**SUN BURST DIAGRAM**

Sunburst chart is typically used to highlight hierarchy in the data, via the concentric circle patterns. The granularity of the data increase as you move outwards, with the outermost

Circle indicating the highest level of granularity. This type of chart is used to compare relative sizes between various dimensions.

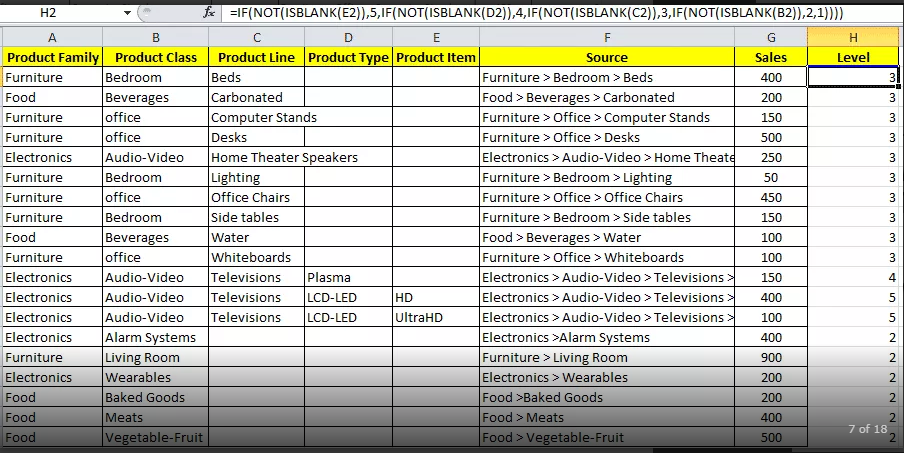
For example: using the global superstore data, we can conclude that southern US have the highest sales, with Office supplies serving as the Primary contributor.

Sun burst charts are a complex chart type using several advanced techniques like data densification, nested table calculations combined with math concepts like quadratic equations.

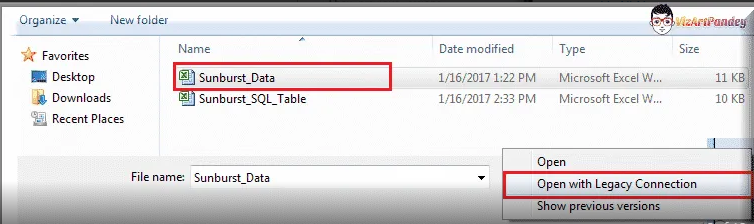
A Sunburst Chart requires more Data structuring prior to designing the viz in tableau.

A sunburst chart is really just a treemap which uses a radial layout (thus the alternative name, “Radial Treemap”). Sunbursts are a series of rings, which represent the different hierarchical levels. The innermost ring is the first level, followed by the second level which shows a breakdown of the components of the first, and so on. Like the more common rectangular treemap, the size of each section represents the magnitude of some measure.

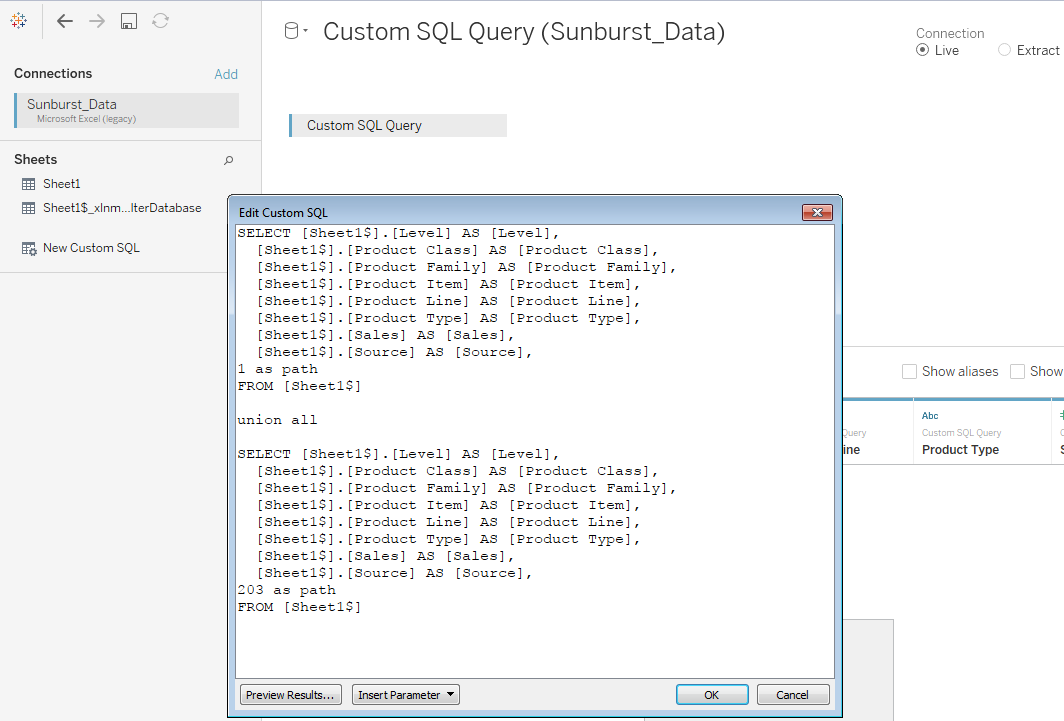
Step 1: Prepare Excel Data with new Column as Level.



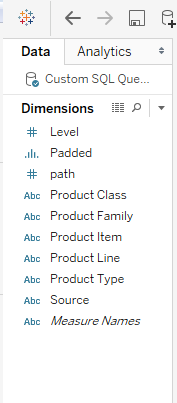
Step 2: Upload our Data and create a Path by using Legacy Connection.



Here you can either use SQL Query or Excel formula for generating the level. I have already generated the level inside my excel file, so I will use the Union All the Path=1 with the Second Query having Path=203.



Add a new sheet, and then convert **Level** and **Path** to Dimensions, if they are not already Dimensions.



Here I am converting the path and level as dimensions.

Calculations Used for Sun Burst Chart:



Edges: IF [Max-Level] > LOOKUP([Max-Level],-1) THEN PREVIOUS\_VALUE(0)

ELSEIF [Max-Level] <= LOOKUP([Max-Level],-1) THEN PREVIOUS\_VALUE(0) + LOOKUP([Slice-Size],-1)

ELSE PREVIOUS\_VALUE(0) END

Index: INDEX()

Max-Level: WINDOW\_MAX(MAX([Level]))

Max-Sales: WINDOW\_MAX(MAX([Sales]))

Slice-Level: WINDOW\_MAX(MAX([Level]))

Slice-Size: [Max-Sales]/WINDOW\_SUM(IIF([Max-Level]=1,[Max-Sales]/203,0))

X: IF([Index]<>WINDOW\_MAX([Index]) AND [Index]>=(WINDOW\_MAX([Index])+1)/2)

THEN

([Slice-level]+5.8)

\* COS(WINDOW\_MAX(2\*PI())\*[Edges]+

(WINDOW\_MAX([Index])-([Index]+1))\*WINDOW\_MAX(2\*PI())\*[Slice-Size]/(((WINDOW\_MAX([Index])-1)/2)-1))

ELSEIF([Index]=WINDOW\_MAX([Index]) OR [Index]<(WINDOW\_MAX([Index])+1)/2)

THEN

([Slice-level] + 5)

\* COS(WINDOW\_MAX(2\*PI())\*[Edges]+

(((IIF([Index]=WINDOW\_MAX([Index]), 1,[Index])-1)\*WINDOW\_MAX(2\*PI())\*[Slice-Size]/((((WINDOW\_MAX([Index])-1)/2)-1)))))

END

Y: IF([Index]<>WINDOW\_MAX([Index]) AND [Index]>=(WINDOW\_MAX([Index])+1)/2)

THEN

([Slice-level]+5.8)

\* SIN(WINDOW\_MAX(2\*PI())\*[Edges]+

(WINDOW\_MAX([Index])-([Index]+1))\*WINDOW\_MAX(2\*PI())\*[Slice-Size]/((((WINDOW\_MAX([Index])-1)/2)-1)))

ELSEIF([Index]=WINDOW\_MAX([Index]) OR [Index]<(WINDOW\_MAX([Index])+1)/2)

THEN

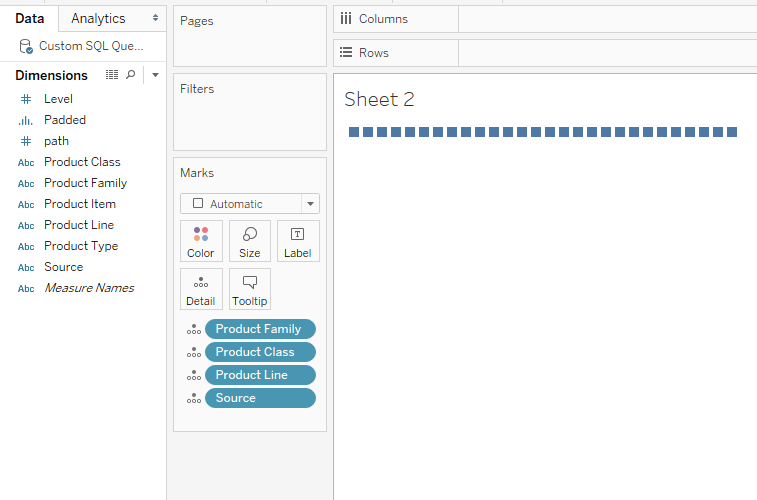
([Slice-level] + 5)

\* SIN(WINDOW\_MAX(2\*PI())\*[Edges]+

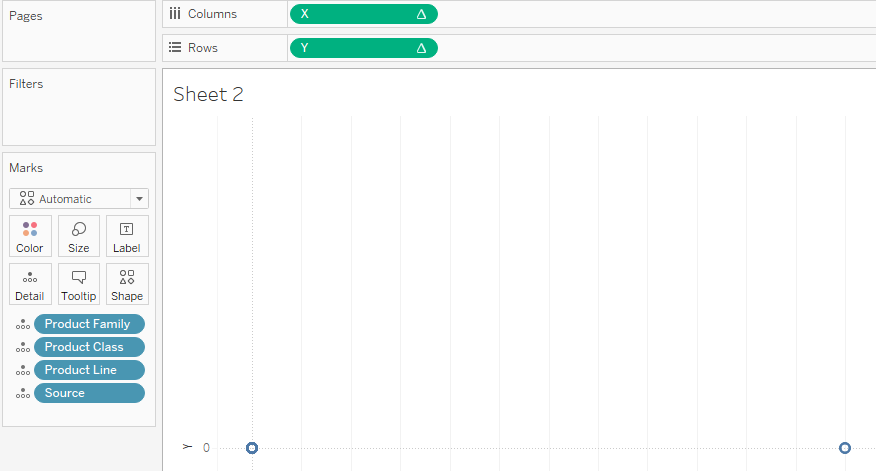
(((IIF([Index]=WINDOW\_MAX([Index]), 1,[Index])-1)\*WINDOW\_MAX(2\*PI())\*[Slice-Size]/(((WINDOW\_MAX([Index])-1)/2)-1))))

END

Step 3: Drag the fields in the hierarchy to detail shelf on the marks card. Start with the highest level to the lowest level( Product family is the highest one, source is the lowest one)

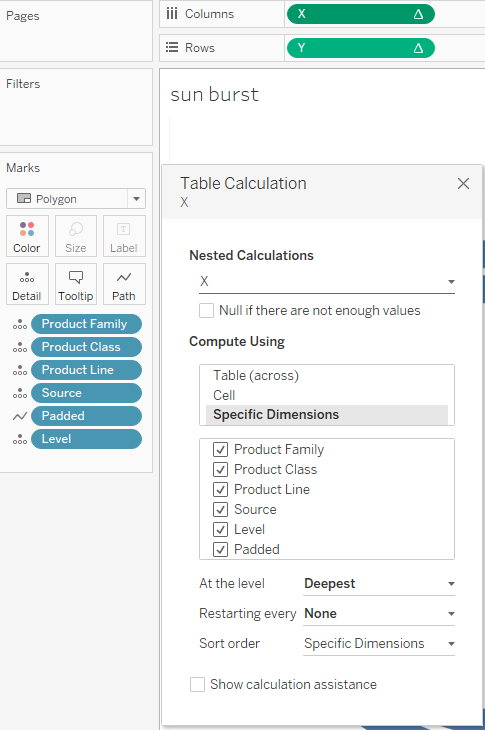


Drag X and Y coordinate to the columns and row shelves respectively. Also Add padding to the path shelf on the marks card. Note To Change the mark type to Polygon.



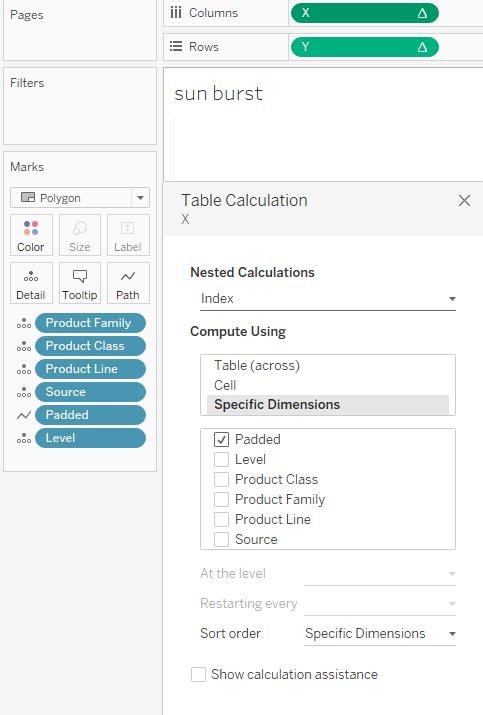
Right click on ‘X’ and select edit table calculations, here we are creating Nested table Calculations. Select all dimensions keeping the same order on the marks shelf also.

**X: X**

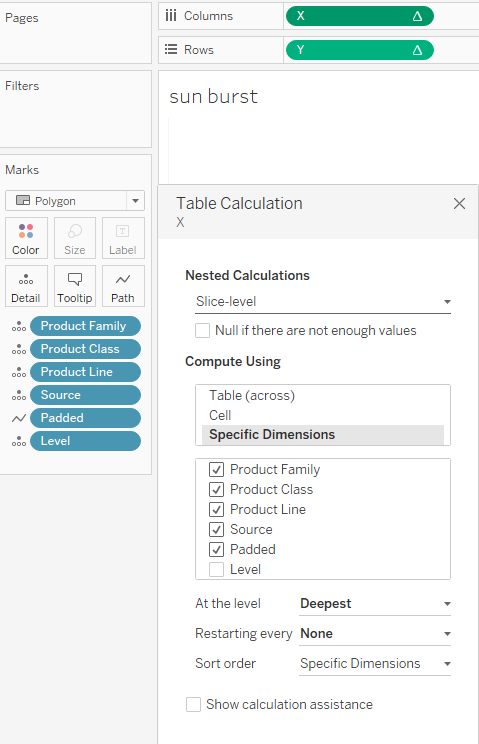


**X: Index**

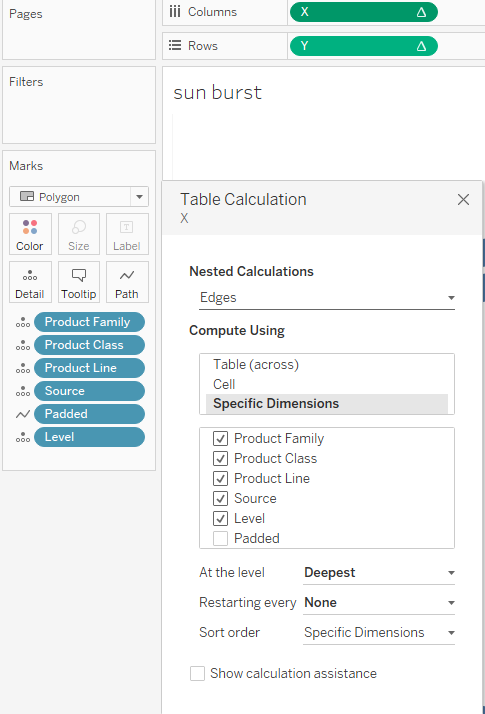
Repeat the same steps for other nested calculations.



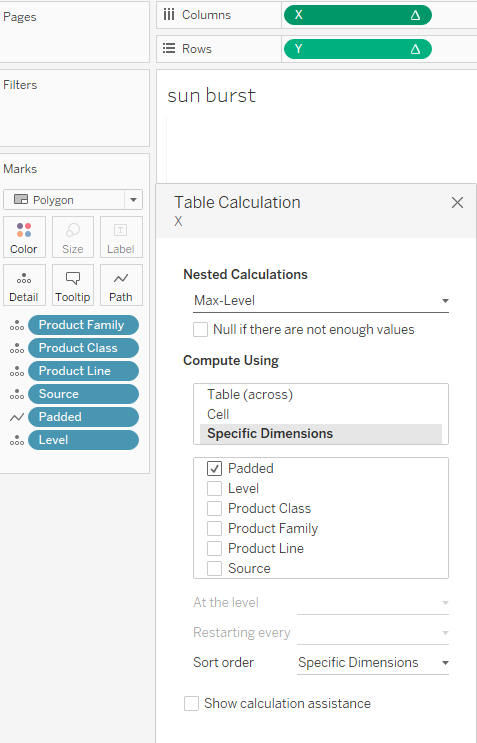
**X : Slice-Level**



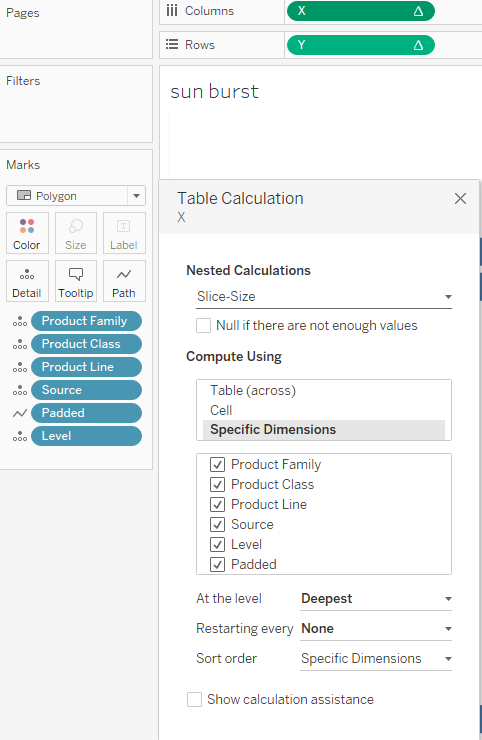
**X : Edges**



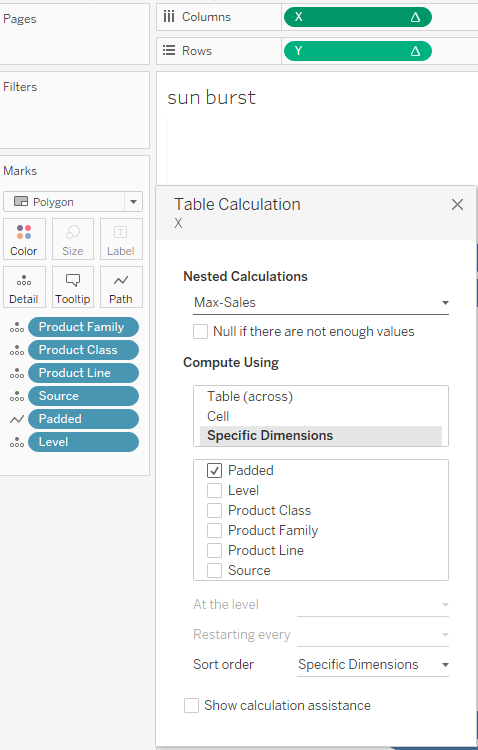
**X: Max-Level**



**X: Slice-Size**



**X: Max-Sales**



Follow the same steps above for the Y- Coordinate also editing same nested calculations on same of x-coordinate..

