

Query Folding in Power BI

Who Are You? <http://bit.ly/query-folding>

Nicky van Vroenhoven

Lead Expert Data management
Van Lanschot Kempen



@NickyvV



nickyvv.com



Power BI Super User
community.powerbi.com



Power BI Days Dutch chapter co-founder
powerbidays.com



Data Platform MVP

Who Are You? <http://bit.ly/query-folding>




Who Are You?

Munich P

ent Date
tember 15, 2020

rt: 10:00 am
l: 5:00 pm



Query Folding - Who Are You? #PowerBIDays

September 15th, 2020 - Webinar @ Power BI Days Munich

* Required

1. What is your name? *

Check out my blog for a how-to:
www.nickyvv.com/2019/12

Who Are You? <http://bit.ly/query-folding>

Agenda

- Power Query
- Query Folding
- Constraints
- Demo

Agenda

- Power Query
- Query Folding
- Constraints
- Demo

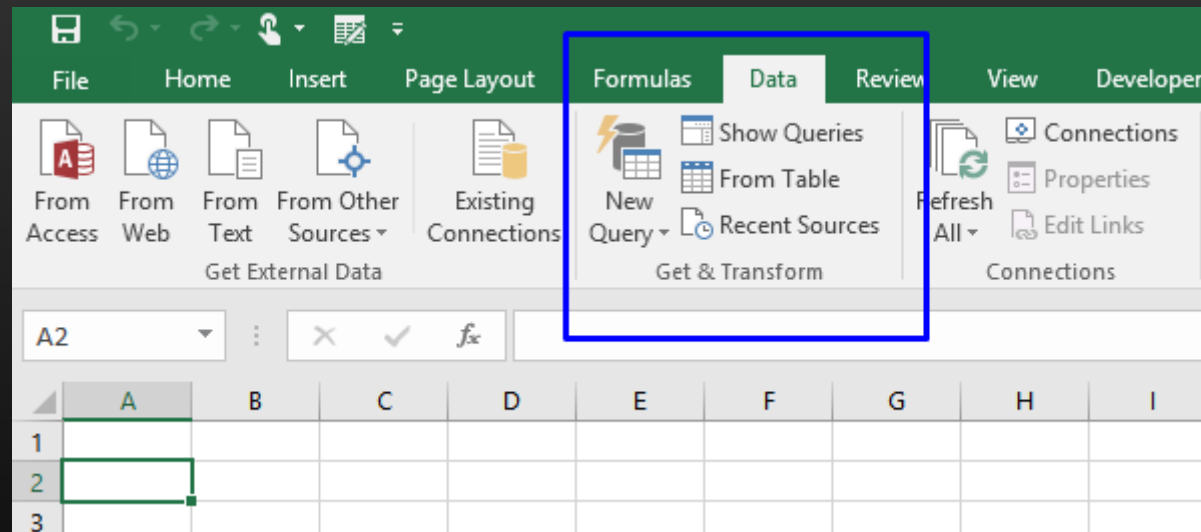
Power Query



+



Power Query



Power Query

New dataflow (unsaved)

Power Query

Edit queries

Get data Refresh Options Manage columns Transform table Reduce rows Add column

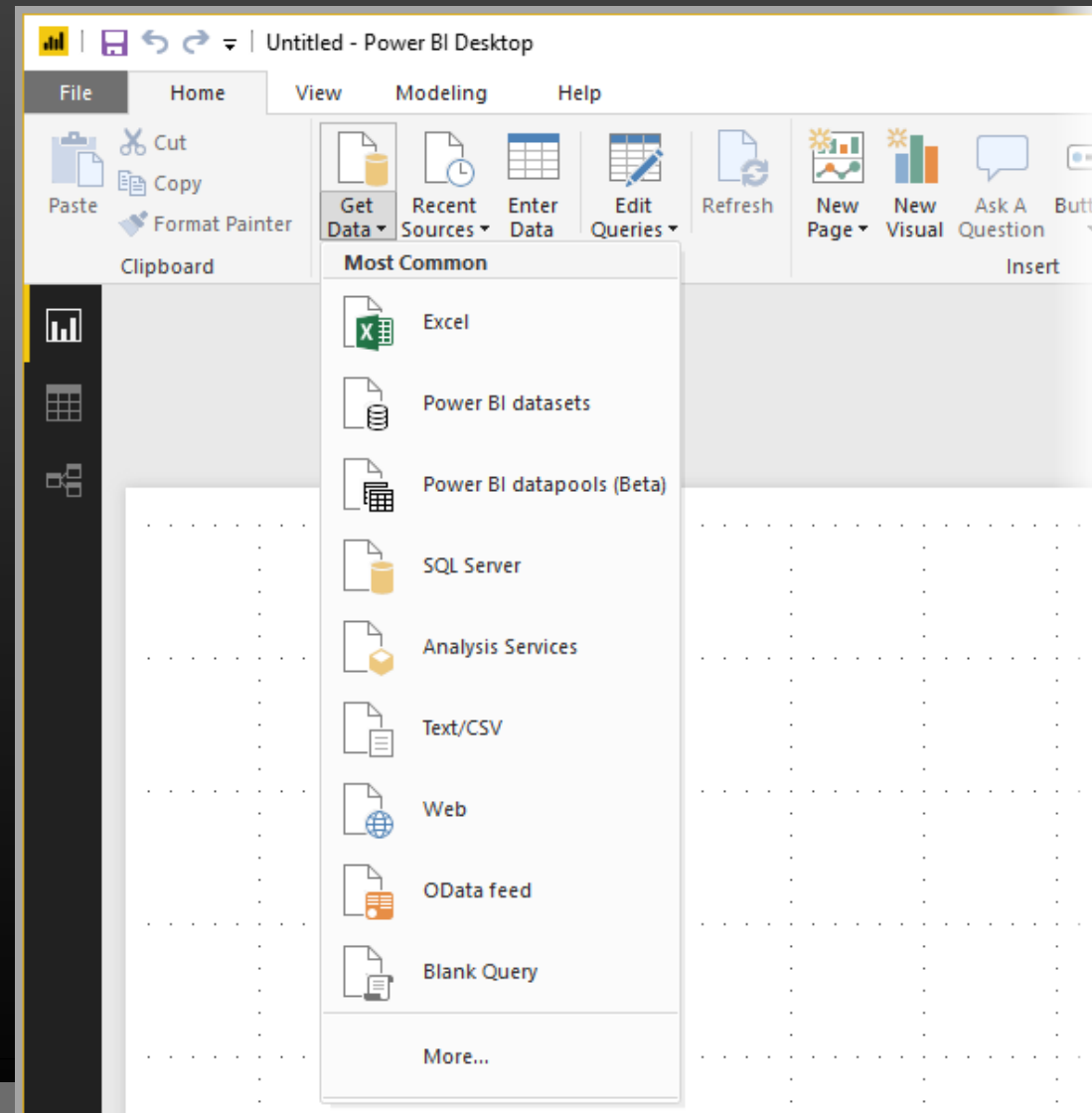
COR transaction_status

Source{[Schema, ..._status"]}[Data]

- Group by
- Use first row as headers
- Use headers as first row
- Transpose
- Reverse rows
- Count rows

	meta_dt_load	meta_nm	meta_dt_reportdate	meta_corr_nr
1	10/3/2018, 8:14:00 PM	EM_MIFID_cor	10/3/2018	
2	10/2/2018, 3:01:00 PM	EM_MIFID_xm	10/3/2018	
3	10/2/2018, 3:01:00 PM	EM_MIFID_xm	10/3/2018	
4	10/2/2018, 3:01:00 PM	EM_MIFID_xm	10/3/2018	

Power Query



Agenda

- Power Query
- Query Folding
- Constraints
- Demo

Converts M to native query

Fact Movement

```

1 let
2     Source = Sql.Databases("."),
3     WideWorldImportersDW = Source{[Name="WideWorldImportersDW"]}[Data],
4     Fact_Movement = WideWorldImportersDW{[Schema="Fact",Item="Movement"]}[Data],
5     #"Filtered Rows" = Table.SelectRows(Fact_Movement, each Date.QuarterOfYear([Date Key]) = 1),
6     #"Expanded Dimension.Customer" = Table.ExpandRecordColumn(#"Filtered Rows", "Dimension.Customer", {"Customer"}, {1}),
7     #"Uppercased Text" = Table.TransformColumns(#"Expanded Dimension.Customer",{{"Dimension.Customer.Customer", Text.Uppercase}}, 1),
8     #"Expanded Dimension.Supplier" = Table.ExpandRecordColumn(#"Uppercased Text", "Dimension.Supplier", {"Primary"}, {1}),
9     #"Merged Columns" = Table.CombineColumns(Table.TransformColumnTypes(#"Expanded Dimension.Supplier", {{"Dimension.Supplier", Text.Uppercase}}, 1), {"Customer", "Supplier"}, 1),
10 in
11     #"Merged Columns"

```

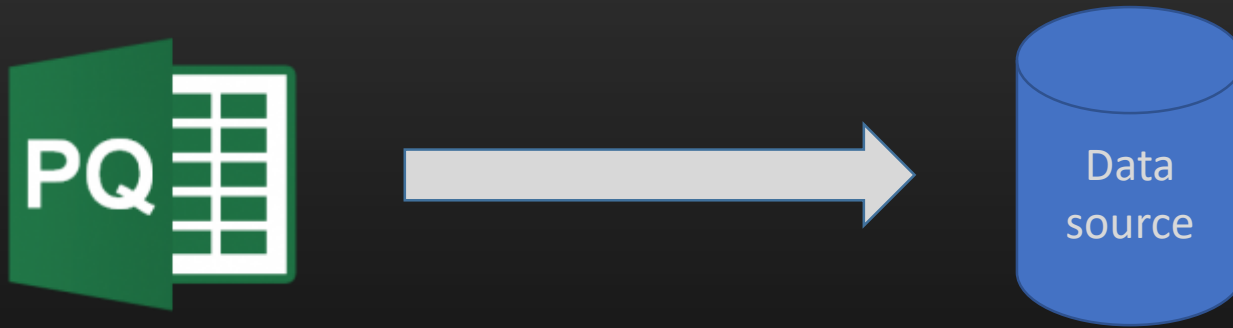
Native Query

```

select [$Outer].[Movement Key] as [Movement Key],
[$Outer].[Date Key] as [Date Key],
[$Outer].[Stock Item Key] as [Stock Item Key],
[$Outer].[Customer Key2] as [Customer Key],
[$Outer].[Supplier Key2] as [Supplier Key],
[$Outer].[Transaction Type Key] as [Transaction Type Key],
[$Outer].[WWI Stock Item Transaction ID] as [WWI Stock Item Transaction ID],
[$Outer].[WWI Invoice ID] as [WWI Invoice ID],
[$Outer].[WWI Purchase Order ID] as [WWI Purchase Order ID],
[$Outer].[Quantity] as [Quantity],
[$Outer].[Lineage Key2] as [Lineage Key],
[$Outer].[t0_0] as [Dimension.Customer.Customer],
[$Outer].[Category2] as [Dimension.Customer.Category],
[$Inner].[Primary Contact] as [Dimension.Supplier.Primary Contact]
from
(
    select [_].[Movement Key] as [Movement Key],
        [_].[Date Key] as [Date Key],
        [_].[Stock Item Key] as [Stock Item Key],
        [_].[Customer Key2] as [Customer Key2],
        [_].[Supplier Key] as [Supplier Key2],
        [_].[Transaction Type Key] as [Transaction Type Key],
        [_].[WWI Stock Item Transaction ID] as [WWI Stock Item Transaction ID],
        [_].[WWI Invoice ID] as [WWI Invoice ID],
        [_].[WWI Purchase Order ID] as [WWI Purchase Order ID],
        [_].[Quantity] as [Quantity],
        [_].[Lineage Key2] as [Lineage Key2],
        [_].[Category] as [Category2],
        upper([_].[Customer]) as [t0_0]
    from
    (
        select [$Outer].[Movement Key],
            [$Outer].[Date Key],
            [$Outer].[Stock Item Key],

```

Transformation pushed back to data source



From M to SQL, MySQL, oData, etc

More efficient data processing



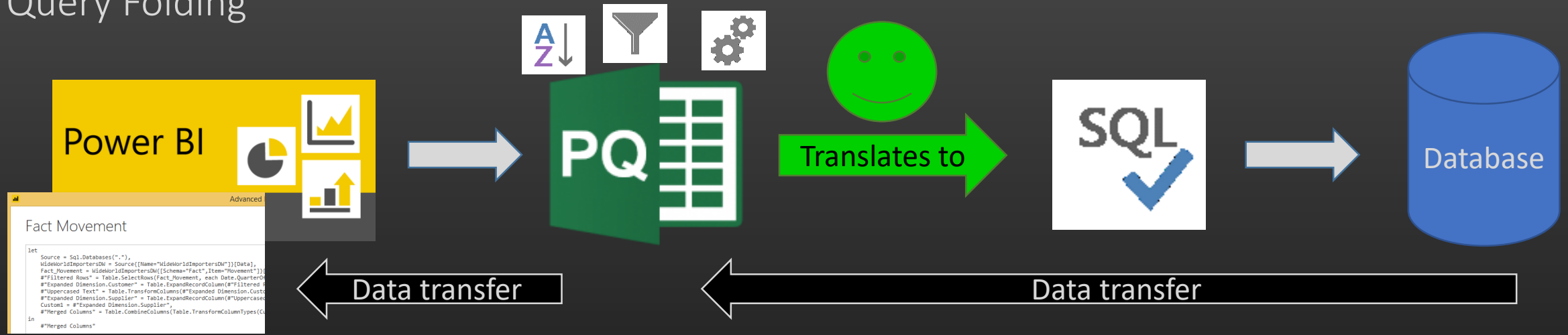
Partial folding



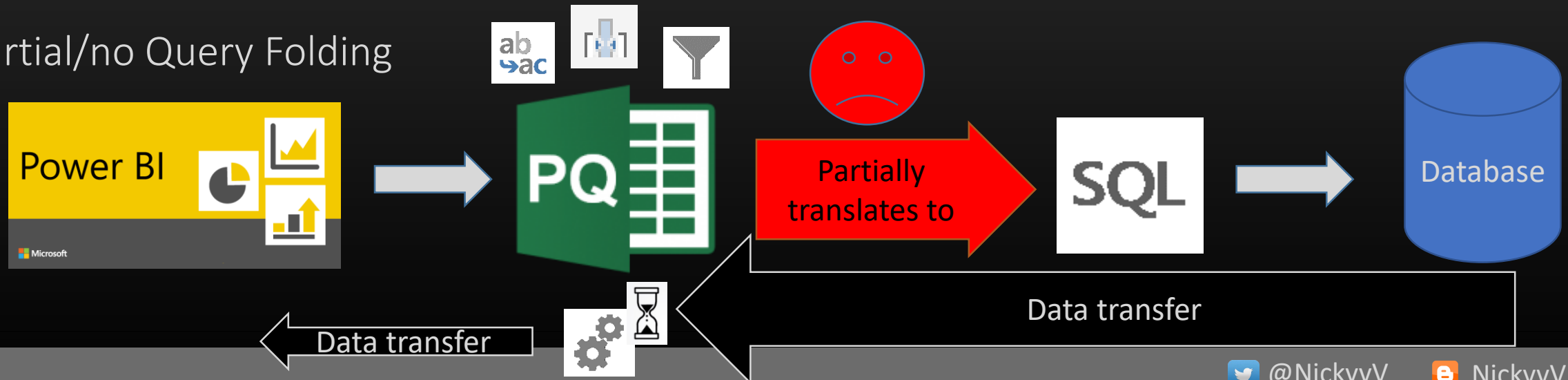
Importance

- Import model tables
- DirectQuery and Dual storage mode tables
- Incremental refresh

Query Folding



Partial/no Query Folding



To Fold or Not To Fold

- Import mode only

SQL Server database

Server ①

Database (optional)

Data Connectivity mode ①

☒ Import

☐ DirectQuery

Advanced options

Command timeout in minutes (optional)

SQL statement (optional, requires database)

☒ Include relationship columns

☐ Navigate using full hierarchy

☐ Enable SQL Server Failover support

OK Cancel

Supported Data Sources



ORACLE®



Supported Transformations

(most commonly used)



Filter



Sort



(Un)Pivot



String
Manipulations



Top Rows

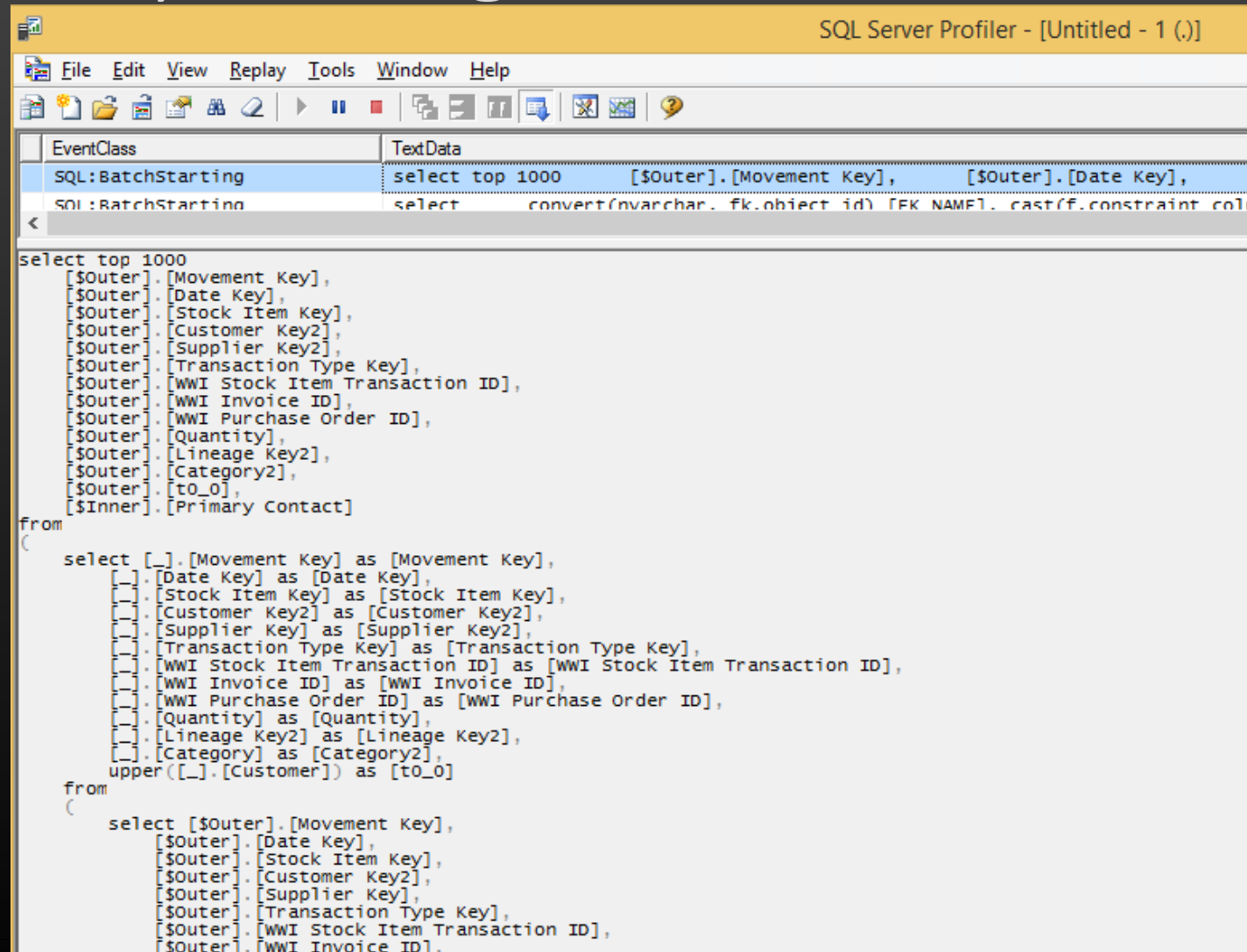


Join
(Append)



Group By

Is My Query Folding? - SQL

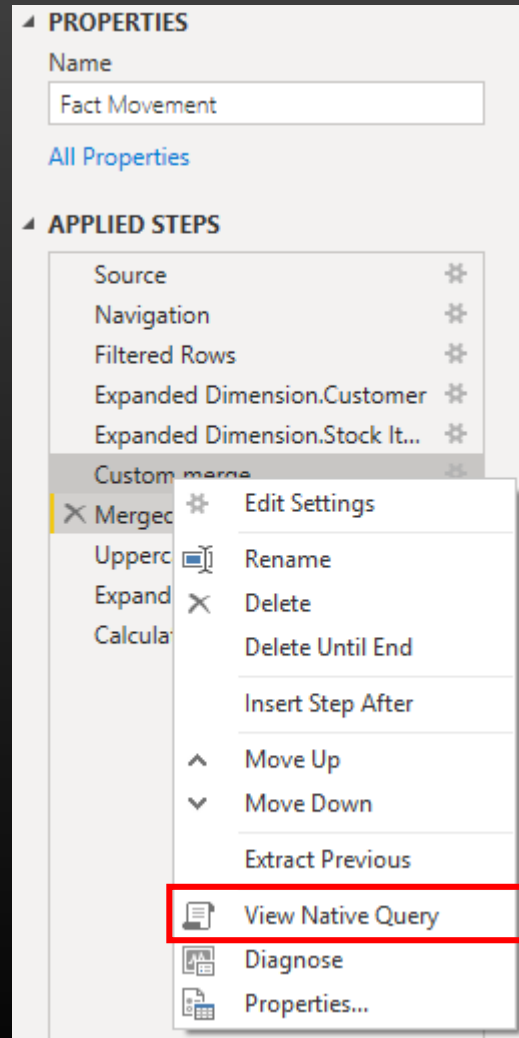


The screenshot shows the SQL Server Profiler interface. The top menu bar includes File, Edit, View, Replay, Tools, Window, and Help. Below the menu is a toolbar with various icons. The main window is divided into two panes. The left pane, titled 'EventClass', shows a list of events with 'SQL:BatchStarting' selected. The right pane, titled 'TextData', shows the SQL query text for the selected event. The query is a complex SELECT statement with multiple columns and a subquery in the FROM clause.

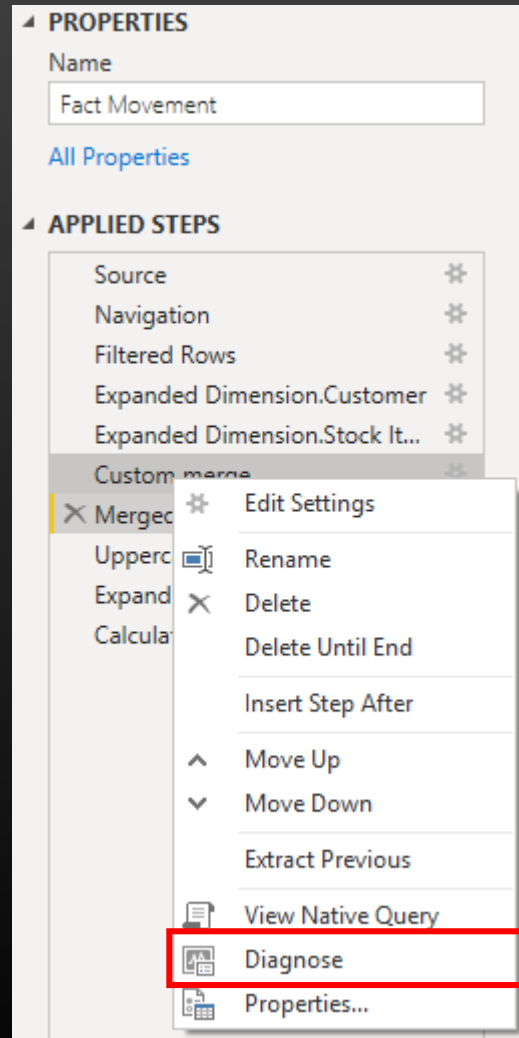
```
select top 1000
    [$Outer].[Movement Key],
    [$Outer].[Date Key],
    [$Outer].[Stock Item Key],
    [$Outer].[Customer Key2],
    [$Outer].[Supplier Key2],
    [$Outer].[Transaction Type Key],
    [$Outer].[WWI Stock Item Transaction ID],
    [$Outer].[WWI Invoice ID],
    [$Outer].[WWI Purchase Order ID],
    [$Outer].[Quantity],
    [$Outer].[Lineage Key2],
    [$Outer].[Category2],
    [$Outer].[t0_0],
    [$Inner].[Primary Contact]
from
(
    select [_].[Movement Key] as [Movement Key],
        [_].[Date Key] as [Date Key],
        [_].[Stock Item Key] as [Stock Item Key],
        [_].[Customer Key2] as [Customer Key2],
        [_].[Supplier Key] as [Supplier Key2],
        [_].[Transaction Type Key] as [Transaction Type Key],
        [_].[WWI Stock Item Transaction ID] as [WWI Stock Item Transaction ID],
        [_].[WWI Invoice ID] as [WWI Invoice ID],
        [_].[WWI Purchase Order ID] as [WWI Purchase Order ID],
        [_].[Quantity] as [Quantity],
        [_].[Lineage Key2] as [Lineage Key2],
        [_].[Category] as [Category2],
        upper([_].[Customer]) as [t0_0]
    from
    (
        select [$Outer].[Movement Key],
            [$Outer].[Date Key],
            [$Outer].[Stock Item Key],
            [$Outer].[Customer Key2],
            [$Outer].[Supplier Key],
            [$Outer].[Transaction Type Key],
            [$Outer].[WWI Stock Item Transaction ID],
            [$Outer].[WWI Invoice ID],
```



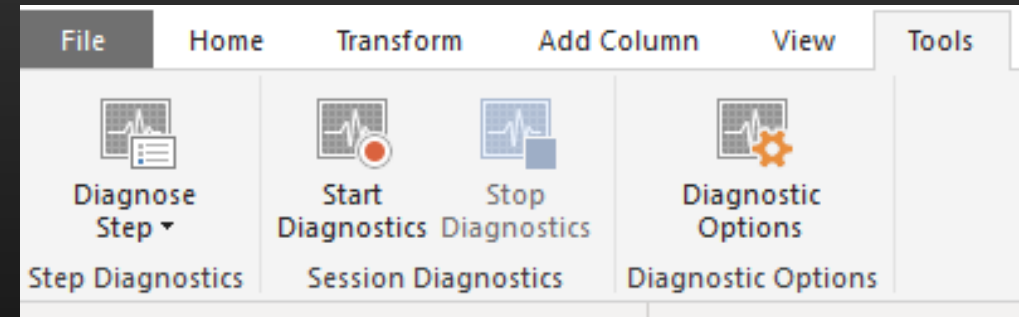
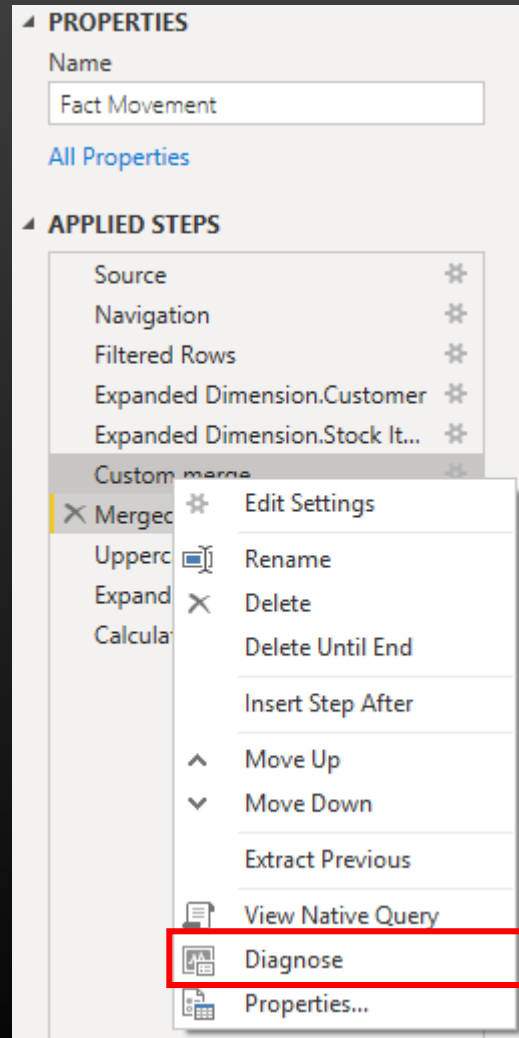
Is My Query Folding? - SQL



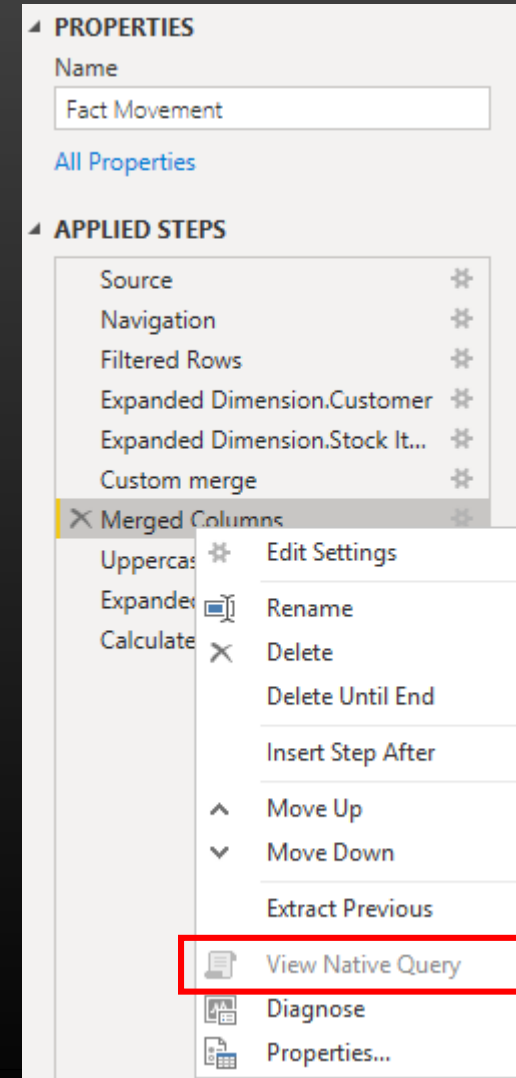
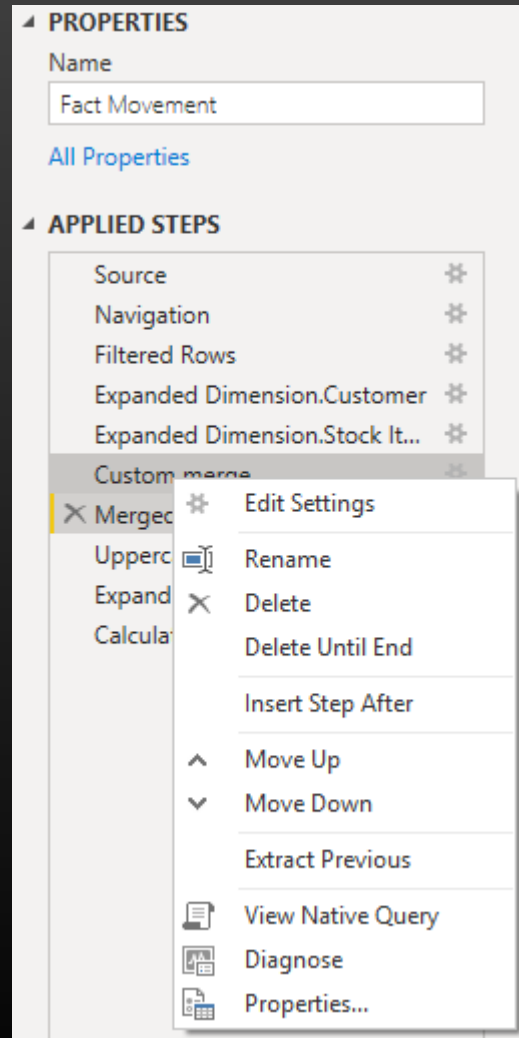
Is My Query Folding? - SQL



Is My Query Folding? - SQL



Is My Query Folding? - SQL



Is My Query Folding? - OData

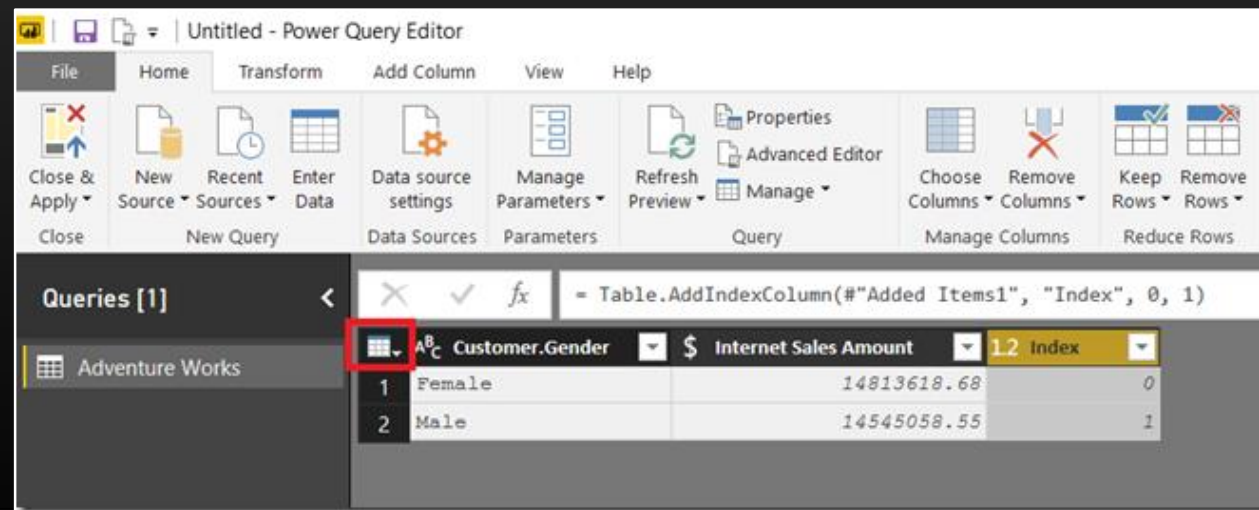
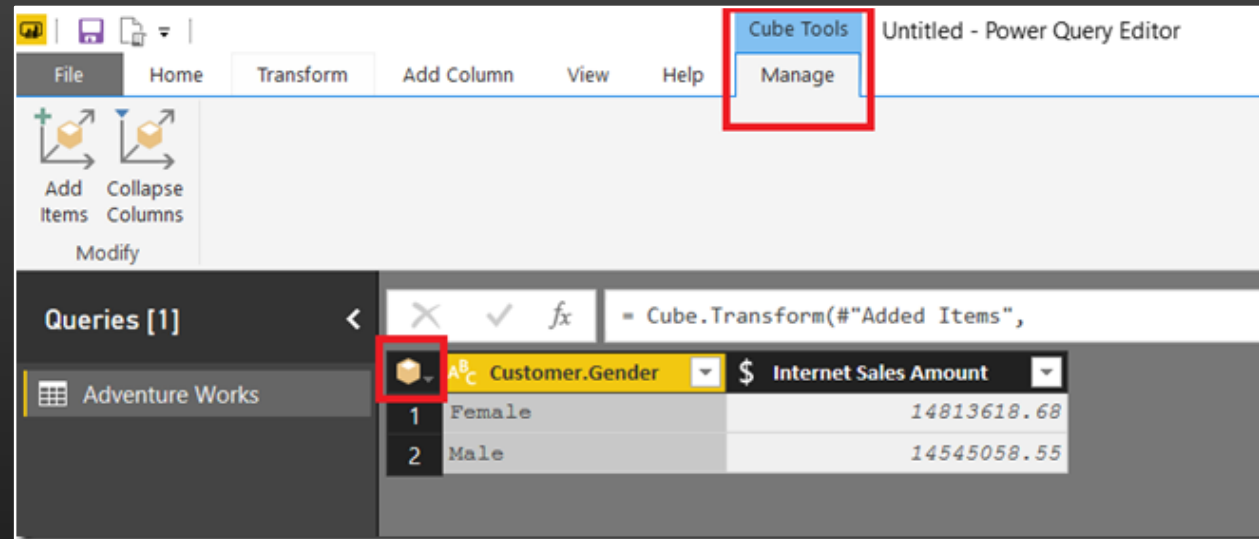
Progress Telerik Fiddler Web Debugger

File Edit Rules Tools View Help

WinConfig Replay Go Stream Decode Keep: All sessions Any Process

#	Result	Protocol	Host	URL
1	200	HTTPS	api.parliament.uk	/Staging/odata
2	200	HTTPS	api.parliament.uk	/odata/\$metadata
3	200	HTTPS	api.parliament.uk	/Staging/odata/GovernmentOrganisation?\$top=1000
4	200	HTTPS	api.parliament.uk	/Staging/odata
5	200	HTTPS	api.parliament.uk	/odata/\$metadata

Is My Query Folding? - SSAS



Agenda

- Power Query
- Query Folding
- **Constraints**
- Demo

Limitations

- Unsupported data sources
- Unsupported transformations
- Data source privacy level
- Native database query

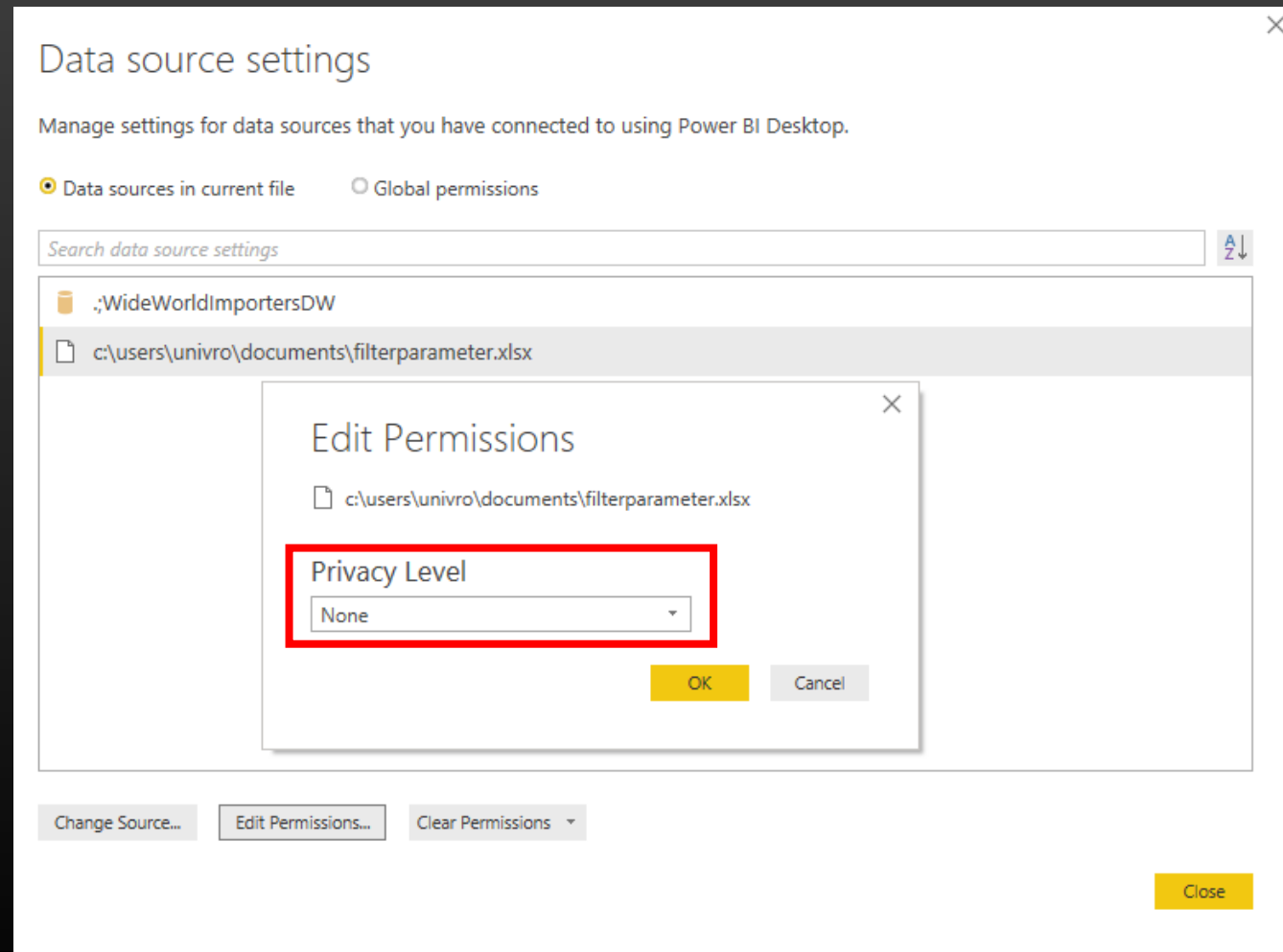
Limitations – Unsupported Transformations

- Merge columns
 - *Solved with a custom column*
- Split column
- Add Index column
- Table.Buffer() M function
- Complex Group By
- Remove errors, duplicates
- DateTime to Date (Transform vs Date Only)
- And more...

Limitations – Unsupported Transformations

- **Merge columns**
 - *Solved with a custom column*
- Split column
- Add Index column
- Table.Buffer() M function
- Complex Group By
- Remove errors, duplicates
- **DateTime to Date (Transform or Date Only)**
- And more...

Limitations – Data source privacy level



Limitations – Data source privacy level

```
FROM
(
    select [$Outer].[Movement Key],
           [$Outer].[Date Key],
           [$Outer].[Stock Item Key],
           [$Outer].[Customer Key],
           [$Outer].[Supplier Key],
           [$Outer].[Transaction Type Key],
           [$Outer].[WWI Stock Item Transaction ID],
           [$Outer].[WWI Invoice ID],
           [$Outer].[WWI Purchase Order ID],
           [$Outer].[Quantity],
           [$Outer].[Lineage Key],
           [$Inner].[Calendar Year]
    from [Fact].[Movement] as [$Outer]
    left outer join [Dimension].[Date] as [$Inner] on ([$Outer].[Date Key] = [$Inner].[Date Key])
) as [_]
where [_].[Calendar Year] = 2018 and [_].[Calendar Year] is not null
) as [_]
as [$Outer]
left outer join [Dimension].[Customer] as [$Inner] on ($Outer.[Customer Key] = $Inner.[Customer Key])
```


Limitations – Native Database Queries

Query the source directly

- PQ isn't sufficient
- Specific requirements/optimizations
- Existing SQL-query
- Stored Procedures

Limitations – Native Database Queries

- Functions on top of the query results aren't folded
- No parsing takes place
- No guarantees on frequency of execution





Demo Time!

Demo Recap

- Import data from SQL Server
- Transform data in several ways

Take-aways:

- Filter first, transform later
- View Native Query



Improvements

- Desktop June 2018:
 - ODBC connector improvements
 - Folding support for top rows
 - OData V4 connector enhancements:
 - Use [Implementation="2.0"] in OData.Feed()
- Relative date slicers now also work
- Desktop February 2019:
 - PBI Premium (& pro!): Incremental refresh query verification
- Desktop June 2019:
 - Cosmos DB connector enhancements
- Desktop September 2019:
 - PostgreSQL connector enhancements over a native query

Improvements

- Desktop June 2018:
 - ODBC connector improvements
 - Folding support for top rows
 - OData V4 connector enhancements:
 - Use [Implementation="2.0"] in OData.Feed()
- **Relative date slicers now also work**
- **Desktop February 2019:**
 - **PBI Premium (& pro!): Incremental refresh query verification**
- Desktop June 2019:
 - Cosmos DB connector enhancements
- Desktop September 2019:
 - PostgreSQL connector enhancements over a native query

Wrap-up

- Concept of Query Folding
- Supported data sources, transformations
- Don't write native SQL and add steps afterwards
- View Native Query!

Resources

Power Query documentation:

- <https://docs.microsoft.com/en-us/power-query/power-query-folding>
- <https://docs.microsoft.com/en-us/power-query/querydiagnosticsfolding>
- <https://docs.microsoft.com/en-us/power-query/handlingqueryfolding>

Blogs on Query Folding:

- <https://www.mssqltips.com/sqlservertip/3635/query-folding-in-power-query-to-improve-performance/>
- <http://geekswithblogs.net/darengosbell/archive/2014/05/16/function-folding-in-powerquery.aspx>
- <http://www.mattmasson.com/2013/07/filtering-in-data-explorer/>
- <https://www.mssqltips.com/sqlservertip/4563/power-bi-native-query-and-query-folding/>
- <https://blog.crossjoin.co.uk/2018/06/27/odata-performance-power-bi/>

Power BI Desktop updates

- <https://powerbi.microsoft.com/en-us/blog/power-bi-desktop-june-2018-feature-summary/#folding>

Power BI Premium Incremental refresh

- <https://powerbi.microsoft.com/en-us/blog/incremental-refresh-query-folding/>
- <https://docs.microsoft.com/en-us/power-bi/service-premium-incremental-refresh#query-folding>
- <https://www.youtube.com/watch?v=RnrCKn5yKcU>

Thank you

and let's connect!



NickyvV.com



@NickyvV



www.linkedin.com/in/nickyvanvroenhoven/