

ArcGIS and Python Programming

1. Project Idea
2. General Processes in Script
3. Code Snips (slides 7,12,13,14,16,17, 19, 20)
4. Interesting Elements of the Script
5. Project Outcomes

ArcGIS and Python Programming

Project Idea

Automate creation of mapbook products for

- Cal EMA (California Emergency Management Agency) sub- unit
- STTAC (State Terrorism Threat Assessment Center)
- Provide mapbook products to reduce response and recovery times in the event of a natural or terrorist-caused disaster within California. A python scripting process can automate part or all of the process of transforming data into map products.

ArcGIS and Python Programming

Project Idea

Probable data sources for Mapbooks

- California Basemap(s) – projection = ?
- Most current national “HSIP Freedom” data sets (license-free data sets describing features in 19 “Critical Infrastructure” sectors)
- California “Suspicious Incidents” Reports data from “eGuardian”
http://www.fbi.gov/page2/sept08/eguardian_091908.html
- California Monthly Summary report of Encounters with TIDE matches (persons whose names match **Terrorist Identities Datamart Environment** records)

ArcGIS and Python Programming

General processes in script

Real World Task	Script task
Acquire 3 data sources if not current	Automate periodic import (unzip) process
Clip 200+ national “Critical Infrastructure” (CIKR) data sets (Data source 1)	Automate clip of all HSIP Freedom feature classes to California
Assign 200 new data sets a common projection	Automate reprojection of all data to California State Plane (etc)
Annually Determine California UASI shapefiles	Select, dissolve buffer identified areas (8 areas in 2010)
Clip new California “Critical Infrastructure” (CIKR) data	Automate clip of all HSIP Freedom feature classes to CA - UASI areas
Create <u>Current Incidents</u> base map(s) with current Incidents- Data source 2 and 3	Automate maps for 8 CA - UASI areas (Urban Areas Security Initiative)
Create <u>Current Incidents +CIKR</u> map(s) (base maps x 19)	Automate loop to create clone maps with 1 CIKR sector layer(s) only added per clone (several feature classes per sector)
?? Create ARCGIS TOOL to allow user to select ONLY CIKR data desired	Design Interface in ARC Map

ArcGIS and Python Programming

General processes in script -accomplished

- Import system modules including zipfile, zlib
- Create the Geoprocessor object
- Set the workspace
- Use raw_input(“ displays during processing”) statements
- Create .txt log files to record file names, count files
- Call 2 bat files that
 - Create folders / subfolders for zip files and ~1500 ESRI shapefile “component “ files (.shp, .xml, .dbf, .prj, .sbn, .sbx, .shx)
 - Unzip all 227 HSIP data zip files from 19 parent zip files
- Clip all 227 .shp files by shape of California
 - Creates another ~1500 ESRI shapefile “component “ files (.shp, .xml, .dbf, .prj, .sbn, .sbx, .shx) PreFIXED “CA_
- Re -project all shp files to NAD 1983 State Plane California II FIPS 0402 (Feet)
 - Creates another ~1500 ESRI shapefile “component “ files (.shp, .xml, .dbf, .prj, .sbn, .sbx, .shx) SUFFIXED “_SP_CAI”

ArcGIS and Python Programming

Project's General Goals

- Use python processes to
 - Automate data import (HSIP data - CIKR)
 - Build shapefiles from other existing files (UASI Areas)
 - Create “Mapbooks”
 - Clipped Current Incidents/ TIDE-Data sources “Basemap”
 - Clone Incident basemap 19x, each with 1 of 19 CIKR sectors data
- Learn more about file folder management
- Learn about Python module zipfile

ArcGIS and Python Programming

Code Snip

Functional Code: use for first pass through filelist of top level HSIP folder

```
def getfilenames_In_Folder(dir_name, *args):
    fileList = []
    for file in os.listdir(dir_name):

        fullfilepath = os.path.join(dir_name, file)
        if os.path.isfile(fullfilepath):
            if zipfile.is_zipfile(fullfilepath):
                if len(args) == 0:
                    fileList.append(fullfilepath)
                    # I have a full path, and want to record
                    # the name of the files found
                    shortpath = os.path.basename(fullfilepath)
                    print "*** ", shortpath
                else:
                    if os.path.splitext(fullfilepath)[1][1:] in args:
                        fileList.append(fullfilepath)

            elif os.path.isdir(fullfilepath):
                print >> log, "Found sub-folder:", fullfilepath
                fileList += getfilenames_In_Folder(fullfilepath, *args)

    return fileList
```

ArcGIS and Python Programming

“HSIP Freedom” data sets

CIKR Sector	Number of Feature Classes
Agriculture and Food	3
Banking and Finance	10
Chemical and Hazardous Materials Industry	7
Commercial Facilities	30
Communications	12
Dams	2
Defense Industrial Base	2
Emergency Services	10
Energy	11
Government Facilities	29
Healthcare and Public Health	1
Information Technology	4
Manufacturing	11
National Monuments and Icons	1
Nuclear	2
Other	50
Postal and Shipping	7
Transportation	31
Water	5
Grand Count (19 sectors)	227

ArcGIS and Python Programming

HSIP Data(2009)

Address: H:\los_rios_course_376_python\Project\Data\HSIP Freedom 2010 4-5 old

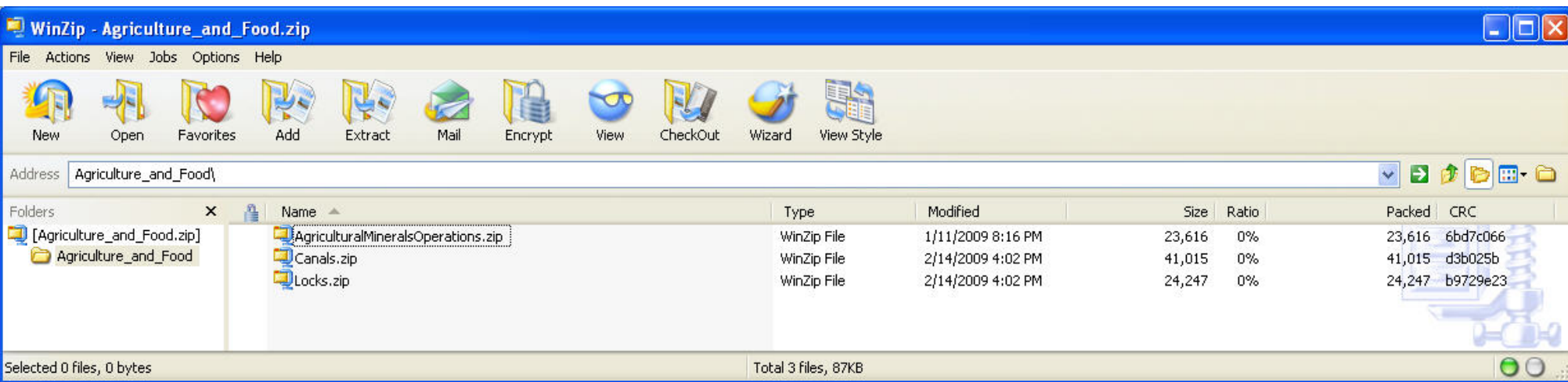
Name	Size	Type	Date Modified
Agriculture_and_Food.zip	88 KB	WinZip File	2/23/2009 7:32 PM
Banking_and_Finance.zip	12,988 KB	WinZip File	1/11/2009 8:04 PM
Chemical_and_Hazardous_Materials_Industry.zip	44,024 KB	WinZip File	2/23/2009 7:37 PM
Commercial_Facilities.zip	39,744 KB	WinZip File	1/11/2009 8:05 PM
Communications.zip	47,307 KB	WinZip File	1/11/2009 8:06 PM
Dams.zip	64 KB	WinZip File	2/23/2009 7:38 PM
Defense_Industrial_Base.zip	66 KB	WinZip File	1/11/2009 8:06 PM
Emergency_Services.zip	43,880 KB	WinZip File	1/11/2009 8:07 PM
Energy.zip	8,141 KB	WinZip File	1/11/2009 8:07 PM
Government_Facilities.zip	25,292 KB	WinZip File	2/23/2009 7:40 PM
Healthcare_and_Public_Health.zip	488 KB	WinZip File	1/11/2009 8:07 PM
Information_Technology.zip	229 KB	WinZip File	1/11/2009 8:07 PM
Manufacturing.zip	618 KB	WinZip File	1/11/2009 8:07 PM
National_Monuments_and_Icons.zip	42,387 KB	WinZip File	1/11/2009 8:07 PM
Nuclear.zip	58 KB	WinZip File	2/23/2009 7:40 PM
Other.zip	572,958 KB	WinZip File	2/23/2009 7:45 PM
Postal_and_Shipping.zip	11,267 KB	WinZip File	2/23/2009 7:47 PM
Transportation.zip	13,112 KB	WinZip File	2/23/2009 7:48 PM
Water.zip	37,893 KB	WinZip File	1/11/2009 8:16 PM

19 objects (Disk free space: 158 GB) 879 MB My Computer

19 zip files have 227 subzips

ArcGIS and Python Programming

HSIP Data(2009)



Click on a zip and most resulting displays show they are the result of zipping a folder with further multiple zip files inside

ArcGIS and Python Programming

Initial Problem with Zip files

- Python has a module “zipfile” which works to create compressed files “zipfile.ZipFile”, named for convenience I suppose, “x.zip”
- They are not exactly what winzip™ creates in terms of hidden headers, “magic numbers” etc.
- I could not get a code sequence to work where I “python-unzipped” winzip - created files .

ArcGIS and Python Programming

Code Snip

Functional Code: calls two .bat files

```
# this does a big piece of PART1 work to unzip the top level ZIPS
# displays new sub-zips names and paths
raw_input("2 That was show and tell. Batchfiles will now do their work to unzip all sub-level zip files contents. This will take some time. Press any key to continue")

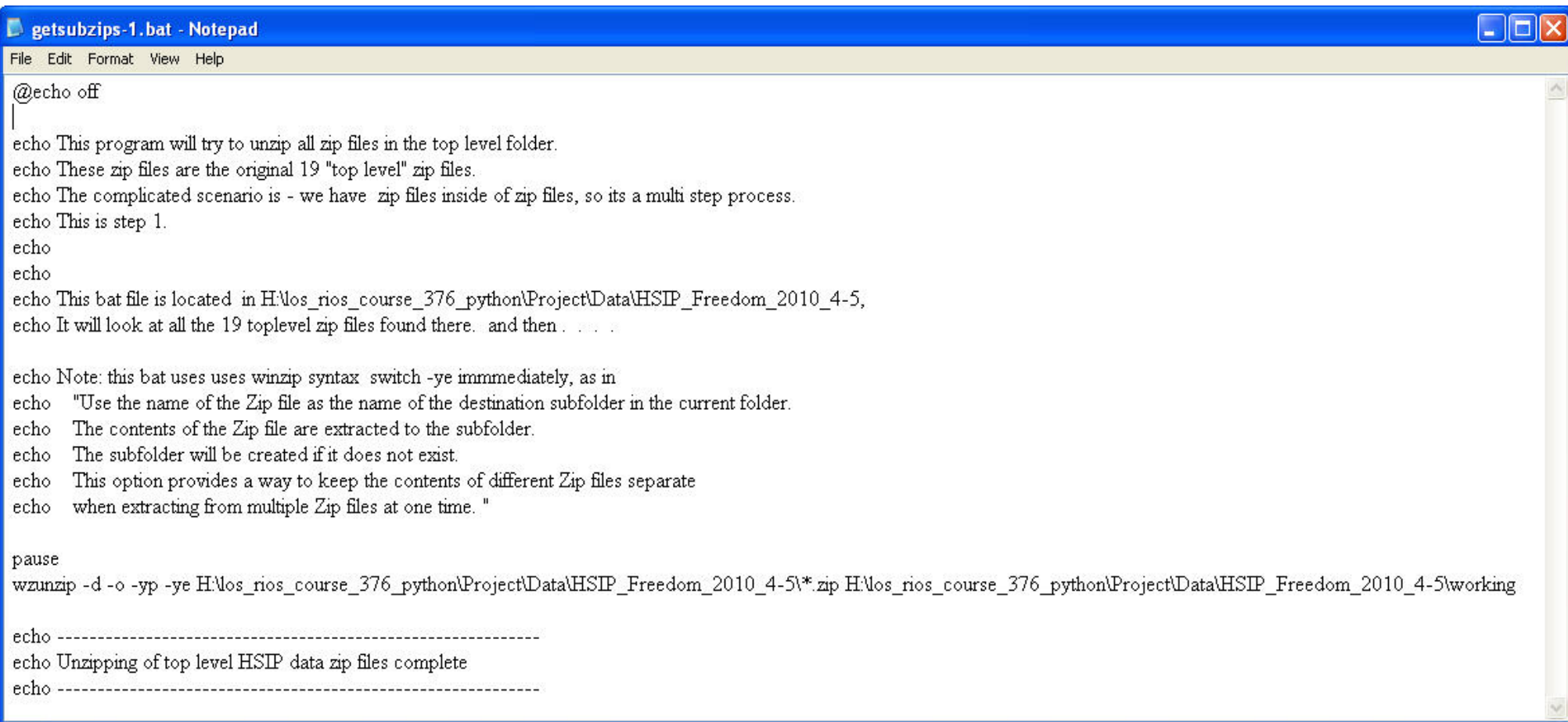
#call 2 bat files that unzip the tl and sub zips
BatOne = "H:\\los_rios_course_376_python\\Project\\workingCode\\get_TLzips-1.bat"
BatTwo = "H:\\los_rios_course_376_python\\Project\\workingCode\\getsubzips-501.bat"

os.system(BatOne)
os.system(BatTwo)
print " "
print "Part one of this routine complete."
print "All 227 zip files unzipped to H:\\los_rios_course_376_python\\Project\\Data\\HSIP_Freedom_2010_4-5\\working"
print "and subfolders" # this all works

raw_input("3 Sublevel zips now expanded in separate folders. Next we clip all shape files by california.shp Press any key to continue")
```

ArcGIS and Python Programming

First batch file to automate unzipping using winzip syntax



```
getsubzips-1.bat - Notepad
File Edit Format View Help

@echo off
|
echo This program will try to unzip all zip files in the top level folder.
echo These zip files are the original 19 "top level" zip files.
echo The complicated scenario is - we have zip files inside of zip files, so its a multi step process.
echo This is step 1.
echo
echo
echo This bat file is located in H:\os_rios_course_376_python\Project\Data\HSIP_Freedom_2010_4-5,
echo It will look at all the 19 toplevel zip files found there. and then . . . .

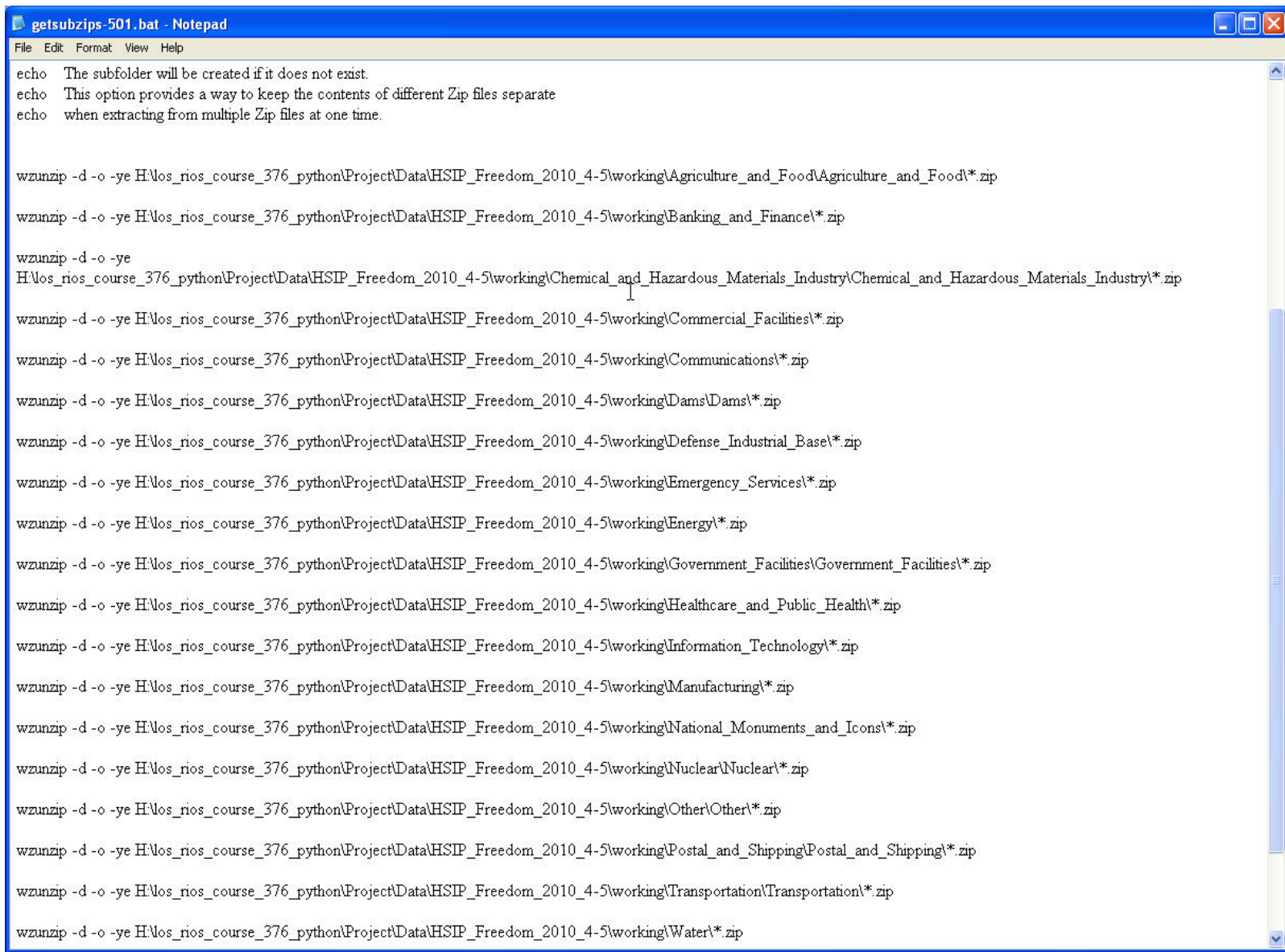
echo Note: this bat uses uses winzip syntax switch -ye immmediately, as in
echo "Use the name of the Zip file as the name of the destination subfolder in the current folder.
echo The contents of the Zip file are extracted to the subfolder.
echo The subfolder will be created if it does not exist.
echo This option provides a way to keep the contents of different Zip files separate
echo when extracting from multiple Zip files at one time. "

pause
wzunzip -d -o -yp -ye H:\os_rios_course_376_python\Project\Data\HSIP_Freedom_2010_4-5\*.zip H:\os_rios_course_376_python\Project\Data\HSIP_Freedom_2010_4-5\working

echo -----
echo Unzipping of top level HSIP data zip files complete
echo -----
```

ArcGIS and Python Programming

Second batch file to automate unzipping using winzip syntax



```
getsubzips-501.bat - Notepad
File Edit Format View Help
echo The subfolder will be created if it does not exist.
echo This option provides a way to keep the contents of different Zip files separate
echo when extracting from multiple Zip files at one time.

wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Agriculture_and_Food\Agriculture_and_Food\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Banking_and_Finance\*.zip

wzunzip -d -o -ye
H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Chemical_and_Hazardous_Materials_Industry\Chemical_and_Hazardous_Materials_Industry\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Commercial_Facilities\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Communications\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Dams\Dams\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Defense_Industrial_Base\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Emergency_Services\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Energy\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Government_Facilities\Government_Facilities\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Healthcare_and_Public_Health\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Information_Technology\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Manufacturing\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\National_Monuments_and_Icons\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Nuclear\Nuclear\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Other\Other\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Postal_and_Shipping\Postal_and_Shipping\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Transportation\Transportation\*.zip
wzunzip -d -o -ye H:\os_rios_course_376_python\ProjectData\HSIP_Freedom_2010_4-5\working\Water\*.zip
```

ArcGIS and Python Programming

Second batch file - results

- Successful
- Creates very long path – I am not sure this is the most desirable result
- H:\los_rios_course_376_python\Project\Data\HSIP_Freedom_2010_4-5\working\Agriculture_and_Food\Agriculture_and_Food\Locks
- H:\los_rios_course_376_python\Project\Data\HSIP_Freedom_2010_4-5\working\Dams\Dams\Locks **also exists**

ArcGIS and Python Programming

Code Snip

Functional Code: clip by California.shp

```
# Process: now we need to Clip all shp files by shape of ca
# put new shape file in same folder?

newSearchFolder = "H:\\los_rios_course_376_python\\Project\\Data\\HSIP_Freedom_2010_4-5\\working"
for root, dirs, files in os.walk(newSearchFolder):
    # uses "H:\\los_rios_course_376_python\\Project\\Data\\HSIP_Freedom_2010_4-5\\working"
    for name in files:
        extension = os.path.splitext(name)[1]
        if extension == ".shp" :
            filename = os.path.join(root, name)
            #print "first we look in this dir " +os.path.dirname(filename) #print " for " +filename
            baseRoot = "H:\\los_rios_course_376_python\\Project\\Data\\HSIP_Freedom_2010_4-5\\working" #static
            shortpathTBC = os.path.basename(name) # ?? Nuclear\\NuclearFuel\\NuclearFuel.shp #print shortpathTBC
            tobeClipped = filename #print "to be Clipped = " + tobeClipped
            Ca_Clipper = "H:\\los_rios_course_376_python\\Project\\Data\\base\\California.shp" #static
            new_outputName = "\\Ca_" +shortpathTBC # print "new_output Name = " +new_outputName
            clipOutputSHP = root + new_outputName #print "clip Output SHP = " +clipOutputSHP
            #raw_input("Press any key to continue.")
            gp.Clip_analysis(tobeClipped,Ca_Clipper,clipOutputSHP,"") # print "new feature at - " + clipOutputSHP
            #raw_input("Press any key to continue.") I
print "All new features -clipped by california shape - created."
```

First use of gp.Something

ArcGIS and Python Programming

Code Snip: Reprojection

```
print "Starting re -Projection for Ca-clipped shape files ONLY"
print "whose current projection"
print "does not match NAD 1983 StatePlane California II FIPS 0402 (Feet)."
print " "

num =1
Reproject_workspace = newSearchFolder

# Set the desired spatial reference variable - cs
cs = "F:\Program Files\ArcGIS\Coordinate Systems\Projected Coordinate Systems\State Plane\NAD 1983 (Feet)\NAD 1983 StatePlane California II FIPS 040
#cs doesnt work to be passed in as parameter # so I used modelbuilder to get a bit o'script
cs2 = "PROJCS['NAD_1983_StatePlane_California_II_FIPS_0402_Feet',GEOGCS['GCS_North_American_1983',DATUM['D_North_American_1983',SPHEROID['GRS_1980',6378137.

for root, dirs, files in os.walk(Reproject_workspace):
    for shp in files:
        extension = os.path.splitext(shp)[1]
        if extension == ".shp" : #obviously we are looking for shapefiles here
            if shp[:3] == "Ca_" : #look for Ca_ prefix to indicate clipped files
                if not shp[-12:] == "_SP_CAII.shp" : #exclude already created files as script will ther wise continue to add suffix as below
                    print " " #but gp.OverwriteOutput allows overwrite
                    print "Shapefile ",str(num), " : ", str(shp)
                    filename = os.path.join(root, shp) #print "filename = ", filename
                    baseoutput = os.path.basename(filename)[:4] + "_SP_CAII.shp" #creates a tag identifying the new projection
                    # we could do a batch rename later I suppose
                    output = os.path.join(root, baseoutput) #print "output = ", output
                    if gp.Exists(output):
                        gp.Delete_management(output)
                    desc = gp.describe(filename) #print desc")
                    sr = desc.SpatialReference #print sr
                    SR_name = desc.spatialreference.name

                    if SR_name == "Unknown": # 3 possibilities re SR_name, improbable that this data will be without spatial reference
                        print "Unknown Projection!"
                        gp.AddError("Unknown Projection!")# more serious that different SR
                        gp.DefineProjection(filename, cs2) # never actually had this option run
                        print "Completed adding SP_CAII coordinate system to " + shp
                    elif SR_name == 'NAD 1983 StatePlane California II FIPS 0402 (Feet)':
                        print " "
                        gp.AddMessage('Projection matches desired NAD 1983 StatePlane California II FIPS 0402 (Feet).')
                        print "Projection matches desired NAD 1983 StatePlane California II FIPS 0402 (Feet)."
```

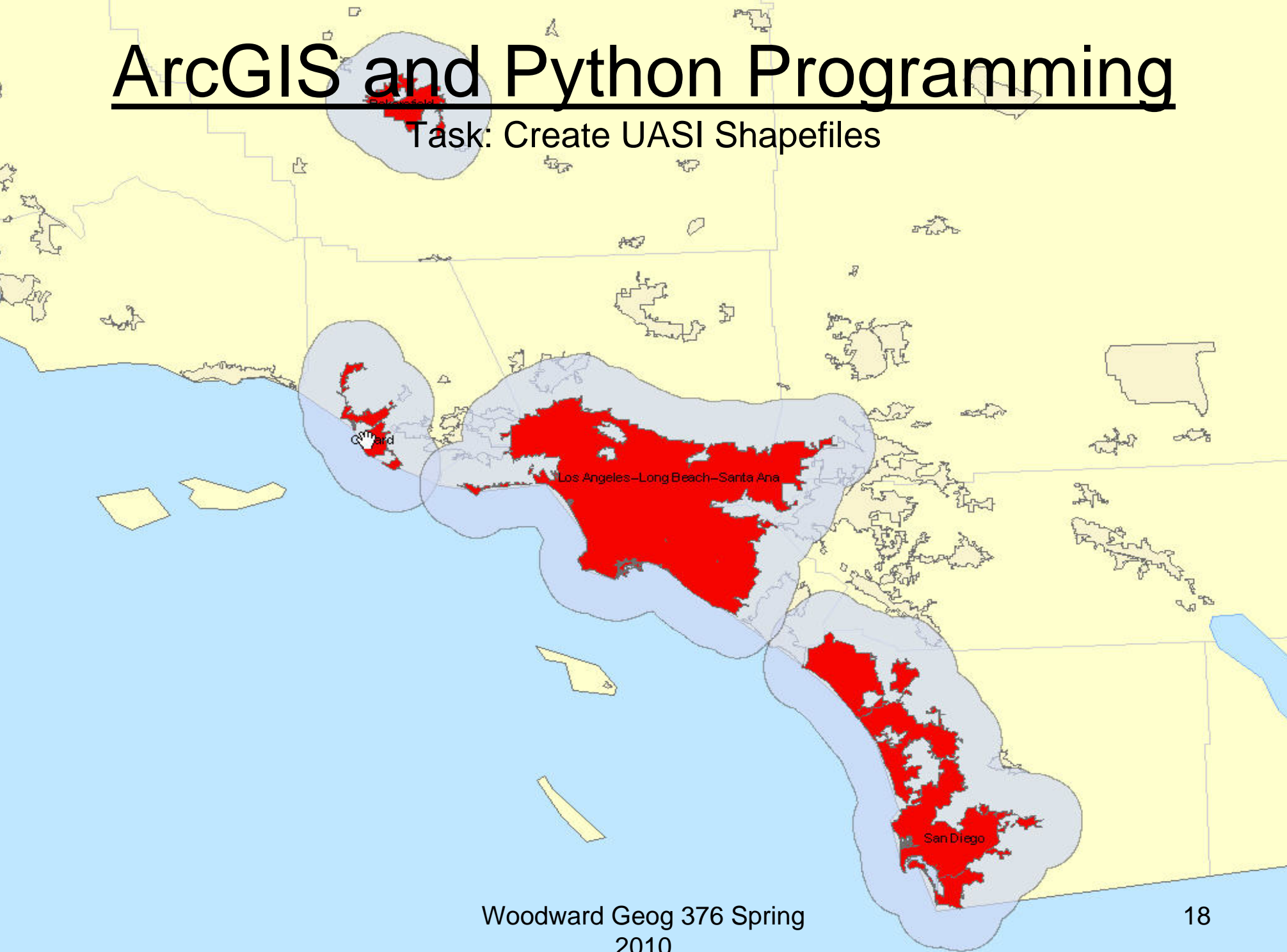
```
                    else :
                        print shp + " -Projection does not match NAD 1983 StatePlane California II FIPS 0402 (Feet)!"
                        gp.AddWarning('Projection does not match NAD 1983 StatePlane California II FIPS 0402 (Feet)!')
                        gp.AddWarning(SR_name)
                        if SR_name.find("WGS_1984")==-1:
                            trans = "#" # or remove this line
                        else:
                            trans = "NAD_1983_To_WGS_1984_1"

                    #gp.AddMessage('Parameters:'+ shp+', '+output+', '+cs+', '+trans)
                    #print "Parameters: "+ shp+", "+output+", "+cs+", "+trans
                    #raw_input("3 Press any key to perform reprojection, or press x, top right, to stop here.")
                    print "Starting reprojection."
                    #syntax Project_management (in_dataset, out_dataset, out_coor_system, transform_method(optional), in_coor_system)
                    gp.Project_management(filename, output, cs2, trans)
                    print "Reprojection complete for - ", filename
                    print "new file at -", output
```

gp.Project

ArcGIS and Python Programming

Task: Create UASI Shapefiles



ArcGIS and Python Programming

Code Snip

```
def Process_UASI_Counties (UASI_name, querystring):
    newfolder = UASI_name + " UASI_data" #\\BAUA_UASI_data etc
    outputFOLDER = gp.workspace + newfolder
    if not os.path.exists(outputFOLDER):
        os.makedirs(outputFOLDER)
    print outputFOLDER
    Input_Shape_Layer = outputFOLDER + UASI_name + "_Counties.lyr"

    FinalOutput = outputFOLDER + UASI_name + "_Counties.shp"
    print FinalOutput
    print Input_Shape_Layer
    raw_input("1 Press any key to continue.")#remove this later

    #Process Make a layer from the feature class
    if gp.exists(Input_Shape_Layer):
        gp.delete_management(Input_Shape_Layer)
    # Make a layer from the feature class arb ca counties
    gp.MakeFeatureLayer_management(Input_ShapeFile,Input_Shape_Layer)
    print Input_Shape_Layer

    print "Make FeatureLayer for counties . . ."
    print " - done."

    raw_input("2 Press any key to continue.")#remove this later

    # Process: Select Layer By Attribute...
    # syntax SelectLayerByAttribute_management (in_layer_or_view, selection_type, where_clause)
    New_Selection = "NEW_SELECTION"
    gp.SelectLayerByAttribute(Input_Shape_Layer, "NEW_SELECTION", querystring)
    result = gp.GetCount_management(Input_Shape_Layer)
    print "For " + UASI_name[1: ]+ " UASI shapefile production, The total number of selected counties is " + str(result) + "."
    print "Expected 13 for BAUA, 1 for Sacramento, 2 for Riverside. aha!"
    raw_input("3 Press any key to continue.")#remove this later

    # Write the selected features to a new featureclass
    gp.CopyFeatures(Input_Shape_Layer, FinalOutput)
    print "Done."

    if result > 1:
        # Process: Dissolve...
        Counties_Dissolve_shp = FinalOutput[ :-4] + "_Dissolve.shp"
        raw_input("4 About to dissolve shapes to " +Counties_Dissolve_shp+ " . Press any key to continue.")#remove this later
        gp.Dissolve_management(FinalOutput, Counties_Dissolve_shp, "", "", "SINGLE_PART", "DISSOLVE_LINES")

        # Process: Buffer...
        Final_Buffer = FinalOutput[ :-4] + "_Buffer.shp"
        gp.Buffer_analysis(Counties_Dissolve_shp,Final_Buffer , "10 Miles", "FULL", "ROUND", "ALL", "")
        raw_input("5 About to buffer shapes to " +Final_Buffer+ " . Press any key to continue.")#remove this later
    elif result == 1: # no dissolve needed
        # Process: Buffer...
        Final_Buffer = FinalOutput[ :-4] + "_Buffer.shp"
        gp.Buffer_analysis(FinalOutput,Final_Buffer , "10 Miles", "FULL", "ROUND", "ALL", "")
        raw_input("4 About to buffer 1 county to " +Final_Buffer+ " . Press any key to continue.")#remove this later

    print "Completed processing"
    print " "
```

Functional Code

Create new feature classes describing UASI areas
(_Counties.shp, _Area.shp), select, maybe dissolve
SELECT county or counties polygon from Ca Counties (given
sql query defining target counties)

OR (duplicate code not shown here)
SELECT city polygon from ESRI data (given sql defining
target cities)

Dissolve to one polygon, if multiple area "parts" found
Buffer the identified UASI area (county, merged county,
"urban area") features by 10 miles.

gp.MakeFeatureLayer
gp.SelectLayerByAttribute
gp.GetCount
gp.CopyFeatures
gp.Dissolve
gp.Buffer

ArcGIS and Python Programming

Code Snip

Functional Code: calls function to Process UASI Counties

```
Suffix = "\\BAUA"
query = "\"NAME\" = 'ALAMEDA' OR \"NAME\" = 'CONTRA COSTA' OR \"NAME\" = 'MARIN' OR \"NAME\" = 'MAPA' OR \"NAME\" = 'SANTA CLARA' OR \"NAME\" = 'SA
Process_UASI_Counties (Suffix, query)

Suffix = "\\Sacramento"
query = "\"NAME\" = 'SACRAMENTO'"
Process_UASI_Counties (Suffix, query)

Suffix = "\\Riverside"
query = "\"NAME\" = 'RIVERSIDE' OR \"NAME\" = 'SAN BERNARDINO'"
Process_UASI_Counties (Suffix, query)

"""
now
|
Bakersfield
LA LB Anaheim Santa Ana Area
Oxnard Area
San Diego Area
"""

Suffix = "\\Bakersfield"
query = "\"NAME\" = 'Bakersfield'"
Process_UASI_Cities (Suffix, query)

Suffix = "\\LA LB Anaheim Santa Ana"
query = "\"NAME\" = 'Los Angeles--Long Beach--Santa Ana'"
Process_UASI_Cities (Suffix, query)

Suffix = "\\Oxnard"
query = "\"NAME\" = 'Oxnard'"
Process_UASI_Cities (Suffix, query)

Suffix = "\\San_Diego"
query = "\"NAME\" = 'San Diego' AND \"UA_ID\" = '78661'"
Process_UASI_Cities (Suffix, query)
```

Function is

Process_UASI_counties

These lines use the call to a duplicate function

Process_UASI_Cities

ArcGIS and Python Programming

Interesting elements of the application

- Batch file use very easy /efficient, minimal learning curve
- Python “zipFile” module is not mature nor predictable in interaction with WinZip created files
- New code structures to me
 - “for root, dirs, files in os.walk(datafolder): “
 - raw_input (“text to be read here. Press any key to continue.”)
 - for f in fileList():
 - For names in namelist()
 - gp.extent = "MAXOF" # Set the Geoprocessing environment
 - Getfilenames_In_Folder function is recursive
 - if `__name__ == '__main__'`: can conditionally fire “self test” code

ArcGIS and Python Programming

Project Outcomes

- Folder creation and Unzip All HSIP data sets process - automated
- Clipped all HSIP data by california.shp -automated
- Reprojected all HSIP data to California State Plane (etc)
- Create 7 UASI Area shapefiles from existing county /city polygon data -automated
- Log HSIP_Zip_Files_Discovery_date.txt documents numbers of folders and file counts.

ArcGIS and Python Programming

Project Outcomes

- Time wasted pursuing Python zipfile module
- Project not achievable -Too large a process demanding building a business process before writing the code
- So much data (HSIP) whose value is not inherently obvious – We probably would not want to create 19 maps per sector per week per (UASI)urban cluster. Real world task needs delimiting by defining mgmt needs /information needs.
- Trial and error would be needed to determine if steps are better in a different sequence, as in “re-project then clip” or “clip then re-project”

ArcGIS and Python Programming

Project Outcomes

- Too ambitious a concept for the time allowed outside of work in student role
- Mapbook final outcome not achieved. Functionality may be exposed in ArcGIS 10.

ArcGIS and Python Programming

Project Future Improved Outcomes

- Review reminders to # replace with sysarg later ?
- Review idea –delete “Ca_filename” files as intermediary
- Review idea – use cursor to determine if projected files “ca_filename_SP_CAll” have a zero count of records, if so delete files and folder.
- Determine differences between 3 zip files with same name -“locks.zip” but in different HSIP sectors. Possible unnecessary processing of redundant files.
- Review - Count of re-projected shp files reports as 232; expected 227 ???