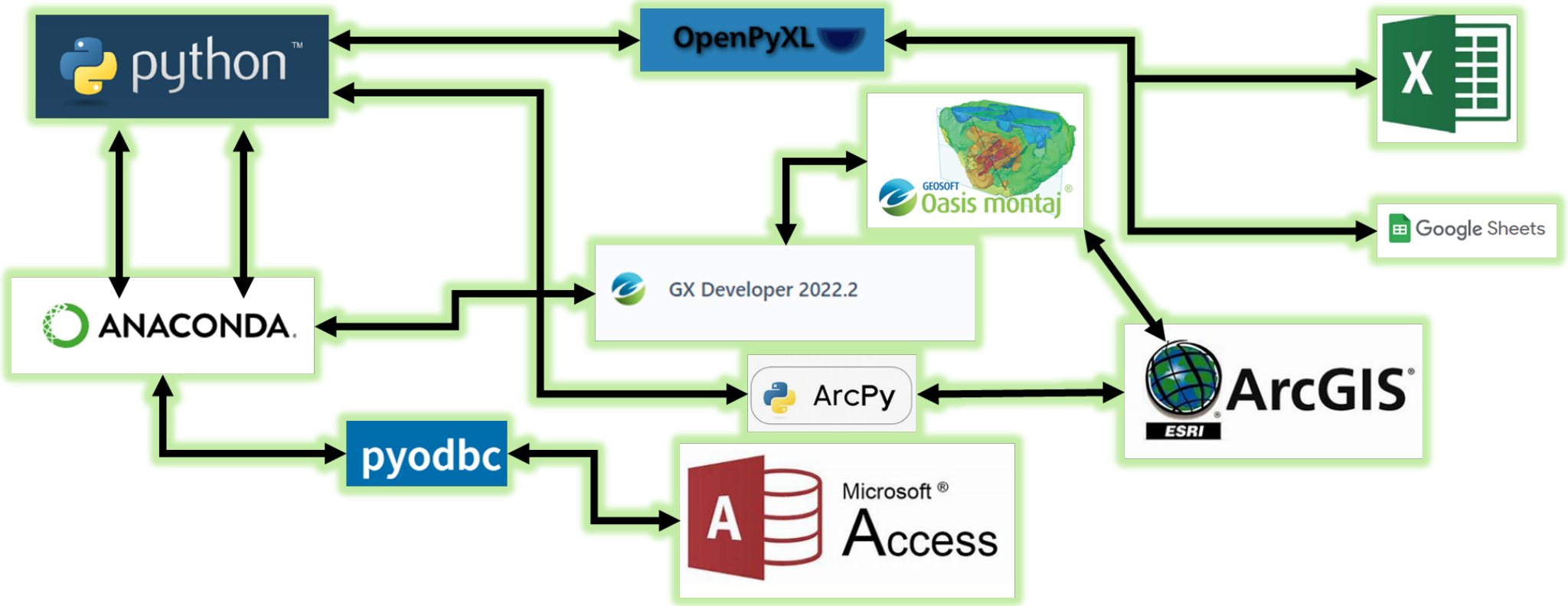
Geophysical data management and automation within the context of a large scale MMRP Project

April 5, 2023



Joshua Celestine – Weston Solutions
J. R. Candlish – Weston Solutions

Trust. Performance. People.

Committed to Creating a Better Tomorrow

Formerly Used Defense Site: Camp Croft

Project 7 Munitions Response Site

In the past year Weston has employed robust data management practices at scale with respect to the remediation effort at Camp Croft, Spartanburg, South Carolina. The Project 7 Munitions Response Site (MRS) is a sizeable remediation effort, as the project includes 2,870 grids and a survey area of approximately 609 acres located within a rugged and densely wooded area of Croft State Park. Owing to this projects unique scale and sophistication, Weston would like to share challenges faced and learnings from managing the voluminous data set.

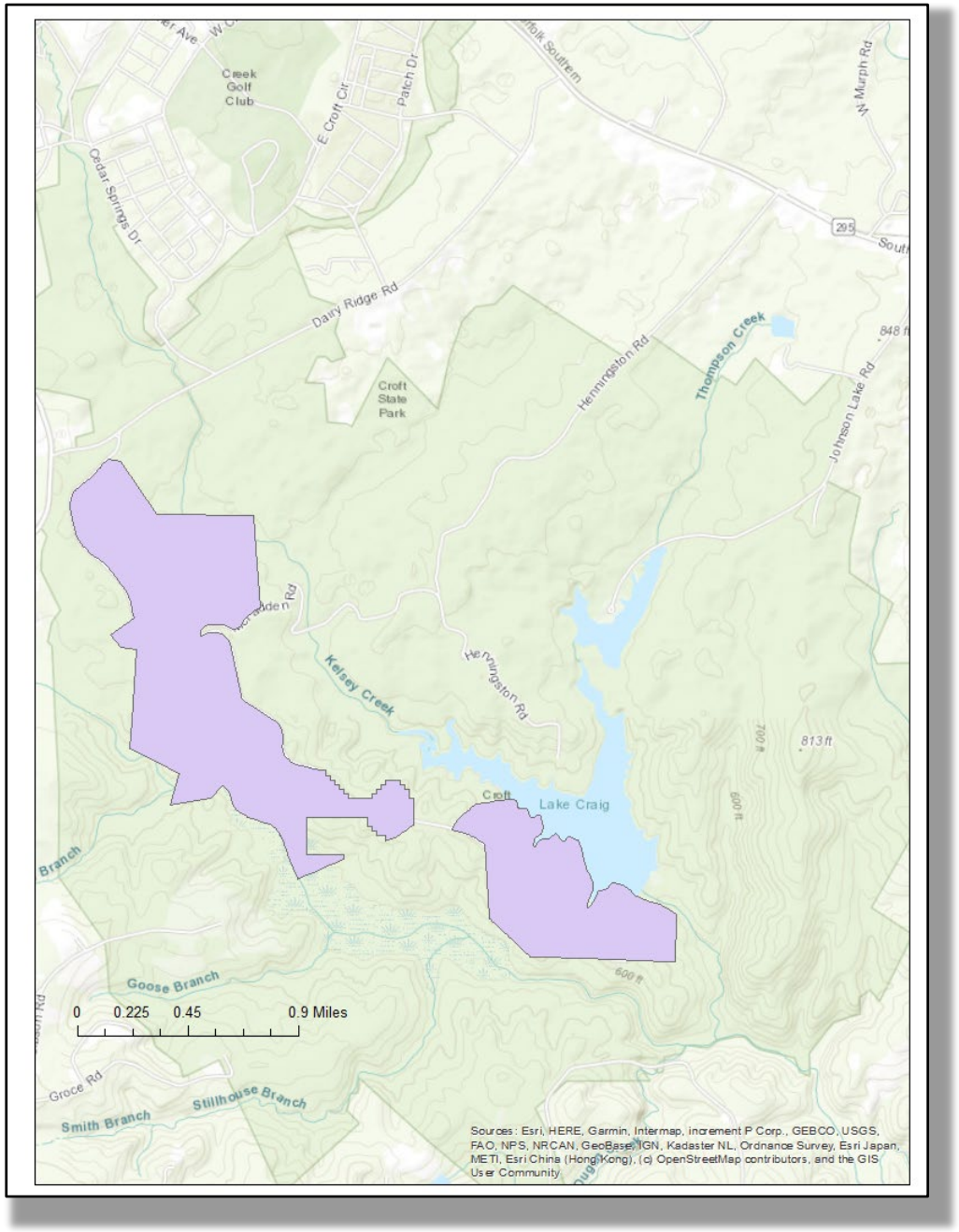
Project Overview

Goal: MEC Remediation

Location: Spartanburg

Stakeholders: Federal

State: SC



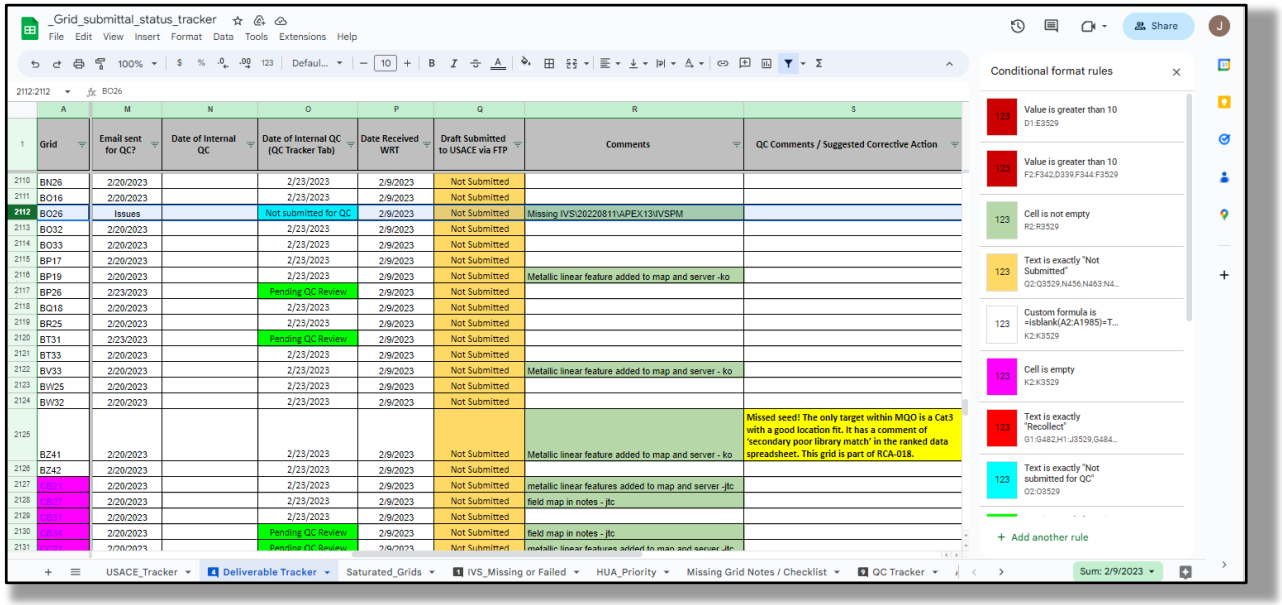
Data Management Tools and Project Execution

Outline Part 1

- Spreadsheet Software: MS Excel and Gsheets
- Relational Database Management
- Relational Database Management (Coverage Data Case Example)
- File Formatting and Schema
- Data Access, Storage and Delivery
- Data Access, Storage and Delivery (Cont.)

Spreadsheet Software: MS Excel and Gsheets

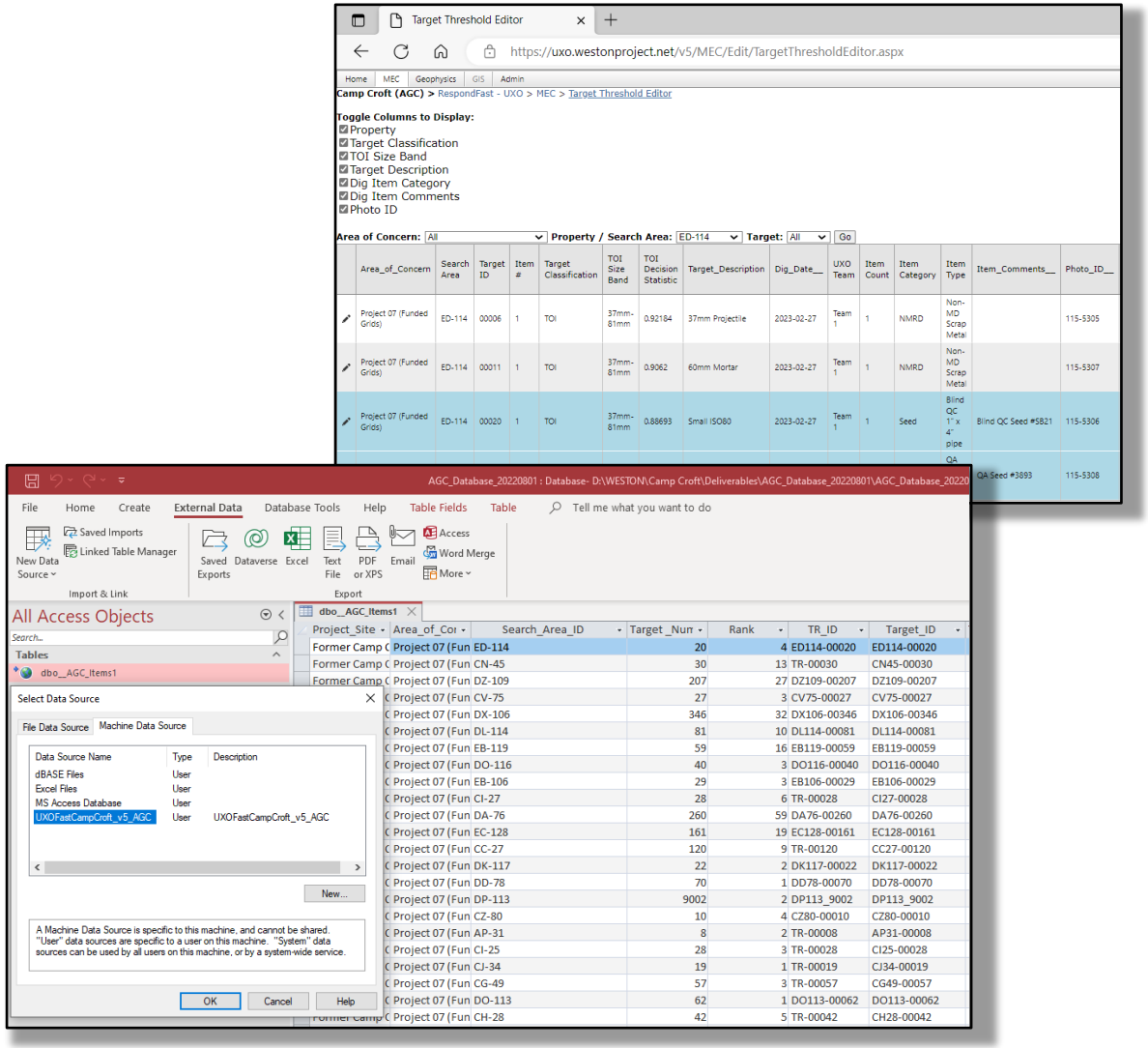
- Trade Offs:
  - Excel functionality (Formulas, Keyboard Interface, Pivot Tables, etc…)
  - Gsheets shareability (Cloud based, Simultaneous Filtering/Editing for multiple users, advanced conditional formatting, etc…)
- Project execution use cases:
  - Deliverable pipeline progress tracking. (Gsheets)
  - Intrusive investigations review process tracking. (Gsheets)
  - Deliverable formatting. (Excel)
  - Data analysis including field work progression forecasting. (Excel)



Grid	Email sent for QC?	Date of Internal QC	Date of Internal QC (QC Tracker Tab)	Date Received WRT	Draft Submitted to USACE via FTP	Comments	QC Comments / Suggested Corrective Action
BN26	2/20/2023		2/23/2023	2/9/2023	Not Submitted		
BO16	2/20/2023		2/23/2023	2/9/2023	Not Submitted		
BO26	Issues		Not submitted for QC	2/9/2023	Not Submitted	Missing IVS\20220811\APEX13\IVSPM	
BO32	2/20/2023		2/23/2023	2/9/2023	Not Submitted		
BO33	2/20/2023		2/23/2023	2/9/2023	Not Submitted		
BP17	2/20/2023		2/23/2023	2/9/2023	Not Submitted		
BP19	2/20/2023		2/23/2023	2/9/2023	Not Submitted	Metallic linear feature added to map and server -ko	
BP26	2/23/2023		Pending QC Review	2/9/2023	Not Submitted		
BQ18	2/20/2023		2/23/2023	2/9/2023	Not Submitted		
BR25	2/20/2023		2/23/2023	2/9/2023	Not Submitted		
BT31	2/23/2023		Pending QC Review	2/9/2023	Not Submitted		
BT33	2/20/2023		2/23/2023	2/9/2023	Not Submitted		
BV33	2/20/2023		2/23/2023	2/9/2023	Not Submitted	Metallic linear feature added to map and server - ko	
BW25	2/20/2023		2/23/2023	2/9/2023	Not Submitted		
BW32	2/20/2023		2/23/2023	2/9/2023	Not Submitted		
BZ41	2/20/2023		2/23/2023	2/9/2023	Not Submitted	Metallic linear feature added to map and server - ko	Missed seed! The only target within MQO is a Cat3 with a good location fit. It has a comment of 'secondary poor library match' in the ranked data spreadsheet. This grid is part of RCA-018.
BZ42	2/20/2023		2/23/2023	2/9/2023	Not Submitted		
CB41	2/20/2023		2/23/2023	2/9/2023	Not Submitted	metallic linear features added to map and server -jtc	
CB47	2/20/2023		2/23/2023	2/9/2023	Not Submitted	field map in notes - jtc	
CB31	2/20/2023		2/23/2023	2/9/2023	Not Submitted		
CB34	2/20/2023		Pending QC Review	2/9/2023	Not Submitted	field map in notes - jtc	

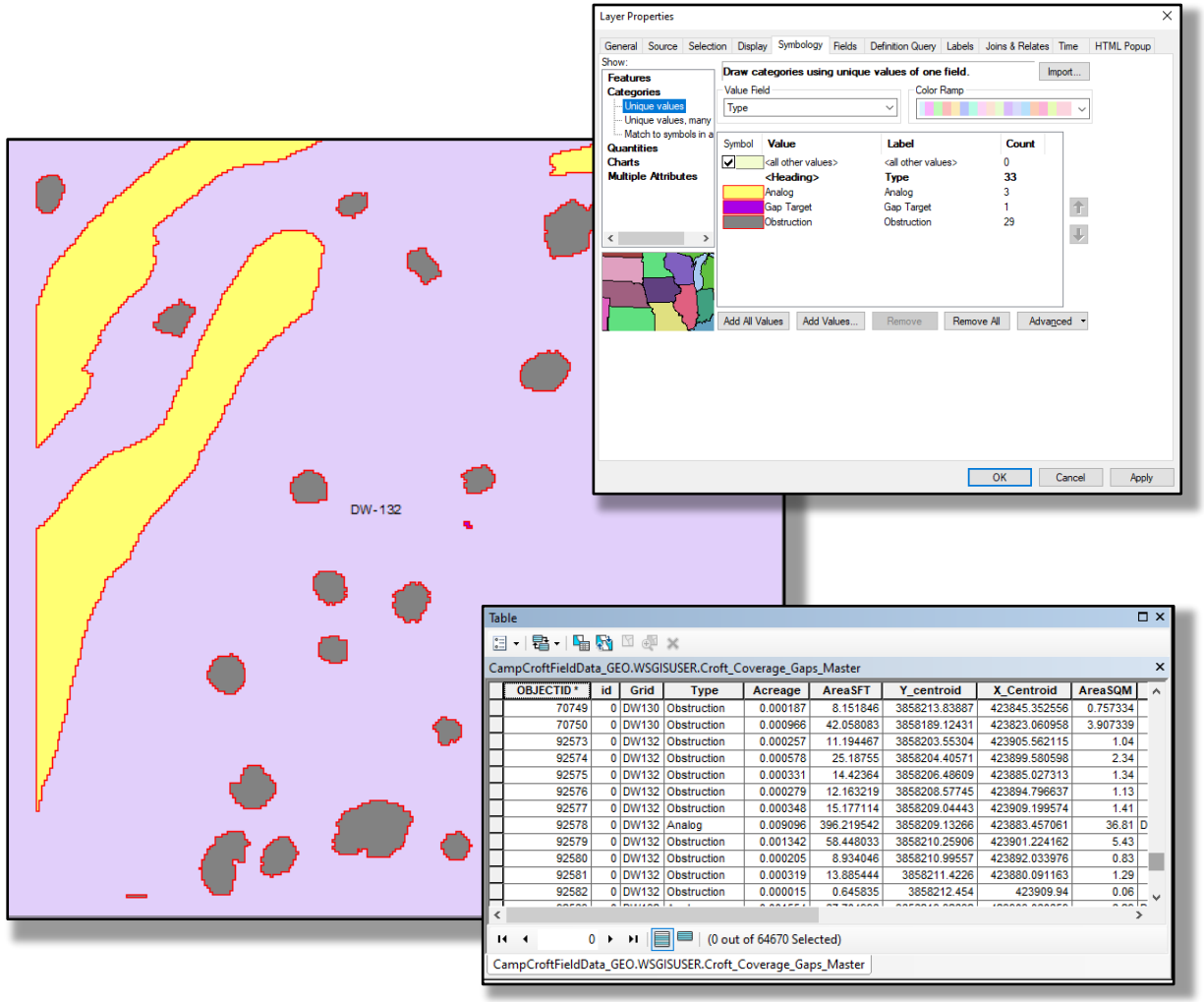
Relational Database Management

- Relational database software enabled with Structure Query Language (SQL) allowed Weston to maintain large data sets and maintain data integrity.
    - Tools:
        - Access database (Target and intrusive result deliverables)
        - ESRI Spatial Database Engine (Master coverage shapes)
    - Advantages:
        - Query creation / Accessibility
        - Remote table linking. (Respond Fast - UXO®, WESTON)



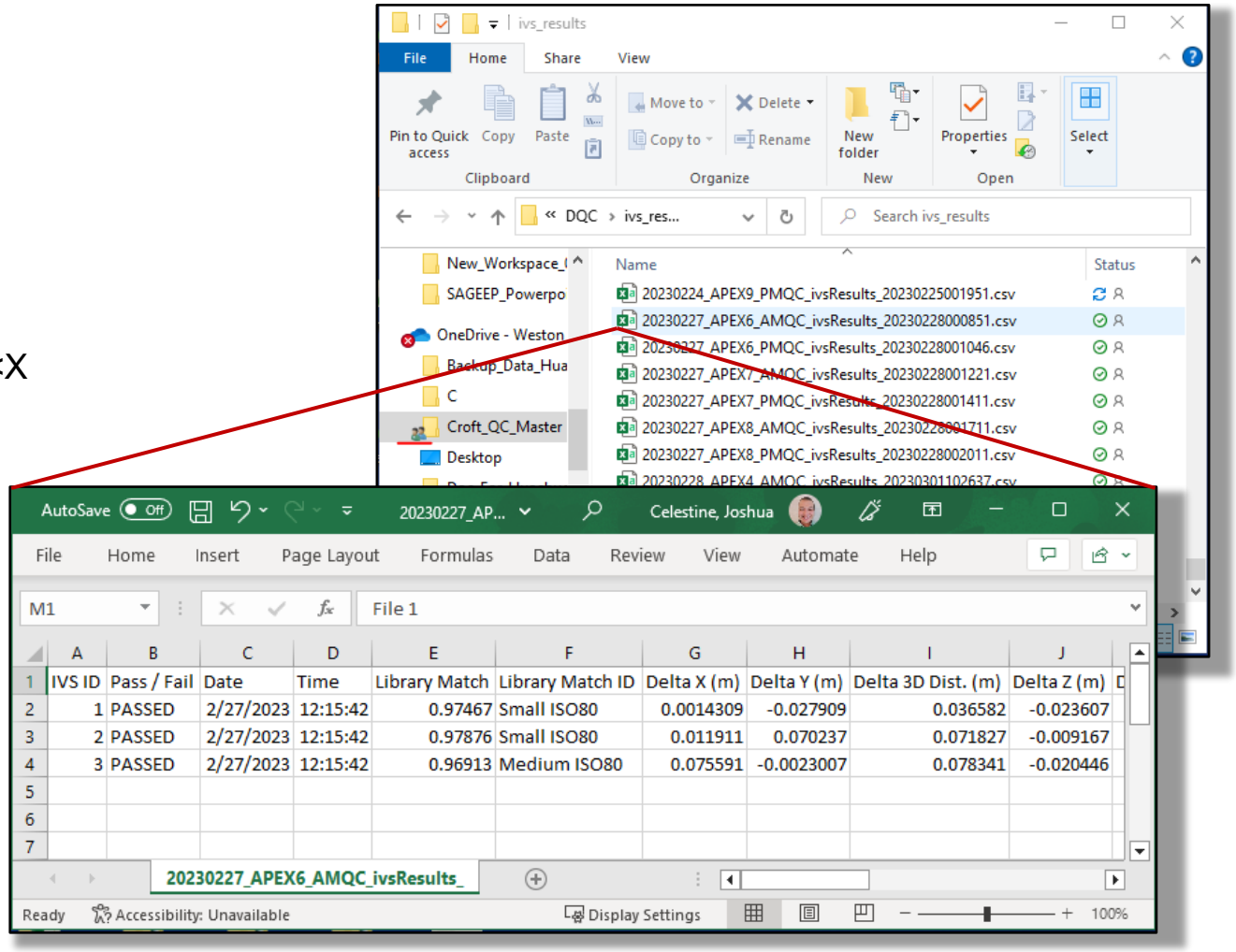
Relational Database Management (Coverage Data Case Example)

- Simpler can be better for data integrity and usability (Coverage data example):
  - Problem:
    - Limitation on automation and data usability by using variable coverage feature descriptions.
  - Solution:
    - 3 feature class “Types” standardization (Analog, Obstruction and Gap Target)



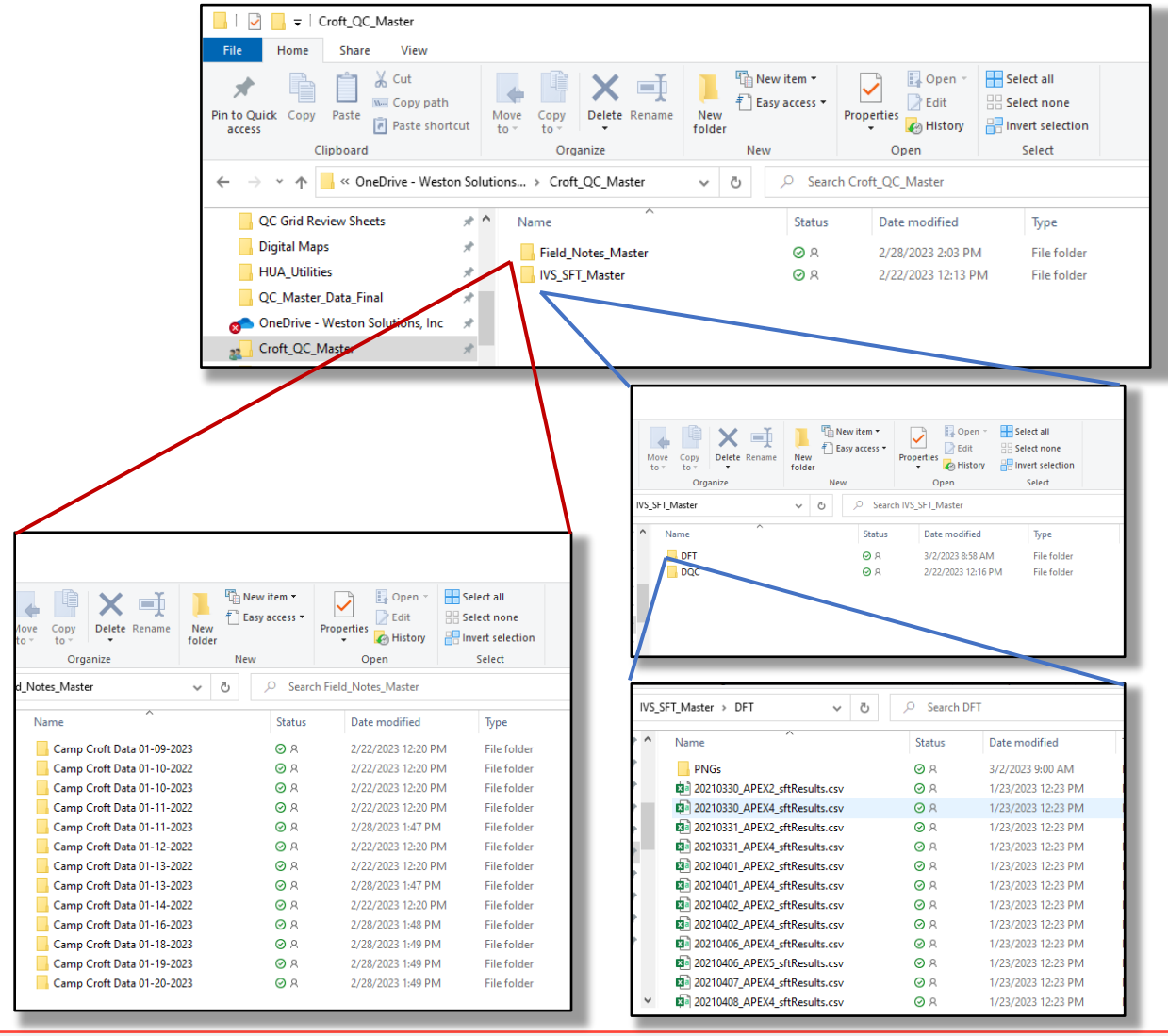
File Formatting and Schema

- File formatting and schema consistency was key for the project's execution.
  - IVS / SFT Result deliverables:
    - Consistent file names. (Example : 20<XXXXXX>_APEX<XX>_AMQC_ivsresults …<XXX>)
    - Consistent file schemas. (Field / Column headers)
    - Consistent file formats were used to maintain data integrity and end user accessibility. (.csv, .xls, .png , .jpg, etc…).



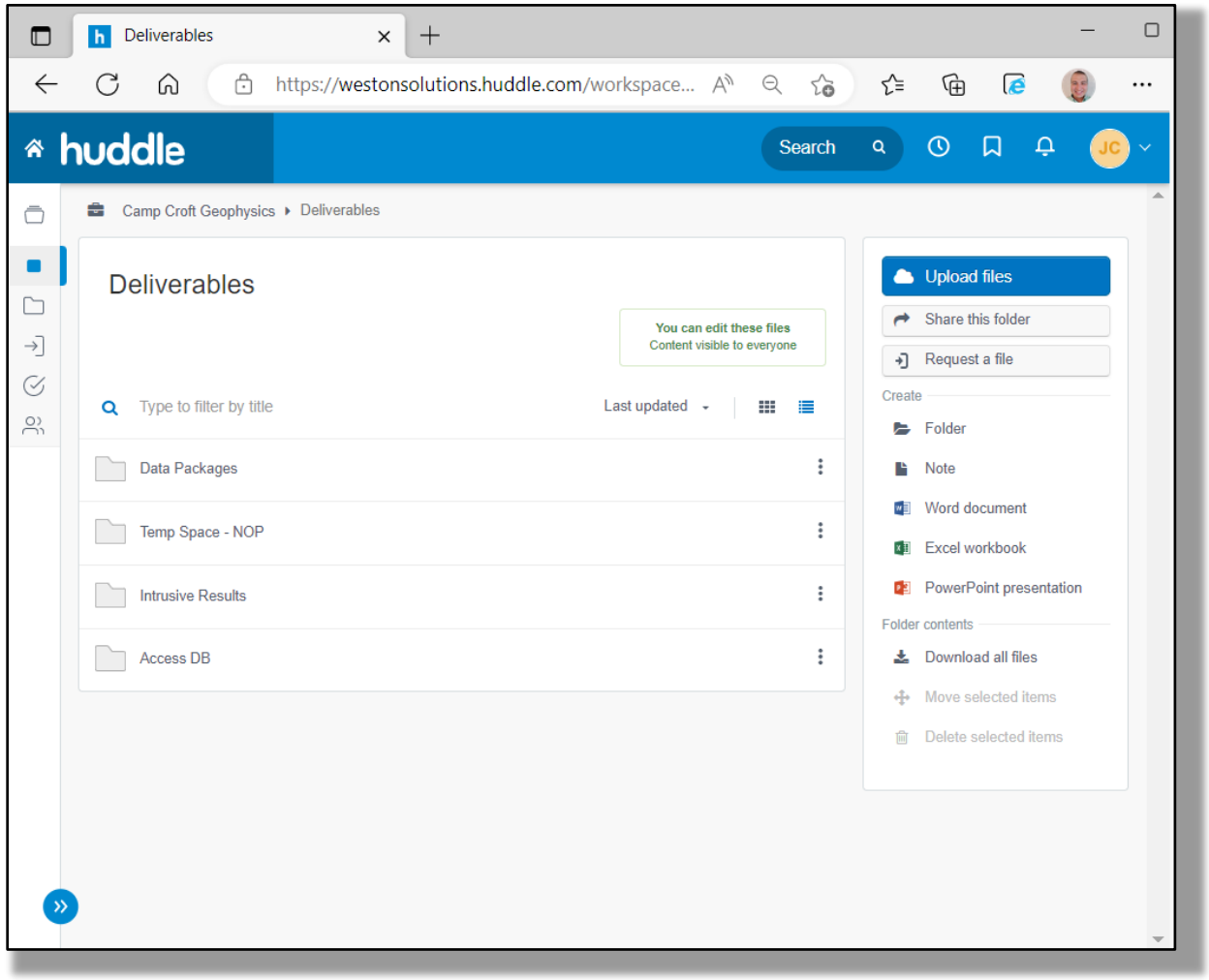
Data Access, Storage and Delivery

- Weston found secure and creative ways to streamline data delivery and access between project stakeholders in a remote work setting.
    - Daily instrument test and field data were stored in a windows file structure within a shared one drive folder.
    - Script runtime was reduced when comparing runtimes with files stored on a Network attached -storage (NAS) device.



Data Access, Storage and Delivery (Cont.)

- Weston found that delivering the data represented challenges as well due to the volume of deliverable materials.
    - Problem:
        - Voluminous data, multiple stake holders (internal/external) and remote work environment.
    - Solution:
        - Huddle File Transfer Protocol (FTP) Site with shared access.



Automation with Python

Outline Part 2

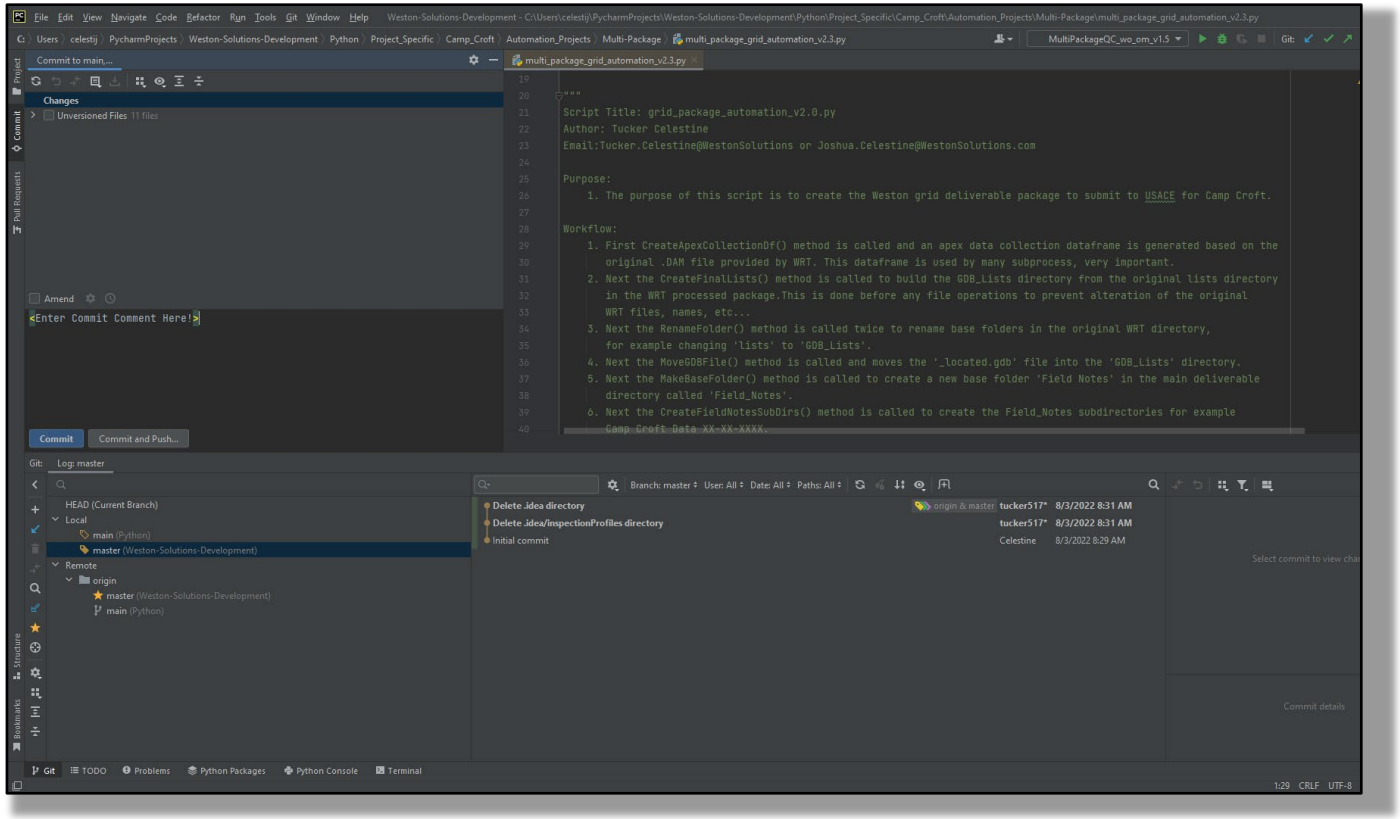
- Python development process overview.
- What is an API?
- Case example (Grid deliverable preparation)
- Case example (Geosoft map automation).
- Case example (Dig sheet generator)
- Case example (Polygon to shape conversion)
- Conclusions

```
    # Apply text wrapping to QC Comments field 11/11/2022 - jtc
    for row in ws[5:ws.max_row]:
        cell = row[7]
        cell.alignment = openpyxl.styles.Alignment(horizontal='center', vertical='ce

    # Set row 2 height so (cm) is visible
    ws.row_dimensions[5].height = 30

    # Apply number formatting to number easting and northing
    for row in ws.iter_rows(min_row=5, min_col=3, max_row=ws.max_row, max_col=4):
        for cell in row:
            cell.number_format = '####0.000'

    # Apply number formatting to depths
    for row in ws.iter_rows(min_row=5, min_col=6, max_row=ws.max_row, max_col=6):
        for cell in row:
            cell.number_format = '##0.0'

    # Add footer with 1 of ...
    ws.HeaderFooter.differentFirst = True
    ws.HeaderFooter.firstFooter.center.text = '1 of 1'

    # Get range start for moving cells
    range_start = ws.max_row + 1

    for k, v in unique_analog_dfs.items():
        if k == grid_name:
            for r in dataframe_to_rows(v, index=False, header=True):
                ws.append(r)
                pass

            # Get the final range for moving cells down
            range_final = ws.max_row
            range_string_start = 'A{}:'.format(range_start)
            # Change range to include new field, 11/11/2022 - jtc
            range_string_end = 'H{}'.format(range_final)
            move_range = range_string_start + range_string_end
```

Python development process overview.

- Tools:
  - Anaconda / Python (Virtual Environments)
  - Pycharm (IDE)
  - GitHub (VCS)

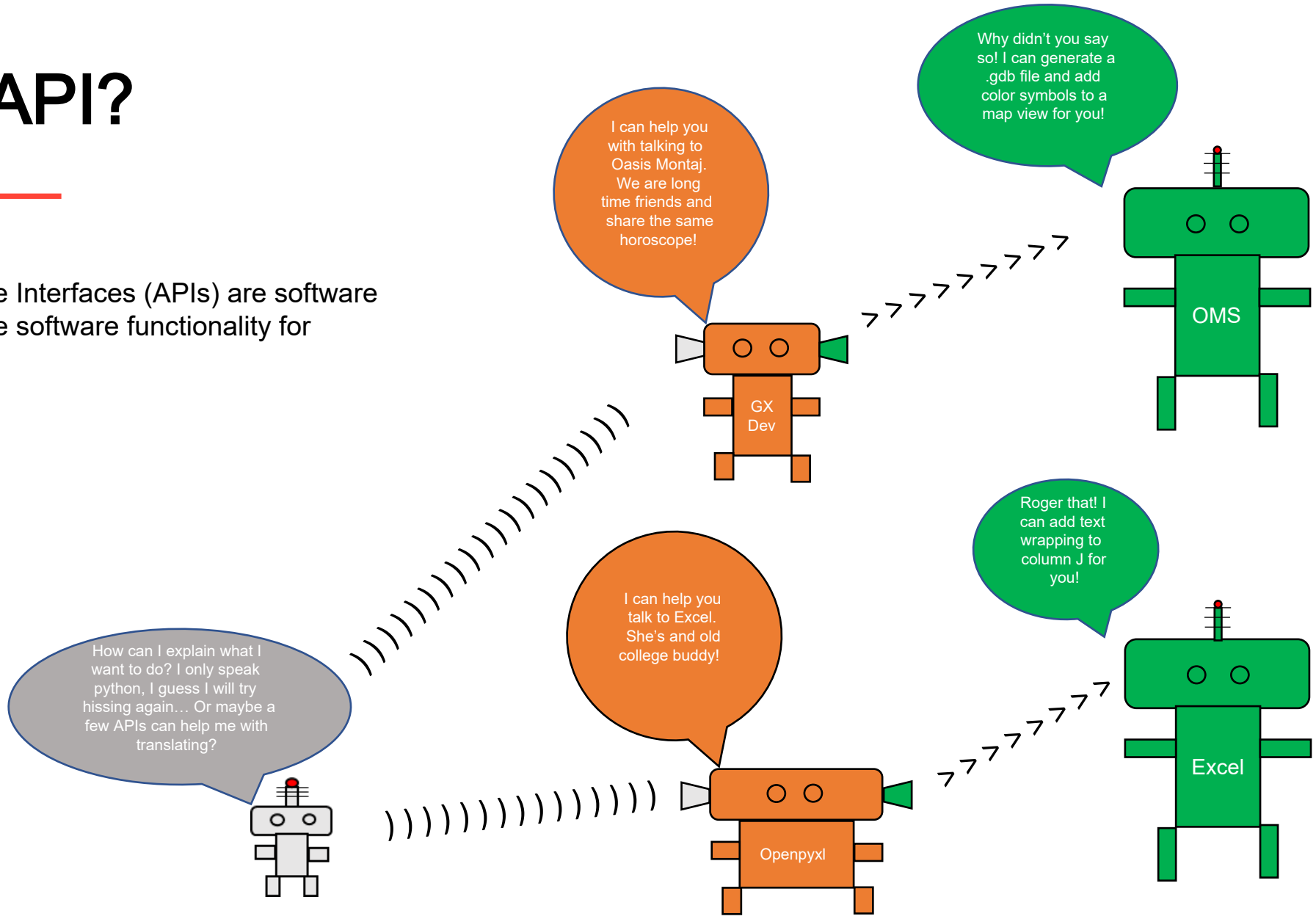
- Advantages:
  - Dependency requirement tradeoffs. (Anaconda/Python)
  - Additional functionality. (Anaconda)
  - Python Enhancement Proposal (PEP) suggestions. (IDE)
  - Streamlined version tracking and repository syncing. (VCS)



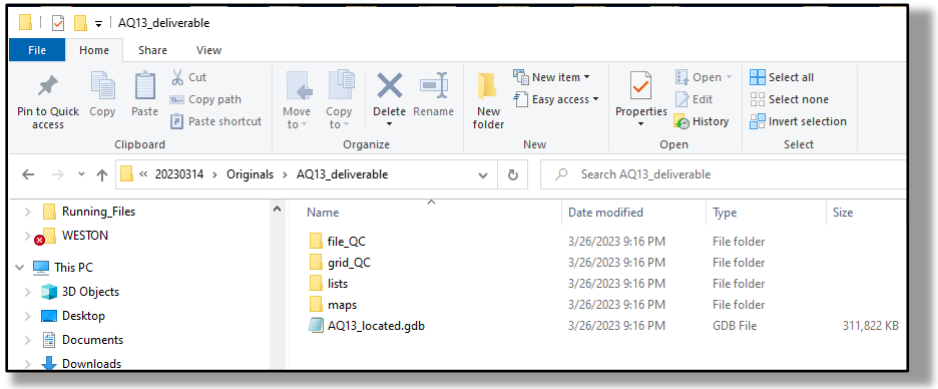
What is an API?

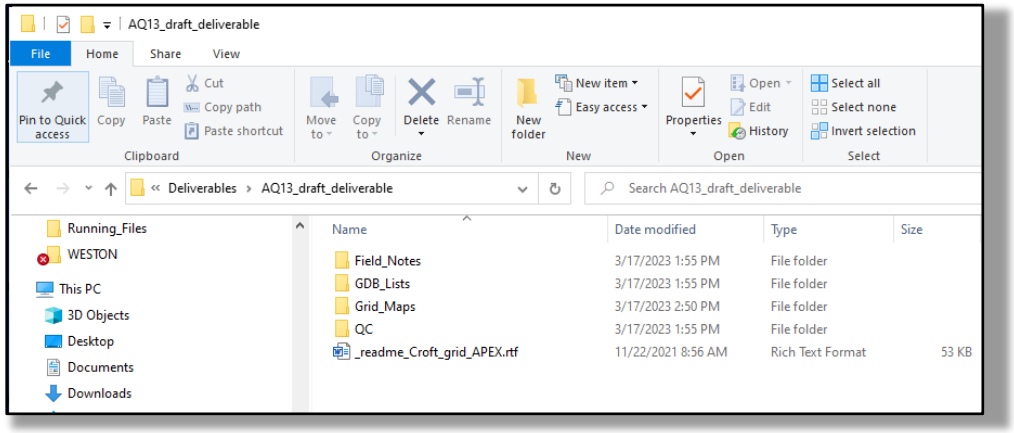
- Application Programmable Interfaces (APIs) are software intermediaries that expose software functionality for developers to utilize.

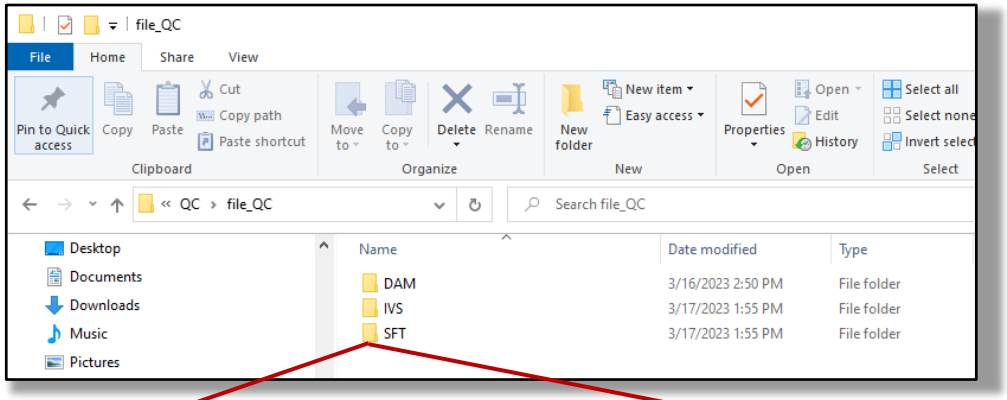
- Types of API's:
  - Python
  - Java
  - C++

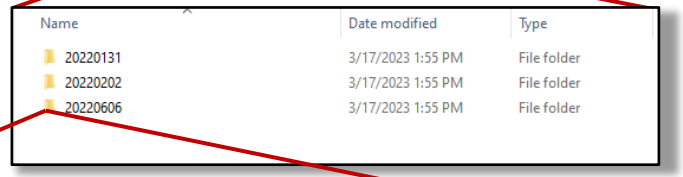


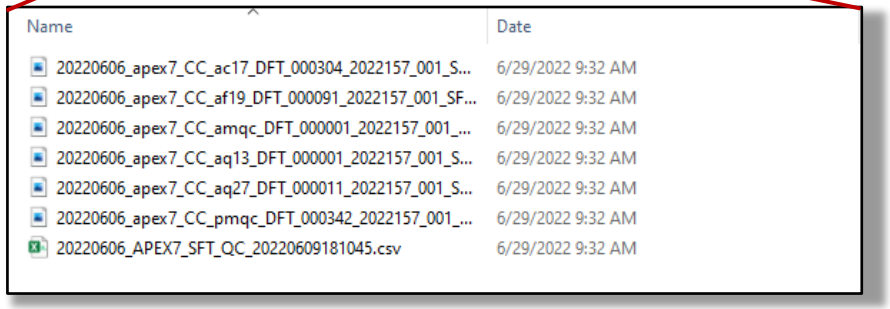
Case example (Grid deliverable preparation)





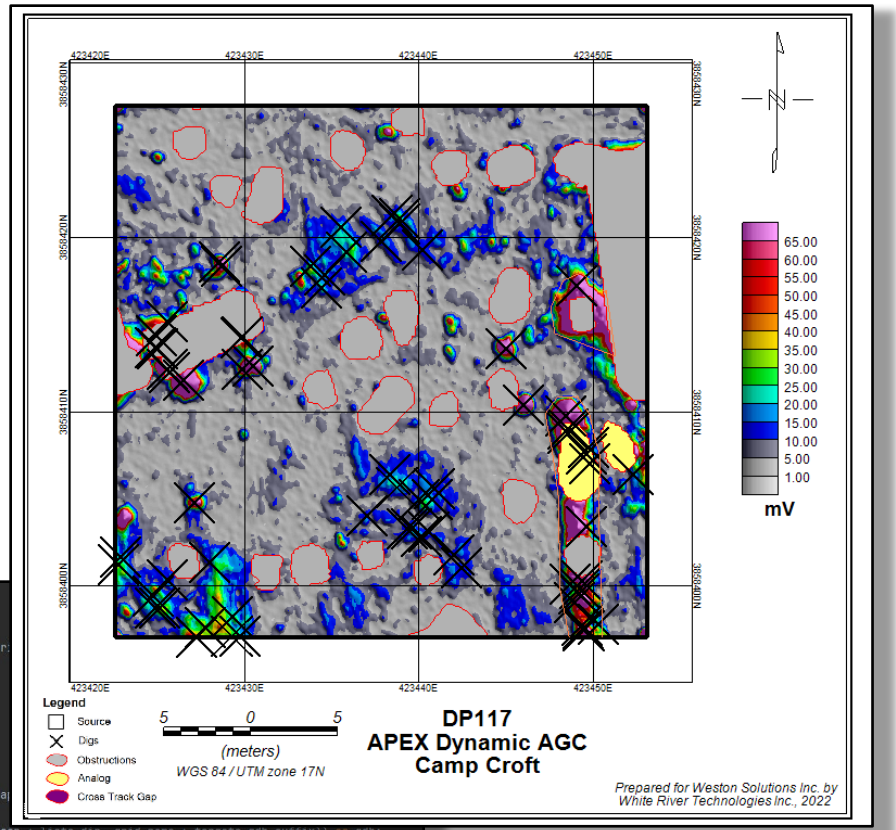






Case example (Geosoft map automation)

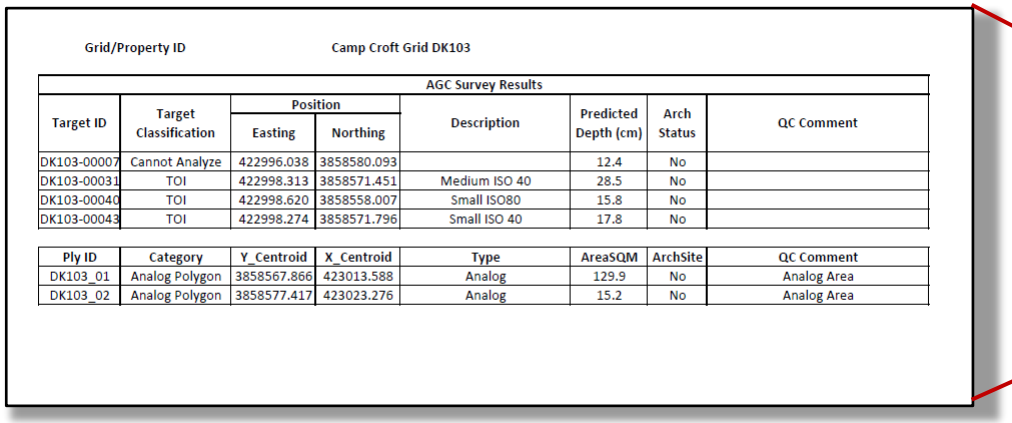
- Problem:
  - Automate generation of supplemental targets Geosoft database and plot on map view.
  - Import ESRI layer file into map view.
  - Import coverage shape legend.
- Solution:
  - Weston utilized Geosoft's GX Developer to generate functions within our pre -existing automation.
  - Functional or modular programming was key here in the larger scheme of the automation.




```
def PlotTargetsGDB():
    global map_file
    global map_folder
    map_file = os.path.join(cwd + os.sep + 'Grid_Maps', gr
    map_folder = os.path.join(cwd, 'Grid_Maps')
    cmap_zn = gxgroup.Color_map(color_zn_file)
    plotting_grp_name = grid_name + '_Digs'
    error = gxapi.str_ref("Problem unpacking map!")

    with gxmap.Map.open(map_file) as gmap:
        # Unpack the map files
        gmap.gxmap.un_pack_files_to_folder(force=1, dir=ma

        with gxdb.Geosoft_gdb.open(os.path.join(cwd + os.sep + lists_dir, grid_name + targets_gdb_suffix)) as gdb:
            # defines data to store in numpy array from x,y,z channels to plot
            npd, ch, fid = gdb.read_line('Digs', channels=['UTM_E_m_', 'UTM_N_m_', 'Depth_m_'])
            # Open the view to put the new color symbols group in.
            with gxview.View.open(gmap, 'data') as v:
                gxgroup.Color_symbols_group.new(v, plotting_grp_name, data=npd, color_map=cmap_zn, symbol=3)
                # Open the new color symbols group to change the scaling of the x's
                with gxgroup.Color_symbols_group.open(v, plotting_grp_name) as cg:
                    cg._gxcsymb.set_scale(2.24)
                cg.close()
            v.close()
        gdb.close()
    gmap.close()
```

Case example (Dig sheet generator)

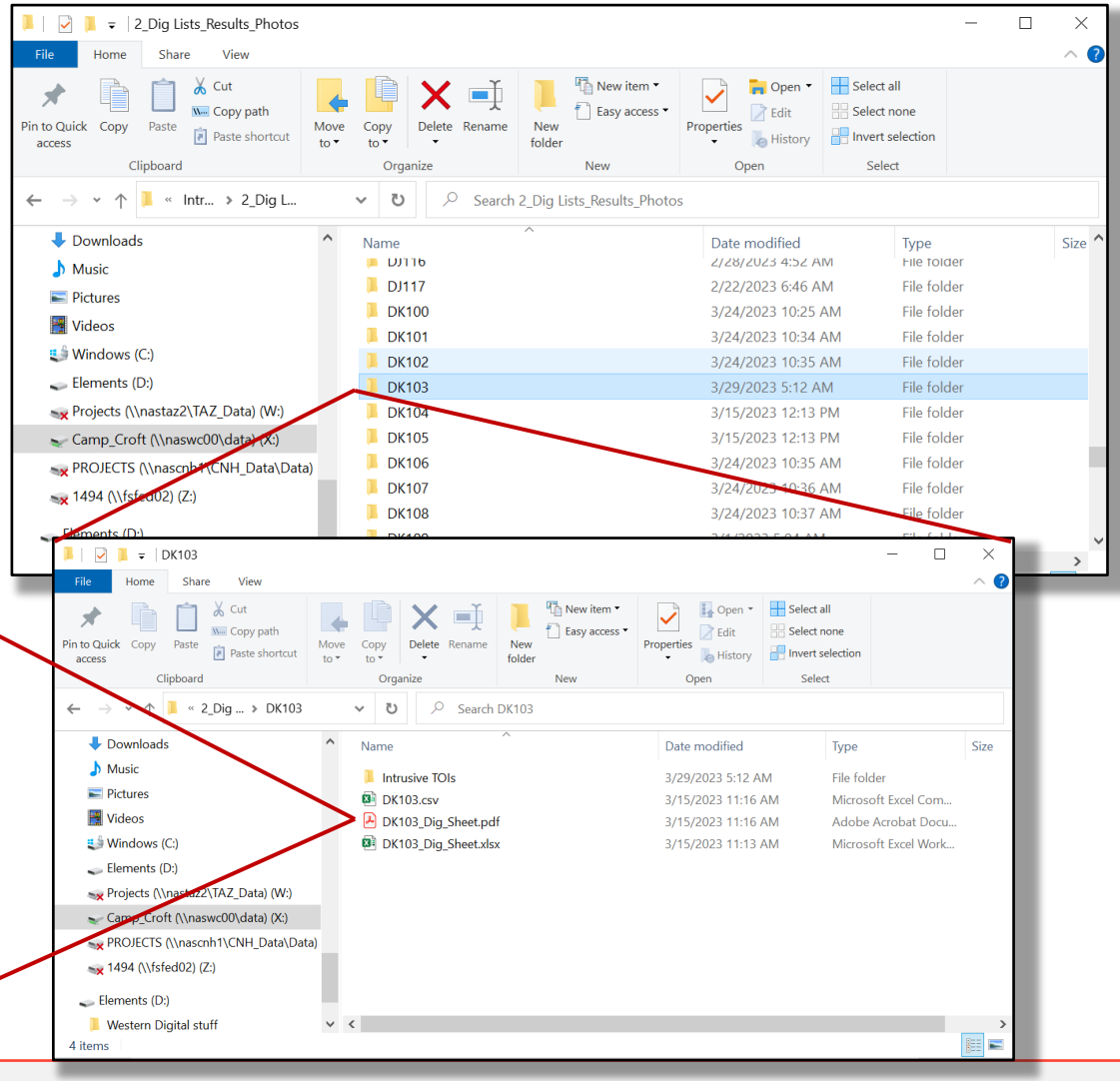
- Problem:
  - Weston needed an automated solution to create dig sheets for our intrusive teams.
- Solution:
  - Multi -functional python automation using pandas with Openpyxl.



Grid/Property ID **Camp Croft Grid DK103**

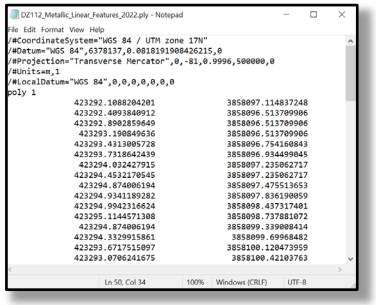
AGC Survey Results							
Target ID	Target Classification	Position		Description	Predicted Depth (cm)	Arch Status	QC Comment
		Easting	Northing				
DK103-00007	Cannot Analyze	422996.038	3858580.093		12.4	No	
DK103-00031	TOI	422998.313	3858571.451	Medium ISO 40	28.5	No	
DK103-00040	TOI	422998.620	3858558.007	Small ISO80	15.8	No	
DK103-00043	TOI	422998.274	3858571.796	Small ISO 40	17.8	No	

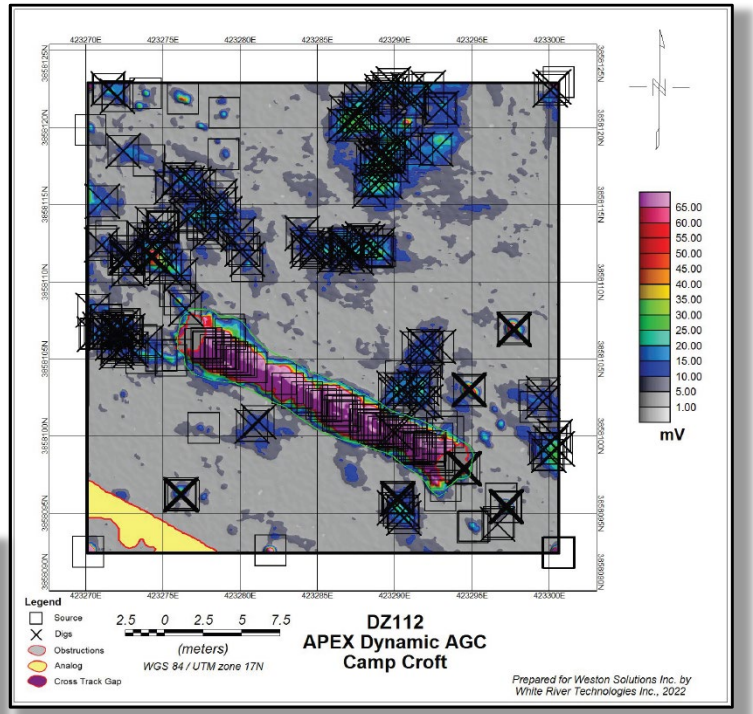
Ply ID	Category	Y_Centroid	X_Centroid	Type	AreaSQM	ArchSite	QC Comment
DK103_01	Analog Polygon	3858567.866	423013.588	Analog	129.9	No	Analog Area
DK103_02	Analog Polygon	3858577.417	423023.276	Analog	15.2	No	Analog Area

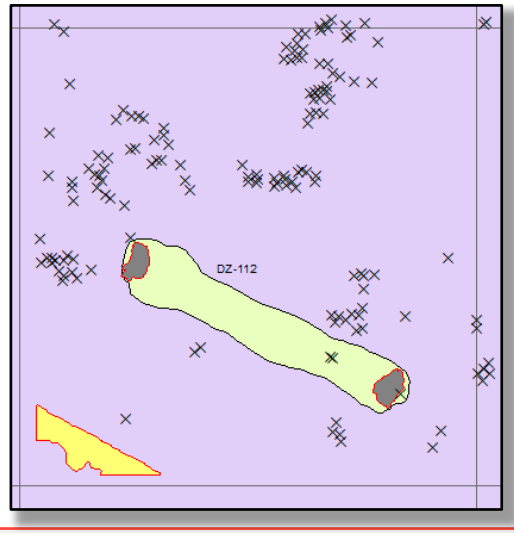


Case example (Polygon to shape conversion)

- Problem:
  - Identifying targets that intersect metallic linear features located within high use areas of the site.
- Solution:
  - Convert Geosoft polygon files to cumulative polygon feature class in order to identify target intersections in ArcMap.

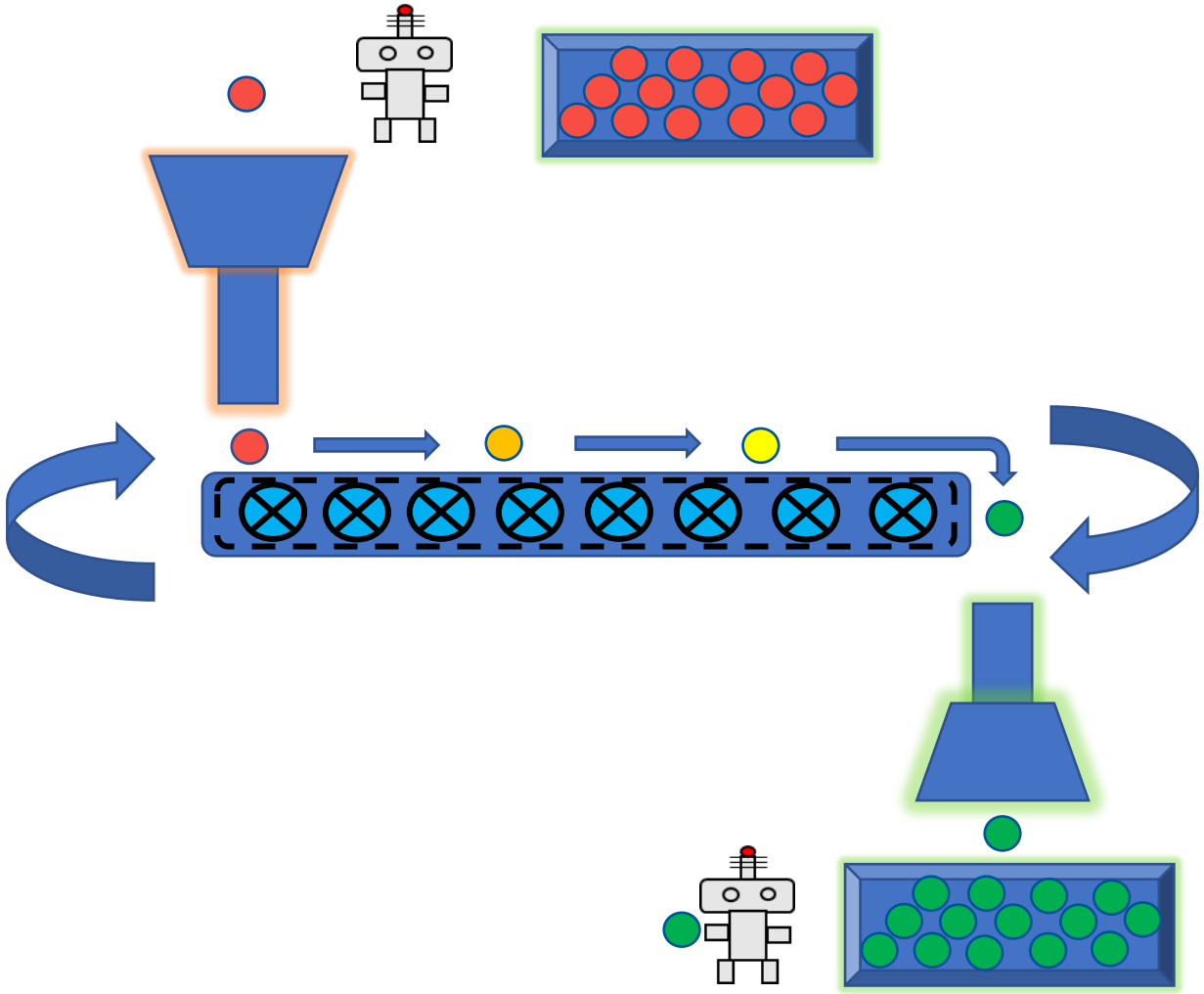






Conclusions

- Python - A vital piece of the puzzle…
  - Data quality and integrity are of paramount importance.
  - Automation can improve data quality and maintain data integrity.
  - Python is great for scripting.
  - There are many python APIs out there and we can use them in many ways.



Questions & Answers

65+ Years of Environmental and Infrastructure Excellence
WESTON SOLUTIONS, INC.

Want to Know More?
Contact Us

Joshua Tucker Celestine
Associate Geoscientist 2

337-202-9370
Joshua.Celestine@WestonSolutions.com

JR Candlish
Senior Analyst (GIS/Data)

865-405-9149
J.Candlish@WestonSolutions.com

65+ Years of Environmental and Infrastructure Excellence
WESTON SOLUTIONS, INC.