ArcMap and Python: Closing the VBA Gap

Mark Cederholm UniSource Energy Services

Esri Developer Summit 2012

Using ArcObjects in Python Esri Developer Summit 2010

ArcMap and Python: Closing the VBA Gap Tuesday 4:30 Mesquite C

Extend Python Using C++ and ArcObjects Wednesday 11:15 Mesquite B

Download presentations and code: http://www.pierssen.com/arcgis/misc.htm http://www.pierssen.com/arcgis10/python.htm

2:23:00 PM

Why Python?

ArcGIS VBA support ends after 10.0 Python is integrated with ArcMap Geoprocessing tasks: arcpy ArcObjects manipulation: comtypes Custom forms: wxPython At 10.1, the ArcGIS add-in framework supports Python However, debugging can be difficult in ArcMap

The comtypes site package:

http://starship.python.net/crew/theller/comtypes/

The wxPython site package: http://wxpython.org/

2:23:00 PM

Modifying comtypes for 10.1:

Loading common ArcMap modules

At the Python prompt: >>> from comtypes.client import GetModule >>> GetModule("c:/program files/arcgis/desktop10.1/com/esriArcMapUI.olb")

[TIP: If loading one or modules fails, delete all files in the comtypes/gen folder before trying again.]

See Snippets.py for examples of using comtypes

2:23:00 PM

Workaround: Array object

- >>> import comtypes.gen.esriSystem as
 esriSystem
 >>> from comtypes.client import CreateObject
 >>> pArray = CreateObject("{8F2B6061-AB00-11D2-
 - 87F4-0000F8751720}", interface=esriSystem.IArray)

wxPython and ArcMap

- Will not work out-of-the-box in the ArcMap Python window
- Use PySimpleApp in a dedicated extension for the application MainLoop
- Destroy does not work: use Show instead
- The "print" command will not output to the Python window from a form

Customizing ArcMap, Method 1: Creating a Framework component (9.x and 10.0)

[Sample Code: ArcMap_Python\DemoCOM]

2:23:00 PM

Requires MIDL.EXE

Obtain by downloading and installing Windows SDK 7.1:

http://www.microsoft.com/download/en/details.aspx?id=8279

2:23:00 PM

Edit DemoExtension.idl and DemoTool.idl to set the correct ESRI paths:

```
DemoExtension.idl - Notepad
File Edit Format View Help
library DemoExtensionLib
   importlib("stdole32.tlb");
   importlib("stdole2.tlb");
   // Edit the following line as needed to fix the path
   importlib("\Program Files\ArcGIS\Desktop10.1\com\esriSystem.olb");
 DemoTool.idl - Notepad
File Edit Format View Help
library DemoToolLib
{
   importlib("stdole32.tlb");
   importlib("stdole2.tlb");
   // Edit the following line as needed to fix the path
   importlib("\Program Files\ArcGIS\Desktop10.1\com\esriSystemUI.olb");
```

Open Windows SDK 7.1 Command Prompt:



Microsoft Windows XP x86 Debug Build Environment
Setting SDK environment relative to C:\Program Files\Microsoft SDKs\Windows\v7.1
Targeting Windows XP x86 Debug
C:\Program Files\Microsoft SDKs\Windows\v7.1>_

 Use MIDL.EXE to produce DemoExtension.tlb and DemoTool.tlb: midl DemoExtension.idl midl DemoTool.idl

Use Python to register the COM objects*: python DemoExtension.py -regserver python DemoTool.py -regserver

*WARNING: The file/module name is case sensitive!

In ArcMap, add the tool in Customize Mode:

ize <u>W</u> indows <u>H</u> elp				
💽 🛫 🖽 🧊 🐺 🚳 🖸 🎾	24 🗸			
i 📸 🕺 💽 🗨 🖕				
Customize	Customize			
Toolbars Commands Options				
Show commands containing:				
Categories:	Commands:			
Python COM	8/ DemoTool			
Raster				
Descube				

 Activate the tool, select a quote, and draw a line:



Use Python to unregister the COM objects:

python DemoExtension.py -unregserver
python DemoTool.py -unregserver

Customizing ArcMap, Method 2: Creating an Add-in (10.1)

[Sample Code: ArcMap_Python\DemoAddin]

2:23:01 PM

Python Add-In Wizard — see "Obtaining the Python Add-In Wizard" in the ArcGIS 10.1 for Desktop Help:

Obtaining the Python Add-In Wizard

ArcGIS 10.1

Locate topic

To simplify the development of add-ins, it is recommended that you use **Python Add-In Wizard** to declare the type of customization. **Python Add-In Wizard** reduces development time and possible errors by creating the files and folders necessary for the add-in to work.

Download the Python Add-In Wizard.

The download is a compressed ZIP file (.zip) containing all the files necessary to support the wizard. To use the wizard, unzip the contents to a folder and locate the executable file named addin_assistant.exe in the bin folder; double-click this executable to launch the wizard. Examples of using **Python Add-In Wizard** are provided in the topics for creating an add-in.

Tip: Create a shortcut of the executable file on your desktop or quick launch toolbar.

Select or create a working folder and set other properties as desired:

ArcGIS P	ArcGIS Python Add-In Wizard					
Pythor	Python Add-In Wizard					
	Project Settings	Add-In Contents				
	Working Folder:	C:\apps\Development\ArcMap_Python\DemoAddin				
	Select Product:	ArcMap 🗾				
	Project Propert	ies:				
	Name*:	Python Addin				
	Version*:	0.1				
	Company:	Untitled				
	Description:	New Addin				
	Author:	Untitled				
	Image:	Select Image				

Add an extension (right-click on "EXTENSIONS"):

Python Add-In Wizard					
Extension					
Name: Demo Extension					
Class Name: DemoExtension					
ID (Variable Name): DemoAddin_addin.DemoExtension					
Description:					
Methods to Implement: Startup activeViewChanged mapsChanged newDocument openDocument beforeCloseDocument closeDocument beforePageIndexExtentChange pageIndexExtentChanged contentsChanged spatialReferenceChanged itemAdded itemDeleted itemReordered					

Add a toolbar:

Project Settings Add-In Contents	
Er EXTENSIONS	Toolbar
MENUS	Caption: Demo
Denio	ID (Variable Name): DemoAddin_addin.DemoToolbar
	Show Initially

Add a tool:

	Project Settings	Add-In Contents				
	EXTENSIONS	ension	Py	thon To	bol	
	- MENUS				Caption:	Demo Tool
	Demo	Tool			Class Name:	DemoTool
	Domo	100.		ID	(Variable Name):	DemoAddin_addin.DemoTool
					Tooltip:	AddIn: Add Quote
					Message:	Click on map to add quote
\backslash					Help Heading:	
$\left\{ \right\}$					Help Content:	
				I	mage for control:	2

2:23:01 PM

Click "Save", navigate to folder, and add files from sample code:



Double-click makeaddin.py to create addin:

	×	Name 🔺	Size	Туре
DemoAddin		🛅 Images		File Folder
Images		🛅 Install		File Folder
E 🔂 Install		📄 config.×ml	2 KB	XML Document
		Remakeaddin.py	2 KB	Python File
		🗐 README.txt	1 KB	Text Document
PythonC		🔁 DemoAddin.esriaddin	15 KB	Esri AddIn File

Double-click DemoAddin.esriaddin to install add-in:

Esri ArcGIS Ad	d-In Installation Utility 🛛 🔀	
	Please confirm Add-In file installation.	
6	Active content, such as Macros and Add-In files, can contain viruses or other security hazards. Do not install this content unless you trust the source of this file.	
Name:	Python Addin	
Version:	0.1	
Author:	Untitled	
Description:	New Addin	
Digital Signat	ure/s	
Signed By:		
Signed date:	Show Certificate	
	Source is trusted Signature is valid	
	Install Add-In Cancel	

In ArcMap, make sure extension is checked:

E	xtensions	×
	Select the extensions you want to use.	
	I 3D Analyst	-
	ArcScan	
	Demo Extension	
	Geostatistical Analyst	
	Network Analyst	
	Publisher	
	Schematics	
	Spatial Analyst	
	I International	

Toolbar automatically appears:

189 - 121	🚯 🥖 💷 🔛 🏦 🛍 🕺 💽 🗨 💂
×	Oh Fred, I think we've got an eater.
	💽 Add Quote 🛛 🗙
	Oh Fred, I think we've got an eater.
	Point = (106.911447084,930.885529158)
	Angle = 0.0

2:23:03 PM

TIP: Add script path to sys.path to enable local imports

import os import sys sMyPath = os.path.dirname(__file__) sys.path.insert(0, sMyPath)

```
class DemoExtension(object):
    _wxApp = None
    def __init__(self):
        sel f. enabled = True
    def startup(self):
        try:
             from wx import PySimpleApp
            self._wxApp = PySimpleApp()
            sel f. _wxApp. Mai nLoop()
        except:
             sMsg = "Error starting extension: \n" + \
                 traceback.format_exc()
            pythonaddi ns. MessageBox(sMsg, "DemoAddI n")
```

```
class DemoTool (object):
               _pApp = None
               _geometry = None
               _sQuote = None
               dl g = None
               def __i ni t__(sel f):
                    self.enabled = True
                   self.cursor = 3
Not stubbed out
                   self.shape = "Line" # Can set to . . .
  by default:
               def onClick(self):
                   if self._dlg is None:
                        from QuoteDialog import QuoteDialog
                        self. dlg = QuoteDialog(sMyPath, self)
                    sel f. _dl g. Show(True)
               def deactivate(self):
                    if self._dlg is None:
                        return
                    sel f. _dl g. Show(Fal se)
               def onLine(self, line_geometry):
                    self._geometry = line_geometry
                    self.DoIt()
```

2:23:03 PM

COM Interop: relative speed



Benchmark = 500+K ShapeCopy operations

2:22:58 PM

Some final tips:

- When in doubt, check the wrapper code: Python../Lib/site-packages/comtypes/gen
- Avoid intensive use of fine-grained ArcObjects in Python
- For best performance, use C++ to create coarsegrained objects
- Use geoprocessing objects and tools to simplify supported tasks – watch for performance, though
- Read the desktop help to check out available functionality in arcgisscripting (and arcpy at 10.x)

Questions?

Mark Cederholm mcederholm@uesaz.com This presentation and sample code may be downloaded at: http://www.pierssen.com/arcgis10/python.htm For 9.x examples, see: http://www.pierssen.com/arcgis/misc.htm

2:23:03 PM