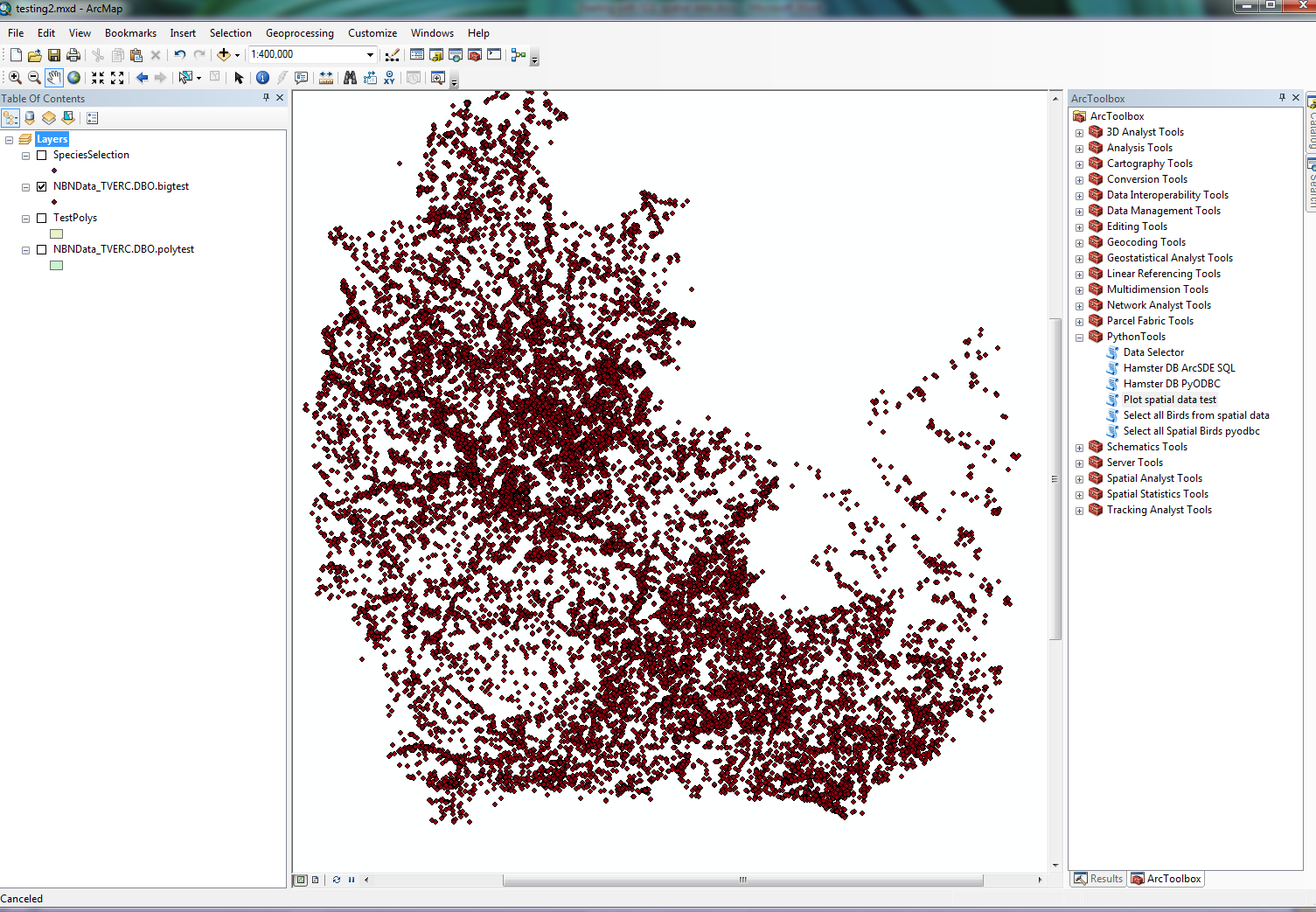
<https://tereshenkov.wordpress.com/2015/02/05/sql-server-spatial-functions-for-gis-users/>

This is a really useful link showing how to get a grip on the spatial objects.

**Issue 1: neither ArcSDE nor Pyodbc seem to work well with the geometry data held in SQL Server.**

Initial solution: Add query layer. Manual add of the entire database shows that this is not successful – polygon data are either not included or included as points (have not verified which):



**Full data layer of the spatial data held in the TV LERC NBN database.**

The explanation for this is that ArcGIS does not support flat layers that have multiple geometry types.

**Suggested solution:**

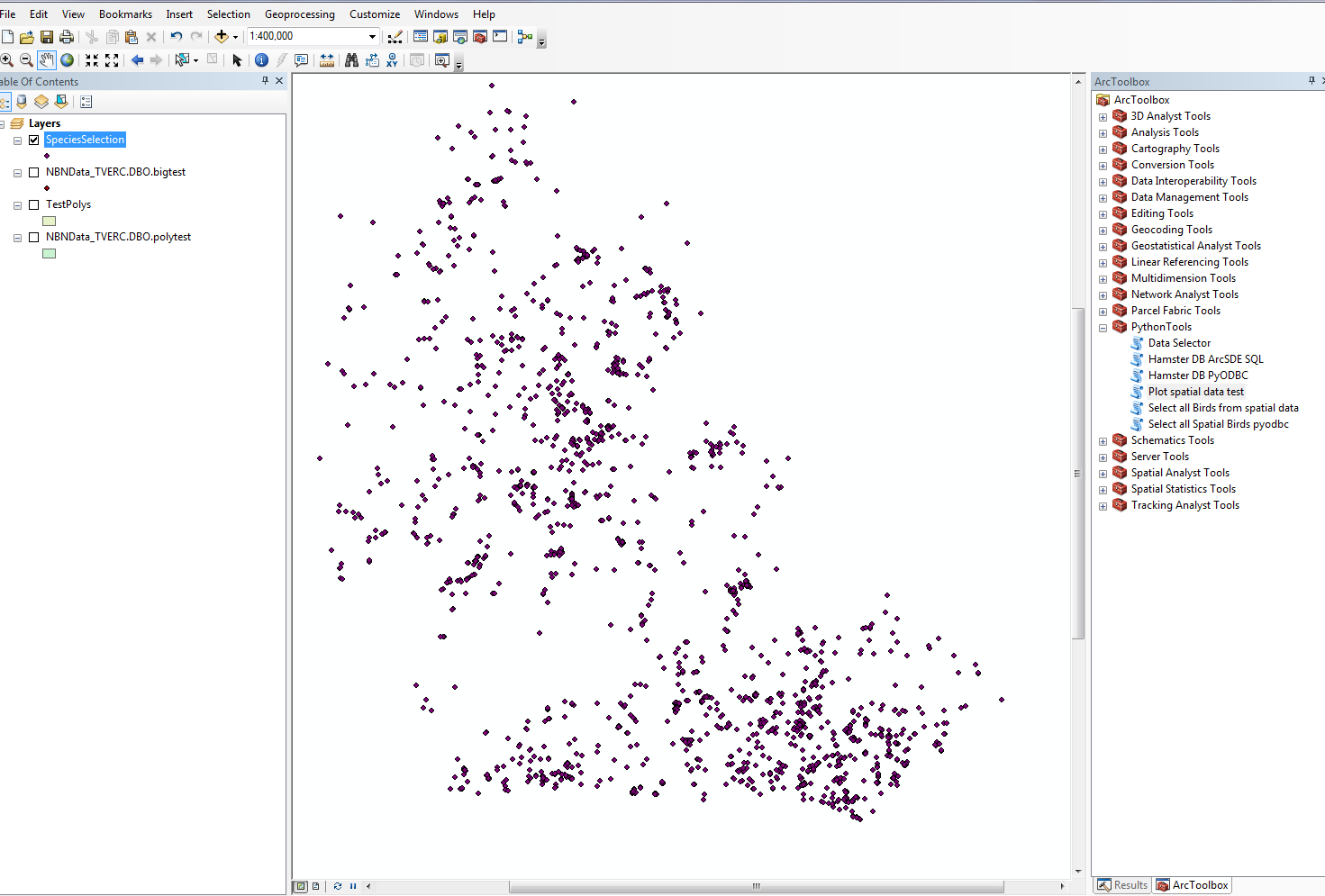
1. Create two temporary tables from the user specifications, one with point and one with polygon data. The data description of the geometry object gives access to this:

SELECT SP\_GEOMETRY, SP\_GEOMETRY.STAsText() as GeoDesc, TaxonName   
INTO HLPolyTest  
FROM dbo.TVERC\_Spp\_Full  
WHERE SP\_GEOMETRY.STAsText() LIKE 'POLY%'

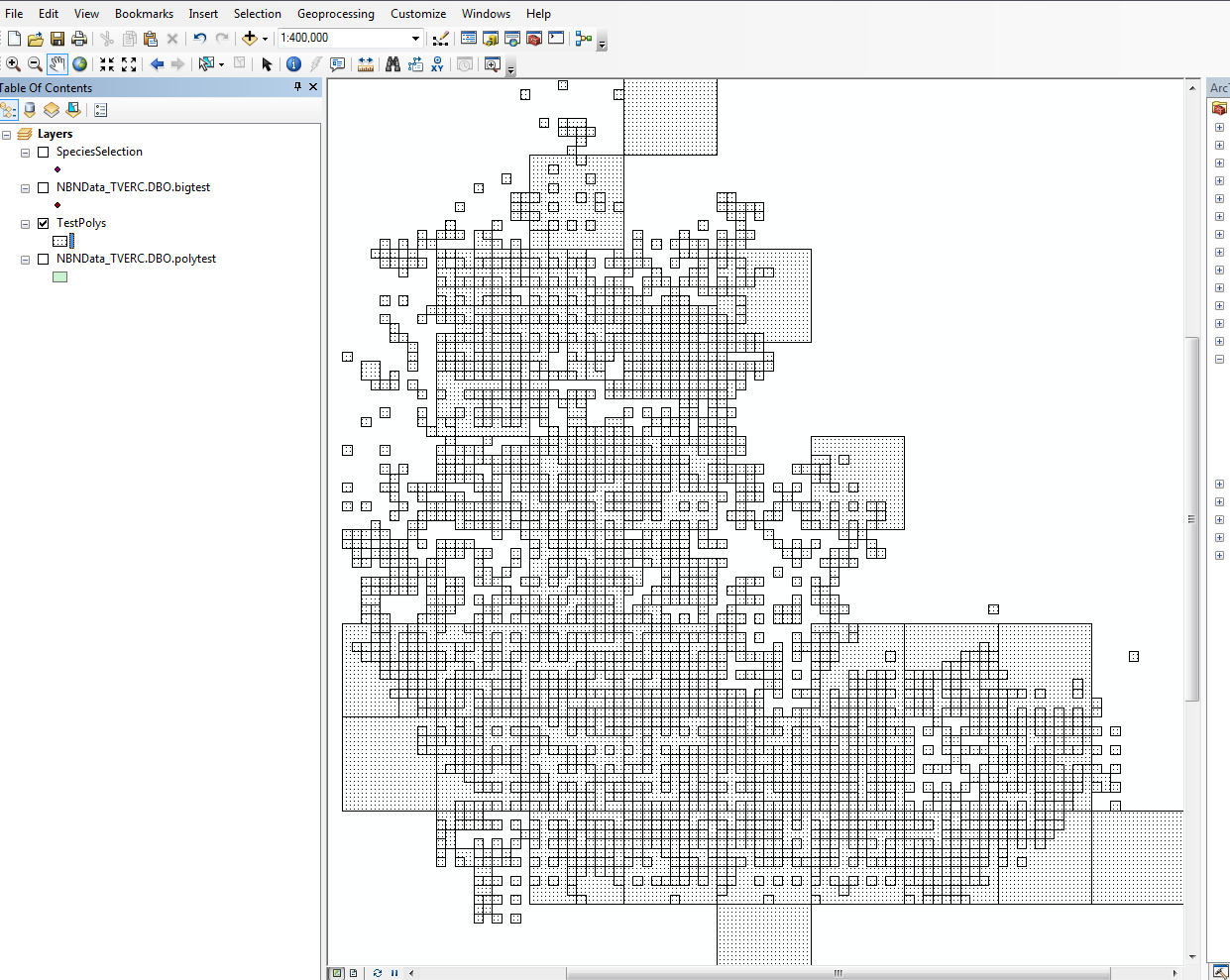
The clause to return point only data would be “ LIKE ‘POI%’ “

These two tables can then be grabbed as a query layer and added to the view, this time showing the points and polygons correctly (proven).

The two datasets could be presented as a group layer with two contributing layers in order to make the display logical; also naming convention could be [outputname]\_point and [outputname]\_poly for the two files.



Point data from test table



Polygon data from test table.