www.sqlbi.com

sqlbi

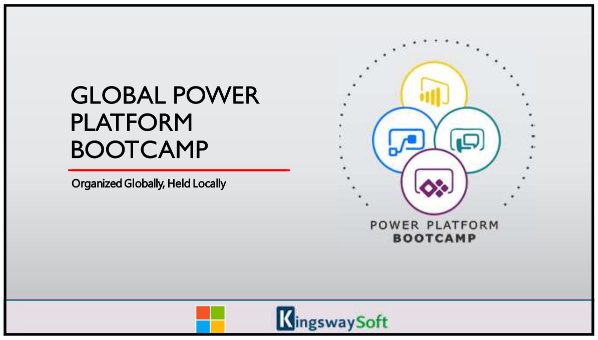
Loader

Microsoft Partner
Gold Business Intelligence
Gold Data Platform

SSAS MAESTRO by Microsoft

MVP Microsoft Most Valuable Professional

1



GLOBAL POWER PLATFORM BOOTCAMP

Organized Globally, Held Locally

POWER PLATFORM
BOOTCAMP
KingswaySoft

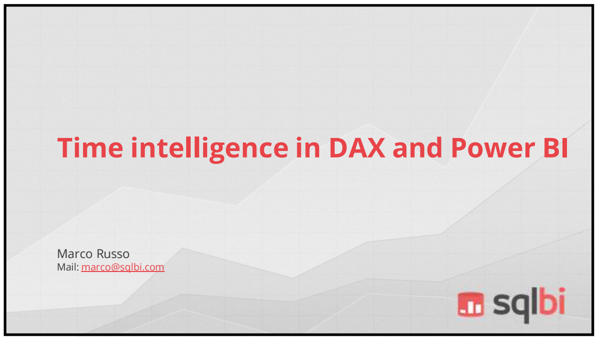
2

Thanks To Our Local Sponsor's

OverNet
EDUCATION

#GlobalPowerPlatformBootcamp

3



Time intelligence in DAX and Power BI

Marco Russo
Mail: marco@sqlbi.com

sqlbi

4

sqlbi.
www.sqlbi.com

DAX Guide
dax.guide

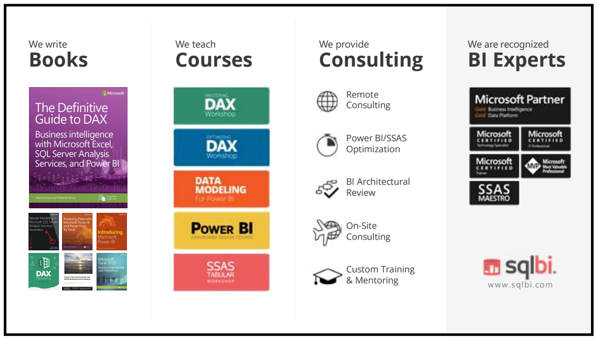
DAX Patterns
www.daxpatterns.com

DAX FORMATTER
www.daxformatter.com

OKVIZ
okviz.com

SYNOPTIC DESIGNER
synoptic.design

5



We write
Books

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

We teach
Courses

- Mastering DAX Workshop
- Optimizing DAX Workshop
- Data Modeling For Power BI
- Power BI Dashboard Design Course
- SSAS Tabular Workshop

We provide
Consulting

- Remote Consulting
- Power BI/SSAS Optimization
- BI Architectural Review
- On-Site Consulting
- Custom Training & Mentoring

We are recognized
BI Experts

Microsoft Partner
Gold Business Intelligence
Gold Data Platform

Microsoft Certified
Microsoft Certified
Microsoft Certified
Microsoft Most Valuable Professional
SSAS MAESTRO

sqlbi.
www.sqlbi.com

6

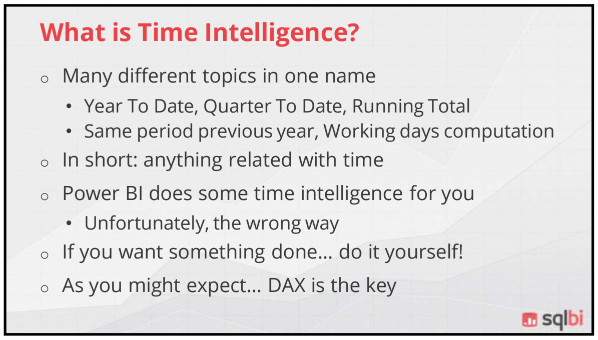
Agenda

- Power BI and dates
- Building a Date table
- Aggregating over time
- Comparison over time
- Custom calendars

Slides and demo

https://sql.bi/651969

7



What is Time Intelligence?

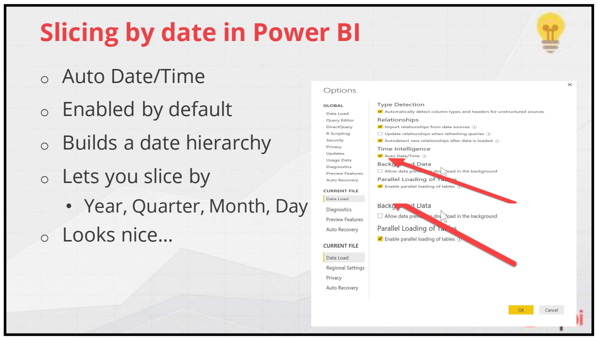
- Many different topics in one name
  - Year To Date, Quarter To Date, Running Total
  - Same period previous year, Working days computation
- In short: anything related with time
- Power BI does some time intelligence for you
  - Unfortunately, the wrong way
- If you want something done… do it yourself!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

8

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

Power BI and dates

9



Slicing by date in Power BI

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

Options

GLOBAL
Data Load
Query Editor
DirectQuery
R Scripting
Security
Privacy
Updates
Usage Data
Diagnostics
Preview Features
Auto Recovery

CURRENT FILE
Data Load
Diagnostics
Preview Features
Auto Recovery

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

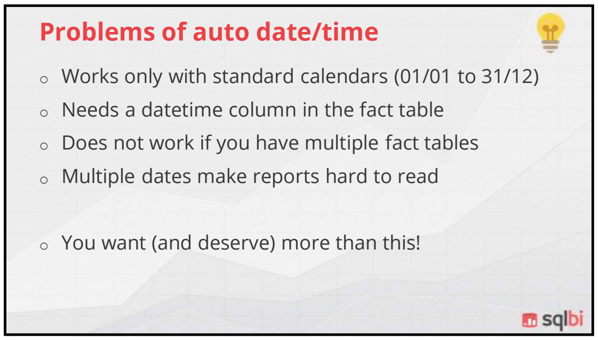
Parallel Loading of Tables
☑ Enable parallel loading of tables

Background Data
□ Allow data preview to download in the background

Parallel Loading of Tables
☑ Enable parallel loading of tables

CURRENT FILE
Data Load
Regional Settings
Privacy
Auto Recovery

10



Problems of auto date/time

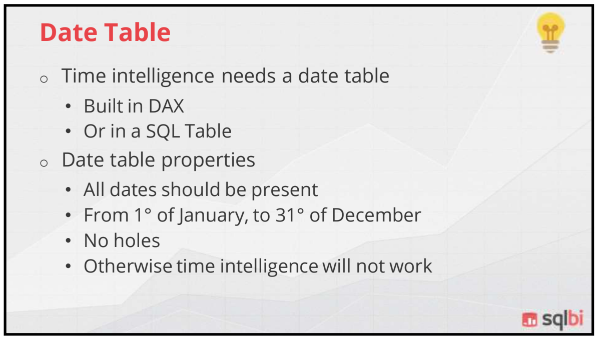
- Works only with standard calendars (01/01 to 31/12)
- Needs a datetime column in the fact table
- Does not work if you have multiple fact tables
- Multiple dates make reports hard to read
- You want (and deserve) more than this!

11

Probably the most important table in your model

Building a Date Table

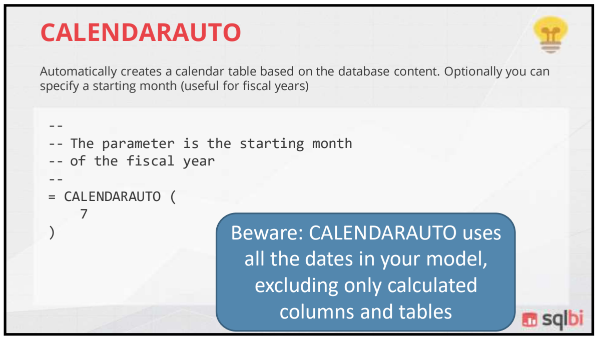
12



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

13



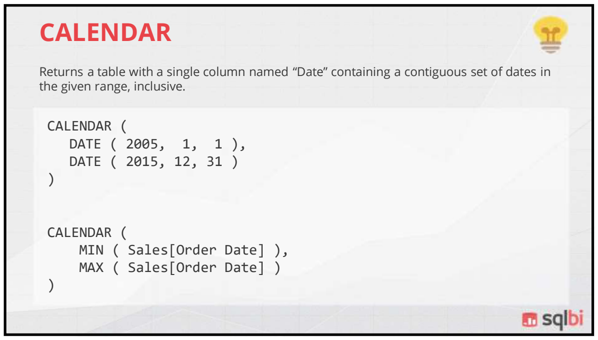
CALENDARAUTO

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

14

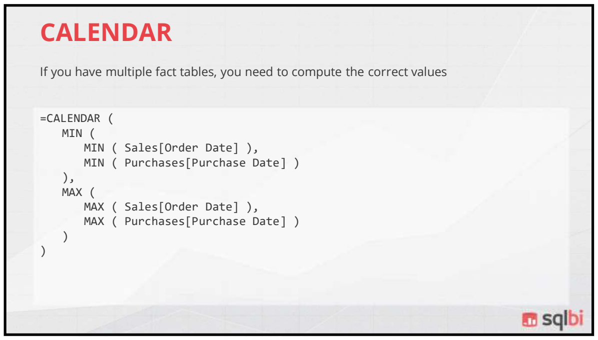


CALENDAR

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] )
)
```

15

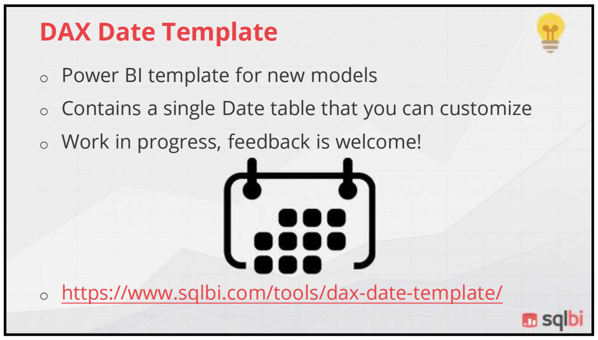


CALENDAR

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] )
    )
)
```

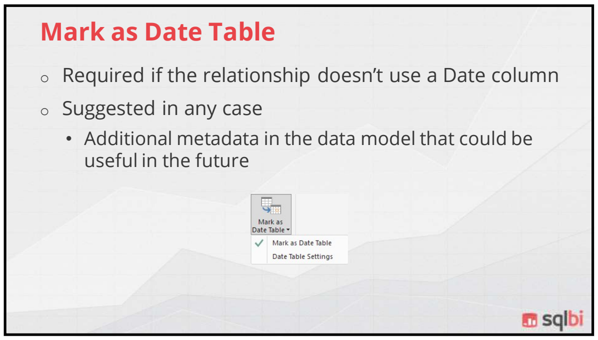
16



DAX Date Template

- Power BI template for new models
- Contains a single Date table that you can customize
- Work in progress, feedback is welcome!
- https://www.sqlbi.com/tools/dax-date-template/

17

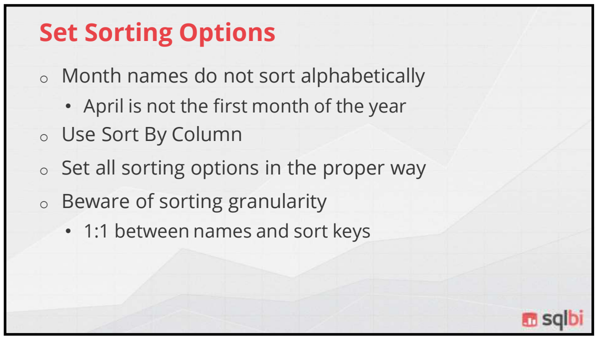


Mark as Date Table

- Required if the relationship doesn't use a Date column
- Suggested in any case
  - Additional metadata in the data model that could be useful in the future

Mark as Date Table
Mark as Date Table
Date Table Settings

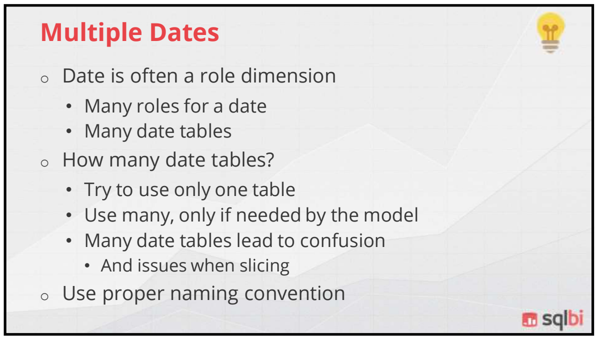
18



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

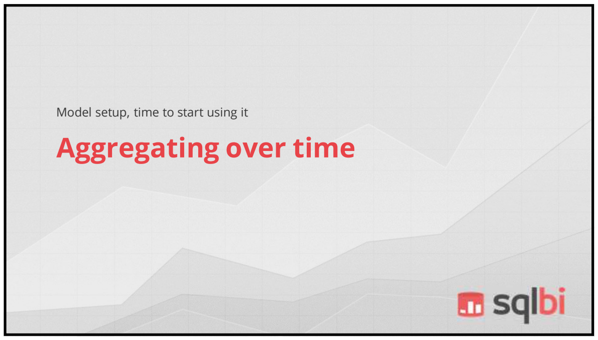
19



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

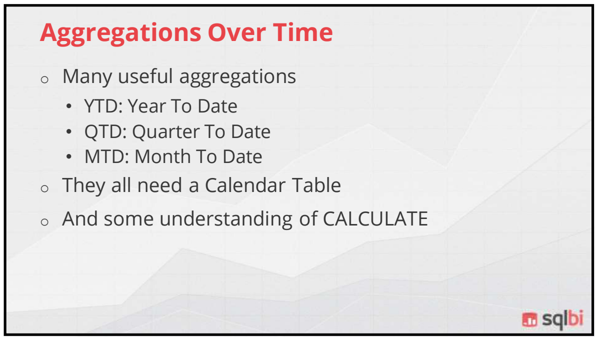
20



Model setup, time to start using it

Aggregating over time

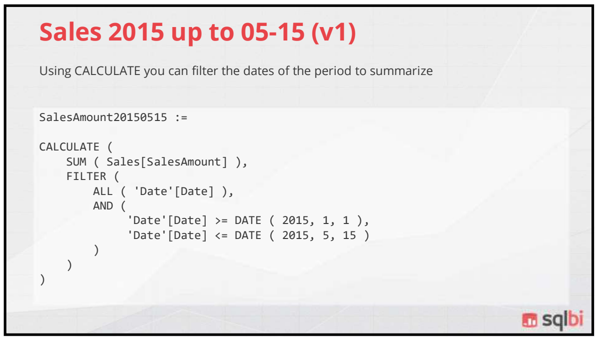
21



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

22

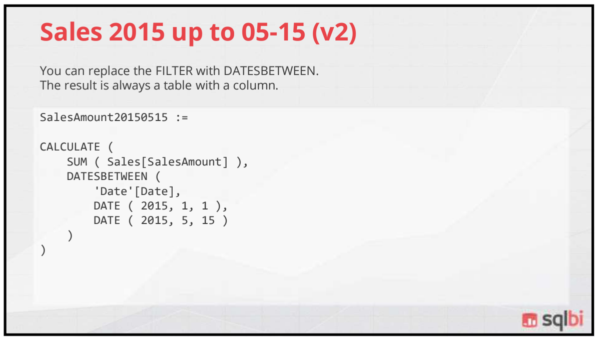


Sales 2015 up to 05-15 (v1)

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 )
        )
    )
)
```

23

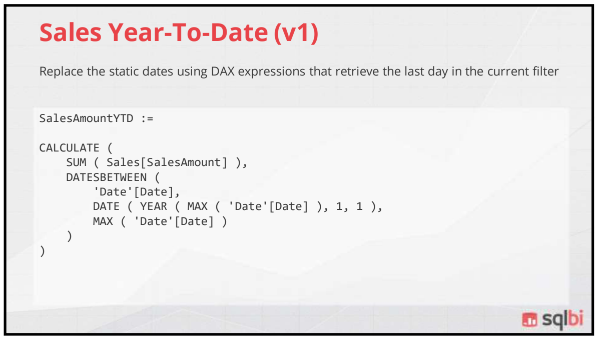


Sales 2015 up to 05-15 (v2)

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 )
    )
)
```

24

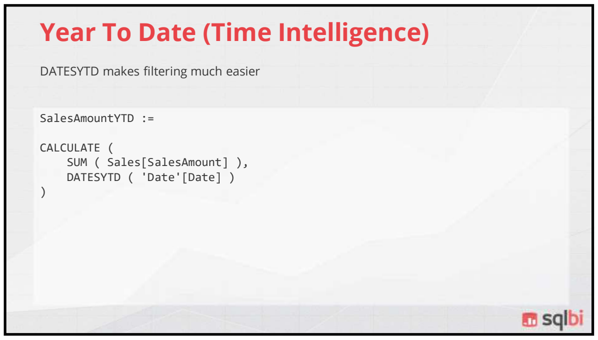


Sales Year-To-Date (v1)

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] )
    )
)
```

25

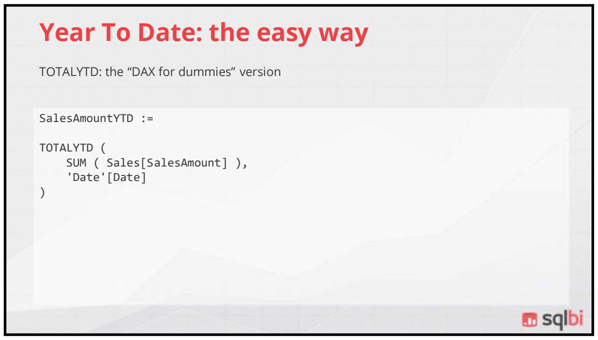


Year To Date (Time Intelligence)

DATESYTD makes filtering much easier

```
SalesAmountYTD :=

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

26

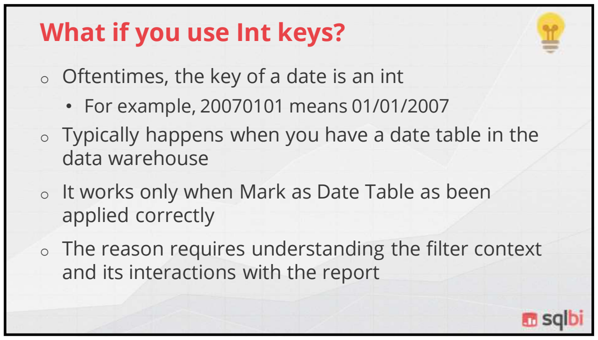


Year To Date: the easy way

TOTALYTD: the “DAX for dummies” version

```
SalesAmountYTD :=

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

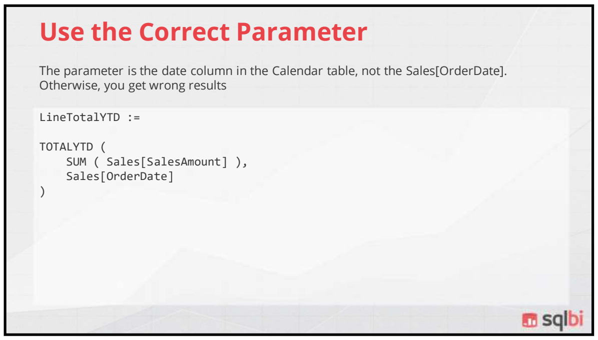
27



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 data warehouse
- It works only when Mark as Date Table as been applied correctly
- The reason requires understanding the filter context and its interactions with the report

28

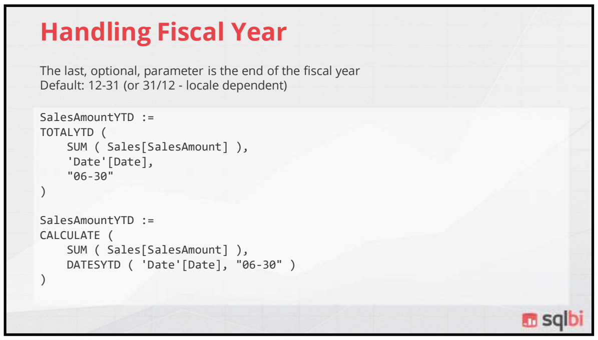


Use the Correct Parameter

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]
)
```

29



Handling Fiscal Year

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

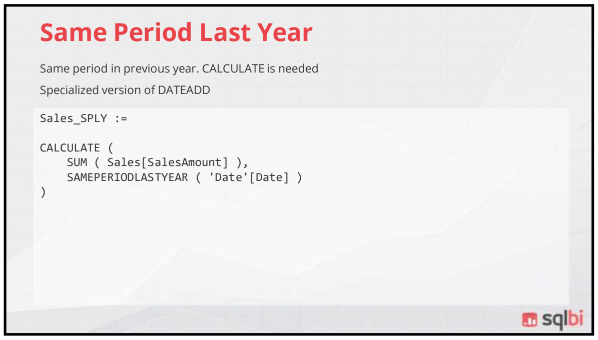
```
SalesAmountYTD :=
TOTALYTD (
    SUM ( Sales[SalesAmount] ),
    'Date'[Date],
    "06-30"
)

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

30

Another type of Time Intelligence calculation

Comparison over time

31



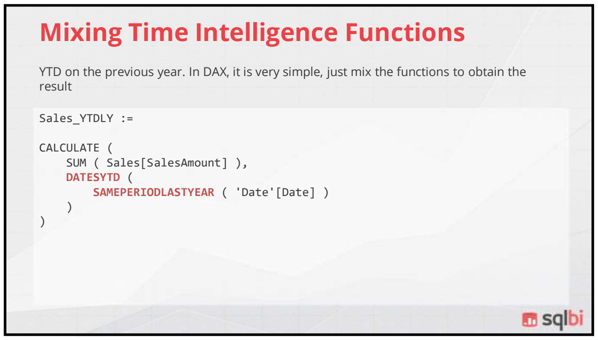
Same Period Last Year

Same period in previous year. CALCULATE is needed

Specialized version of DATEADD

```
Sales_SPLY :=

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

32

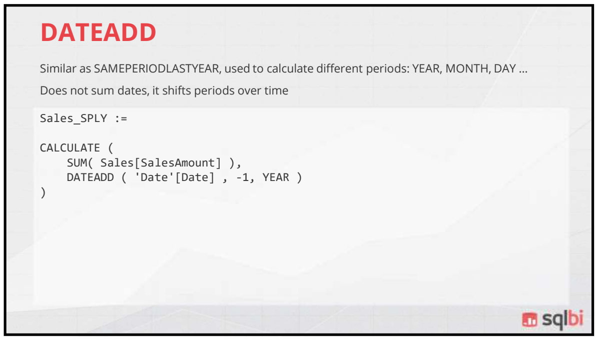


Mixing Time Intelligence Functions

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] )
    )
)
```

33



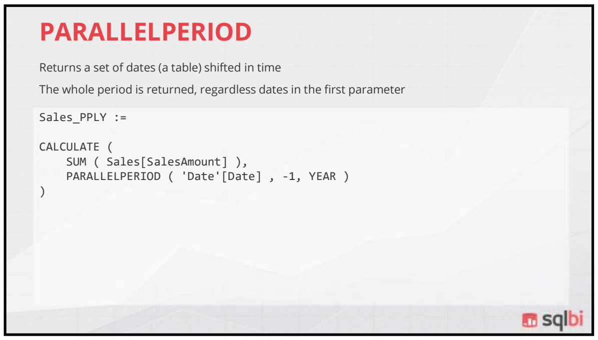
DATEADD

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 )
)
```

34



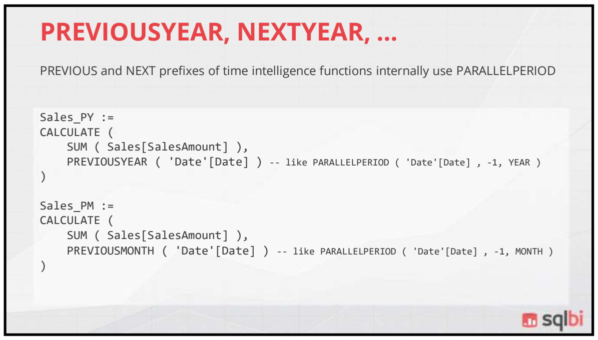
PARALLELPERIOD

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 )
)
```

35

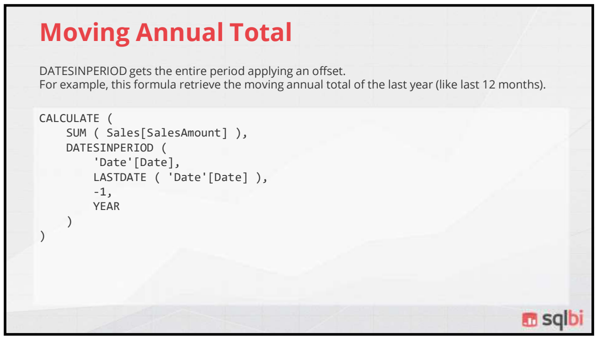


PREVIOUSYEAR, NEXTYEAR, …

PREVIOUS and NEXT prefixes of time intelligence functions internally use PARALLELPERIOD

```
Sales_PY :=
CALCULATE (
    SUM ( Sales[SalesAmount] ),
    PREVIOUSYEAR ( 'Date'[Date] ) -- like PARALLELPERIOD ( 'Date'[Date] , -1, YEAR )
)

Sales_PM :=
CALCULATE (
    SUM ( Sales[SalesAmount] ),
    PREVIOUSMONTH ( 'Date'[Date] ) -- like PARALLELPERIOD ( 'Date'[Date] , -1, MONTH )
)
```

36

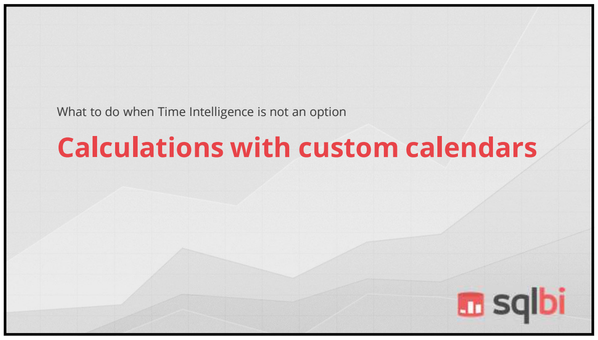


Moving Annual Total

DATESINPERIOD gets the entire period applying an offset.
For example, this formula retrieve the moving annual total of the last year (like last 12 months).

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

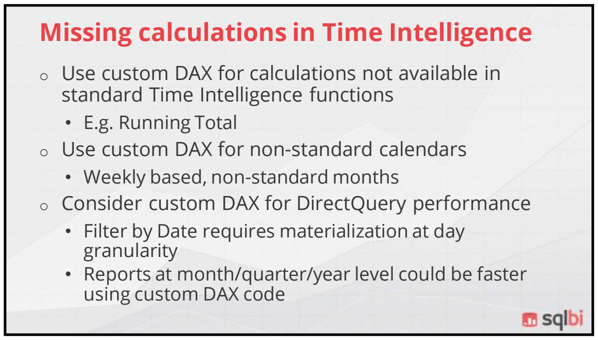
37



What to do when Time Intelligence is not an option

Calculations with custom calendars

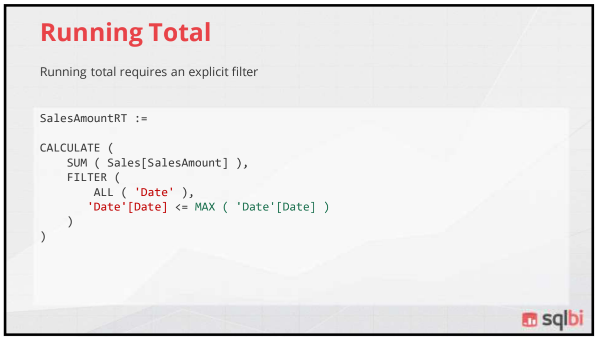
38



Missing calculations in Time Intelligence

- Use custom DAX for calculations not available in standard Time Intelligence functions
  - E.g. Running Total
- Use custom DAX for non-standard calendars
  - Weekly based, non-standard months
- Consider custom DAX for DirectQuery performance
  - Filter by Date requires materialization at day granularity
  - Reports at month/quarter/year level could be faster using custom DAX code

39

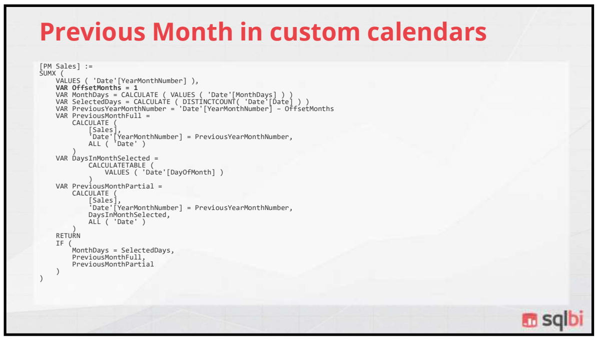


Running Total

Running total requires an explicit filter

```
SalesAmountRT :=

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

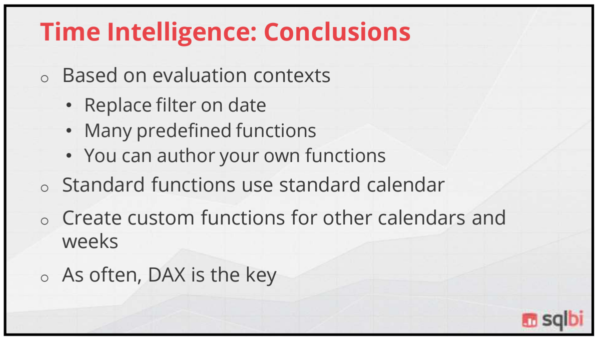
40



Previous Month in custom calendars

```
[PM Sales] :=
SUMX (
    VALUES ( 'Date'[YearMonthNumber] ),
    VAR OffsetMonths = 1
    VAR MonthDays = CALCULATE ( VALUES ( 'Date'[MonthDays] ) )
    VAR SelectedDays = CALCULATE ( DISTINCTCOUNT( 'Date'[Date] ) )
    VAR PreviousYearMonthNumber = 'Date'[YearMonthNumber] - OffsetMonths
    VAR PreviousMonthFull =
        CALCULATE (
            [Sales],
            'Date'[YearMonthNumber] = PreviousYearMonthNumber,
            ALL ( 'Date' )
        )
    VAR DaysInMonthSelected =
            CALCULATETABLE (
                VALUES ( 'Date'[DayOfMonth] )
            )
    VAR PreviousMonthPartial =
        CALCULATE (
            [Sales],
            'Date'[YearMonthNumber] = PreviousYearMonthNumber,
            DaysInMonthSelected,
            ALL ( 'Date' )
        )
    RETURN
    IF (
        MonthDays = SelectedDays,
        PreviousMonthFull,
        PreviousMonthPartial
    )
)
```

41



Time Intelligence: Conclusions

- Based on evaluation contexts
  - Replace filter on date
  - Many predefined functions
  - You can author your own functions
- Standard functions use standard calendar
- Create custom functions for other calendars and weeks
- As often, DAX is the key

42

Thank you!

Check our articles, whitepapers and courses on
www.sqlbi.com

Slides and demo

https://sql.bi/651969

43