



# Python Map Automation – Introduction to arcpy.mapping

Michael Grossman

Jeff Barrette

Technical Workshop

# What is arcpy.mapping?

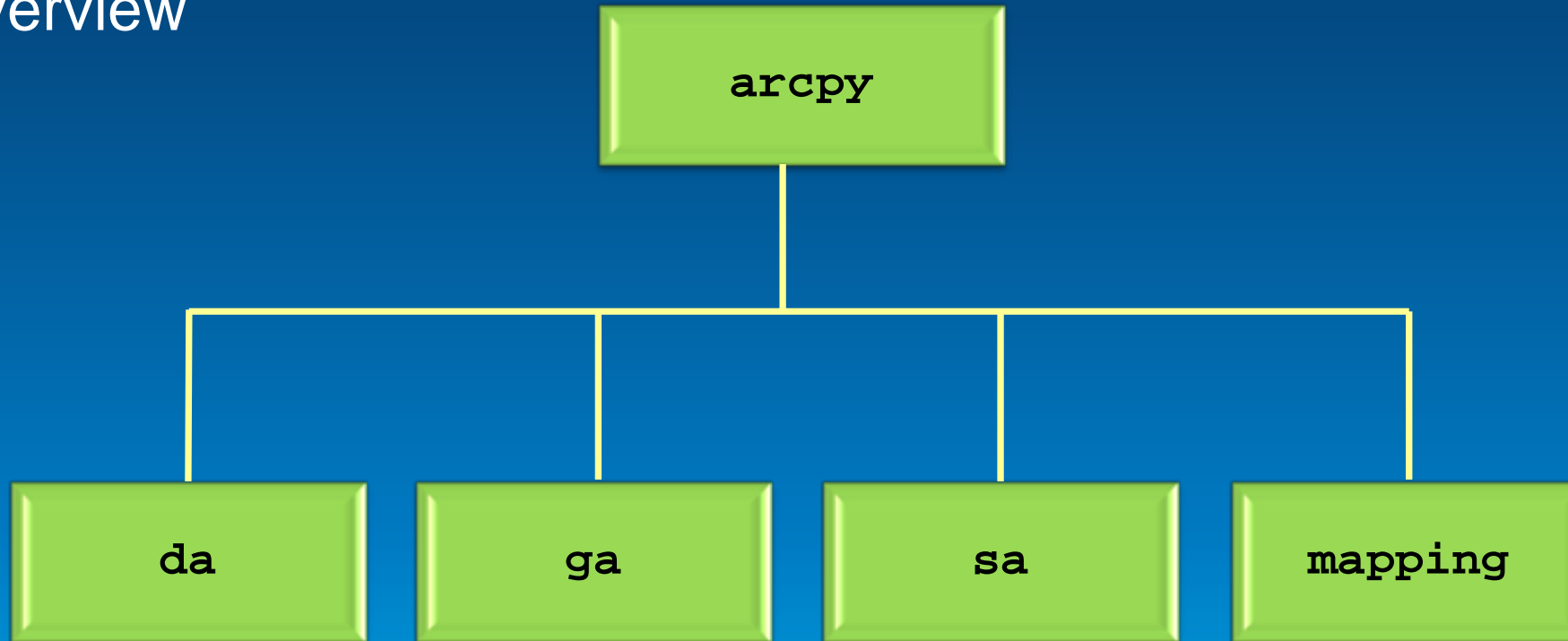
- Python mapping module that is part of the ArcPy site-package
- An API that allows users to:
  - manage map documents, layer files, and their contents
    - find a layer with data source X and replace with Y
    - update a layer's symbology in many MXDs
    - generate reports that lists document information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

# Who is arcpy.mapping for? Why was it built?

- An easy to use, productive scripting environment for the GIS Analyst
  - courser grained object model
  - not a complete replacement for ArcObjects
- An environment to use for basic map/layer management and map automation tasks
- A simple way to publish mapping tasks to the server environment
  - arcpy.mapping scripts can be easily published as geoprocessing tools

# Tour of arcpy.mapping

## Overview



# Functions provide access to classes

## **FUNCTIONS**

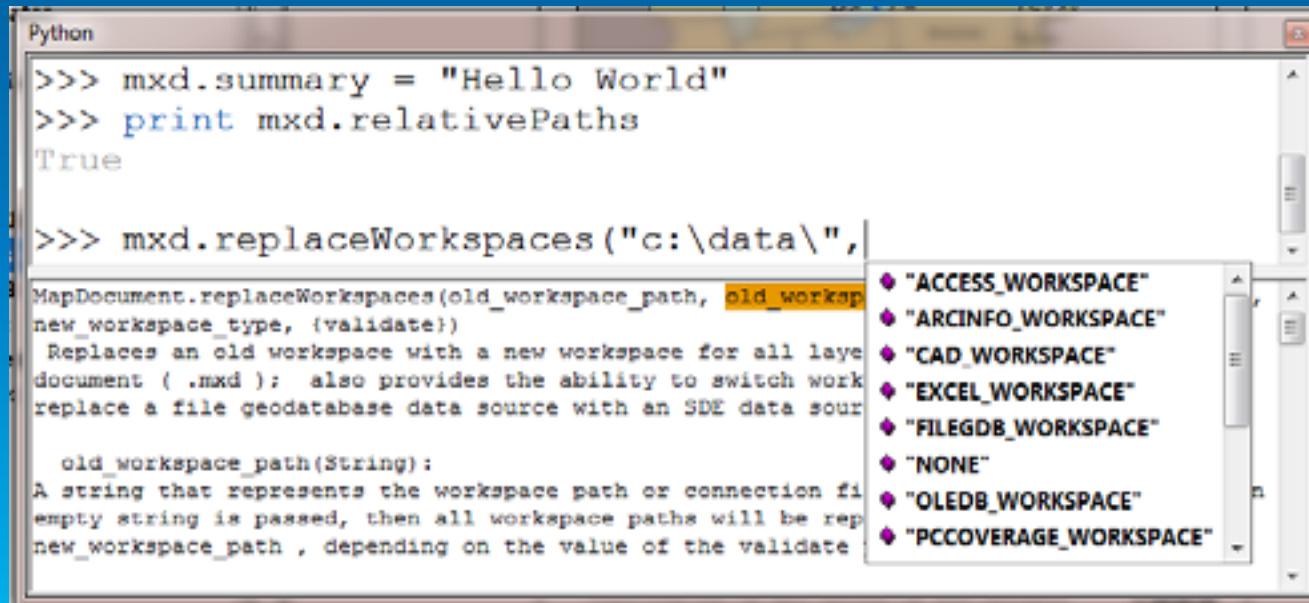
AddLayer  
AddLayerToGroup  
CreateMapSDDraft  
InsertLayer  
Layer  
ListBrokenDataSources  
ListDataFrames  
ListLayers  
ListLayoutElements  
ListTableViews  
MapDocument  
MoveLayer  
RemoveLayer  
UpdateLayer  
...

## **CLASSES**

DataFrame/Time  
GraduatedColorSym  
GraphicElement  
LabelClass  
Layer/Time  
LegendElement  
MapDocument  
MapSurroundElement  
PictureElement  
RasterClassified  
StyleItems  
TableView  
TextElement  
UniqueValueSym  
...

# Python window

- Quick and easy access to Python and arcpy
  - Gateway for new users to learn Python
  - Intellisense for all tools, methods and properties & help window
  - Quickly and efficiently execute tools



The screenshot shows a Python window with the following code and output:

```
>>> mxd.summary = "Hello World"
>>> print mxd.relativePaths
True

>>> mxd.replaceWorkspaces("c:\data\",
```

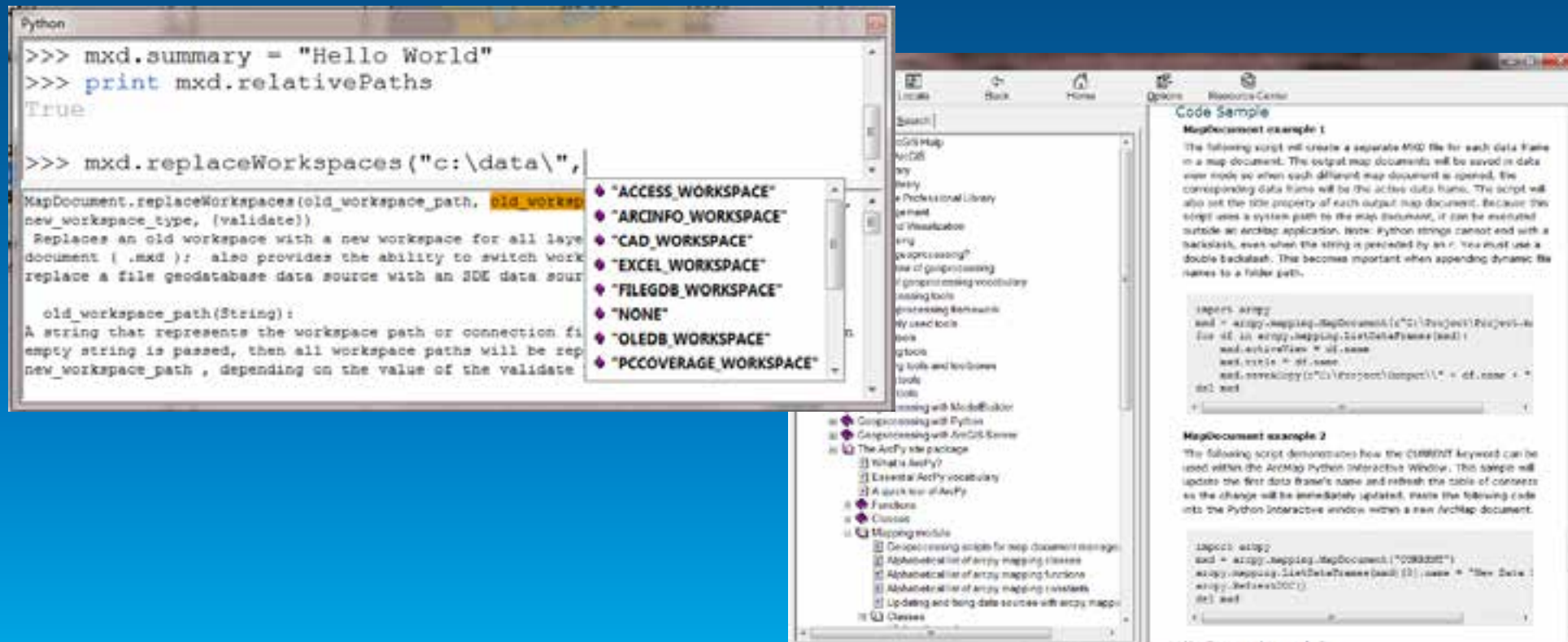
The dropdown menu lists the following workspace types:

- ◆ "ACCESS\_WORKSPACE"
- ◆ "ARCINFO\_WORKSPACE"
- ◆ "CAD\_WORKSPACE"
- ◆ "EXCEL\_WORKSPACE"
- ◆ "FILEGDB\_WORKSPACE"
- ◆ "NONE"
- ◆ "OLEDB\_WORKSPACE"
- ◆ "PCCOVERAGE\_WORKSPACE"



# Demonstration

- The Python Window and using the Desktop Help System





# Referencing map documents

## MapDocument function

```
MapDocument (mxd_path)
```

## MapDocument class

### Methods

```
replaceWorkspaces
```

```
save
```

```
saveAsCopy
```

```
...
```

### Properties:

```
activeDataFrame
```

```
author
```

```
credits
```

```
relativePaths
```

```
...
```

# Referencing map documents (MXDs)

- Opening Map Documents (MXD) with `arcpy.mapping`
- Use the `MapDocument` function
  - Takes a path to MXD file on disk or special keyword "CURRENT"

- Reference map on disk

```
mxd = arcpy.mapping.MapDocument(r"C:\some.mxd")
```

- Get map from current ArcMap session

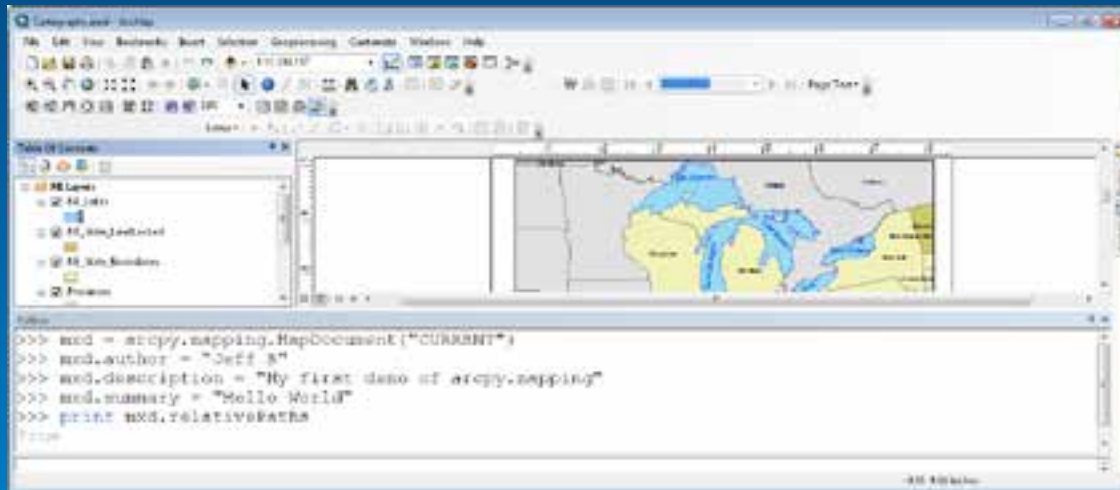
```
mxd = arcpy.mapping.MapDocument("CURRENT")
```

# Referencing map documents (MXDs), cont.

- When using CURRENT
  - Always run in foreground (checkbox in script tool properties)
  - May need to refresh the application  
`arcpy.RefreshActiveView()`
- Limitations and pre-authoring
  - No "New Map" function, so keep an empty MXD available
  - Can't create new objects (e.g., north arrow, data frame)

# Demonstration

- Working with Map Documents (MXDs)



- Use Python Window to change map document property info
- Evaluate relative paths, last saved, etc.
- Change the active view
- Save changes out to a new file

# Layers and data frames

- The “List” functions

- ListLayers and ListDataFrames
- Watch the list indexes (it is easy to forget to use [0])

```
df = arcpy.mapping.ListDataFrames(MXD)[0]
```

- Layer properties

- Common properties are available (e.g., def query, visible)
- All properties can be updated via layer (.lyr) files

- DataFrame properties and methods

- Basic map navigation and settings

# Layers and Data Frames

## Layer functions

**Layer**  
**ListLayers**  
**ListTableViews**

**AddLayer**  
**AddLayerToGroup**  
**InsertLayer**  
**MoveLayer**  
**RemoveLayer**  
**UpdateLayer**

...

## Data Frame Class

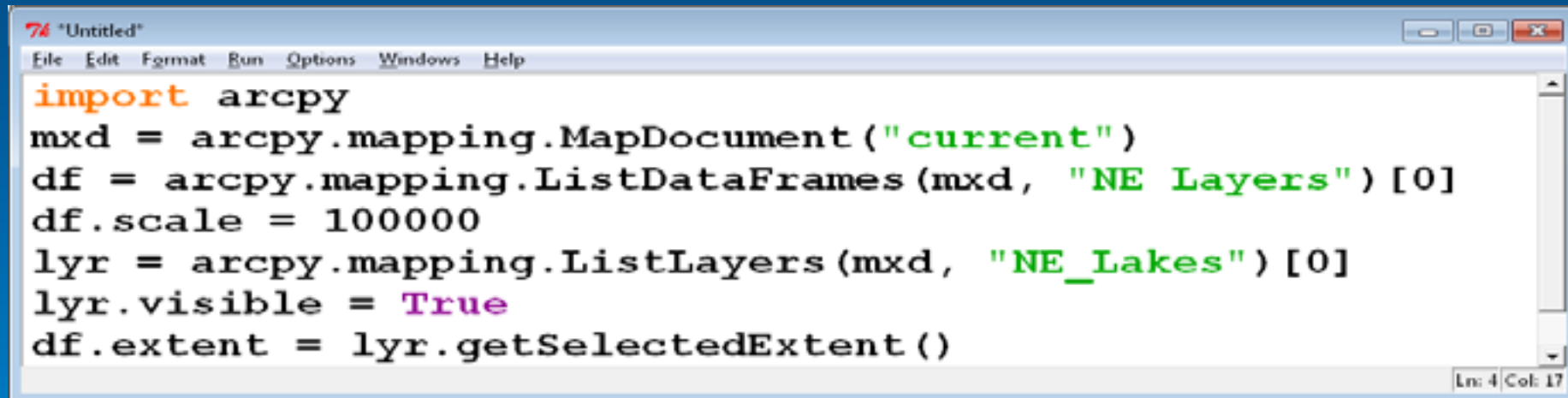
**Methods**  
**panToExtent(extent)**  
**zoomToSelectedFeatures()**

**Properties:**  
**credits**  
**description**  
**displayUnits**  
**elementHeight**  
**elementPositionX**  
**extent**  
**scale**

...

# Demonstration

- Working with Layers and Data Frames

A screenshot of a Python script editor window titled "Untitled". The window has a menu bar with "File", "Edit", "Format", "Run", "Options", "Windows", and "Help". The script content is as follows:

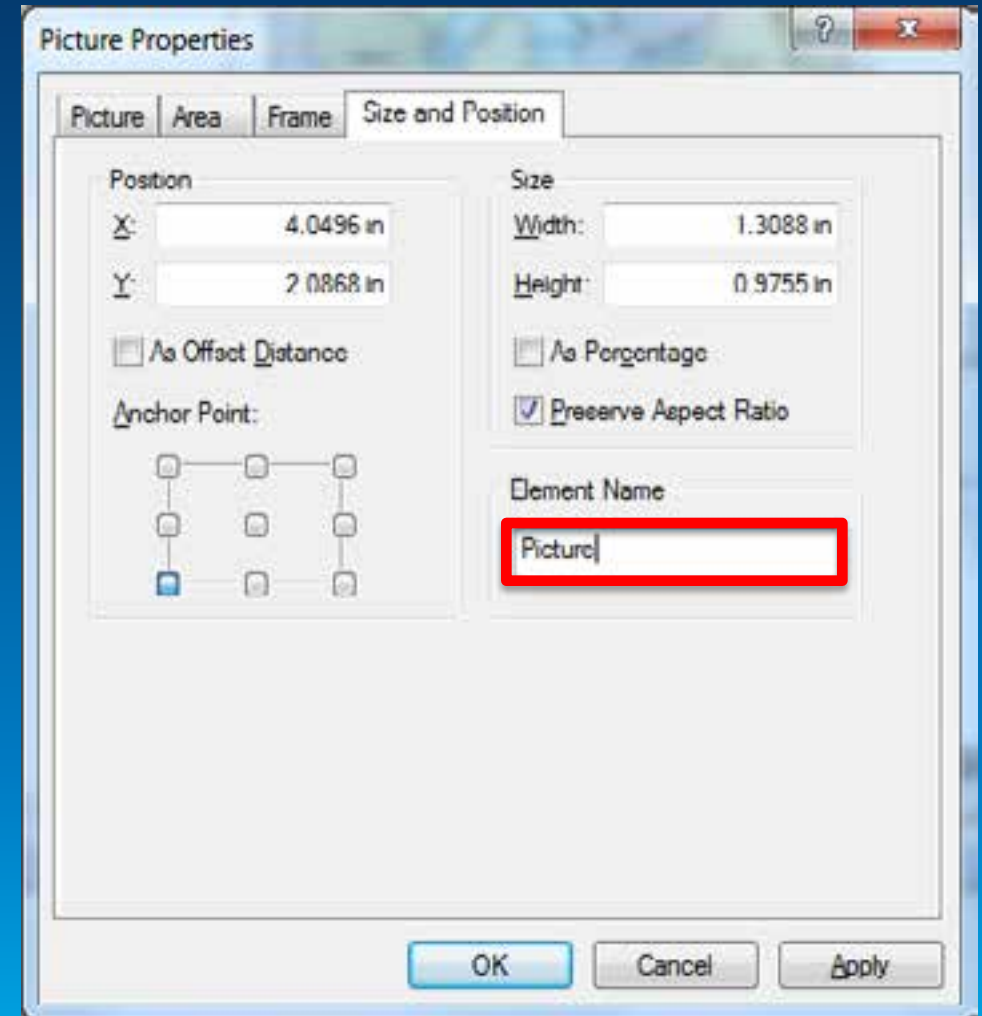
```
import arcpy
mxd = arcpy.mapping.MapDocument("current")
df = arcpy.mapping.ListDataFrames(mxd, "NE Layers")[0]
df.scale = 100000
lyr = arcpy.mapping.ListLayers(mxd, "NE_Lakes")[0]
lyr.visible = True
df.extent = lyr.getSelectedExtent()
```

The status bar at the bottom right of the window shows "Ln: 4 Col: 17".

- Find a layer and turns it on or off
- Modify the scale/rotation of a data frame
- Zoom to selected features

# Page Layout Elements

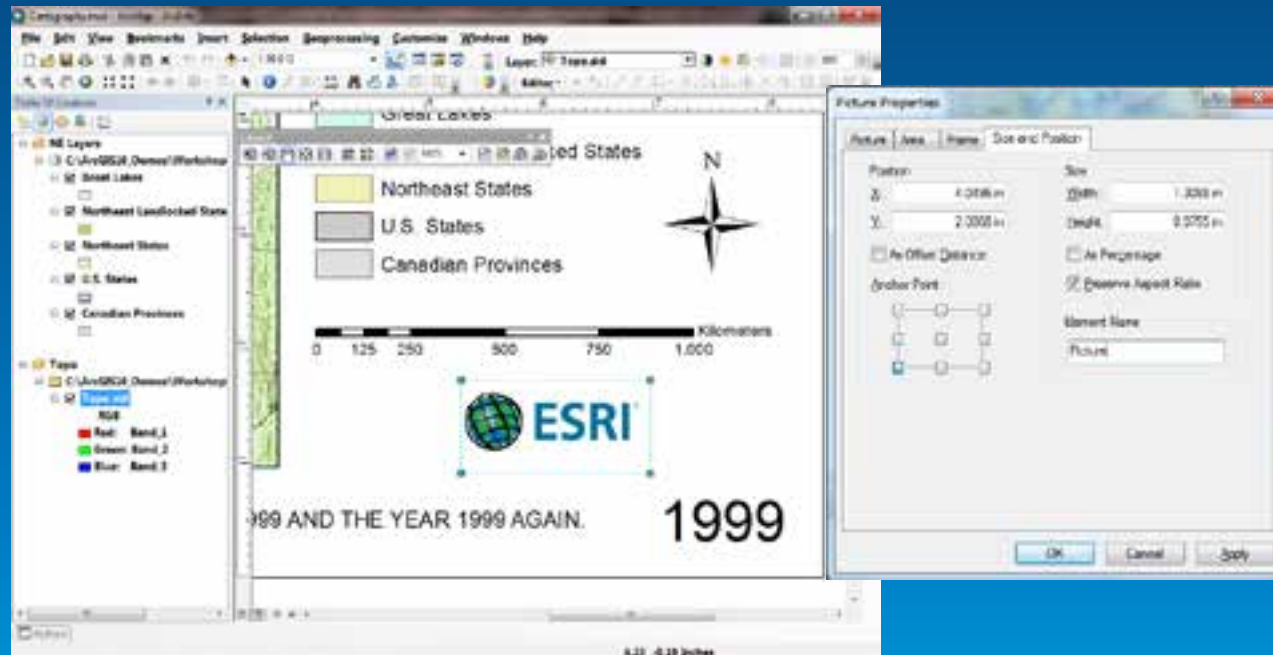
- Must pre-author most elements
  - Uniquely name your layout elements
  - Set the appropriate anchor
- Only graphics and text can be cloned
- Move elements off the page





# Demonstration

- Working with layout elements



- Find a picture element and change its data source

# Printing, exporting, server publishing, map books

## CLASSES

DataDrivenPages  
PDFDocument

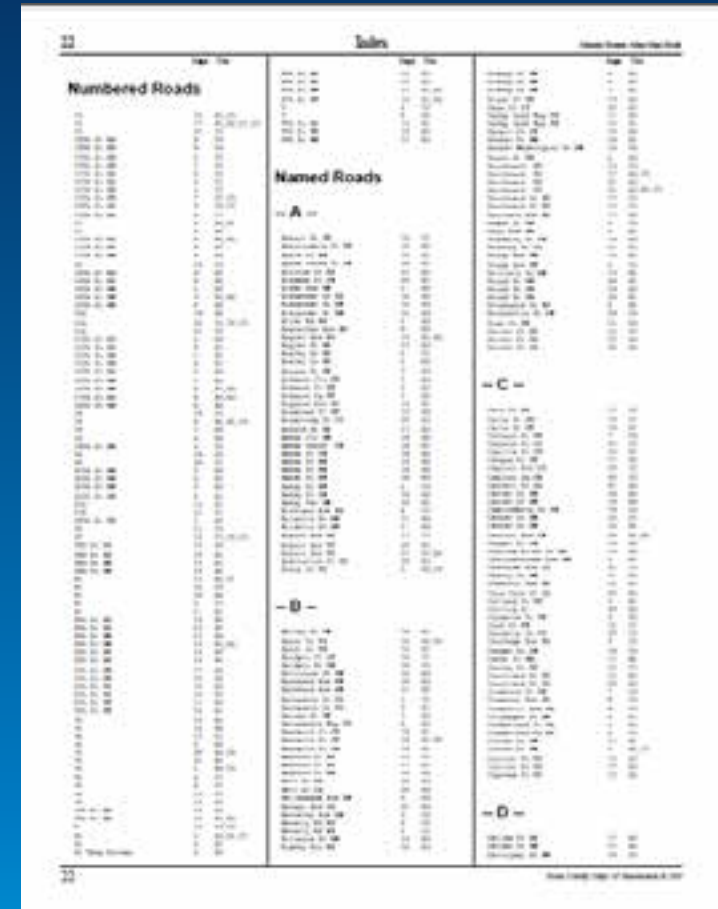
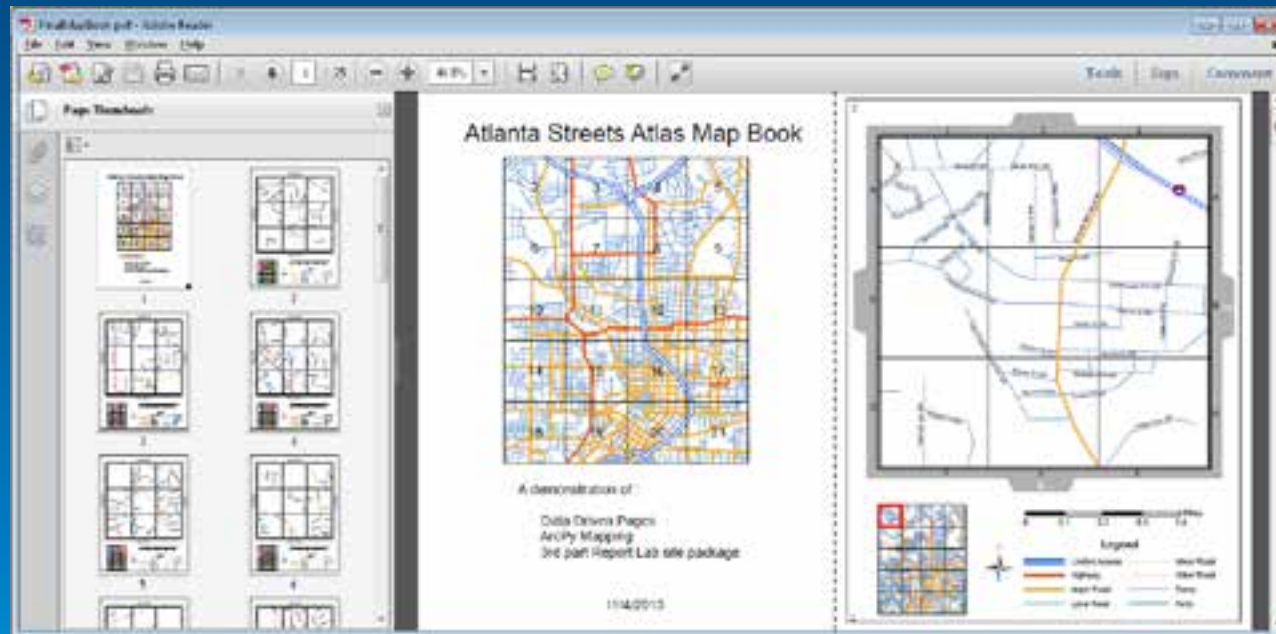
## FUNCTIONS

ExportReport  
ExportToAI  
ExportToBMP  
ExportToEPS  
ExportToGIF  
ExportToJPEG  
ExportToPDF  
ExportToPNG  
ExportToSVG  
ExportToTIFF  
PDFDocumentCreate  
PDFDocumentOpen  
PrintMap  
PublishMSDToServer

...

# Demonstration

- Map output and map books



Numbered Roads	Named Roads	Numbered Roads
101	101	101
102	102	102
103	103	103
104	104	104
105	105	105
106	106	106
107	107	107
108	108	108
109	109	109
110	110	110
111	111	111
112	112	112
113	113	113
114	114	114
115	115	115
116	116	116
117	117	117
118	118	118
119	119	119
120	120	120
121	121	121
122	122	122
123	123	123
124	124	124
125	125	125
126	126	126
127	127	127
128	128	128
129	129	129
130	130	130
131	131	131
132	132	132
133	133	133
134	134	134
135	135	135
136	136	136
137	137	137
138	138	138
139	139	139
140	140	140
141	141	141
142	142	142
143	143	143
144	144	144
145	145	145
146	146	146
147	147	147
148	148	148
149	149	149
150	150	150
151	151	151
152	152	152
153	153	153
154	154	154
155	155	155
156	156	156
157	157	157
158	158	158
159	159	159
160	160	160
161	161	161
162	162	162
163	163	163
164	164	164
165	165	165
166	166	166
167	167	167
168	168	168
169	169	169
170	170	170
171	171	171
172	172	172
173	173	173
174	174	174
175	175	175
176	176	176
177	177	177
178	178	178
179	179	179
180	180	180
181	181	181
182	182	182
183	183	183
184	184	184
185	185	185
186	186	186
187	187	187
188	188	188
189	189	189
190	190	190
191	191	191
192	192	192
193	193	193
194	194	194
195	195	195
196	196	196
197	197	197
198	198	198
199	199	199
200	200	200

Map book that includes index pages using Python ReportLab

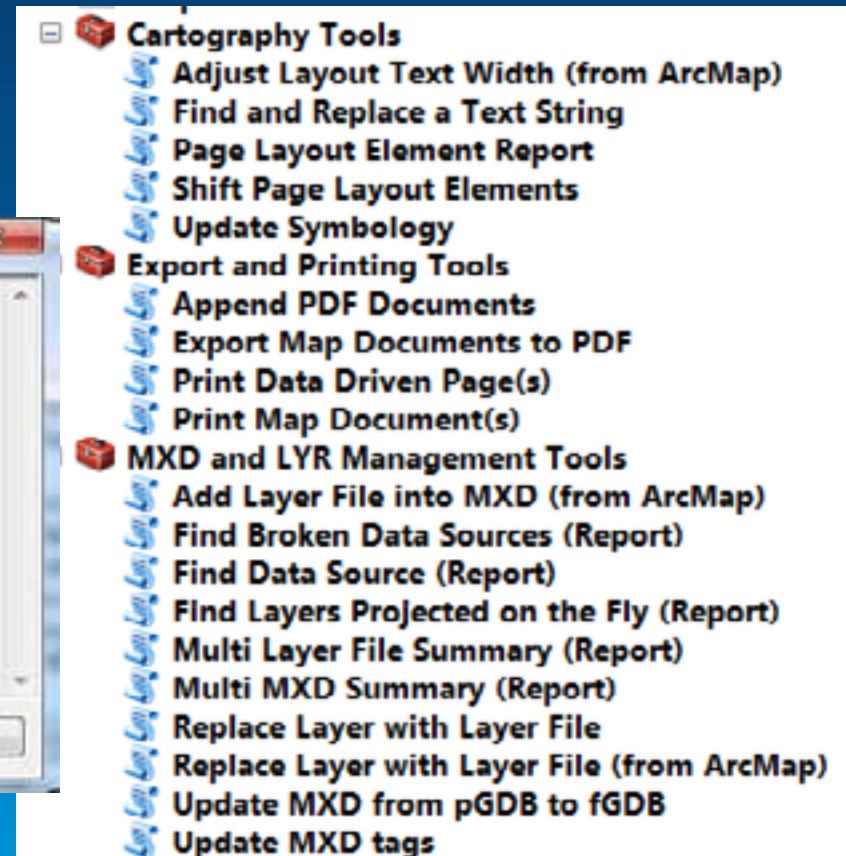
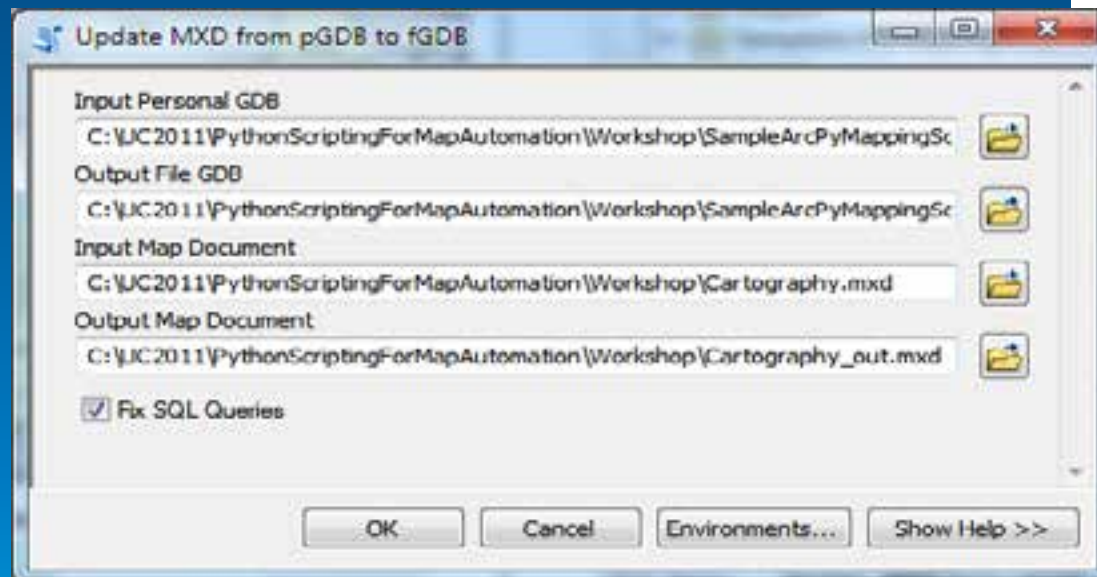
Sample: <http://esriurl.com/4629>

# Updating data sources

- Use `arcpy.mapping` for migrating Map Documents and Layer files to new data sources
- Fancier scripts can help mitigate migration pain: SQL syntax changes, field name changes, etc
- A complete concept document is dedicated to this topic
  - “Updating and fixing data sources with `arcpy.mapping`”
  - <http://esriurl.com/8149>

# Demonstration

- arcpy.mapping sample script tools



- Sample: <http://esriurl.com/4622>

# Resources available

- ArcGIS Resource Center (web help) <http://esriurl.com/8148>
  - Alphabetical lists of classes and functions
  - Detailed discussions
  - Multiple sample scripts for each class and function topic
- ArcGIS Resource Center (forums)
  - Map Automation: <http://esriurl.com/4624>
  - Python: <http://esriurl.com/4625>
- ArcGIS Online – arcpy.mapping / Map Automation group
  - Download sample scripts <http://esriurl.com/4626>

# Migrating to ArcGIS Pro

- Help Topic: Migrating arcpy.mapping from ArcMap to ArcGIS Pro
  - Python 3.4
  - ArcGIS project file (.aprx)
  - Stand-alone functions have moved to appropriate classes
    - `mapFrame.exportToPDF()`
    - `map.addLayer()`, `map.insertLayer()`, etc
  - Layer files have changed
  - `DataFrame` replaced by `Map`, `MapFrame`, and `Camera`
  - New `Layout` object
  - Application always refreshes when using `CURRENT`

***Thank you...***



**Please fill out the session evaluation**

***First Offering ID: 1301***

***Second Offering ID: 1398***

**Online – [www.esri.com/ucsessionsurveys](http://www.esri.com/ucsessionsurveys)**

**Paper – pick up and put in drop box**





Understanding our world.