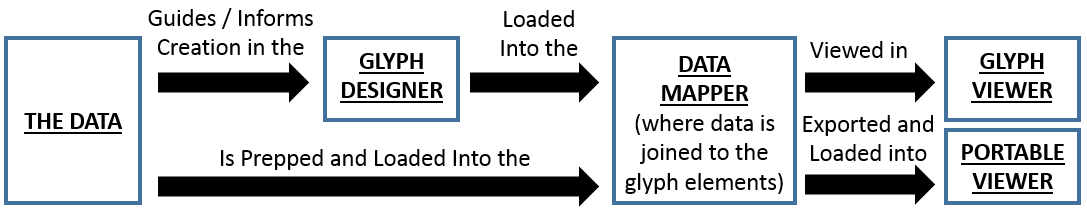
**The SynGlyphX Quick Start Guide (For V.12)**

**Part i: Overview**

This guide will establish a baseline competency to get from raw data to an accurate 3D visualization. Each section (The Data, Glyph Designer, Data Mapper, Glyph Visualizer, and Portable Viewer (in development) corresponds to a specific piece of the SynGlyphX package and the image below shows how each piece interacts:



As you navigate through each section, there will be several tips and tricks to get the most out of the V.12 toolset as well as common pitfalls and solutions.

**Part ii: License Installation**

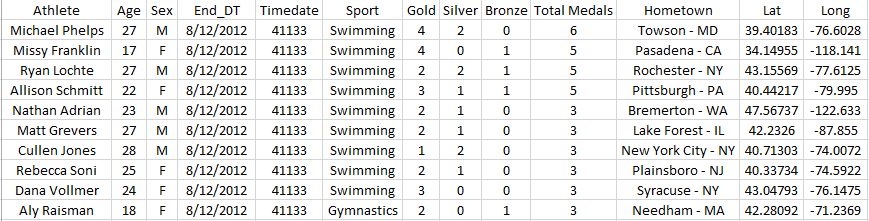
Open either Glyph Designer, Data Mapper, or Glyph Viewer and you will be prompted to view the SynGlyphX EULA and install a license key. After viewing and accepting the EULA, you will need to point to the \*.lic file given to you (for example sgx\_trial.lic). Once that information is mapped, you will be free to open any of the SynGlyphX products for the duration of your license period.

**Part 1: The Data**

As of V.12, the following relational databases can be properly loaded into Data Mapper:

* CSV files
* SQLite files

While it is perfectly fine to begin playing with the Glyph Designer, once you decide to create a project from start to finish we highly suggest that you first figure out what your data source will be in order to minimize rework. The data below will be used in this initial walkthrough and provided in the drop box location with this guide.

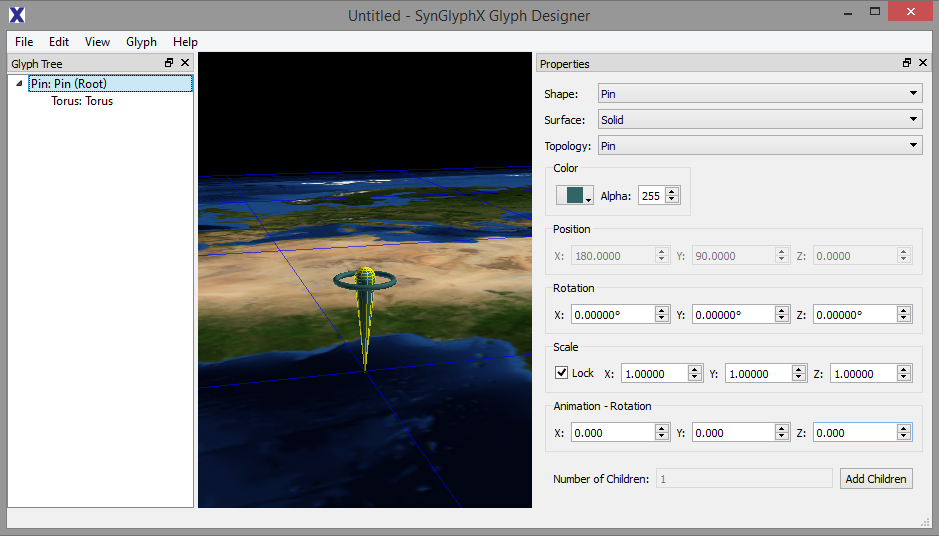


Some things to note in pre-prep:

1. The End\_DT field has been converted to an integer. This will allow you to utilize the Z-axis for time in Data Mapper
2. There are lat and long values. Every glyph must be positioned somewhere on the 3D field of view. If there are no position coordinates, the system will assign everything to a default and all glyphs will end up stacked on top of each other.

**Part 2: Glyph Designer**

1. Open up Glyph Designer and you should see the following below:



We now have a canvas to plan and map the primary fields of interest which are:

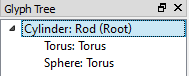
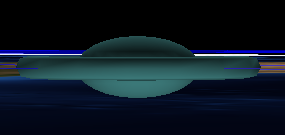
* Position (long / lat / time) <=> (X-axis / Y-axis / Z-axis)
* Age
* Sex (M/F)
* Medal Count (Gold/Silver/Bronze/Total)

And secondary fields of interest which are:

* Name
* Location

**NOTE**: If you look at the primary fields you will notice that they are mostly **numeric**. The general guiding principle will be to create a glyph design around these numerical properties and then as you get more comfortable, you can begin to overlay / map text to value.

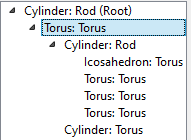
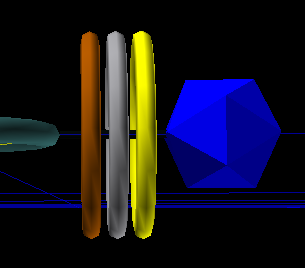
1. It is now time to begin building the base glyph. The following steps will be used to set up the following properties: **Name**, **Location**, and **Sex**
   1. Go to the “Glyph Tree” bar on the left and you will see all of the elements. We are going to create a base that looks something like this:

* 1. The shape will be formed using these steps
     1. Click on the (Root) of the glyph tree
     2. On the “Properties” bar change “Shape: Cylinder”, “Topology: Rod”, “Scale: 0/0/0” 🡸 Because the default is “Lock” you can just change one of the X/Y/Z values and it will make everything zero. You may also change the transparency to 0 from 255 but in this example it makes no difference.
     3. Click “Add Children” below and set the number to 1 and click OK
     4. Click on the first “Torus:Torus” and change its Scale to 0.6
     5. Click on the second “Torus:Torus” and change its Shape to “Sphere”. Then go to Scale, uncheck “Lock”, and make X=0.5, Y=0.5, Z=0.25

**NOTE**: Changing the X/Y/Z will carry over to all Children of the glyph element. We are using a pill shape here because it’s easier to stack glyphs over time with a narrower shape profile and nothing else will be mapped to the sphere in our example.

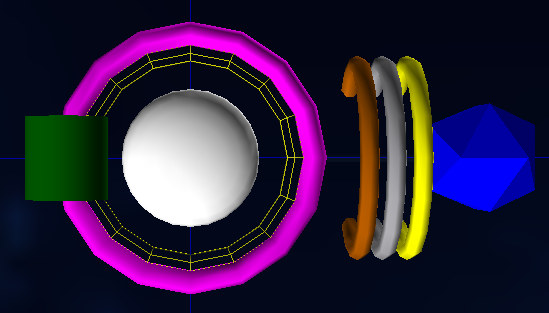
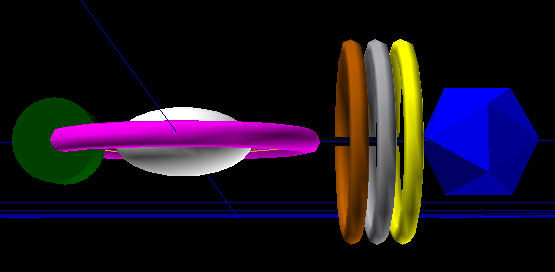
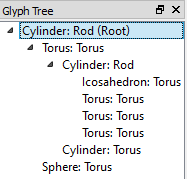
1. Create a branch / child for the medal counts.
   1. You are free to creatively decide how to display multiple elements. In this example we will be using an arm + elements + summary end to create a glyph that looks like this:

* 1. As you can see, there is a base arm created with a Cylinder using Rod topology. We have exaggerated several of the features using the following steps:
     1. Torus: Torus 🡺 Add Children: 1
     2. Shape: Cylinder, Topology: Rod, Color:Alpha:25 (opacity), Scale:0.75
     3. Cylinder: Rod 🡺 Add Children: 4
     4. Shape: Icosahedron, Topology: Torus, Color: Blue, Scale: 3.0
     5. Shape/Top: Torus, Color: Bronze, Position: X= -150°, Scale:3.0
     6. Shape/Top: Torus, Color: Silver, Position: X= -120°, Scale:3.0
     7. Shape/Top: Torus, Color: Gold, Position: X= -90°, Scale:3.0

**NOTE**: The X position in Torus topology will let you move glyph elements up and down on the -180 ⬄ 180 degree range.

1. Create an element for Age
   1. Torus: Torus 🡺 Add Children:1
   2. Shape: Cylinder, Topology: Torus, Scale: 1.0, Position: X=180°
2. Add in a few more default colors and you should see this design and Glyph Tree

1. Save the glyph under any name. In this example we use “Sample glyph.sgt”

At this point in time we are ready to move to the Data Mapper stage

**Part 3: Data Mapper**

This is where the fun begins and you can start to assign data and character fields to specific glyph elements.

**Part 3a: MapQuest Key**

Before we begin though, you will need to download a MapQuest key so the lat/long coordinates from earlier can be properly placed on an underlying image.

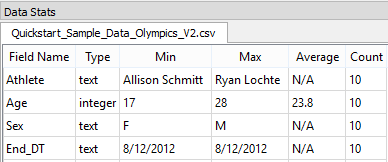
1. Navigate to the following site:

<http://developer.mapquest.com/web/products/quick_start>

1. Click on the link that says "Sign up for an AppKey"
2. Follow the directions that it gives you and MapQuest will email the key (which can take up to an hour but is typically sent quickly)
3. Once you have received the key, open **Data Mapper**
   1. Navigate to Tools 🡺 Map Download Settings
   2. Paste your MapQuest Open Key into the field and click “OK”
4. While you have the key on hand, open **Glyph Viewer**
   1. Navigate to Tools 🡺 Map Download Settings
   2. Paste your MapQuest Open Key into the field and click “OK”

**Part 3b: Mapping the Data**

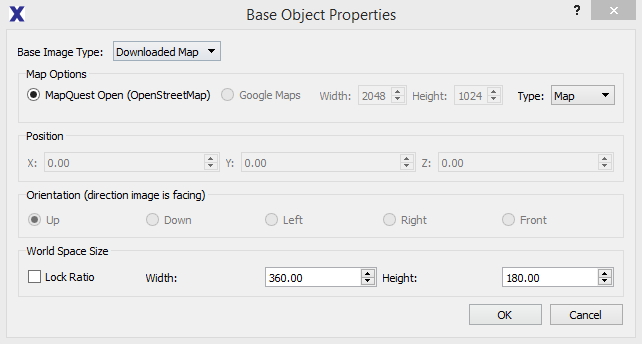
1. Open Data Mapper 🡺 Data Source 🡺 Add Data Sources 🡺 “Sample data.csv”
2. Select the proper elements for each field (you can have the base file open simultaneously if you are unsure about each column). Note that you cannot assign a Field Name to “text” and then later drag that element to a Base Properties Input unless you are using the Text Field to Value function.
3. If all goes well, you should have a “Data Stats” window appear on the right with some summary information such as Type, Min, Max, Average, and Count. It should look something like the sample below:



* 1. **NOTE**: Trouble Shooting – if there are issues loading in the data, the following checks can be done.
     1. Make sure there are no random populated fields in your data source
     2. Remove any commas from the fields to prevent row mismatching
     3. Make sure each column name is unique
     4. Check that each type field is uniform (i.e. no real values in the integer col)
     5. Check that numeric fields are not assigned as free text in your data source

1. Open Glyph 🡺 Add Glyph Templates 🡺 “Sample glyph.sgt”. This will populate the central region with your previously created glyph.
2. Click on all of the arrows on the Glyph Tree to open all shapes. This will be your working template as you map each field. You can also click directly on the glyph elements themselves (but as you graduate to much more complex visualizations it is easier to mistakenly select things).
3. Right click on “Default” under the “Grids/Base Images” window to set up a base image
   1. Right Click 🡺 Properties 🡺 Base Image Type: Downloaded Map 🡺 Type: Map

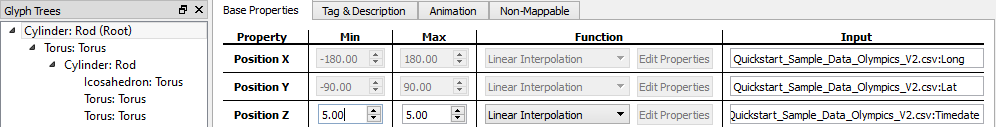
**NOTE**: You can load all sorts of interesting images and legends in this section. Once you have gained more familiarity with this field you should try and experiment with new maps and grids.



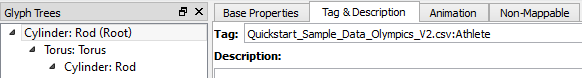
It is now time to drag, drop, and modify the Base Properties and Tag/Description fields.

1. Cylinder: Rod (Root), **Lat** 🡺 Position Y, **Long** 🡺 Position X, **Timedate** 🡺 Position Z

**NOTE**: Currently this sample data is from one period of time, but if you add more information to the data set later on you will not have to go back and remap this value. Also notice that the Min/Max values. For now if you change Max=5 or some other value it will float the glyphs above the map layer.

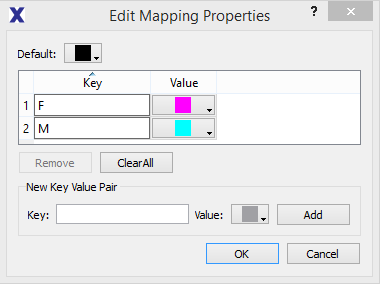


1. Tag&Description 🡺 **Athlete**



1. Sphere: Torus, Tag&Description 🡺 **Hometown**
2. Torus:Torus, Function: Color: Text Field to Value, Color: **Sex**
   1. Edit Properties, Key: M – Value: Blue, Key: F – Value: Pink
   2. This will map the females to pink and males to blue





**NOTE**: If you have a typo or forget to map all of your unique text values in the text to value mapping it will default to your Default color (in this example it is Black) which can be changed to user specification.

1. Cylinder:Torus, ScaleX/ScaleY/ScaleZ: **Age** (this requires three drags), Min X/Y/Z = .61
   1. **SIZING**: At the end of this guide there is a special section on sizing, but currently we will assume that the Max values established from the initial setup are correct. Data Mapper defaults to Min (in data) = Min (size) which is typically not what you want. In this case, relative Min = 17/28 = .61.
2. Cylinder:Torus, Tag&Description 🡺 **Age**

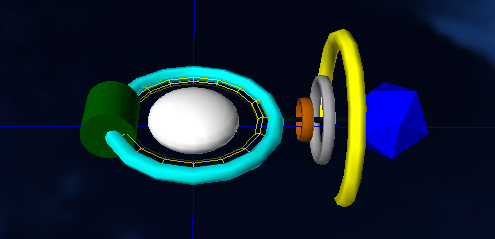
And finally we will map the medal counts and totals

1. Icosahedron: Torus, ScaleX/Y/Z: **Total Medals**, Scale Min X/Y/Z = 3/6 \*(3) = 1.5, Max = 3.0, Tag
2. Torus: Torus (1), ScaleX/Y/Z: **Bronze**, Scale Min X/Y = 0, Scale Max X/Y = (1/4)\*(4) = 1.0.

**NOTE**: Since we are comparing multiple columns, the min and max baselines are for the entire cluster rather than just the individual field (i.e. why 4 not 1). We have also chosen to not modify the Z scale here to make it easier to view small values later on.

1. Torus: Torus (2), ScaleX/Y: **Silver**, Scale Min X/Y= 0, Scale Max X/Y = (2/4)\*(4) = 2.0
2. Torus: Torus (3), ScaleX/Y: **Gold**, Scale Min X/Y= .75, Scale Max X/Y = (4/4)\*(4) = 4.0

**NOTE**: You will notice that the glyph elements will grow or shrink as you modify the Min/Max values. Data Mapper is set up to show how large the Max will be for each element. If you are unsure whether a Min is too small, you can play around with that value in the Max column to get quick visual feedback before finalizing everything.



After that last click you should be finished! Save the file.   
We have used “Sample datamap.sdt.”

**Part 3c: Exporting to Portable Viewer**

While the Glyph Viewer can natively read in \*.sdt files, Portable Viewer needs you to create a new folder and export from Data Mapper. Once you have an empty folder created on your desktop, go to File 🡺 Create Portable Visualization 🡺 select the desired folder. After that, you can go to glyphviewerportable.exe, press “1”, and the data will load.

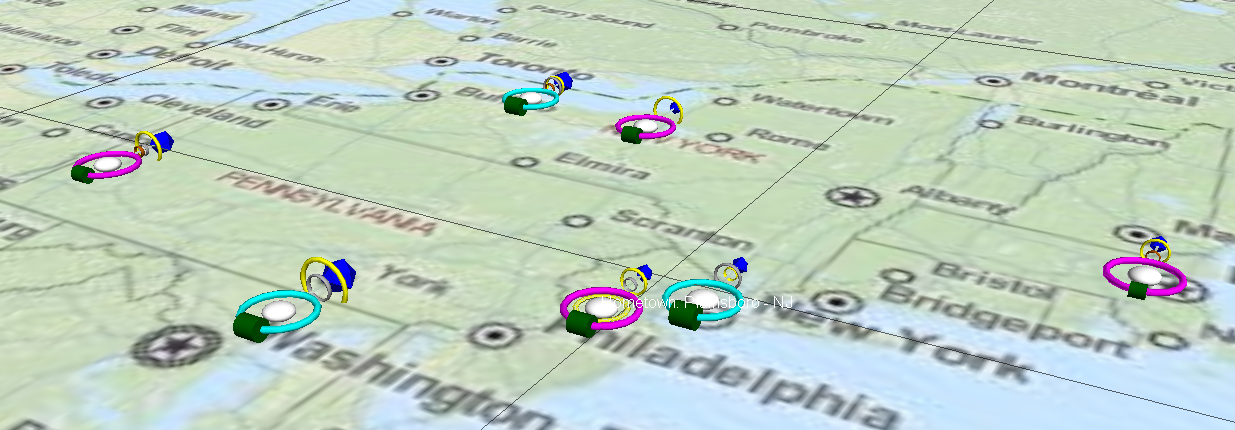
**Part 4: Glyph Viewer**

This section is fairly short.

Simply open Glyph Viewer 🡺 File 🡺 Open Visualization 🡺 “Sample datamap.sdt.”

Depending on the file size it may take a bit of time to view, and oftentimes this may also zoom into a single element once loaded.

The nice thing about Glyph Viewer is that all data will be shown in the Glyph List. Clicking on each element will snap to location, and that make navigating very easy. A sample screenshot of the New England region is shown below



For now the main navigation commands are:

* Drag Left/Right = Rotate view
* L+R Mouse + Forward = Zoom in
* L+R Mouse + Backward = Zoom out

**Part 5: Portable Visualizer**

As mentioned in Part 3c, your data can be viewed in Portable Visualizer by simply opening up the \*.exe file and pressing “1”. If you are using the sample data you will see something like this:



**Part 5a: General Navigation**

Notice that you are in “mode:Pin”. This default mode will let you select elements and pull their information as well as navigate the viewing area. The navigation commands are:

* Mouse Drag Left/Right = Rotate view
* L+R Mouse + Forward = Zoom in
* L+R Mouse + Backward = Zoom out

And the selection / exploration commands are:

* Left Click (on element) = select single
* Right Click (on elements) = select multiple
* “I” (after selecting) = display information
* Toggle “I” (after displaying information) = change color / hide information

**Part 5b: Changing the Background Image**

If the user would like to have the glyphs “pop” a bit more, a quick way to do so is to change the background image to greyscale. This can be done by going to the Portable Visualizer folder and navigating to usr 🡺 images. The main background is named “map00002.jpg”, and opening that image in a program such as paint.net, changing to black/white, and resaving will make the visualization have a greyscale base image during subsequent times opened.

**Part 6: Modifying Glyphs and Adding new information**

Once you have your Data Set, Glyph Design, and Data Map elements in place, updating and changing information is extremely fast.

***Example 1: I want to modify the Glyph Design and move elements from one parent location***

Version 11 allows the user to natively create and modify the loaded glyph template in Data Mapper. Once that program is loaded, simply right clicking on a parent glyph element will open the “Add Children” option. This will bring up the standard GUI used in Glyph Designer.

If you accidentally create a glyph element in the wrong location, you can drag and drop it in its proper place. This action can only be done in the “Glyph Tree” location, not in the sample view.

***Example 2: I want to modify the Glyph Sizes and heights***

Let’s say the output looks a bit too close to the map (base image), and the glyphs themselves are too small.

Open Data Mapper 🡺 Load the Sample 🡺 Cylinder:Rod(Root), Change Max Position Z: 10 🡺 Torus: Torus, Scale X/Y/Z: 1.00 🡺 Sphere:Torus, Change Scale X/Y: 0.75

This will make the glyphs larger (and you only need to modify the top level because everything underneath (the child glyphs) scale) and rise above the map a bit more.

***Example 3: I want to add more data and remove some rows***

As long as you keep to the data format originally mapped in Data Mapper, you should be able to go back to your original Data Set and add/remove rows at will. Once you re-save the base file everything else will be taken care of.

**NOTE**: You cannot do the following and have auto-update work

* Add or remove columns
* Leave blank fields
* Add in elements that do not correspond to the mapped types (i.e. real in integer)

**Part 7: Glyph Legends, Rotation Series Calculations, Max/Min Guidelines for Data Mapper, and Alignment Charts (Additional Information)**

Some things have come up over and over again when dealing with Glyph Designer and Data Mapper –

* how can I add a legend that shows up where I want in Glyph Viewer / Portable Visualizer
* how do I modify the rotation values to make my glyph elements to orient them correctly
* I have several related glyph elements but their sizes don’t seem to match the underlying values. How can I size each element so they are proportional as a whole
* how many degrees should I input to make things evenly spaced in Glyph Designer

**Part 7a: Creating Legends for Glyph Viewer and Portable Viewer**

Legends are a quick way to convey exactly what each glyph element means without depending too much on the Glyph List (in Glyph Viewer) or “I” key in Portable Viewer. To set one up:

1. Create a **2048 x 1024** image file. This can be done in several programs such as MS Paint.
2. Grab a screen cap or two (or three depending on the complexity) of your glyph design

**NOTE**: Glyph Designer pulls the “world.png” file as its background. If you want a neutral color, this file will have to be replaced with your desired “world.png” image. Typically a neural solid color is a good choice to minimize distraction.

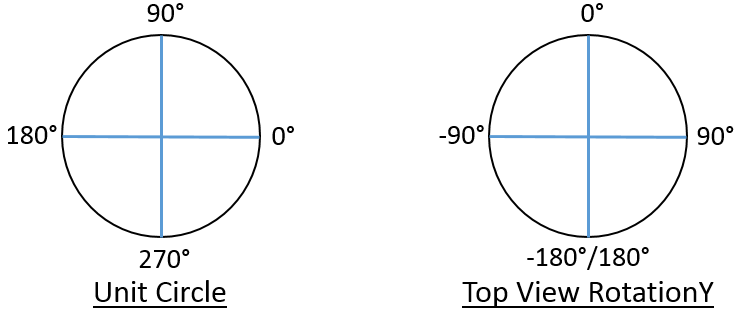
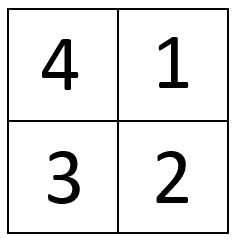
1. Label each element and paste the results into your image file. Save the file as \*.png
2. Open the Data Mapper file where you want to place the legend.
   1. Go to Base Object 🡺 Add Base Object
   2. Base Image Type: Local Image 🡺 Navigate to your Legend
   3. Position and orient the Legend to your choosing. For example if you want the image facing the **front** then X: 0.00, Y: 90.00, Z: 90.00 and Orientation: Front
   4. Click “OK” and then your Legend should be loaded and positioned properly
3. If you want to create a legend on **Left** or **Right**, use the following XYZ values
   1. Left: X:-180.00, Y: 0.00, Z: 90.00 – Orientation: Right – Height:180.00
   2. Right: X:180.00, Y:0.00, Z:90.00 – Orientation: Left – Height: 180:00

**NOTE**: Left/Right use square image spaces (1:1) instead of rectangular ones (2:1)

**Part 7b: Rotation Series Calculations**

Sometimes a glyph will have directional information attached and you will want to “point” it a specific way (i.e. planes, boats, and cars as they move around over time). The following information below will help you in data processing.

1. Rotation Y controls directional pointing in DataMapper and is different from a unit circle, and each directional vector can be placed in a quadrant (unless it points straight U/D/L/R)

1. Using a standard trigonometric function, one can extract a degree (or radian) value
2. Latitude change corresponds to ΔY, and longitude change corresponds to ΔX
3. Calculating a few derivative values will allow one to transform a vector defined by ΔY and ΔX into a specific RotationY value that can be directly mapped in DataMapper

The following information is used to define RotationY

1. ΔY, ΔX, and HYP
2. Quadrant Indicator and Quadrant Adjuster
3. Plus/Minus Adjuster

And the equations below are used (in Excel currently) to create the elements above

1. **ΔY** = Lat 2 – Lat 1, **ΔX** = Long 2 – Long 1, **HYP** = SQRT(ΔY2 + ΔX2)
2. **Quadrant Indicator** = IF(AND(ΔY >0, ΔX >0),1,0)+IF(AND(ΔY >0, ΔX <0),4,0)+IF(AND(ΔY <0, ΔX <0),3,0)+IF(AND(ΔY <0, ΔX >0),2,0)
3. **Quadrant Adjuster** = IF(QI=1,90,0)+IF(QI=2,90,0)+IF(QI=3,-90,0)+IF(QI=4,-90,0)
4. **Plus/Minus** = IF(QI=1,-1,0)+IF(QI=2,-1,0)+IF(QI=3,1,0)+IF(QI=4,1,0)

**NOTE**: QI = Quadrant Indicator from (b.)

Finally we take these elements and compute the RotationY value.

RotationY = IF(QI>0,(QA+(P/M)\*(DEGREES(ASIN(ΔY /HYP)))),0)

If done correctly, you will notice that all vectors falling into Q1 will have a RotationY value between 0 and 90, Q2 between 90 and 180, Q3 between -90 and -180, and Q4 between 0 and -90.

**Part 7c: Max/Min Guidelines for Data Mapper**

The examples below will go through how to assign proportional Max/Min values to each glyph. This section is needed when the user wants to have normalized glyph sizes across multiple related elements but the actual values are stored in different base data columns. Note that Data Mapper values are currently limited to 2 decimal places, so rounding will occur.

Example 1 (Min > 0): Default Glyph Size = 4, Max = 30, Min = 5

This is the simplest case where you may have something like an Age column and do not want the smallest value to be invisible in Glyph Viewer. Because you are only comparing one column, if the default sizing is correct then only the Min values will need to be changed.

Min = (5/30)\*(4) = **.67**

Example 2 (Min > 0, resize max): Default Glyph Size = 15, Max = 30, Min =5

Suppose the default size is too large. Once you have decided on the Max Base (i.e. 2) then

Max = (30/30)\*(Max Base = 2) = **2**  
Min = (5/30) \* (2) = **.33**

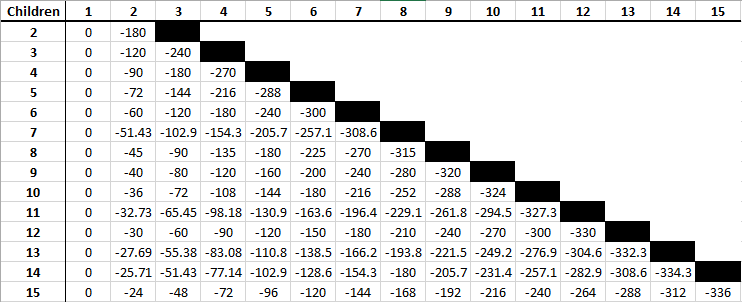
Example 3 (Multiple columns): Col 1 [0, 10] / Col 2 [5, 15] / Col 3 [15, 25] / Col 4 [0, 40]

In this last example you are normalizing across several related columns. Instead of using the local Max as a baseline, this process will using the global Max (which is 40). Let’s use size = 5.

Min 1 = (0/40)\*(5) = **0** Max 1 = (10/40)\*(5) = **1.25**  
Min 2 = (5/40)\*(5) = **.63** Max 2 = (15/40)\*(5) = **1.88**   
Min 3 = (15/40)\*(5) = **1.88** Max 3 = (25/40)\*(5) = **3.13**  
Min 4 = (0/40)\*(5) = **0** Max 4 = (40/40)\*(5) = **5**

**Part 7d: Glyph Designer Alignment Charts**

For each N=# of children for a circular (360) Glyph here are the values to space equivalently



And for each N=# of children on a rod topology (0 to -180) here are the values

