Python: Introduction to Map Automation

Jeff Barrette
What is `arcpy.mapping` or `arcpy.mp`?

- Python mapping module that is part of the ArcPy site-package
- An API that allows users to:
  - manage MXDs/Projects, layer files, and their contents
    - find a layer with data source X and replace with Y
    - update a layer’s symbology in many documents/projects
    - generate reports that lists document information
      - data sources, broken layers, spatial reference info, etc.
  - Automate the exporting and printing of layouts
  - Automate map production and create map books
    - extend Data Driven Pages / Map Series capabilities
Tour of arcpy.mapping / arcpy.mp

- Overview
Help

Code sample

**ArcGISProject example 1**

The following script demonstrates how to import documents into an existing ArcGIS Pro project. It also sets up some of the default project settings and saves the result to a new file.

```python
import arcpy
aprx = arcpy.mp.ArcGISProject(r"C:\Projects\Yosemite\Documents\Yosemite.aprx")
aprx.ImportDocument(r"C:\Projects\Yosemite\Documents\Yosemite.xmd")
aprx.ImportDocument(r"C:\Projects\Yosemite\Documents\Yosemite_ScenicViews.docx")
aprx.defaultToolbox = r"C:\Projects\Yosemite\Data_\Vector\YosemiteData.gdb"
aprx.defaultLayers = r"C:\Projects\Yosemite\Analysis\AnalysisTools.tmx"
aprx.saveAsCopy(r"C:\Projects\Yosemite\Documents\Yosemite.aprx")
```

**ArcGISProject example 2**

The following script uses the keyword current, so it should be run from the Python window. The script prints the name of each map in a project and its layers, as well as the name of each layout and its page size.

```python
aprx = arcpy.mp.ArcGISProject("CURRENT")
for m in aprx.listMaps():
    print("Map: " + m.name)
    for lyr in m.listLayers():
        print("   " + lyr.name)
    print "Layouts:";
    for lyr in aprx.listLayouts():
        print("   \{0}\((1) x (2)\)".format(lyr.name, lyr.pageHeight, lyr.pageWidth, lyr.pageUnits))
```
Python window

- Quick and easy access to Python and arcpy
  - Great way for new users to learn Python
  - Intellisense for all tools (methods and properties), parameter syntax
  - Quickly and efficiently test and execute tools
  - ArcPy module is already imported and fully accessible
You can get there from here

• The starting points are:
  - ArcGISProject
  - LayerFile

• The end result might be:
  - A modified, saved project
  - An exported PDF
Referencing projects

ArcGISProject function

ArcGISProject(aprx_path)

ArcGISProject class

Methods
importDocument
listBrokenDataSources
save
updateConnectionProperties
...

Properties:
defaultGeodatabase
filePath
homeFolder
...
Referencing projects

• Use the **ArcGISProject** function
  - Takes a path to an aprx file on disk or special keyword "CURRENT"

• Reference project on disk (stand-alone script)
  aprx = arcpy.mp.ArcGISProject(r"C:\path\some.aprx")

• Reference project in current application session (in process)
  aprx = arcpy.mp.ArcGISProject("CURRENT")
Where to run your scripts

- Python window or script tool (in process)
- IDE (e.g., IDLE), command line (out of process)
ArcGISProject
Maps and Layers

• Use “List” functions on appropriate objects
  
  ```python
  mp = aprx.listMaps("parc*")[0]
  lyr = mp.listLayers()[0]
  ```
  
  - Warning - it is easy to forget to use [0]

• Common properties are available
  
  - Layers:
    - definitionQuery, saveACopy, visible
  
  - Map:
    - AddLayer, ClearSelection, defaultCamera
Layouts and layout elements

• Support zero to many
  - `lyt = aprx.listLayouts("Landscape")[0]

• Find elements on a layout
  - `txtElm = lyt.listElements("text_element", "title")[0]
  - `txtElm.text = "Some new title"

• Each element has a unique name
Map Frame and Camera

- **Reference map frame on a layout**
  
  \[-mf = lyt.listElements(“mapframe_element”)\[0]\]

- **Zoom to Bookmark / Layer**
  
  \[-bkmk = mp.listBookmarks(“bookmark”)\[0]\]
  
  \[-mf.zoomToBookmark(bkmk)\]
  
  \[-lyrExt = mf.getLayerExtent(lyr, True)\]
  
  \[-mf.camera.setExtent(lyrExt)\]

- **Modify camera**
  
  \[-cam = mf.camera\]
  
  \[-cam.X = x; cam.Y = y; cam.scale = sc\]
  
  \[-mf.camera = cam\]
Layout elements
Managing PDF documents

- Can create or open existing PDFs
  - `myPDF = arcpy.mp.PDFDocumentCreate(r"C:\Temp\new.pdf")`
  - `myPDF.appendPages(r"C:\Temp\someOther.pdf")`
  - `myPDF.saveAndClose()`
PDF Export
Please Take Our Survey on the Esri Events App!

Download the Esri Events app and find your event

Select the session you attended

Scroll down to find the survey

Complete Answers and Select “Submit”