

Advanced Power BI Modeling Techniques

by Martin Schoombee

About Me

“I help people make sense of their data”

My blog: martinschoombee.com

My company: 28twelve.consulting

Tweet me: [@MartinSchoombee](https://twitter.com/MartinSchoombee)

GitHub Repo:

<https://github.com/mschoombee/Presentations/wiki/Presentations>

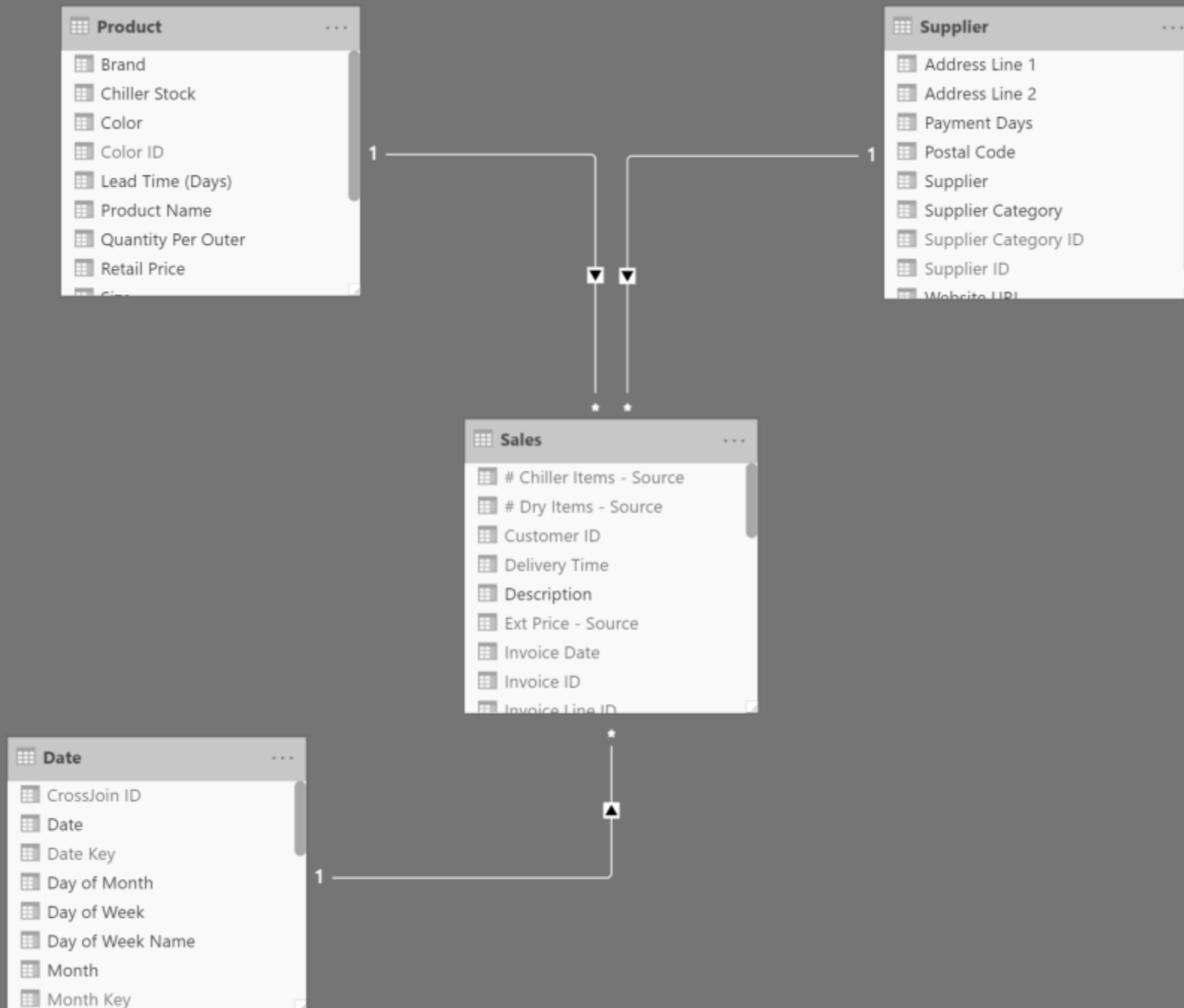


Agenda

- Measure Tables
- Dynamic Currency Conversions
- Role-Playing Dimensions
- (Bonus Content) Dynamic Month Bands

Measure Table?





Supplier Category	Qty	Ext Price	Profit
Clothing Supplier	2,624,338	\$53,407,430.60	\$21,929,432.50
Novelty Goods Supplier	525,778	\$11,849,641.64	\$6,071,706.15
Packaging Supplier	5,696,347	\$115,763,819.96	\$51,663,616.75
Toy Supplier	104,165	\$17,022,547.25	\$6,064,425.50
Total	8,950,628	\$198,043,439.45	\$85,729,180.90

Measure Name	Clothing Supplier	Novelty Goods Supplier	Packaging Supplier	Toy Supplier	Total
Qty	2,624,338	525,778	5,696,347	104,165	8,950,628
Ext Price	53,407,431	11,849,642	115,763,820	17,022,547	198,043,439
Profit	21,929,433	6,071,706	51,663,617	6,064,426	85,729,181

Demo Time



File

Home

Insert

Modeling

View

Help

Format

Data / Drill

Paste

Cut

Copy

Format painter

Get data

Excel

Power BI datasets

SQL Server

Enter data

Recent sources

Transform data

Refresh

New visual

Text box

More visuals

New measure

Quick measure

Publish

Clipboard

Data

Queries

Insert

Calculations

Share

Qty

Ext Price

Profit

8,950,628

\$198,043,439.45

\$85,729,180.90

Visualizations

Fields

Visualizations

Rows

Columns

Values

Drill through

Add data fields here

Add data fields here

Qty

Ext Price

Profit

Add drill-through fields here

Fields

Search

Database Name

Date

Metadata

Product

Sales

Server Name

Source Query - Sales

Source Query - Stock It...

Source Query - Supplier

Supplier

Address Line 1

Address Line 2

Payment Days

Postal Code

Supplier

Supplier Category

Supplier Categor...

Supplier ID

Website URL

Step 1

Create a calculated table

ROW

DAX Function (Table manipulation)

[Syntax](#) | [Return values](#) | [Examples](#) | [Articles](#)

Returns a single row table with new columns specified by the DAX expressions.

Syntax

```
ROW ( <Name>, <Expression> [, <Name>, <Expression> [, ... ] ] )
```

PARAMETER	ATTRIBUTES	DESCRIPTION
Name	Repeatable	Name of the new column.
Expression	Repeatable	The expression for the column.

Return values

TABLE An entire table or a table with one or more columns.

A single row table.

UNION

DAX Function (Table manipulation)

[Syntax](#) | [Return values](#) | [Remarks](#) | [Related](#)

Returns the union of the two tables whose columns match.

Syntax

```
UNION ( <Table> [, <Table> [, ... ] ] )
```

PARAMETER	ATTRIBUTES	DESCRIPTION
Table	Repeatable	A table that will participate in the crossjoin union.

Return values

TABLE

An entire table or a table with one or more columns.

A table that contains all the rows from each of the table expressions.

My Measures =

UNION

```
(  
    ROW("Measure Name", "Qty", "Sort Order", 1)  
,  
    ROW("Measure Name", "Ext Price", "Sort Order", 2)  
,  
    ROW("Measure Name", "Profit", "Sort Order", 3)  
)
```

DATATABLE

DAX Function (Table manipulation)

[Syntax](#) | [Return values](#) | [Remarks](#) | [Examples](#) | [Articles](#)

Returns a table with data defined inline.

Syntax

```
DATATABLE ( <name>, <type> [, <name>, <type> [, ... ] ], <data> )
```

PARAMETER	ATTRIBUTES	DESCRIPTION
name	Repeatable	A column name to be defined.
type	Repeatable	A type name to be associated with the column.
data		The data for the table.

Return values



TABLE

An entire table or a table with one or more columns.

A table declaring an inline set of values.

```
My Measures =  
DATATABLE  
(  
    "Measure Name", STRING  
    , "Sort Order", INTEGER  
    , {  
        {"Qty", 1}  
        , {"Ext Price", 2}  
        , {"Profit", 3}  
    }  
)
```


Table Constructor

12/09/2018 • 2 minutes to read • Contributors  

Returns a table of one or more columns.

Syntax

DAX

 Copy

```
{ <scalarExpr1>, <scalarExpr2>, ... }  
{ ( <scalarExpr1>, <scalarExpr2>, ... ), ( <scalarExpr1>, <scalarExpr2>, ... ), ... }
```

Parameters

Term	Definition
scalarExprN	Any DAX expression that returns a scalar value.

Return value

A table of one or more columns. When there is only one column, the name of the column is Value. When there are N columns where $N > 1$, the names of the columns from left to right are Value1, Value2, ..., ValueN.


```
My Measures =  
{  
    ("Qty", 1)  
    , ("Ext Price", 2)  
    , ("Profit", 3)  
}
```

Step 2

Define a measure

```
value =  
SWITCH  
(  
    FIRSTNONBLANK('My Measures'[Measure Name])  
    , "Qty", [Qty]  
    , "Ext Price", [Ext Price]  
    , "Profit", [Profit]  
    , BLANK()  
)
```

Supplier Category	Qty	Ext Price	Profit
Clothing Supplier	2,624,338	\$53,407,430.60	\$21,929,432.50
Novelty Goods Supplier	525,778	\$11,849,641.64	\$6,071,706.15
Packaging Supplier	5,696,347	\$115,763,819.96	\$51,663,616.75
Toy Supplier	104,165	\$17,022,547.25	\$6,064,425.50
Total	8,950,628	\$198,043,439.45	\$85,729,180.90

Measure Name	Clothing Supplier	Novelty Goods Supplier	Packaging Supplier	Toy Supplier	Total
Qty	2,624,338	525,778	5,696,347	104,165	8,950,628
Ext Price	53,407,431	11,849,642	115,763,820	17,022,547	198,043,439
Profit	21,929,433	6,071,706	51,663,617	6,064,426	85,729,181

```
value (Formatted) =  
SWITCH  
(  
    VALUES('My Measures'[Measure Name])  
    , "Qty", FORMAT([Qty], "#,0")  
    , "Ext Price", FORMAT([Ext Price], "Currency")  
    , "Profit", FORMAT([Profit], "Currency")  
    , BLANK()  
)
```

FileHomeInsertModelingViewHelpFormatData / DrillTable toolsMeasure tools

NameValue (Formatted)

Home tableMy Measures

\$%Text

\$ % 000 Auto

Data categoryUncategorized

New Quick measure measure

Structure

Formatting

Properties

Calculations

1 Value (Formatted) =

Clothing Supplier	2,624,338	\$53,407,430.60	\$21,929,432.50
Novelty Goods Supplier	525,778	\$11,849,641.64	\$6,071,706.15
Packaging Supplier	5,696,347	\$115,763,819.96	\$51,663,616.75
Toy Supplier	104,165	\$17,022,547.25	\$6,064,425.50
Total	8,950,628	\$198,043,439.45	\$85,729,180.90

Measure NameValue (Formatted)

Qty	8,950,628
Ext Price	\$198,043,439.45
Profit	\$85,729,180.90

Visualizations

Fields

Clothing Supplier	2,624,338	\$53,407,430.60	\$21,929,432.50
Novelty Goods Supplier	525,778	\$11,849,641.64	\$6,071,706.15
Packaging Supplier	5,696,347	\$115,763,819.96	\$51,663,616.75
Toy Supplier	104,165	\$17,022,547.25	\$6,064,425.50
Total	8,950,628	\$198,043,439.45	\$85,729,180.90

Measure Name	Value (Formatted)
Qty	8,950,628
Ext Price	\$198,043,439.45
Profit	\$85,729,180.90

Visualizations

Fields

Search

Database Name

Date

Metadata

My Measures

Measure Name

Sort Order

Value

Value (Formatted)

My Measures (Datatable)

My Measures (Row)

My Measures (Table Constructor)

Product

Sales

Server Name

Source Query - Sales

Source Query - Stock Item

Source Query - Supplier

Supplier

Address Line 1

Address Line 2

Payment Days

Postal Code

Supplier

Supplier Category

Supplier Category ID

Supplier ID

Website URL

Rows

Measure Name

Columns

Add data fields here

Values

Value (Formatted)

Drill through

Cross-report

Off

Keep all filters

On

Add drill-through fields here

Dynamic Currency Conversion



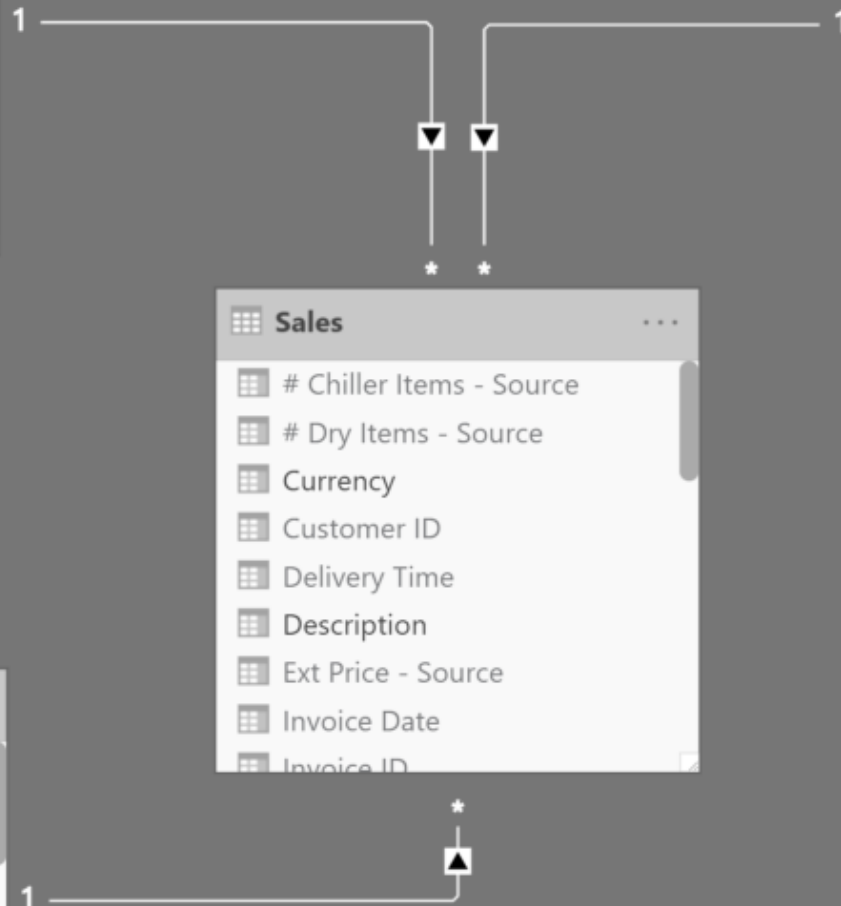
Product
Brand
Chiller Stock
Color
Color ID
Lead Time (Days)
Product Name
Quantity Per Outer
Retail Price
Size

Supplier
Address Line 1
Address Line 2
Payment Days
Postal Code
Supplier
Supplier Category
Supplier Category ID
Supplier ID
Website URL

Sales
Chiller Items - Source
Dry Items - Source
Currency
Customer ID
Delivery Time
Description
Ext Price - Source
Invoice Date
Invoice ID

Date
CrossJoin ID
Date
Date Key
Day of Month
Day of Week
Day of Week Name
Month
Month Key

Currency Exchange Rate
Date
Destination Currency
Exchange Rate
Source Currency

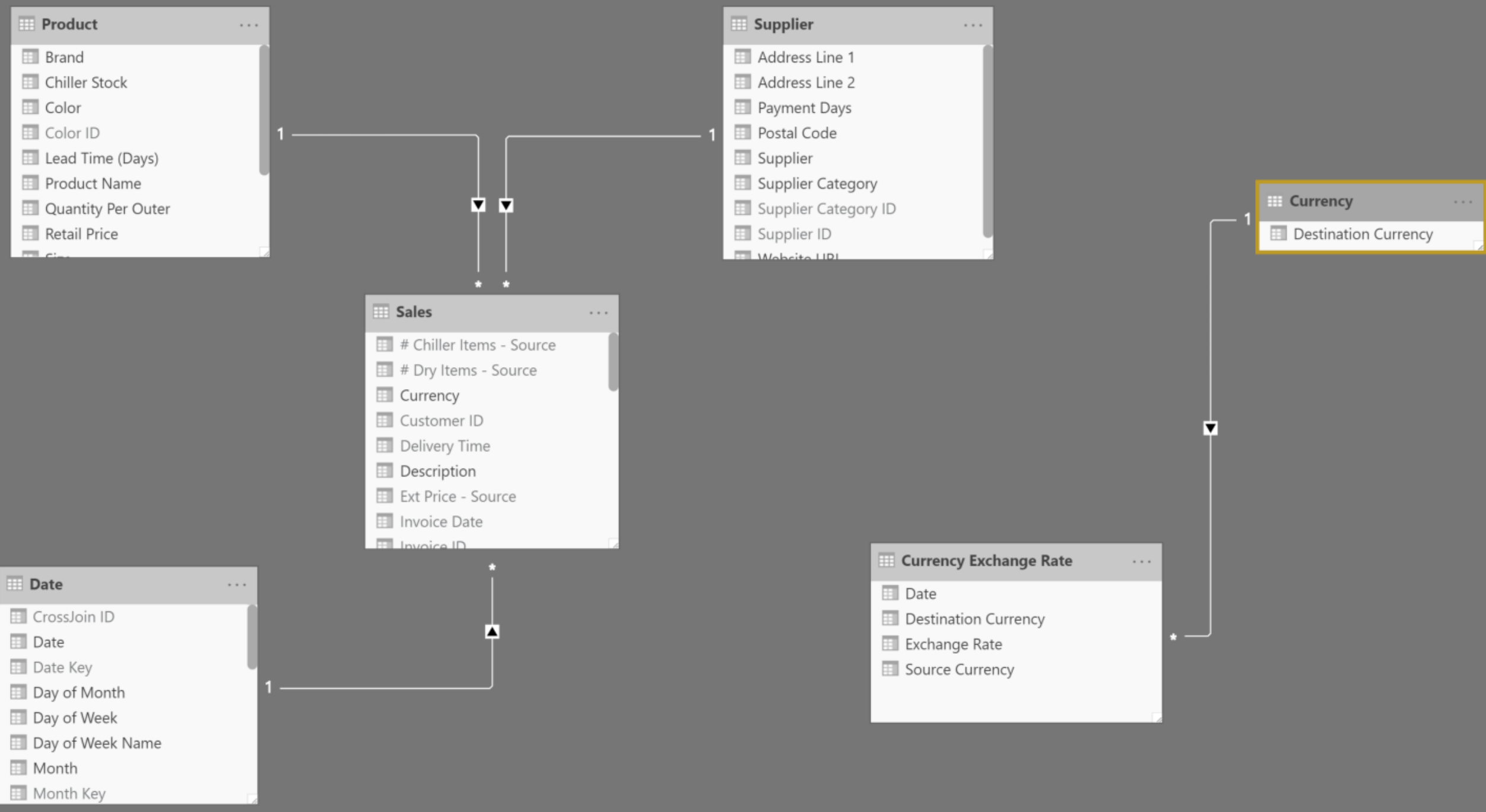



```
Ext Price (CAD) =  
SUMX  
(  
    Sales  
    Sales[Ext Price - Source] * Sales[Exchange Rate (CAD)]  
)
```

Year	Ext Price	Ext Price (CAD)	Exchange Rate (CAD)
2013	\$52,563,272.64	\$36,794,290.85	\$0.70
2014	\$57,418,916.89	\$45,935,133.51	\$0.80
2015	\$62,090,220.81	\$37,254,132.49	\$0.60
2016	\$25,971,029.11	\$23,373,926.20	\$0.90
Total	\$198,043,439.45	\$143,357,483.05	\$0.90

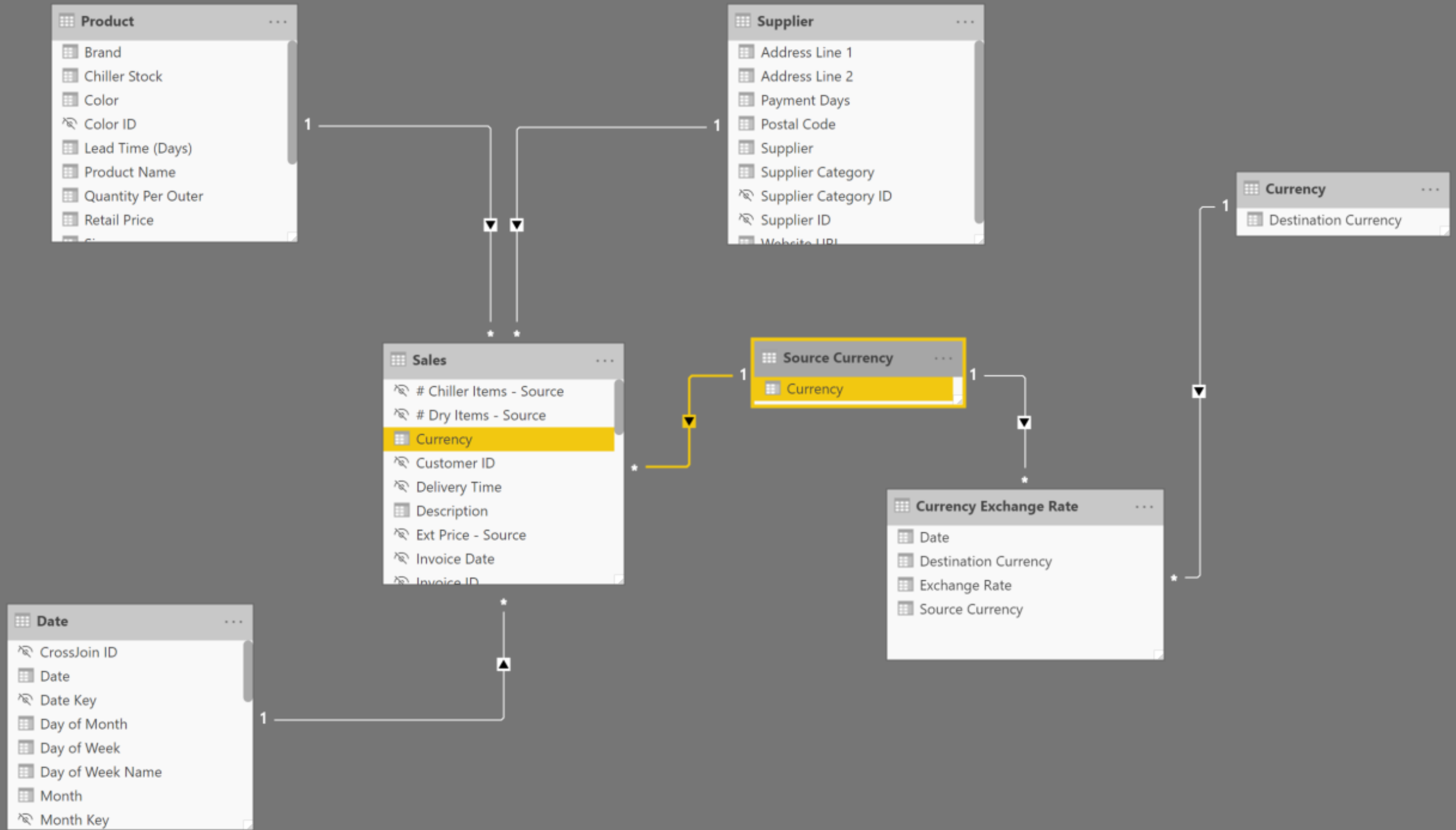
Step 1

Create a filter dimension



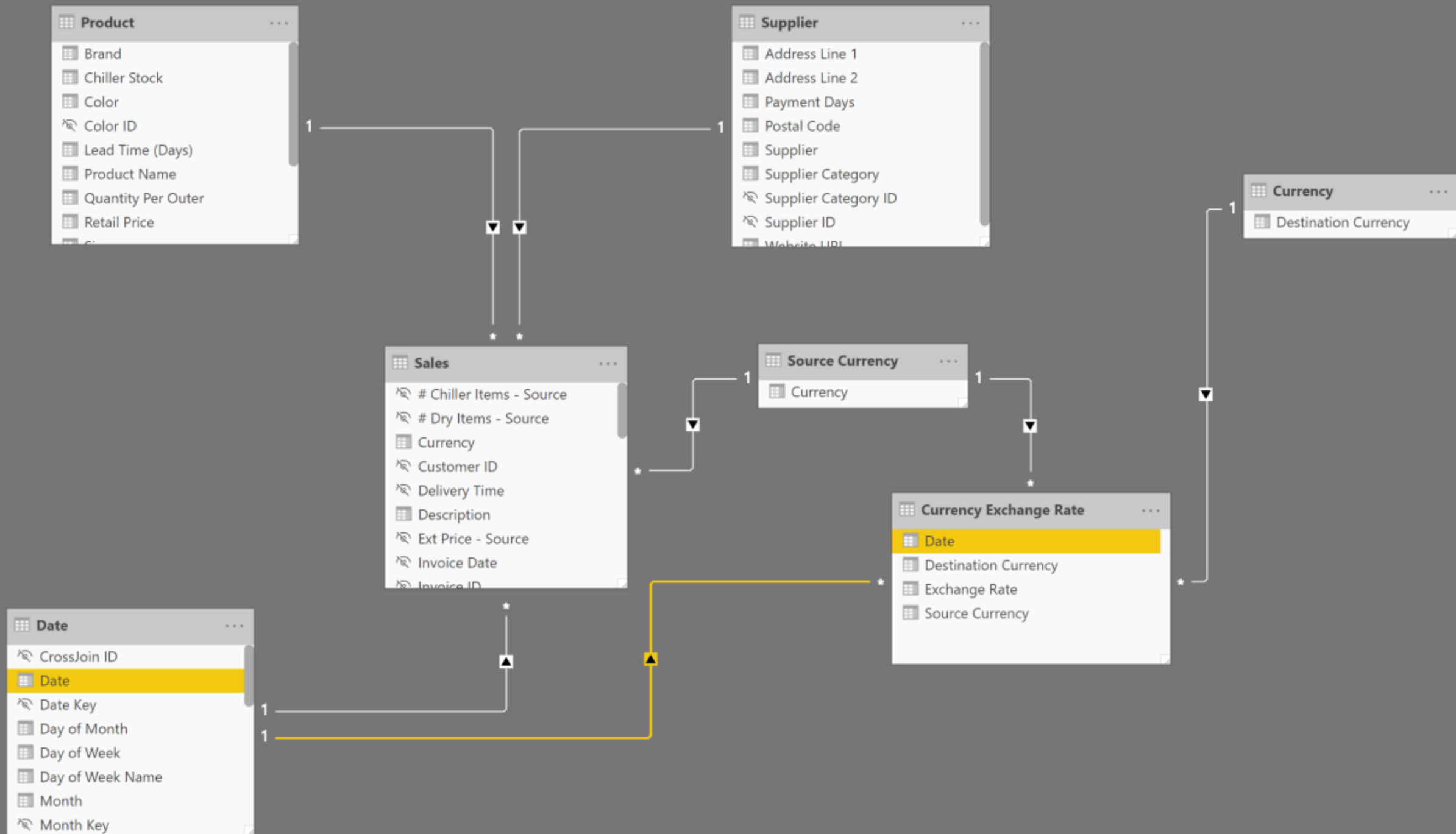
Step 2

Create a bridge table



Step 3

Add a relationship to the
Date entity



Step 4

Create the measures

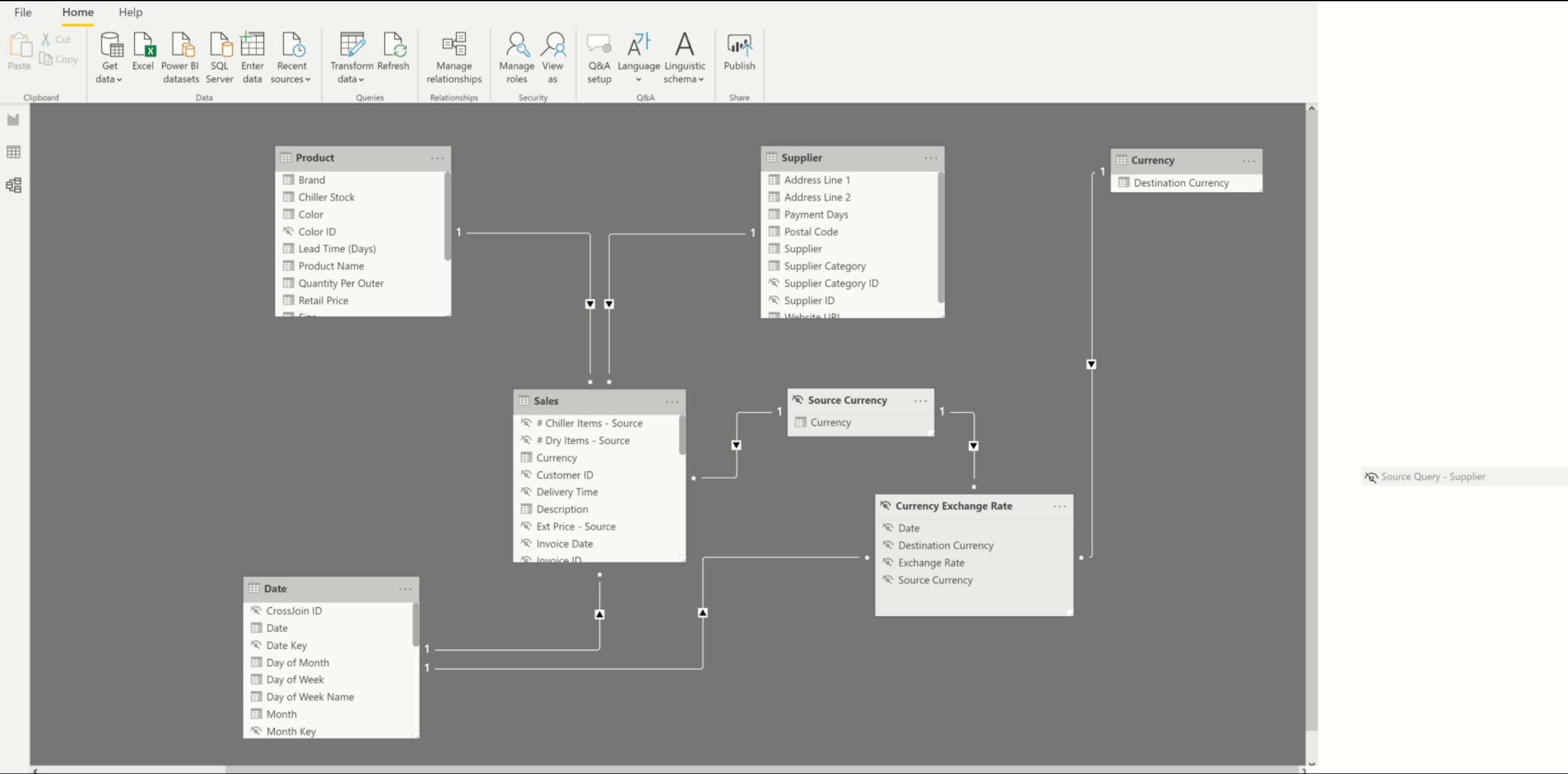
Exchange Rate = MAX('Currency Exchange Rate'[Exchange Rate])

New Ext Price =

```
IF
(
    HASONEVALUE('Currency'[Destination Currency])
    ,
    SUMX
    (
        Sales
        , [Ext Price] * [Exchange Rate]
    )
    ,
    CALCULATE
    (
        SUMX
        (
            Sales
            , [Ext Price] * [Exchange Rate]
        )
        , 'Currency'[Destination Currency] = "USD"
    )
)
```

Demo Time



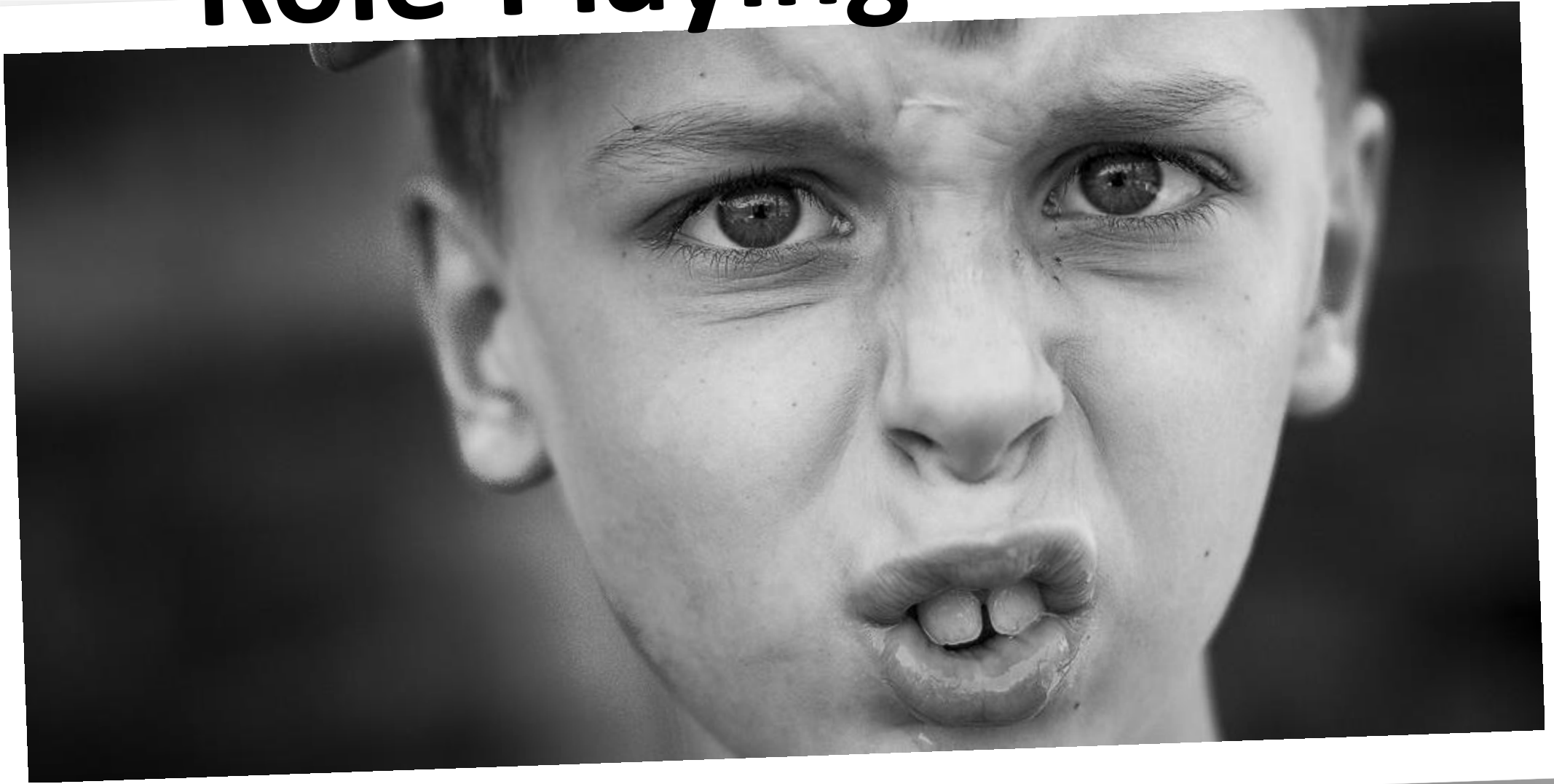


Source Query - Supplier

Role-Playing Dimensions



Role-Playing What?

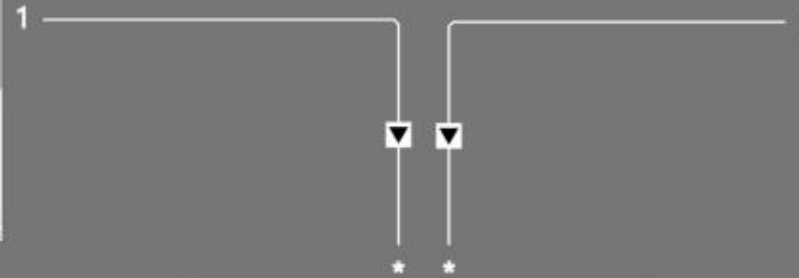


Product
Brand
Chiller Stock
Color
Color ID
Lead Time (Days)
Product Name
Quantity Per Outer
Retail Price
Size

Supplier
Address Line 1
Address Line 2
Payment Days
Postal Code
Supplier
Supplier Category
Supplier Category ID
Supplier ID
Website URL

Sales
Customer ID
Delivery Date
Description
Ext Price - Source
Invoice Date
Invoice ID
Invoice Line ID
Order ID
Profit - Source

Date
CrossJoin ID
Date
Date Key
Day of Month
Day of Week
Day of Week Name
Month
Month Key



Product
Brand
Chiller Stock
Color
Color ID
Lead Time (Days)
Product Name
Quantity Per Outer
Retail Price

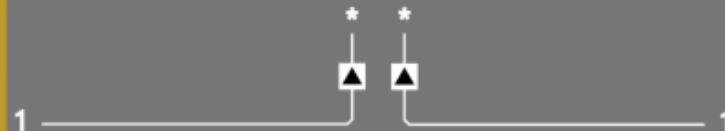
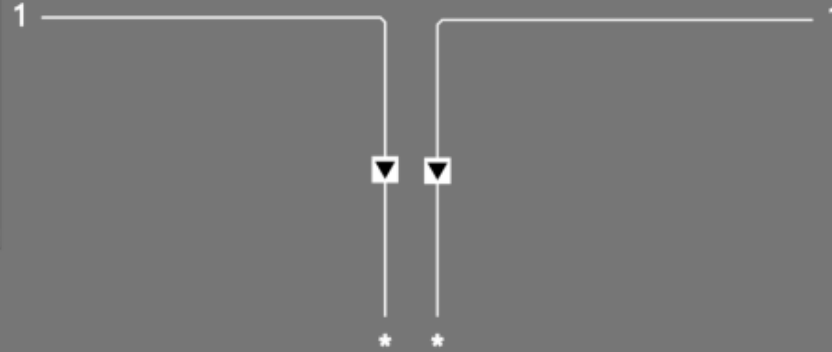
Supplier
Address Line 1
Address Line 2
Payment Days
Postal Code
Supplier
Supplier Category
Supplier Category ID
Supplier ID
Website URL

Sales
Chiller Items - Source
Dry Items - Source
Currency
Customer ID
Delivery Date
Description
Ext Price - Source
Invoice Date
Invoice ID

Date
CrossJoin ID
Date
Date Key
Day of Month
Day of Week
Day of Week Name
Month
Month Key

Delivery Date
CrossJoin ID
Date
Date Key
Day of Month
Day of Week
Day of Week Name

Invoice Date
CrossJoin ID
Date
Date Key
Day of Month
Day of Week
Day of Week Name

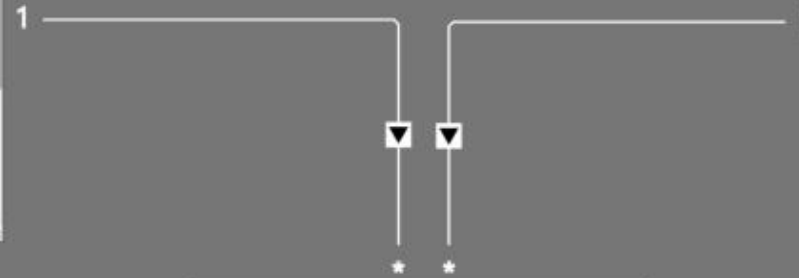


Product
Brand
Chiller Stock
Color
Color ID
Lead Time (Days)
Product Name
Quantity Per Outer
Retail Price
Size

Supplier
Address Line 1
Address Line 2
Payment Days
Postal Code
Supplier
Supplier Category
Supplier Category ID
Supplier ID
Website URL

Sales
Customer ID
Delivery Date
Description
Ext Price - Source
Invoice Date
Invoice ID
Invoice Line ID
Order ID
Profit - Source

Date
CrossJoin ID
Date
Date Key
Day of Month
Day of Week
Day of Week Name
Month
Month Key



```
Ext Price by Delivery Date =  
CALCULATE  
(  
    [Ext Price]  
    , USERRELATIONSHIP(Sales[Delivery Date], 'Date'[Date])  
)
```

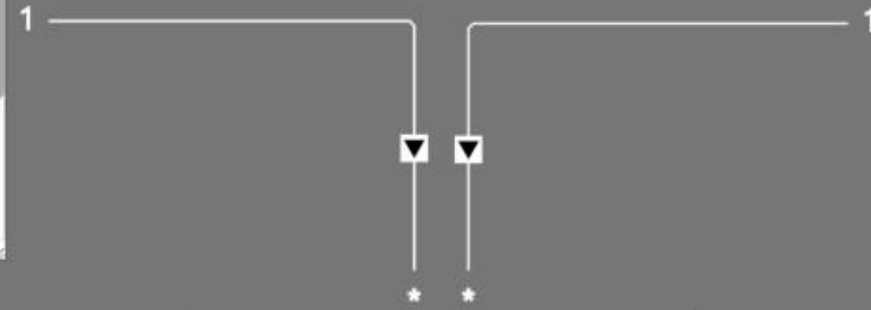
```
Ext Price by Invoice Date =  
CALCULATE  
(  
    [Ext Price]  
    , USERRELATIONSHIP(Sales[Invoice Date], 'Date'[Date])  
)
```

Step 1

Create a filter dimension

Product
Brand
Chiller Stock
Color
Color ID
Lead Time (Days)
Product Name
Quantity Per Outer
Retail Price
Size

Supplier
Address Line 1
Address Line 2
Payment Days
Postal Code
Supplier
Supplier Category
Supplier Category ID
Supplier ID
Website URL



Date Filter
Date To Filter

Sales
Chiller Items - Source
Dry Items - Source
Currency
Customer ID
Delivery Date
Description
Ext Price - Source
Invoice Date
Invoice ID

Date
CrossJoin ID
Date
Date Key
Day of Month
Day of Week
Day of Week Name
Month
Month Key



Step 2

Create a measure that's
“date aware”

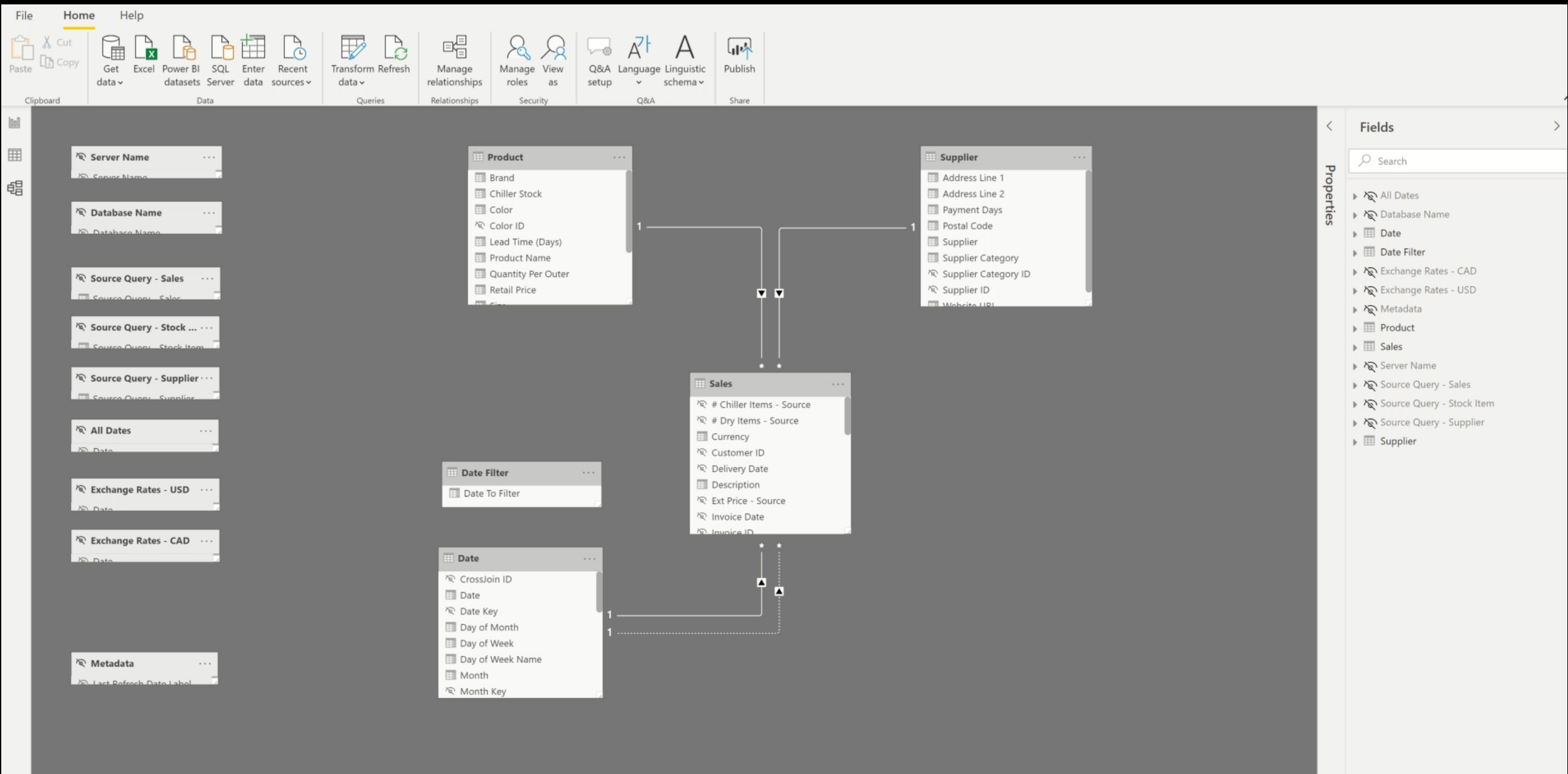
```

Ext Price - Date Aware =
VAR
    SelectedDate = SELECTEDVALUE('Date Filter'[Date To Filter], "Invoice Date")
VAR
    UseInvoiceDate = CALCULATE
        (
            [Ext Price]
        , USERRELATIONSHIP(Sales[Invoice Date], 'Date'[Date])
        )
VAR
    UseDeliveryDate = CALCULATE
        (
            [Ext Price]
        , USERRELATIONSHIP(Sales[Delivery Date], 'Date'[Date])
        )
RETURN
    SWITCH
        (
            SelectedDate
        , "Invoice Date"
        , UseInvoiceDate
        , "Delivery Date"
        , UseDeliveryDate
        , UseInvoiceDate
        )

```

Demo Time





Dynamic Month Bands



Current vs. Historic Data : Jan 2020 - Apr 2020

Partnering for Care Since

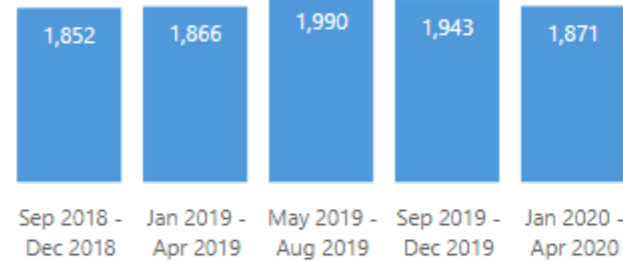
Total Activity Locations : 1,871

Work Site	1,271
Chaplain Home/Office	559
Other Location (Specify in Setting Notes)	17
Hospital	10
Funeral Home	7
Employee Home	6
Jail	1

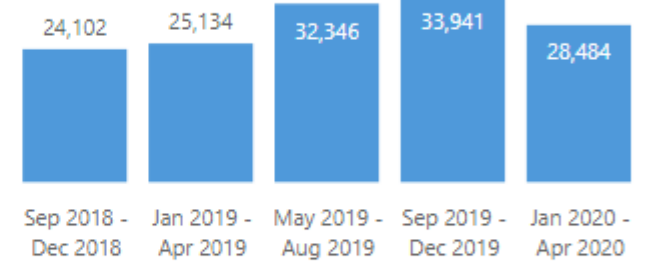
Total Care Activities : 28,484

Chaplain Engagement	10,302
Confidential Discussions	9,289
Distributed Resources	4,251
Communications	3,865
Referrals	413
Employee Care Program Orientation	292
Crisis Event	30
Job Related Issues	21
Inspirational Activity	9
Funeral	7
Special Activity	5

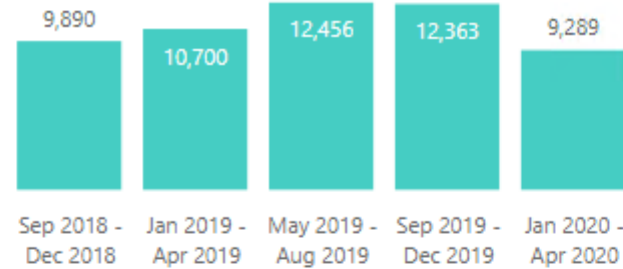
Activity Locations



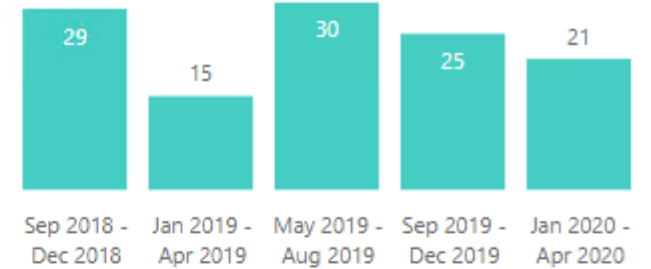
Total Care Activities



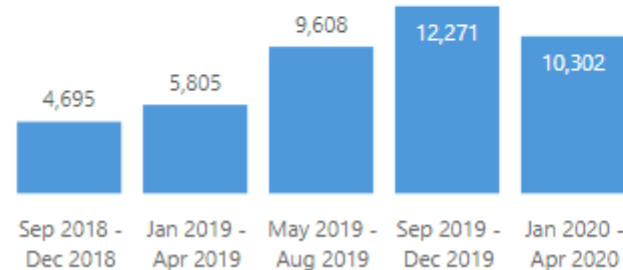
Total CDs



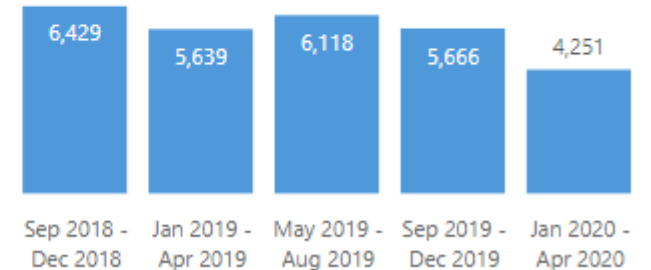
Job Related Issues



Chaplain Engagement



Distributed Resources



Reporting Month



- ☒ Apr 2020
- ☐ Mar 2020
- ☐ Feb 2020
- ☐ Jan 2020
- ☐ Dec 2019
- ☐ Nov 2019
- ☐ Oct 2019
- ☐ Sep 2019
- ☐ Aug 2019
- ☐ Jul 2019
- ☐ Jun 2019
- ☐ May 2019

Band Size (Months)

- ☐ 1
- ☐ 2
- ☐ 3
- ☒ 4
- ☐ 5
- ☐ 6
- ☐ 7
- ☐ 8
- ☐ 9
- ☐ 10
- ☐ 11
- ☐ 12


Step 1


Create a table for the band
(range) size

	Band Size (Months) 	<i>Add Column</i>
1	1	
2	2	
3	3	
4	4	
5	5	
6	6	
7	7	
8	8	
9	9	
10	10	
11	11	
12	12	

Step 2

Create a Cartesian product
table

	Reporting Month ▾	RelativeMonthOffset - Reporting Month ▾	Month  ▾	RelativeMonthOffset ▾	Band Size (Months) ▴ ▾	Month Band ▾	MonthBandKey ▾	MonthBandOffset ▾	Add Column
1	Apr 2020	-1	Dec 2019	-5	1	Dec 2019	201912201912	-5	
2	Apr 2020	-1	Jan 2020	-4	1	Jan 2020	202001202001	-4	
3	Apr 2020	-1	Feb 2020	-3	1	Feb 2020	202002202002	-3	
4	Apr 2020	-1	Mar 2020	-2	1	Mar 2020	202003202003	-2	
5	Apr 2020	-1	Apr 2020	-1	1	Apr 2020	202004202004	-1	

	Reporting Month ▾	RelativeMonthOffset - Reporting Month ▾	Month  ▾	RelativeMonthOffset ▾	Band Size (Months) ▴ ▾	Month Band ▾	MonthBandKey ▾	MonthBandOffset ▾	Add Column
1	Apr 2020	-1	Jul 2019	-10	2	Jul 2019 - Aug 2019	201907201908	-5	
2	Apr 2020	-1	Aug 2019	-9	2	Jul 2019 - Aug 2019	201907201908	-5	
3	Apr 2020	-1	Sep 2019	-8	2	Sep 2019 - Oct 2019	201909201910	-4	
4	Apr 2020	-1	Oct 2019	-7	2	Sep 2019 - Oct 2019	201909201910	-4	
5	Apr 2020	-1	Nov 2019	-6	2	Nov 2019 - Dec 2019	201911201912	-3	
6	Apr 2020	-1	Dec 2019	-5	2	Nov 2019 - Dec 2019	201911201912	-3	
7	Apr 2020	-1	Jan 2020	-4	2	Jan 2020 - Feb 2020	202001202002	-2	
8	Apr 2020	-1	Feb 2020	-3	2	Jan 2020 - Feb 2020	202001202002	-2	
9	Apr 2020	-1	Mar 2020	-2	2	Mar 2020 - Apr 2020	202003202004	-1	
10	Apr 2020	-1	Apr 2020	-1	2	Mar 2020 - Apr 2020	202003202004	-1	

```

VAR
    MaxReportingMonths = 12
VAR
    BandsToReturn = 5
VAR
    ReportingMonths =
        FILTER
        (
            ALL('Date'[Month], 'Date'[RelativeMonthOffset])
            'Date'[RelativeMonthOffset] < 0 && 'Date'[RelativeMonthOffset] >= (MaxReportingMonths * -1)
        )

RETURN
    ReportingMonths

```

Month	RelativeMonthOffset
Apr 2020	-1
Mar 2020	-2
Feb 2020	-3
Jan 2020	-4
Dec 2019	-5
Nov 2019	-6
Oct 2019	-7
Sep 2019	-8
Aug 2019	-9
Jul 2019	-10
Jun 2019	-11
May 2019	-12


```

VAR
    ReportingBandSize =
        ALL('Reporting Band Selection')
VAR
    ReportingMonthAndBandSize =
        GENERATE
        (
            ReportingMonths
            ReportingBandSize
        )
RETURN
    _ReportingMonthAndBandSize

```

Month	RelativeMonthOffset	Band Size (Months)	
May 2019	-12	1	
Jun 2019	-11	1	
Jul 2019	-10	1	
Aug 2019	-9	1	
Sep 2019	-8	1	
Oct 2019	-7	1	
Nov 2019	-6	1	
Dec 2019	-5	1	
Jan 2020	-4	1	
Feb 2020	-3	1	
Mar 2020	-2	1	
Apr 2020	-1	1	
May 2019	-12	2	
Jun 2019	-11	2	
Jul 2019	-10	2	
Aug 2019	-9	2	
Sep 2019	-8	2	

VAR

CalculatedReportingMonths =

GENERATE

(

SELECTCOLUMNS

(

ReportingMonthAndBandSize

, "Reporting Month"

, [Month]

, "Band Size (Months)"

, [Band Size (Months)]

, "RelativeMonthOffset - Reporting Month"

, [RelativeMonthOffset]

)

, FILTER

(

ALL('Date'[Month], 'Date'[RelativeMonthOffset])

, 'Date'[RelativeMonthOffset] <= ([RelativeMonthOffset - Reporting Month])

&& 'Date'[RelativeMonthOffset] > (([Band Size (Months)] * BandsToReturn * -1) + [RelativeMonthOffset - Reporting Month])

)

)

RETURN

CalculatedReportingMonths

Reporting Month	Band Size (Months)	RelativeMonthOffset - Reporting Month	Month	RelativeMonthOffset
Apr 2020	1	-1	Apr 2020	-1
Apr 2020	1	-1	Mar 2020	-2
Apr 2020	1	-1	Feb 2020	-3
Apr 2020	1	-1	Jan 2020	-4
Apr 2020	1	-1	Dec 2019	-5
Apr 2020	2	-1	Apr 2020	-1
Apr 2020	2	-1	Mar 2020	-2
Apr 2020	2	-1	Feb 2020	-3
Apr 2020	2	-1	Jan 2020	-4
Apr 2020	2	-1	Dec 2019	-5
Apr 2020	2	-1	Nov 2019	-6
Apr 2020	2	-1	Oct 2019	-7
Apr 2020	2	-1	Sep 2019	-8
Apr 2020	2	-1	Aug 2019	-9
Apr 2020	2	-1	Jul 2019	-10

Demo Time



Reporting Month Bands

-  Month
-  Reporting Month
-  Band Size (Months)
-  RelativeMonthOffset - Reporting Month
-  RelativeMonthOffset
-  Month Band
-  MonthBandKey
-  MonthBandOffset
-  Selected Reporting Month

Reporting Month Bands Bridge

 Month



1



*

 Date

About Me

“I help people make sense
of their data”

My blog: martinschoombee.com

My company: 28twelve.consulting

Tweet me: [@MartinSchoombee](https://twitter.com/MartinSchoombee)

GitHub Repo:

<https://github.com/mschoombee/Presentations/wiki/Presentations>

