Custom GIS Tools: How & Why

Matthew Basanta, GISP
Nick Soltis, GISP

2015 Ohio GIS Conference
September 22, 2015
Custom GIS Tools: How & Why

Matthew Basanta, GISP
Nick Soltes, GISP

2015 Ohio GIS Conference
September 22, 2015
Introduction

Benefits of custom tools

Identifying when a custom tool is useful

Basic types of GIS tools

Getting started with custom tools

Examples

Questions
When is a Tool Useful

- Frequent tasks
- Infrequent tasks
- Preparing tasks for other staff
- Automating background tasks
Types of GIS Tools
Model Builder
Python
.NET
Model Builder
Combines existing geoprocessing tools
Can incorporate scripts and nested models
Ideal for routine task automation
Getting Starting with Model Builder

ArcCatalog is all you need!

Automate one of your routine tasks

Export as a Python script and schedule
Getting Started with Model Builder

ArcCatalog is all you need!

Automate one of your routine tasks

Export as a Python script and schedule
Model Builder Example

A Good One
Model Builder Example
A Bad One
WARNING
TECHNICAL CONTENT AHEAD
Getting Started with Python

Download an example

Examine and adjust an existing Model Builder export

Online resources
Python Example

Monitor Instrumentation

View real time subsurface profiles
Python Example

Monitor Instrumentation

View real time subsurface profiles
Python Example

Monitor Instrumentation

View real time subsurface profiles
Python Example

Monitor Instrumentation

View real time subsurface profiles
Show me the Code!

Mixture of Python and Psuedo-code

Tryin' to keep it simple

Website sends the line verticies to the server as 2-d array

```python
in_src = '[
    [822.7,311.5],[772.5,142.5],...
]
```
Create the Profile

```
profile_dbf = os.path.join(WORKSPACE, "profile.dbf")
arcpy.CheckOutExtension('3D')
arcpy.StackProfile_3d(PROFILE, SURFACE, profile_dbf)
arcpy.CheckInExtension('3D')
```
Create the Profile Points

Another 2-d Array

```python
with DBF(profile_dbf) as table:
    for row in table:
        data_pt = [row['X'], row['Y1'], row['Y2']]  
        profile_data.append(data_pt)  
        json.dumps(profile_data)
```
Create the Profile Points

Another 2-d Array

```python
with DBF(profile_dbf) as table:
    for row in table:
        data_pt = [row['X'], row['Y1'], row['Y2']]
        profile_data.append(data_pt)
    json.dumps(profile_data)
```
Create the Profile Points

Another 2-d Array

```python
with DBF(profile_dbf) as table:
    for row in table:
        data_pt = [row['X'], row['Y1'], row['Y2']]
        profile_data.append(data_pt)
json.dumps(profile_data)
```
ArcObjects/Runtime SDK

More focused on stand alone applications.
Python can handle most ArcMap AddIns now.

ArcObjects
Old and Broke

- Older .NET/Java/C++
- Convoluted API
- More difficult debugging, logging, & error handling
- Easy to upgrade from VBA

Runtime SDK
New Hotness

- Java/WPF (And More)
- Sleeker and Purpose Built
- The Future
Getting Started with .NET

With Great Power Comes Great Responsibility

**ArcObjects**
- Visual Studio 2010 (Old, Old, Old)
- No Supported Free Version
- Very Developer-Centric Documentation

**Runtime SDK**
- VS 2012/2013 (Probably 2015)
- VS Community Edition is Free (and feature packed)
- Lots of Tutorials
  [http://developers.arcgis.com](http://developers.arcgis.com)
So Where Do I Start?
Not as straight-forward
Download VS 2013 - visualstudio.com
Try a tutorial - developers.arcgis.com
.NET Examples
MapBooks (Data Driven Pages)
Sign Inventory Tool