Conversations with Spatial Data

Image courtesy of fanfiction.net

Michael Olkin, GISP
Springfield Water and Sewer Commission
michael.olkin@waterandsewer.org

What to Expect Today

This Presentation is a SQL

What to Expect Today

Cats included to keep us entertained

Dedicated to Moonbeam
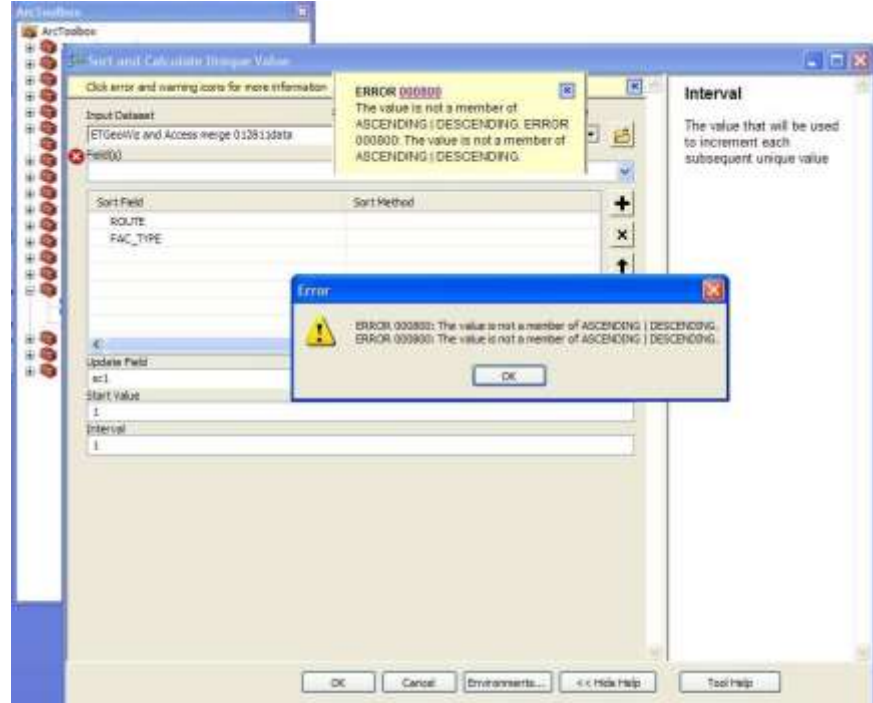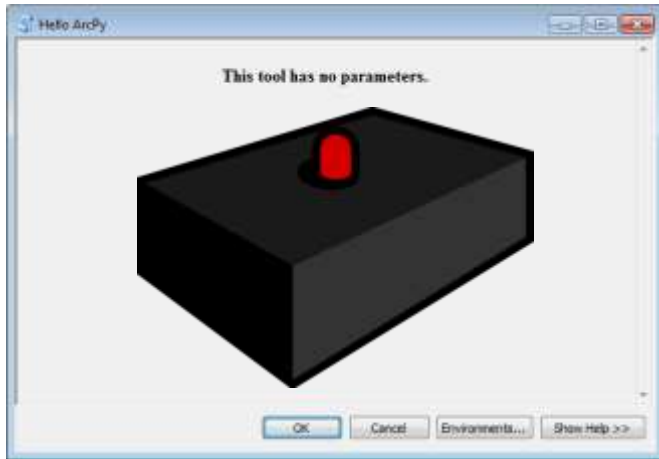
1999-2019

Pow! Sock!
Cats & SQL!

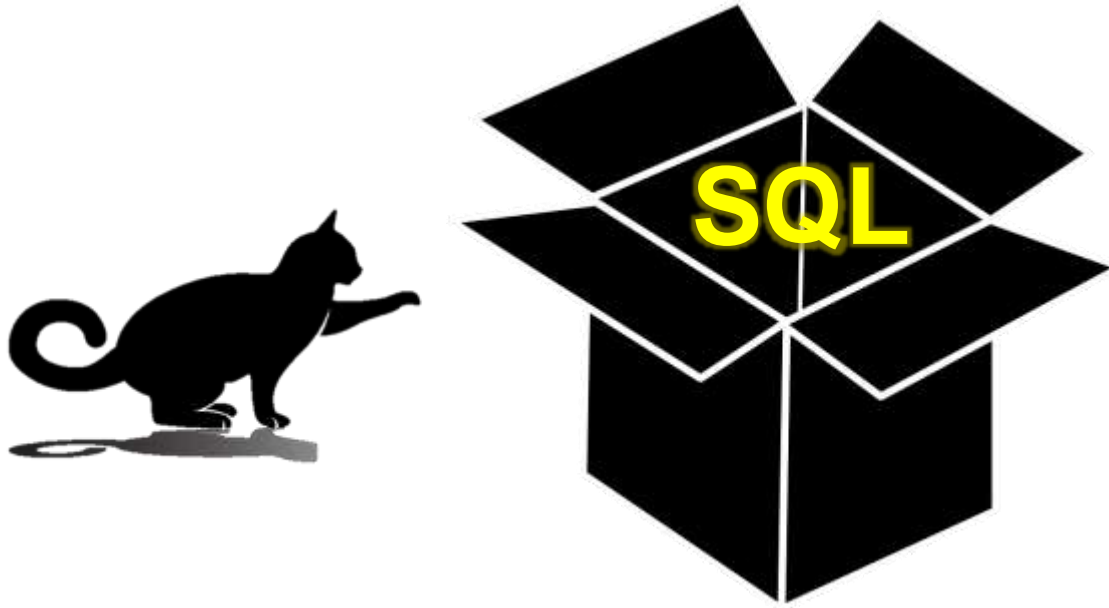# Guiding Principle #1

The Typical End User Doesn't Care About SQL

Why should we?

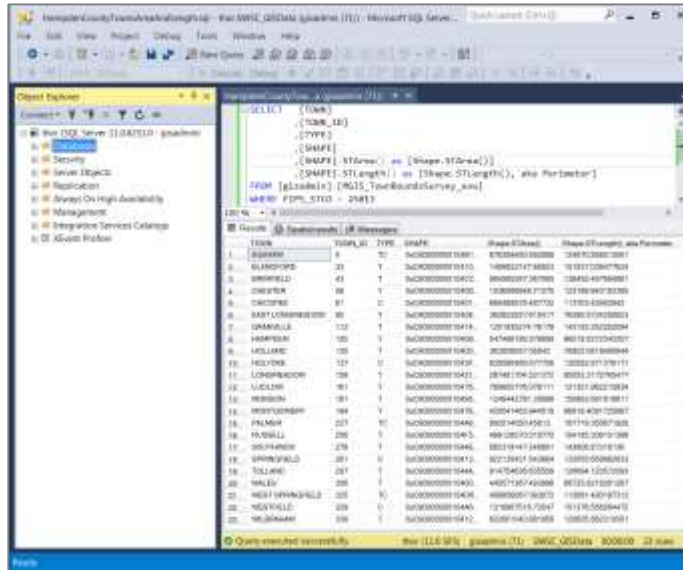# GIS Spatial Querying is a bit of a Black Box

SQL is somewhere inside of the Black Box

We can speak directly with our data

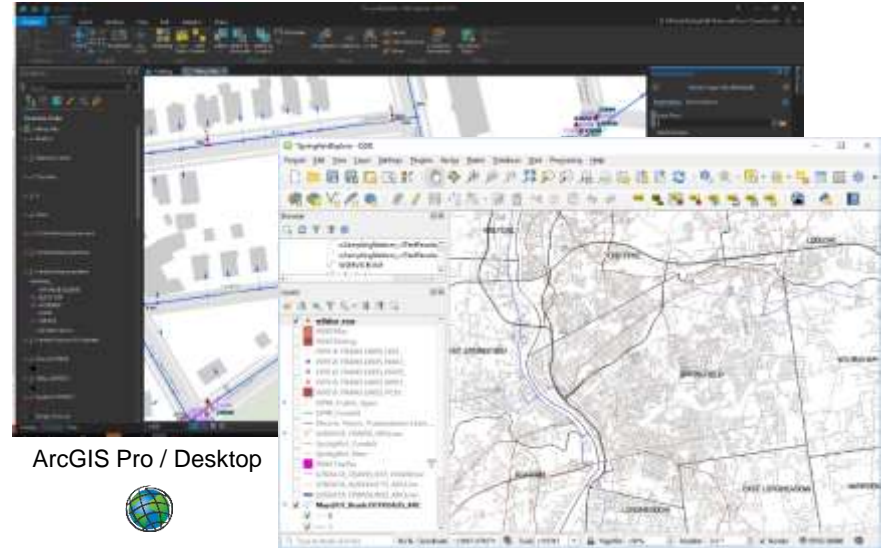# Today: Spatial Queries in Two Environments

Relational Database Management System

SQL Server Management Studio

Desktop GIS

ArcGIS Pro / Desktop

Quantum GIS (qGIS)

Why Bother with Spatial Queries?

Be Cross-Platform: Get Spatial without using GIS Software

Normalize: Avoid hard-coding redundant attributes

Stop the Geoprocessing Madness: cut down on geoprocessed data-scraps

Open Standards Have Provided a Common Language

OGC Simple Feature Specification

Standard for storing & querying spatial data

Longtime standard for various opensource formats
(PostgreSQL PostGIS leads the way)

Default Esri Enterprise Geodatabase storage format
(supported since ArcGIS 9.0)

http://www.opengeospatial.org/standards/sfs

Quiz 1: Sequel or Original?

Answer: **Sequel (Indiana Jones and the Temple of Doom)**

Basics: Line Length & Polygon Area

- GIS software automatically displays calculated lengths & areas
  - We can do the same via queries

(fe)Line Length

SELECT [SHAPE].**STLength()** as [Length (Ft)]
,CAST([SHAPE].**STLength()** as int) as [Rounded (Ft)]
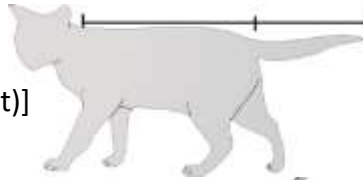
Length (Ft)	Rounded (Ft)
1056.36436184714	1056

Polygon Area

SELECT SHAPE.**STArea()** as [Area (SqFt)]
,CAST([SHAPE].**STArea()** as int) as [Rounded (SqFt)]
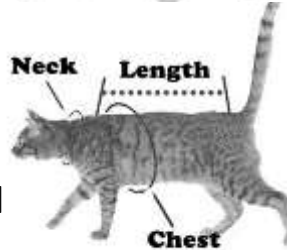
Area (SqFt)	Rounded (SqFt)
5165.25909423828	5165

ArcGIS Attribute Tables Include Queried Length & Area

*Courtesy of the **.STArea()** & **.STLength()** OGC standard SQL methods*

MA Community Boundaries

| TOWN	TOWN_ID *	TYPE	Shape *	Shape.STArea()	Shape.STLength()
AGAWAM	5	TC	Polygon	678359450.582886	124870.856014
BLANDFORD	33	T	Polygon	1489822147.959229	151837.038478
BRIMFIELD	43	T	Polygon	984585267.367065	139455.497565
CHESTER	59	T	Polygon	1036066948.710754	123169.940193
CHICOPEE	61	C	Polygon	665489578.487732	113703.429529
EAST LONGMEADO	85	T	Polygon	362622827.615417	76395.572426
GRANVILLE	112	T	Polygon	1201935274.76178	143192.252252
HAMPDEN	120	T	Polygon	547499195.579956	96518.837234
HOLLAND	135	T	Polygon	363809057.05542	78802.091849
HOLYOKE	137	C	Polygon	635590489.577759	130582.871378
LONGMEADOW	159	T	Polygon	267481704.531372	65052.317277
LUDLOW	161	T	Polygon	789950776.076111	121351.862216

Using .STArea() & .STLength() in our own Query

```sql
SELECT [TOWN]
      ,[TOWN_ID]
      ,[TYPE]
      ,[Shape]
      ,[Shape].STArea() as [Shape.STArea()]
      ,[Shape].STLength() as [Shape.STLength()]
FROM [dbo].[MGIS_TownBoundsSurvey_evw]
WHERE [FIPS_STCO] = 25013
ORDER BY [TOWN]
```


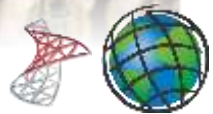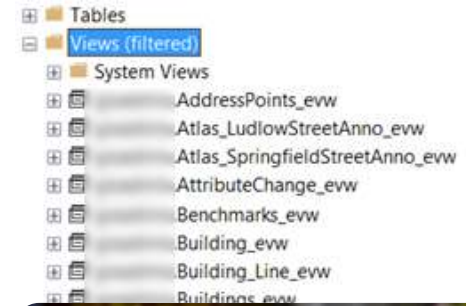|	TOWN	TOWN_ID	TYPE	SHAPE	Shape.STArea()	Shape.STLength(), aka Perimeter
1	AGAWAM	5	TC	0xC9080000104B1...	678359450.582886	124870.856013551
2	BLANDFORD	33	T	0xC9080000010410...	1489822147.95923	151837.038477624
3	BRIMFIELD	43	T	0xC90800000104CC...	984585267.367065	139455.497564861
4	CHESTER	59	T	0xC908000001040D...	1036066948.71075	123169.940193395
5	CHICOPEE	61	C	0xC9080000010401...	665489578.487732	113703.42952942
6	EAST LONGMEADOW	85	T	0xC908000001040E...	362622827.615417	76395.5724258823
7	GRANVILLE	112	T	0xC9080000010414...	1201935274.76178	143192.252252094
8	HAMPDEN	120	T	0xC9080000010409...	547499195.579956	96518.8372343257
9	HOLLAND	135	T	0xC9080000010400...	363809057.05542	78802.0918490844
10	HOLYOKE	137	C	0xC908000001043F...	635590489.577759	130582.871378171
11	LONGMEADOW	159	T	0xC9080000010421...	267481704.531372	65052.3172765477
12	LUDLOW	161	T	0xC9080000010475...	789950776.076111	121351.862215934
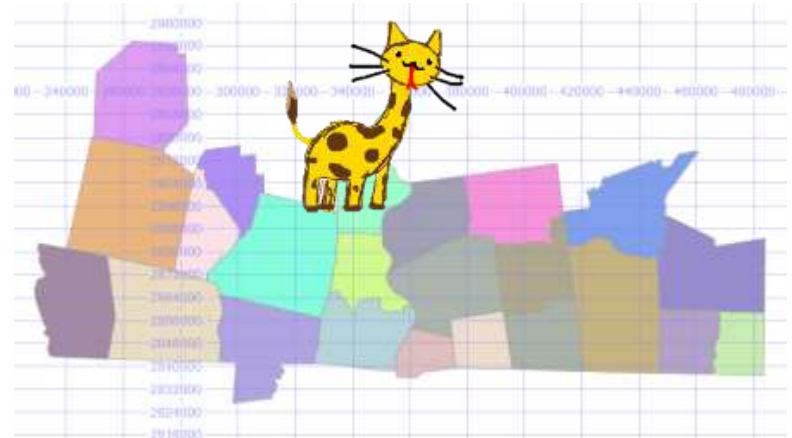
A word about querying Esri Versioned Feature Classes

Versioned Feature Classes have a "Versioned View"

The Versioned View provides the
current posted Default version of the data

Naming convention:
<Feature Class Name>_EVW

https://desktop.arcgis.com/en/arcmap/latest/manage-data/
using-sql-with-gdbs/what-is-a-versioned-view.htm

Tables
Views (filtered)
 System Views
  AddressPoints_evw
  Atlas_LudlowStreetAnno_evw
  Atlas_SpringfieldStreetAnno_evw
  AttributeChange_evw
  Benchmarks_evw
  Building_evw
  Building_Line_evw
  Buildings_evw

Quiz 2: **S**e**q**ue**l** or Original?

Answer: **Sequel (The Godfather Part II)**

Spatial Data Can Be Rendered Via SQL Queries

Points from X/Y or Lat/Lon fields

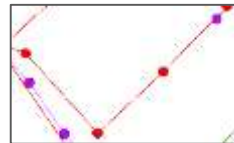| Latitude	Longitude
42.11638925	-72.83398931

Points from Polygon Centroids

Buffers from any features

Points from line Start or End nodes

Basics: Make Your Point

Constructing a GIS-readable point from x/y or lat/long columns

```sql
SELECT Latitude, Longitude, SpatialRef
```

Latitude	Longitude	SpatialRef
42.11638925	-72.83398931	4326

```sql
SELECT GEOMETRY::STGeomFromText('POINT(' +
CONVERT(VARCHAR, Longitude) + ' ' +
CONVERT(VARCHAR, Latitude) + ')', SpatialRef)
as Shape
```

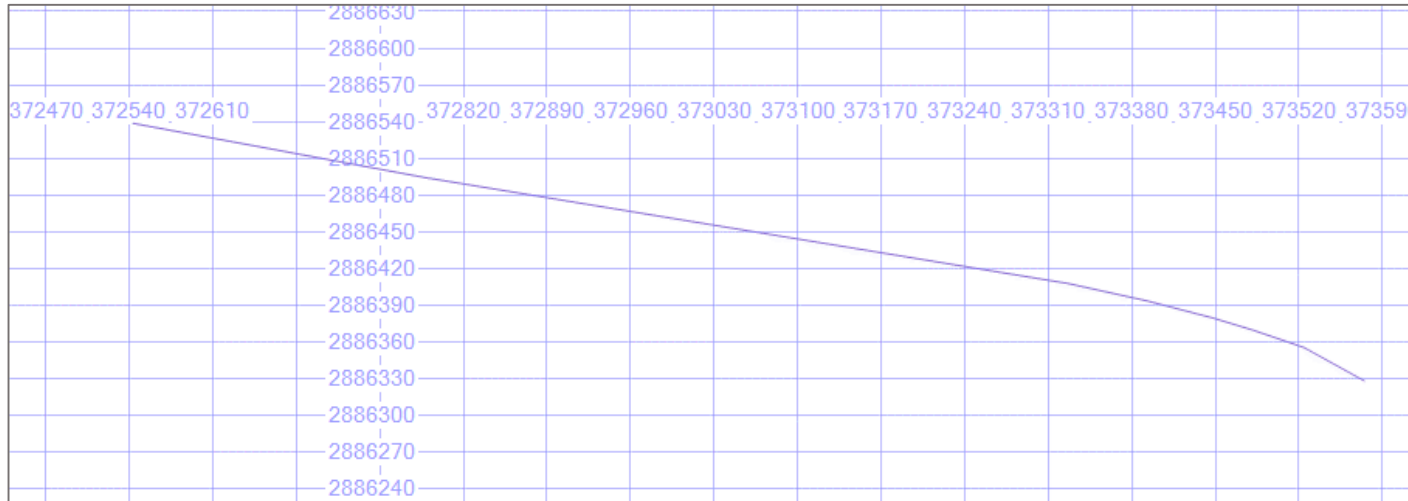| Shape |
|---|
| 0xE6100000010C79EAB214603552C0902DC… |


```sql
SELECT [Shape].ToString() as ShapeString
```

ShapeString
POINT (-72.83398931 42.11638925)

Basics: Draw The Line

LINESTRING (373574.7 2886328.4, 373524.0 2886355.7, 373481.9 2886369.7, 373447.6 2886379.8, 373392.2 2886393.9, 373326.7 2886407.9, 372787.8 2886494.5, 372543.0 2886538.8)

Basics: Create a Polygon

POLYGON ((356803.4 2876290.2, 356789.6 2876393.7, 356739.7 2876390.1, 356753.2 2876288.8, 356803.4 2876290.2))

Another way of looking at ArcToolbox

- ⊟ 📦 **Analysis Tools**
  - ⊞ 📚 Extract
  - ⊟ 📚 Overlay
    - 🔨 Erase
    - 🔨 Identity
    - 🔨 Intersect —————— Layer1.Shape.**STIntersection**(Layer2.Shape)
    - 🔨 Spatial Join
    - 🔨 Symmetrical Difference
    - 🔨 Union
    - 🔨 Update
  - ⊟ 📚 Proximity
    - 🔨 Buffer —————— Layer1.**STBuffer**(<distance>)
    - 🔨 Create Thiessen Polygons
    - 🔨 Generate Near Table
    - 🔨 Graphic Buffer
    - 📜 Multiple Ring Buffer
    - 🔨 Near —————— various ways, using **STDistance**

Distance: Point to Point

Where's the Nearest Hydrant?

```
DECLARE @PT GEOMETRY
SET @PT = (SELECT SHAPE
                FROM AddressPts as a
                WHERE a.Address = '555 MAIN ST')

SELECT TOP 1 h.HydrantID
,h.SHAPE.STDistance(@PT) as Distance
,h.SHAPE.ToString() as HydrantGeometry
FROM Hydrants as h
ORDER BY h.SHAPE.STDistance(@PT)
```

| HydrantID	Distance	HydrantGeometry
888	47.05	POINT (361358.95999999344 2863573.450000003)

Quiz 3: **S**e**q**ue**l** or Original?

Answer: **Sequel (Despicable Me 2)**

Point in Polygon

*In which neighborhood
is this address?*

SELECT a.Address, n.Neighborhood
FROM AddressPts as a
JOIN Neighborhoods as n
ON n.Shape.**STContains(**a.Shape**) = 1**
WHERE a.Address = '555 MAIN ST'

Address	Neighborhood
555 MAIN ST	Happyville

Greentown

Yellowton

555 Main St

Happyville
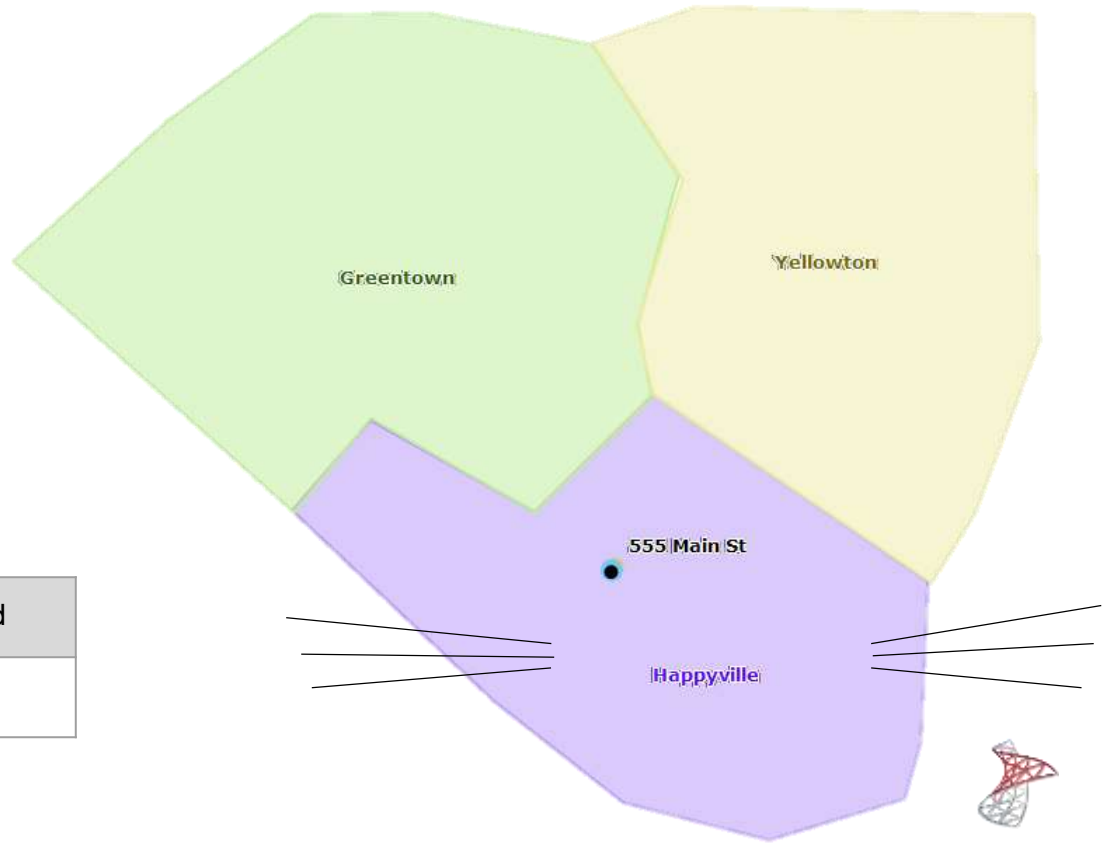
Polygon Center in Polygon

Example: Town in County

SELECT t.TOWN as Community ,c.COUNTY as County
,format((t.SUM_ACRES / c.AREA_ACRES),'P1') as [% Coverage of County]
FROM MGIS_TOWNBOUNDSSURVEY as t
JOIN MGIS_COUNTYBOUNDARIES as c
ON c.shape.**STIntersects(**t.shape.**STCentroid())** = 1
WHERE t.TOWN = 'SPRINGFIELD'

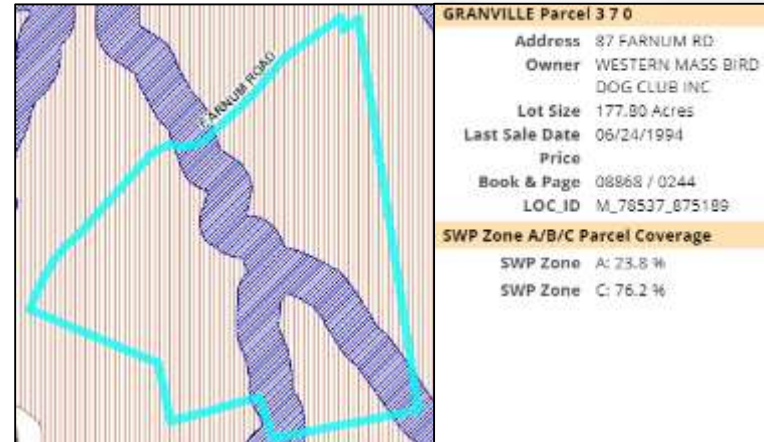| Community	County	% Coverage of County
SPRINGFIELD	HAMPDEN	5.2 %

Polygon Overlap Area

Percentage of parcel in Zone A, B or C watershed protection area

SELECT DISTINCT swp.SWPZONE as [SWP Zone]
,format(sum(swp.Shape.**STIntersection(**p.Shape**).STArea()**) / p.Shape.**STArea()**, 'P1') as [Overlap Pct]
FROM [MGIS_Parcels] as p
JOIN [MGIS_SurfaceWaterProtection_ZoneABC] as swp
ON p.Shape.**STIntersection(**swp.shape**).STArea()** > 0
WHERE p.LOC_ID = 'M_78537_875189'
GROUP BY swp.SWPZONE, p.Shape.**STArea()**

SWP Zone	Overlap Pct
A	23.8 %
C	76.2 %

GRANVILLE Parcel 3 7 0
Address   87 FARNUM RD
Owner   WESTERN MASS BIRD
            DOG CLUB INC
Lot Size   177.80 Acres
Last Sale Date 06/24/1994
Price
Book & Page 08868 / 0244
LOC_ID   M_78537_875189

SWP Zone A/B/C Parcel Coverage
SWP Zone A: 23.8 %
SWP Zone C: 76.2 %

Quiz 4: Sequel or Original?

Answer: **Original (The Terminator)**

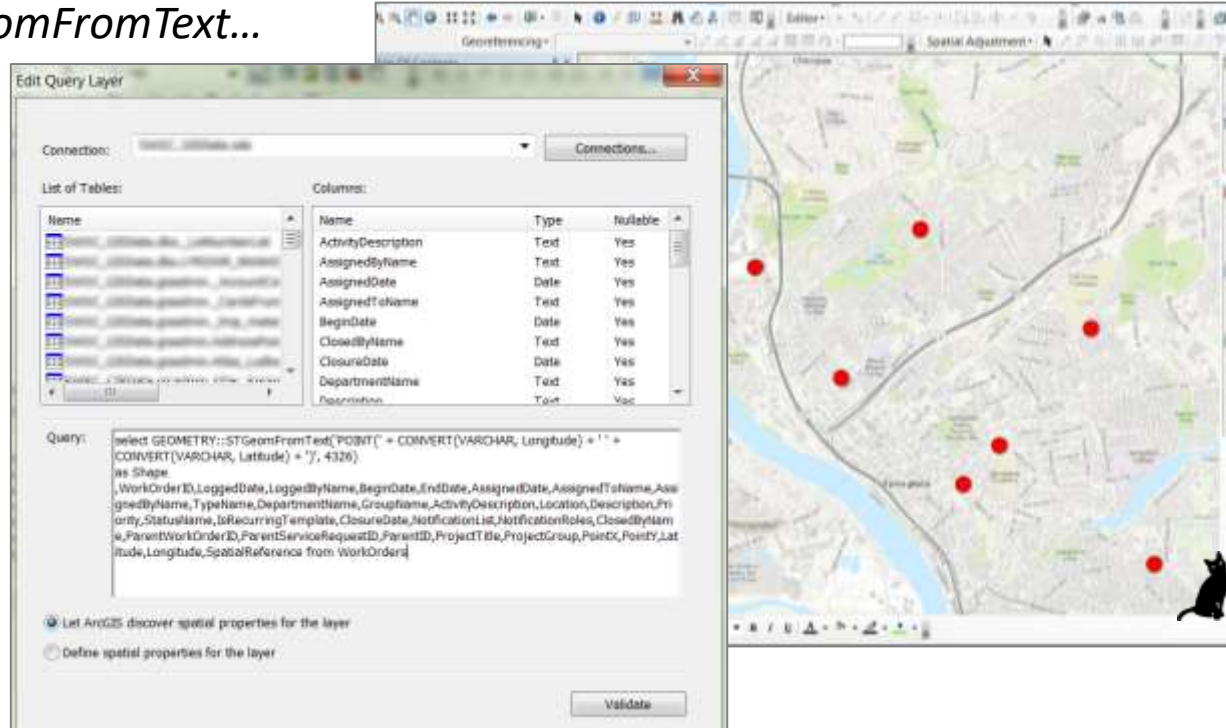# Roll your own SHAPE field from x/y or lat/lon fields

GEOMETRY::STGeomFromText…

"Live"
point data

Virtual Layer in
QGIS

Query Layer in
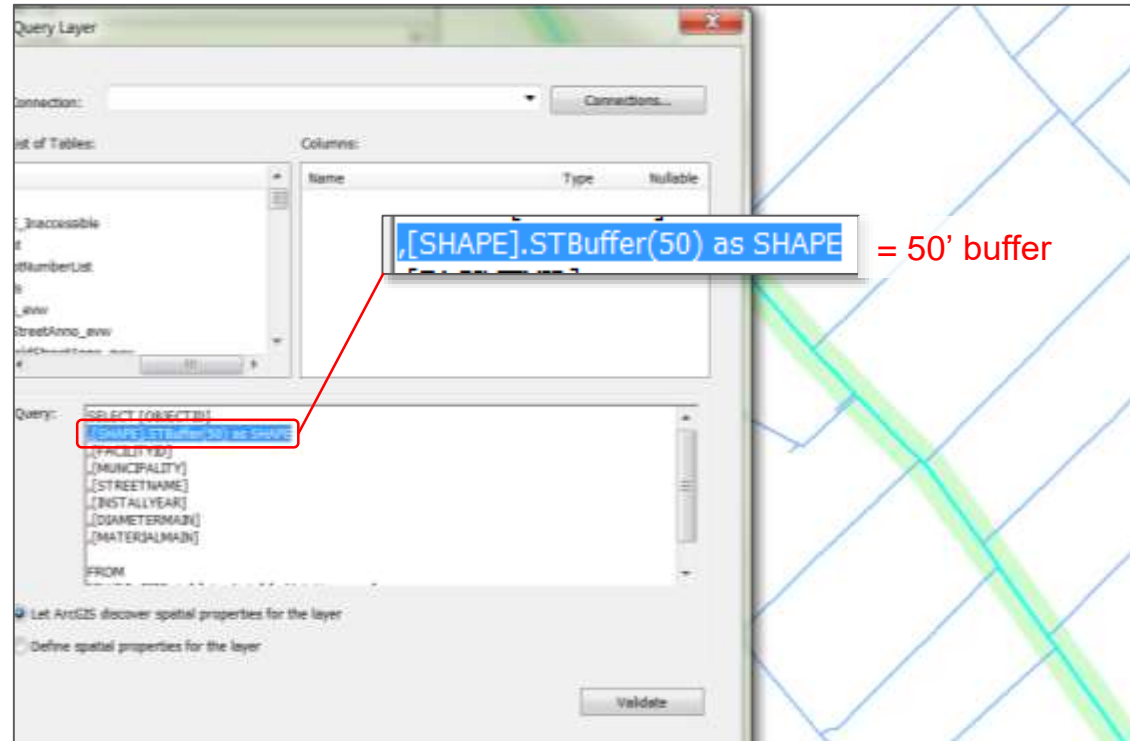ArcMap or
ArcGIS Pro

Compatible with
ArcGIS Server
Map Services

Buffer via an ArcGIS Query Layer

"Dynamic" Buffer

Not recommended for large selection sets

use a good WHERE clause to narrow the scope

`,[SHAPE].STBuffer(50) as SHAPE` = 50' buffer

Line / Point Connectivity

Selecting upstream / downstream manhole IDs by
joining to the coincident start & end nodes of the pipes

```sql
SELECT    g.[ASSETID] as GravityPipeAssetID
         ,F_MH_XY.ASSETID as Calculated_F_Node
         ,T_MH_XY.ASSETID as Calculated_T_Node
  FROM [GRAVITYPIPE] as g
JOIN [MANHOLE] as F_MH_XY
       on        (g.Shape.STStartPoint()).STX = F_MH_XY.Shape.STX
       and       (g.Shape.STStartPoint()).STY = F_MH_XY.Shape.STY
JOIN [MANHOLE] as T_MH_XY
       on        (g.Shape.STEndPoint()).STX = T_MH_XY.Shape.STX
       and       (g.Shape.STEndPoint()).STY = T_MH_XY.Shape.STY
```

GravityPipeAssetID	Calculated_F_Node	Calculated_T_Node
P997B	41C1	2035A
P43E2B	456C	45BE
P13FB	313D	3148
P3EFA	3149	407A
P4365	30DD	4364
P41D2	41C5	41C4
P1501	3146	3149
P41D9	3FBF	41CB
P5011	41C3	41CB
P18C2	4880	3FBF

Line / Point Connectivity

Checking for errors or miscodes with upstream/downstream node values

```
SELECT      g.[ASSETID] as GravityPipeAssetID ,(g.Shape.STEndPoint()).STX as PipeStartX, m.Shape.STX  as ManholeX
            ,(g.Shape.STEndPoint()).STY as PipeStartY, m.Shape.STY as ManholeY, g.[F_NODE] as Current_F_Node
            ,case isnull(mxy.ASSETID,'') WHEN '' THEN 'Spatial Mismatch' ELSE mxy.ASSETID END as Correct_F_Node
FROM GRAVITYPIPE as g
join SMH as m on g.F_NODE = m.ASSETID
join SMH as mxy
on (g.Shape.STStartPoint()).STX = mxy.Shape.STX and (g.Shape.STStartPoint()).STY = mxy.Shape.STY
where m.Shape.STX <> (g.Shape.STStartPoint()).STX and m.Shape.STY <> (g.Shape.STStartPoint()).STY
```

| GravityPipeAssetID	PipeStartX	ManholeX	PipeStartY	ManholeY	Current_F_Node	Correct_F_Node
P47EE	363865.05325	363865.05325	2875992.89775001	2875992.89775001	1B5A	4B01A
P4655	367720.359999999	361807.003999993	2865468.90449999	2875011.37774999	1BA2	Spatial Mismatch
P1122	379845.25	379906.25	2863031.75	2863499.5	24B0	Spatial Mismatch
P1E7B	362723.322750002	362786.249750003	2868109.30374999	2868147.99950001	33C7	Spatial Mismatch
P20CB	356680.140249997	356514.886749998	2866862.92325	2867068.1145	3DD7	Spatial Mismatch
P3E67	389488.972499996	389463.520999998	2879242.9945	2879427.88375001	3FE9	Spatial Mismatch

Practical Benefits of Start/End Node Attributes

Upstream / Downstream Trace

Advanced: Spatial Joining by Start or End of a Line

Problem: How can we detect lines that have improper directionality?
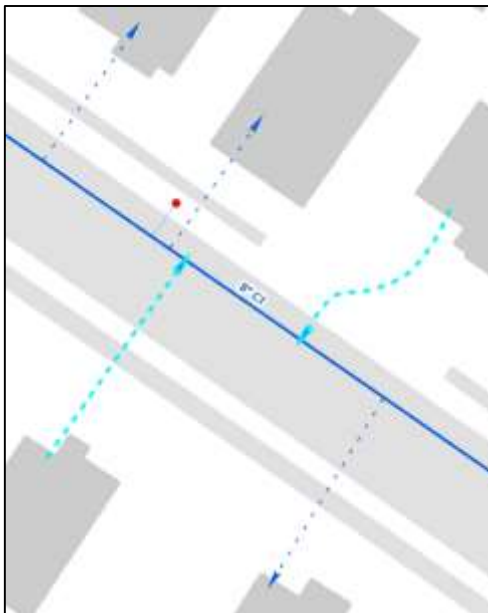
SELECT sL.SVC_LineID ,sL.WtrMainID as SvcLineWaterMainID
,wM.FACILITYID as WaterMainID, sL.DeliveryArea

FROM [dbo].[wServiceLine_evw] sL --*Water Service Lines*
LEFT OUTER JOIN [dbo].[wMainLine_evw] wM --*Water Mains*

--Joining STEndPoint or STStartPoint is processor intensive.
--Best to avoid running in a production environment.
ON sL.Shape.STEndPoint().STIntersects(wM.SHAPE) = 1

--Only show the backward service lines
WHERE wm.FACILITYID is not null

SVC_LineID	SvcLineWater MainID	WaterMainID	DeliveryArea
581	42492	49736	

Practice Data & Sample Code

https://github.com/MikeOlkinSwsc/SpatialSQL

Image courtesy of fanfiction.net

Questions?

Michael Olkin, GISP
Springfield Water and Sewer Commission
michael.olkin@waterandsewer.org   https://github.com/MikeOlkinSwsc/SpatialSQL