- Multiple dates make reports hard to read
- C’mon… you want (and deserve) more than this!

Probably the most important table in your model

Building a Date Table

Date Table

- Time intelligence needs a date table
  - Built in DAX
  - Or in a SQL Table
- Date table properties
  - All dates should be present
  - From 1° of January, to 31° of December
  - No holes
  - Otherwise time intelligence will not work

CALENDARAUTO

Automatically creates a calendar table based on the database content. Optionally you can specify a starting month (useful for fiscal years)

```
--
-- The parameter is the starting month
-- of the fiscal year
--
= CALENDARAUTO (
    7
)
```

Beware: CALENDARAUTO uses all the dates in your model, excluding only calculated columns and tables

CALENDAR

Returns a table with a single column named “Date” containing a contiguous set of dates in the given range, inclusive.

```
CALENDAR (
   DATE ( 2005,  1,  1 ),
   DATE ( 2015, 12, 31 )
)


CALENDAR (
    MIN ( Sales[Order Date] ),
    MAX ( Sales[Order Date] )
)
```

CALENDAR

If you have multiple fact tables, you need to compute the correct values

```
=CALENDAR (
    MIN (
        MIN ( Sales[Order Date] ),
        MIN ( Purchases[Purchase Date] )
    ),
    MAX (
        MAX ( Sales[Order Date] ),
        MAX ( Purchases[Purchase Date] )
    )
)
```

Set Sorting Options

- Month names do not sort alphabetically
  - April is not the first month of the year
- Use Sort By Column
- Set all sorting options in the proper way
- Beware of sorting granularity
  - 1:1 between names and sort keys

Multiple Dates

- Date is often a role dimension
  - Many roles for a date
  - Many date tables
- How many date tables?
  - Try to use only one table
  - Use many, only if needed by the model
  - Many date tables lead to confusion
    - And issues when slicing
- Use proper naming convention

Model setup, time to start using it

Aggregating over time

Aggregations Over Time

- Many useful aggregations
  - YTD: Year To Date
  - QTD: Quarter To Date
  - MTD: Month To Date
- They all need a Calendar Table
- And some understanding of CALCULATE

Sales 2015 up to 05-15 (v1)

Using CALCULATE you can filter the dates of the period to summarize

```
SalesAmount20150515 :=

CALCULATE (
    SUM ( Sales[SalesAmount] ),
    FILTER (
        ALL ( 'Date'[Date] ),
        AND (
             'Date'[Date] >= DATE ( 2015, 1, 1 ),
             'Date'[Date] <= DATE ( 2015, 5, 15 )
        )
    )
)
```

Sales 2015 up to 05-15 (v2)

You can replace the FILTER with DATESBETWEEN.
The result is always a table with a column.

```
SalesAmount20150515 :=

CALCULATE (
    SUM ( Sales[SalesAmount] ),
    DATESBETWEEN (
        'Date'[Date],
        DATE ( 2015, 1, 1 ),
        DATE ( 2015, 5, 15 )
    )
)
```

Sales Year-To-Date (v1)

Replace the static dates using DAX expressions that retrieve the last day in the current filter

```
SalesAmountYTD :=

CALCULATE (
    SUM ( Sales[SalesAmount] ),
    DATESBETWEEN (
        'Date'[Date],
        DATE ( YEAR ( MAX ( 'Date'[Date] ), 1, 1 ),
        MAX ( 'Date'[Date] )
    )
)
```

Year To Date (Time Intelligence)

DATESYTD makes filtering much easier

```
SalesAmountYTD :=

CALCULATE (
    SUM ( Sales[SalesAmount] ),
    DATESYTD ( 'Date'[Date] )
)
```

Year To Date: the easy way

TOTALYTD: the “DAX for dummies” version

```
SalesAmountYTD :=

TOTALYTD (
    SUM ( Sales[SalesAmount] ),
    'Date'[Date]
)
```

What if you use Int keys?

- Oftentimes, the key of a date is an int
  - For example, 20070101 means 01/01/2007
- Typically happens when you have a date table in the dat awarehouse
- Things are a bit more complex, then… in other words, time intelligence does not work
- The key, as usual, is understanding the filter context and its interactions with the report

Use the Correct Parameter

The parameter is the date column in the Calendar table, not the Sales[OrderDate]. Otherwise, you get wrong results

```
LineTotalYTD :=

TOTALYTD (
    SUM ( Sales[SalesAmount] ),
    Sales[OrderDate]
)
```

Handling Fiscal Year

The last, optional, parameter is the end of the fiscal year
Default: 12-31 (or 31/12 - locale dependent)

```
SalesAmountYTD :=
TOTALYTD (
    SUM ( Sales[SalesAmount] ),
    'Date'[Date],
    "06-30"
)

SalesAmountYTD :=
CALCULATE (
    SUM ( Sales[SalesAmount] ),
    DATESYTD ( 'Date'[Date], "06-30" )
)
```

Same Period Last Year

Same period in previous year. CALCULATE is needed

Specialized version of DATEADD

```
Sales_SPLY :=

CALCULATE (
    SUM ( Sales[SalesAmount] ),
    SAMEPERIODLASTYEAR ( 'Date'[Date] )
)
```

Mixing Time Intelligence Functions

YTD on the previous year. In DAX, it is very simple, just mix the functions to obtain the result

```
Sales_YTDLY :=

CALCULATE (
    SUM ( Sales[SalesAmount] ),
    DATESYTD (
        SAMEPERIODLASTYEAR ( 'Date'[Date] )
    )
)
```

DATEADD

Similar as SAMEPERIODLASTYEAR, used to calculate different periods: YEAR, MONTH, DAY …

Does not sum dates, it shifts periods over time

```
Sales_SPLY :=

CALCULATE (
    SUM( Sales[SalesAmount] ),
    DATEADD ( 'Date'[Date] , -1, YEAR )
)
```

PARALLELPERIOD

Returns a set of dates (a table) shifted in time

The whole period is returned, regardless dates in the first parameter

```
Sales_PPLY :=

CALCULATE (
    SUM ( Sales[SalesAmount] ),
    PARALLELPERIOD ( 'Date'[Date] , -1, YEAR )
)
```

Running Total

Running total requires an explicit filter

```
SalesAmountRT :=

CALCULATE (
    SUM ( Sales[SalesAmount] ),
    FILTER (
        ALL ( 'Date' ),
       'Date'[Date] <= MAX ( 'Date'[Date] )
    )
)
```

Moving Annual Total (v1)

Moving window from the current date back one year

```
CALCULATE (
    SUM ( Sales[SalesAmount] ),
    DATESBETWEEN (
        'Date'[Date],
        NEXTDAY (
            SAMEPERIODLASTYEAR (
                LASTDATE ( 'Date'[Date] )
            )
        ),
        LASTDATE ( 'Date'[Date] )
    )
)
```

Beware of function order!

Time intelligence functions return sets of dates, and the set of dates need to exist.

```
CALCULATE (
    SUM ( Sales[SalesAmount] ),
    DATESBETWEEN (
        'Date'[Date],
        SAMEPERIODLASTYEAR (
            NEXTDAY (
                LASTDATE ( 'Date'[Date] )
            )
        ),
        LASTDATE ( 'Date'[Date] )
    )
)
```

Moving Annual Total (v2)

DATESINPERIOD makes everything much easier.

```
CALCULATE (
    SUM ( Sales[SalesAmount] ),
    DATESINPERIOD (
        'Date'[Date],
        LASTDATE ( 'Date'[Date] ),
        -1,
        YEAR
    )
)
```

Time Intelligence: Conclusions

- Based on evaluation contexts
  - Replace filter on date
  - Many predefined functions
  - You can author your own functions
- Basic Time Intelligence
- Semi Additive Measures
- ISO weeks and fancy calendars
- As often, DAX is the key

sqlbi

Thank you!

Check our articles, whitepapers and courses on

www.sqlbi.com