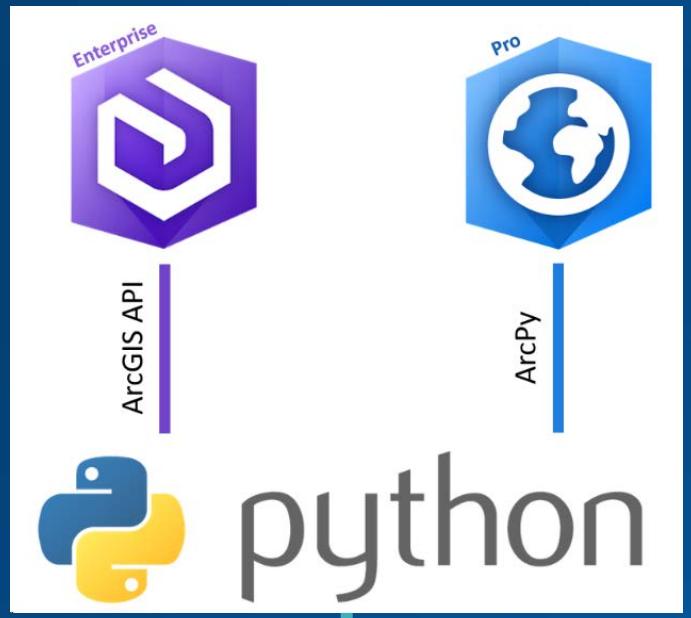
OKLAHOMA CORPORATION COMMISSION: OIL & GAS DIVISION

Colin Brooks – Regulatory Manager

OKLAHOMA Corporation Commission

Oklahoma Corporation Commission

Data Wrangling with Python

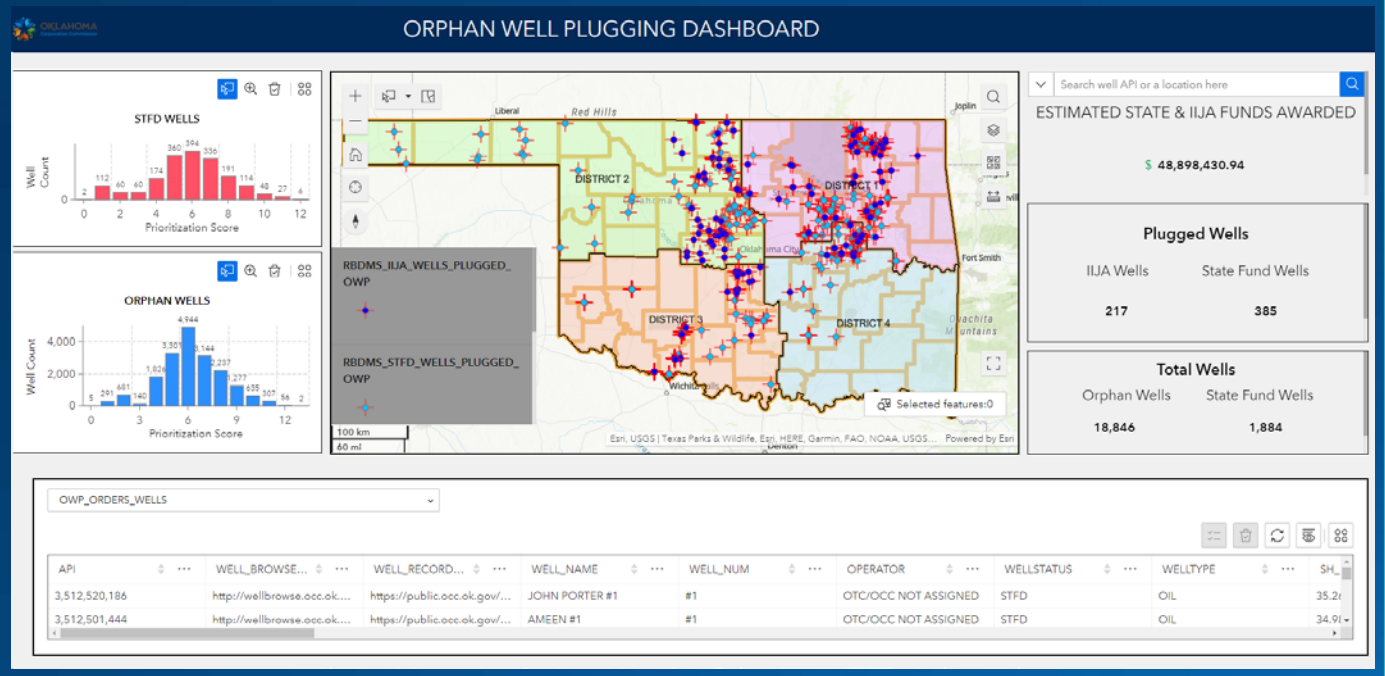


Plan Your Project

- Data sources
- Spatial awareness…what are you trying to display or solve with GIS
- Data prep…data science tools
- Geoprocessing…how we get spatial with analysis
- ESRI tools…present your data in a visually effective manner
- Feedback hub…what works, what doesn't

Oklahoma Corporation Commission

Want to Make an ESRI Dashboard or App? Got Data?

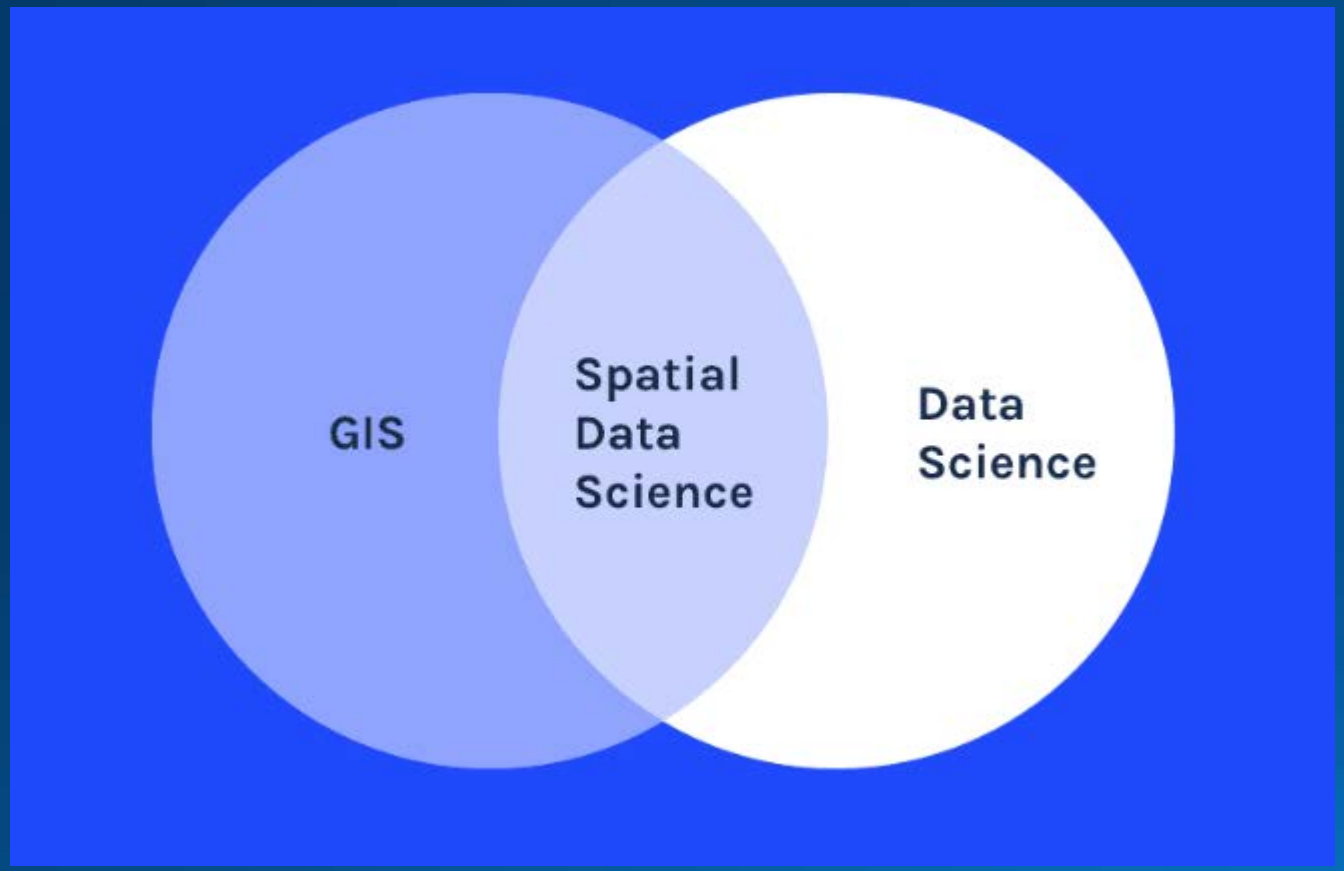


Data Sources for GIS

- Data can come from anywhere… **Databases, websites, network storage, etc.**
- Be prepared to deal with all kinds of data sources and formats… **CSV, Excel, SQL, XML, GDB, etc.**
- Data may be clean or dirty ☹
- Can't do geoprocessing or ESRI visualization until your data is properly **wrangled**!

Oklahoma Corporation Commission

Spatial Data Awareness



Questions to Ask

- What problem are we solving?
- How does GIS complement or enhance the data?
- What kind of processing does the data require?
- Known uncertainties (spatial or in the data)?
- How can we improve the data either with data processing or geoprocessing routines?

Oklahoma Corporation Commission

GIS Data Prep

```python
import pandas as pd
import os,glob, win32com.client,xlsxwriter,shutil,sys
from datetime import datetime
import pyodbc
from sqlalchemy import create_engine

Server='Server_Name'
Database='Database_Name'
UserName='USER'
cnxn = pyodbc.connect("Driver={SQL Server};"
"Server=Server_Name;"
"Database=Database_Name;"
"Trusted_Connection=yes")

Orders='C:\\Data\\flatfile.xlsx'
out_path='C:\\Data\\Manipulated_flatfile.xlsx'
with pd.ExcelWriter(out_path, engine="xlsxwriter", options = {'strings_to_numbers': True, 'strings_to_formulas': False}) as writer:
        try:
                df=pd.read_excel(Orders, sheet_name='WORKING', engine="openpyxl", skiprows=[1])
                df['API']=df['API'].str.replace('-','')
                mask = df['some column'].str.contains('search param',case=False, na=False)
                df[mask].to_excel(writer, sheet_name = "Sheet1",startrow=1, header = False, index = False)
                workbook=writer.book
                worksheet=writer.sheets['Sheet1']
                (max_row,max_col)=df.shape
                column_settings=[{'header':column} for column in df.columns]
                worksheet.add_table(0,0,max_row,max_col - 1,{'columns': column_settings})
        except AssertionError:
                pass
df=pd.read_excel(out_path, sheet_name='Sheet1', engine="openpyxl")
cursor = cnxn.cursor()
cursor.execute('''
                DELETE FROM Table_Name
                ''')
cnxn.commit()
cursor = cnxn.cursor()
engine = create_engine('mssql+pyodbc:///?odbc_connect={}'.format("DRIVER={SQL Server};SERVER=Server_Name;DATABASE=Database_Name;Trusted_Connection=yes;"))
df.to_sql('Table_Name', schema='dbo', con = engine, index=False,if_exists='replace')
```

Wrangling Data Files for GIS

- Not all data is perfect and ready for GIS use. Sometimes you need to manipulate it
- Automating data tasks with Python >> manual imports
- Pandas and other data science tools are very powerful! Use them!

Oklahoma Corporation Commission

GIS Data Prep (cont.)

```
df['some column']=df['some column'].str.replace('character','new character')
```

```
mask = df['some column'].str.contains('search param',case=False, na=False)
```

```
format1 = workbook.add_format({"num_format": "#,##0.00"})
worksheet.set_column(1,1,None, format1)
```

More Examples

- Need to remove a character? Use Pandas!
- Need to filter your active dataset? Use Pandas!
- Need to format specific fields for correct appearance in ESRI? Use Pandas!
- Use any good data science toolset with documentation (Pandas has lots of documentation)

Oklahoma Corporation Commission

Other Data Sources? Wrangle Them with Python Too

```python
import pyodbc
import pandas as pd
import xlsxwriter
import os, shutil,glob,sys
Server='Server_Name'
Database='Database_Name'
UserName='USER'
cnxn = pyodbc.connect("Driver={SQL Server};"
"Server=Server_Name;"
"Database=Database_Name;"
"Trusted_Connection=yes")
query=open('C:\\Data\\Script.sql','r')
out_path=r"C:\Data\wrangled_data.xlsx"
with pd.ExcelWriter(out_path, engine="xlsxwriter", options = {'strings_to_numbers': True, 'strings_to_formulas': False}) as writer:
    try:
        df=pd.read_sql(query.read(), cnxn)
        df.to_excel(writer, sheet_name = "Sheet1",startrow=1, header = False, index = False)
        workbook=writer.book
        worksheet=writer.sheets['Sheet1']
        (max_row,max_col)=df.shape
        column_settings=[{'header':column} for column in df.columns]
        worksheet.add_table(0,0,max_row,max_col - 1,{'columns': column_settings})
        worksheet.set_column(0,max_col - 1, 12)
    except AssertionError:
        pass
```

```python
# Join well data to STFD FUNDS LIST
arcpy.conversion.ExportTable(r"C:\Data\IIJA_WELLS\csv\IIJA_WELLS_Orders_OWP.xlsx\Sheet1$", r"C:\Data\OWP_Orphan_Well_Program\Orphan_GIS\Orphan_GIS.gdb\IIJA_WELLS_ORDERS_OWP"
RBDMS_STFD_WELLS_OWP = "C:\\Data\\OWP_Orphan_Well_Program\\Orphan_GIS\\Orphan_GIS.gdb\\RBDMS_STFD_WELLS_OWP"
IIJA_ORDERS_TEMP = "C:\\Data\\OWP_Orphan_Well_Program\\Orphan_GIS\\Orphan_GIS.gdb\\IIJA_ORDERS_TEMP"
arcpy.management.CopyFeatures(in_features=RBDMS_STFD_WELLS_OWP, out_feature_class=IIJA_ORDERS_TEMP, config_keyword="", spatial_grid_1=None, spatial_grid_2=None, spatial_grid
arcpy.management.JoinField("IIJA_ORDERS_TEMP", "API", "IIJA_WELLS_ORDERS_OWP", "API", "F__OF_WELLS;ATTORNEY_ASSIGNED_TO;CAT;CHANGE_UP_OR_DOWN;CONTRACT_AWARD_DATE;COST;D_;DAT
```

Python Handles Any Data

- Pandas read_sql command can read stored SQL queries directly
- ArcPy Library reads and writes native GIS formats!

OKLAHOMA Corporation Commission

Oklahoma Corporation Commission

ArcPy for Geoprocessing

```python
import arcpy

def Model():  # Model

    # To allow overwriting outputs change overwriteOutput option to True.
    arcpy.env.overwriteOutput = True

    arcpy.ImportToolbox(r"c:\program files\arcgis\pro\Resources\ArcToolbox\toolboxes\Data Management Tools.tbx")
    IIJA_WELLS_OWP_csv = "C:\\Data\\IIJA_WELLS\\csv\\IIJA_WELLS_OWP.csv"

    # Process: XY Table To Point (XY Table To Point) (management)
    RBDMS_IIJA_WELLS_OWP = "c:\\Data\\OWP_Orphan_Well_Program\\Orphan_GIS\\Orphan_GIS.gdb\\RBDMS_IIJA_WELLS_OWP"
    arcpy.management.XYTableToPoint(in_table=IIJA_WELLS_OWP_csv, out_feature_class=RBDMS_IIJA_WELLS_OWP, x_field="SH_LON", y_field="SH_LAT", z_field="", coordinate_system="GEOGC
    arcpy.sa.ExtractMultiValuesToPoints("RBDMS_IIJA_WELLS_OWP", r"C:\Data\OWP_Orphan_Well_Program\Orphan_GIS\Orphan_GIS.gdb\Surface_Water_Combined_Raster OWP_Sur_Water;C:\Data\O
    arcpy.management.CalculateField("RBDMS_IIJA_WELLS_OWP", "OWP_BTW", """IIF($feature.BTW>500,"2",IIF($feature.BTW>100,"1","0"))""", "ARCADE", '', "SHORT", "NO_ENFORCE_DOMAINS"
    arcpy.management.CalculateField("RBDMS_IIJA_WELLS_OWP", "OWP_Priority_Score", "$feature.OWP_Sur_Water+$feature.OWP_Aquifers+$feature.OWP_BTW+$feature.OWP_Disadv+$feature.OWP_
if __name__ == '__main__':
    # Global Environment settings
    with arcpy.EnvManager(scratchWorkspace=r"C:\Data\OWP_Orphan_Well_Program\Orphan_GIS\Orphan_GIS.gdb", workspace=r"C:\Data\OWP_Orphan_Well_Program\Orphan_GIS\Orphan_GIS.gdb"):
        Model()
```

Spatial Wrangling with ArcPy

- Now that we have our data sourced and wrangled maybe we need to do some geoprocessing
- ArcPy Library is capable of powerful spatial data routines!
- Any geoprocessing tool you have ever used in desktop GIS should be available for scripting in ArcPy

Oklahoma Corporation Commission

ArcGIS API for Python

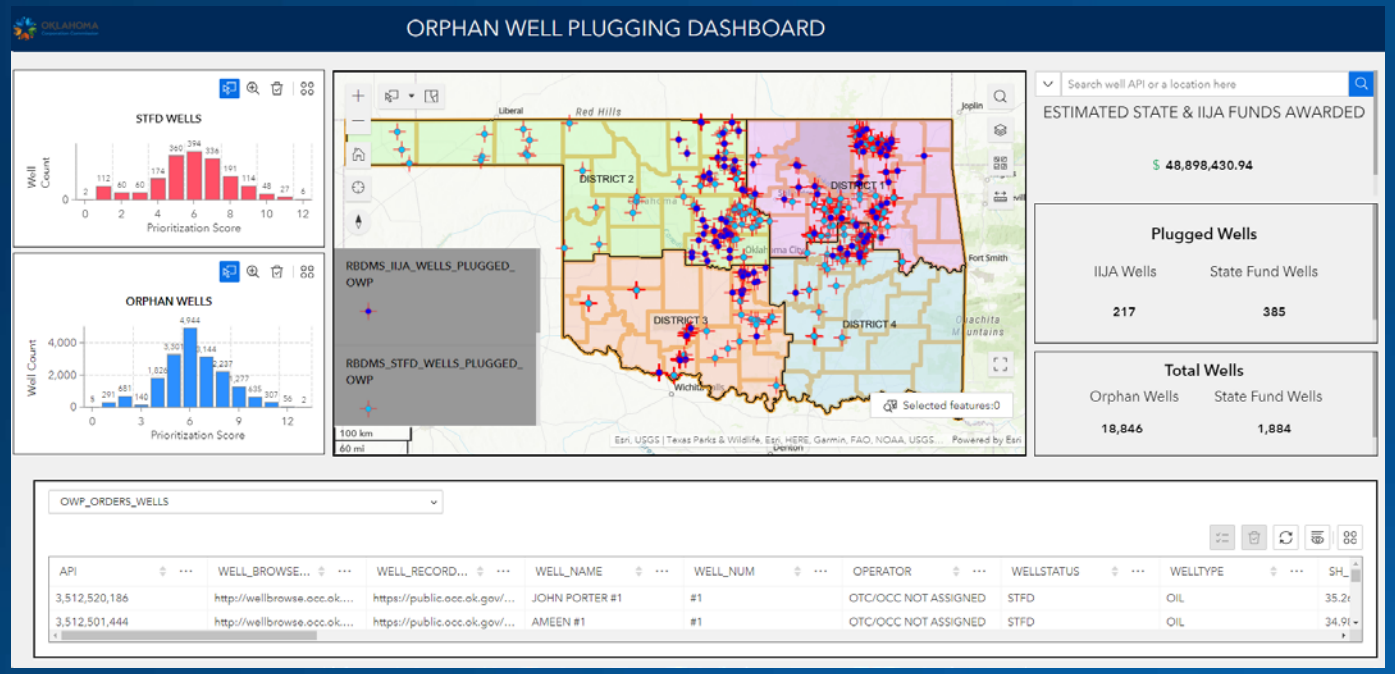
```python
import arcpy
import os, sys, shutil
import keyring
from arcgis.gis import GIS

### Start setting variables
# Set the path to the project
prjPath = r"C:\Data\WP_Orphan_Well_Program\Orphan_GIS\Orphan_GIS.aprx"
folder = r"C:\Data\Enterprise_Update_Temp\test"

# Update the following variables to match:
# Feature service/SD name in Portal, user/password of the owner account
username = "user"
mypw = keyring.get_password("Enterprise", username)
arcpy.SignInToPortal('portalname', username, mypw)
sd_fs_name = "RBDMS_IIJA_FUNDS_WELLS_OWP"
portal = "https://gis.occ.ok.gov/portal/home" # Can also reference a local portal
user = username
password = mypw

# Set sharing options
shrOrg = True
shrEveryone = True
shrGroups = ""

### End setting variables

# Local paths to create temporary content
relPath = r"C:\Data\Enterprise_Update_Temp\test"
sddraft = os.path.join(relPath, "RBDMS_IIJA_FUNDS_WELLS_OWP.sddraft")
sd = os.path.join(relPath, "RBDMS_IIJA_FUNDS_WELLS_OWP.sd")

# Create a new SDDraft and stage to SD
print("Creating SD file")
arcpy.env.overwriteOutput = True
prj = arcpy.mp.ArcGISProject(prjPath)
m = prj.listMaps("Map")[0]
lyr = m.listLayers("RBDMS_IIJA_FUNDS_WELLS_OWP")[0]
arcpy.mp.CreateWebLayerSDDraft(lyr, sddraft, sd_fs_name, "MY_HOSTED_SERVICES", "FEATURE_ACCESS","", True, True)
arcpy.StageService_server(sddraft, sd)



print("Connecting to {}".format(portal))
gis = GIS(portal, user, password)
```

Publishing Your Web Data

- ArcGIS API for Python extends your Python Libraries to ESRI's Web GIS
- ArcGIS API makes scripting automatic updates to Web GIS possible
- Can also retrieve data from GIS Portals for use in Python
- Works for Hosted Feature Layers and Map Image Layers published to Enterprise or AGO Portals

Oklahoma Corporation Commission

Finally Some Results!



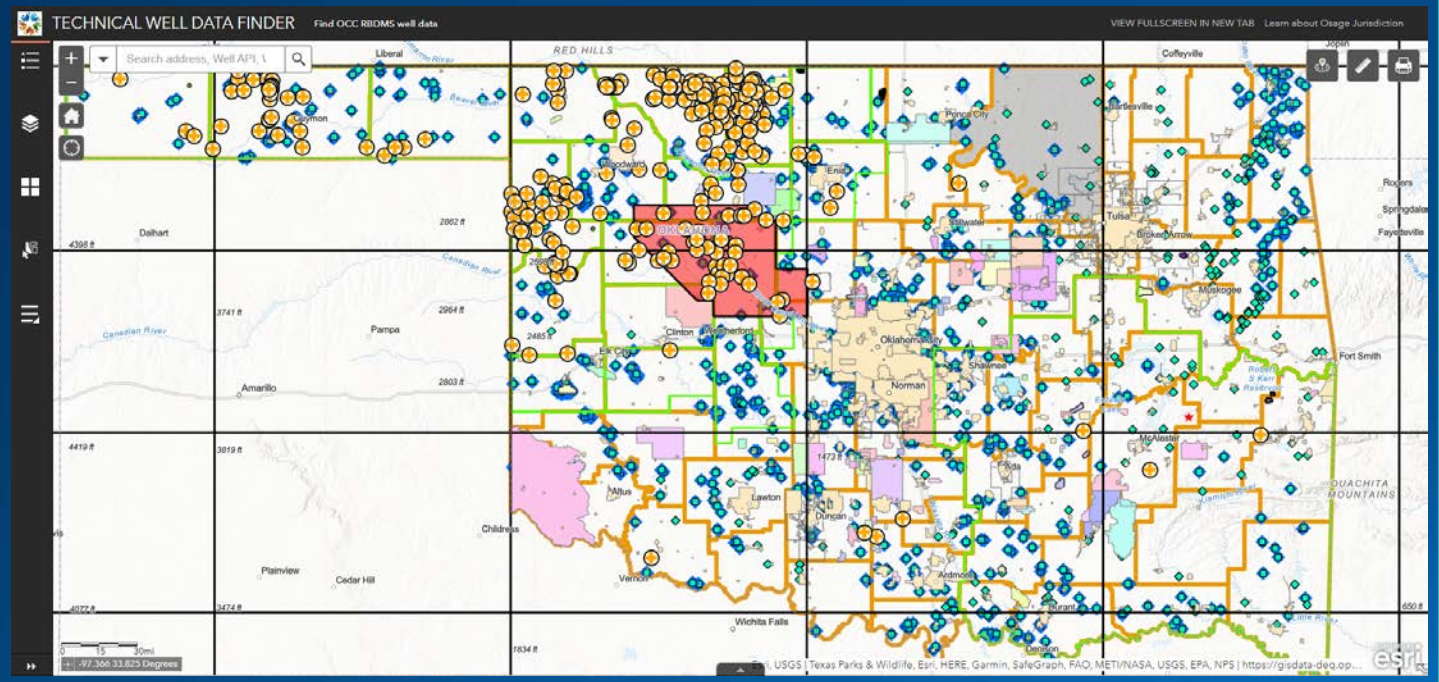
https://gis.occ.ok.gov/portal/apps/experiencebuilder/experience/?id=419e1e0be19245a7931f5aeb8580b1fd

Visualizing Data with ESRI

- Once all the complicated data wrangling, analysis, and publishing is done we can visualize our hard work!
- View your data in a Dashboard or WebApp and make changes if the results are not desired
- Unlikely to get all data formatted correctly the first time…stick with it!

Oklahoma Corporation Commission

Finishing Touches



Listen to Your Audience

- Your users will have helpful suggestions about things to improve
- No data process is perfect on the first attempt. Review your results with stakeholders and correct your data prep or geoprocessing tasks accordingly
- Every App or Dashboard I have made was improved by feedback from users!

OKLAHOMA Corporation Commission

Questions, Comments, Jokes

???