

Stephen F. Austin State University

SFA ScholarWorks

---

Faculty Publications

Spatial Science

---

12-2012

## Using ArcPy to Automate Making Thematic Map Books: Two Cases

Yanli Zhang

*Arthur Temple College of Forestry and Agriculture, Stephen F. Austin State University, zhangy2@sfasu.edu*

Daniel Unger

*Arthur Temple College of Forestry and Agriculture, Stephen F. Austin State University, unger@sfasu.edu*

Jason Grogan

*Arthur Temple College of Forestry and Agriculture, Stephen F. Austin State University, jgrogan@sfasu.edu*

Brian P. Oswald

*Arthur Temple College of Forestry and Agriculture, Stephen F. Austin State University, Nacogdoches, Texas 75962, boswald@sfasu.edu*

John Lanham

Follow this and additional works at: <https://scholarworks.sfasu.edu/spatialsci>



Part of the [Forest Management Commons](#)

[Tell us](#) how this article helped you.

---

### Repository Citation

Zhang, Yanli; Unger, Daniel; Grogan, Jason; Oswald, Brian P.; and Lanham, John, "Using ArcPy to Automate Making Thematic Map Books: Two Cases" (2012). *Faculty Publications*. 19.

<https://scholarworks.sfasu.edu/spatialsci/19>

This Article is brought to you for free and open access by the Spatial Science at SFA ScholarWorks. It has been accepted for inclusion in Faculty Publications by an authorized administrator of SFA ScholarWorks. For more information, please contact [cdsscholarworks@sfasu.edu](mailto:cdsscholarworks@sfasu.edu).

# Using ArcPy to Automate Making Thematic Map Books: Two Cases

By Yanli Zhang, Daniel Unger, Jason Grogan, Brian P. Oswald, and John Lanham

According to Esri, a map book is a collection of maps printed or exported together. The most common example of a map book is a road atlas, in which every map contains the same layer information, but for a different location. Road atlases are simple reference-series map books and can be divided into two groups: each map represents an administrative area such as a state or a county (scale varies), or each map represents a cell of an index grid (scale remains constant).

As they may have tens or hundreds of maps, map book-making can be a tedious and time-consuming job. Esri provides detailed step-by-step instructions to simplify the creation of index grid-driven map books, (<http://help.arcgis.com/en/arcgis-desktop/10.0/help/index.html#//00s90000002p000000.htm>). One important step in these instructions is running a sample ArcPy code provided on that webpage. Esri introduced ArcPy in ArcGIS 10 as a facility for integrating Python programming-language scripts into ArcGIS Desktop. However, not every ArcMap user is an experienced Python programmer. In this article, we offer an introduction to using ArcPy to automate the making of two kinds of thematic map books.

ArcPy offers some exciting features for mapping automation. These features include—but are not limited to—map document and layer management, automating map export, and printing. Thus, ArcPy can facilitate map book-making. To run ArcPy code, the user needs to open a Python IDLE user-interface window (in a Windows 7 system with a default ArcMap installation, select Program – ArcGIS – Python 2.6 – IDLE). Then use the File menu to open a new window (Figure 1), where you can paste and edit the sample ArcPy code that we provide. When the code is finalized, you can run the code to automate GIS analysis or map exporting.

Two cases from our recent research help illustrate making thematic map books. In the first case, each map represents a different layer in the map—in other words, we need to loop through map layers, turning on and off a series of layers in the map document. This is called a layer-driven thematic map book. In the second case, we only have one important layer and need to define its symbology based on different attribute fields. This is called an attribute field-driven thematic map.

## A Layer-Driven Map Book

Our example of a layer-driven thematic map book is shown in Figure 2. There are 13 precipitation layers in the map data frame, each representing monthly precipitation or the annual precipitation. The requirement is to produce a single-file map book in PDF format, with each page showing each month's precipitation. If we do this manually, we need to turn on one layer, change the title, then export the map as that month's PDF file. Finally, the 13 individual files would be combined into one output PDF file. We developed ArcPy code (Figure 3) to automate the procedure. This code can be used in ArcMap 10 and 10.1. As a demonstration, four of the output maps are shown in Figure 4.

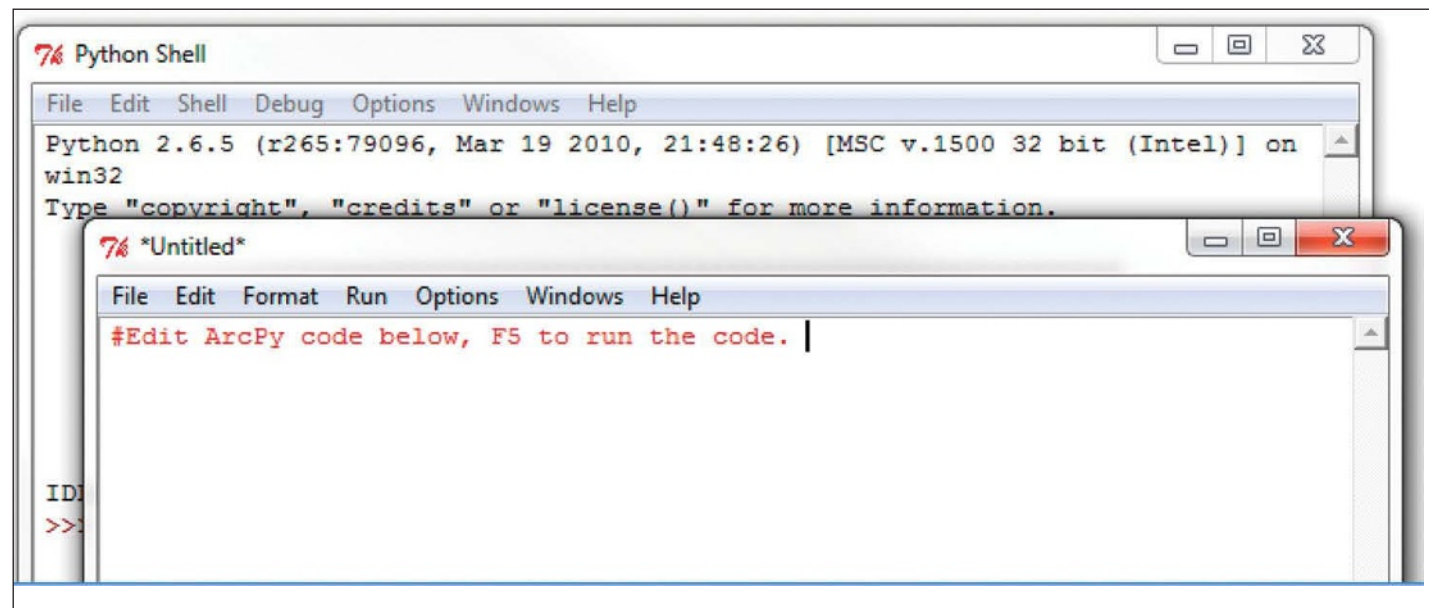


Figure 1. Python 2.6 user interface for editing and running ArcPy code in ArcGIS.

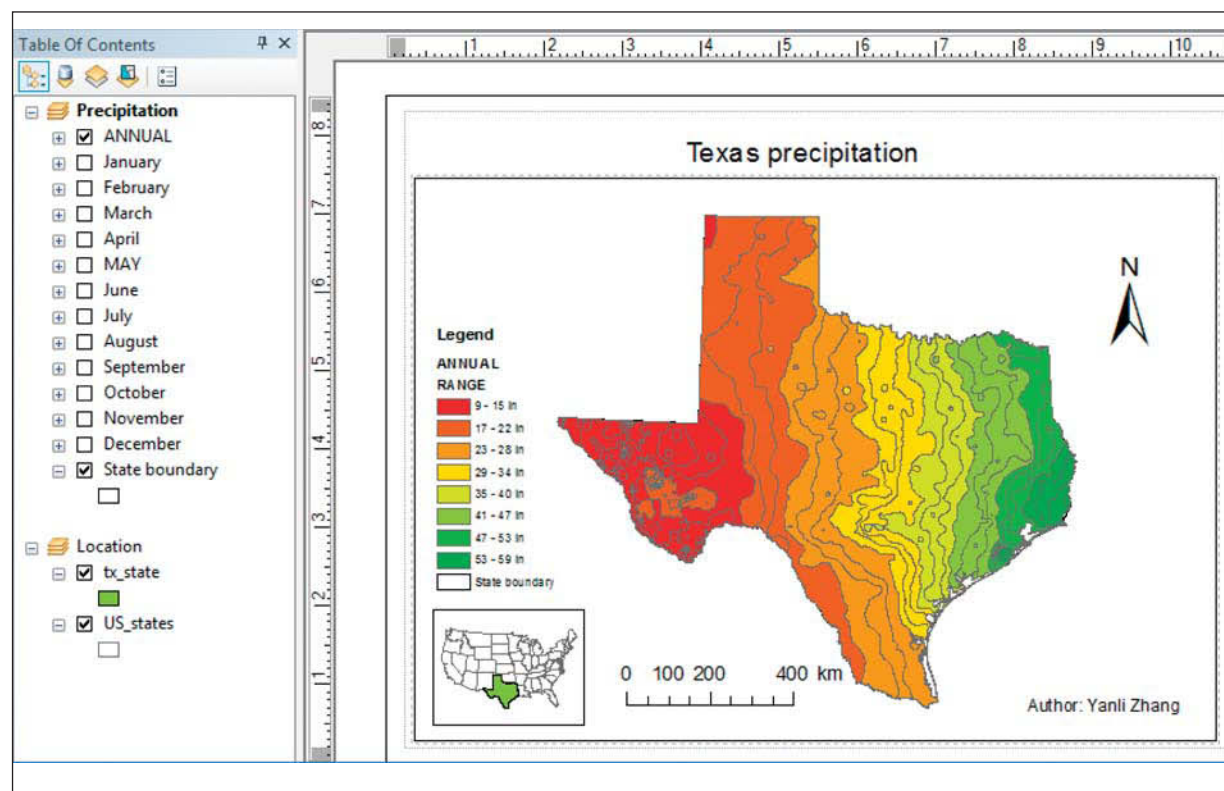


Figure 2 (left): An example map from a layer-driven thematic map book.

Figure 3 (below): ArcPy code to create a layer-driven thematic map book.

When using this code for other layer-driven thematic maps, you need only replace input parameters, such as the input map file name (in the sample code, it is "C:\temp\txPrecipitation.mxd"); data frame name ("Precipitation"); output folder ("C:\temp\output"); output file name ("TX\_precipitation.pdf"); thematic layer name ("ANNUAL," "January," etc.); and map title ("Texas precipitation"). A critical issue is that Python language treats backslash (\) as an escape character. For example, \n represents a line. For a path to a folder, a forward slash (/) can be used in place of a backslash. Our sample code uses the letter "r" before the path to make sure Python interprets it correctly.

## An Attribute Field-Driven Map Book

An example of an attribute field-driven thematic map book is shown in Figure 5. The pine layer's attribute table has about 140 fields to represent tree diameter in different years. We needed to create figures (in this case, JPEG images instead of the default PDF files) to represent each year's diameter. To process this manually, you would need to access layer properties 140 times, change the symbology based on a different attribute field, change the title, then export each map separately. We developed ArcPy code (Figure 6) to automate the procedure. This code can only be used in ArcMap 10.1.

When using this code for other attribute field-driven thematic maps, you'll need to replace input parameters, such as input map file name (in the sample code, it is "C:\temp\pine.mxd"); data frame name ("Layers"); output folder ("C:\temp\output"); thematic layer name ("pine"); and map title ("Plot 4 Year"). For the

```
import arcpy, os

#Specify the input map document and the data frame
mxd = arcpy.mapping.MapDocument("C:\temp\txPrecipitation.mxd")
df = arcpy.mapping.ListDataFrames(mxd, "Precipitation")[0]

#Specify output path and final output PDF
outPath = r"C:\temp\output\\"
finalPdf = arcpy.mapping.PDFDocumentCreate(outPath + "TX_precipitation.pdf")

# Check to see if the PDF file exists, if it does, delete it
if arcpy.Exists(finalPdf):
    arcpy.Delete_management(finalPdf)

#turn off all layers except State boundary in the data frame
layerList = arcpy.mapping.ListLayers(mxd, "", df)
for mapLayer in layerList:
    if mapLayer.name != "State boundary":
        mapLayer.visible = False
arcpy.RefreshTOC()

#Specify the thematic layers' names
lyrNameList = ["ANNUAL", "January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December"]

for lyrName in lyrNameList:
    #Turn on the specific layer
    lyr = arcpy.mapping.ListLayers(mxd, lyrName, df)[0]
    lyr.visible = True

    # Change title for each map
    for elm in arcpy.mapping.ListLayoutElements(mxd, "TEXT_ELEMENT"):
        if elm.text.find("precipitation") >= 0:
            elm.text = "Texas " + lyrName + " precipitation"

    #Export each theme to a temporary PDF and then append to the final PDF
    tmpPdf = outPath + lyrName + ".pdf"
    arcpy.mapping.ExportToPDF(mxd, tmpPdf)
    finalPdf.appendPages(tmpPdf)

#Turn off current layer
lyr.visible = False
del lyr, tmpPdf

del mxd, df, finalPdf
```

example, we used the year number as the attribute field, thus we can use a *while* loop to go through every field (year). Another option is using a list of attribute field names. You need to specify the class break values and labels based on the specific thematic input data.

The sample code used in this article can be downloaded from [www.faculty.sfasu.edu/zhangy2/download.htm](http://www.faculty.sfasu.edu/zhangy2/download.htm). For questions about the code, please contact Yanli Zhang, [zhangy2@sfasu.edu](mailto:zhangy2@sfasu.edu).

Yanli Zhang is an assistant professor; Daniel Unger is an associate professor; Jason Grogan, CF, is a research specialist, Forest Resources Institute; Brian P. Oswald is Denman Distinguished Professor; and John Lanham is a graduate student at Stephen F. Austin State University, Nacogdoches, Texas.

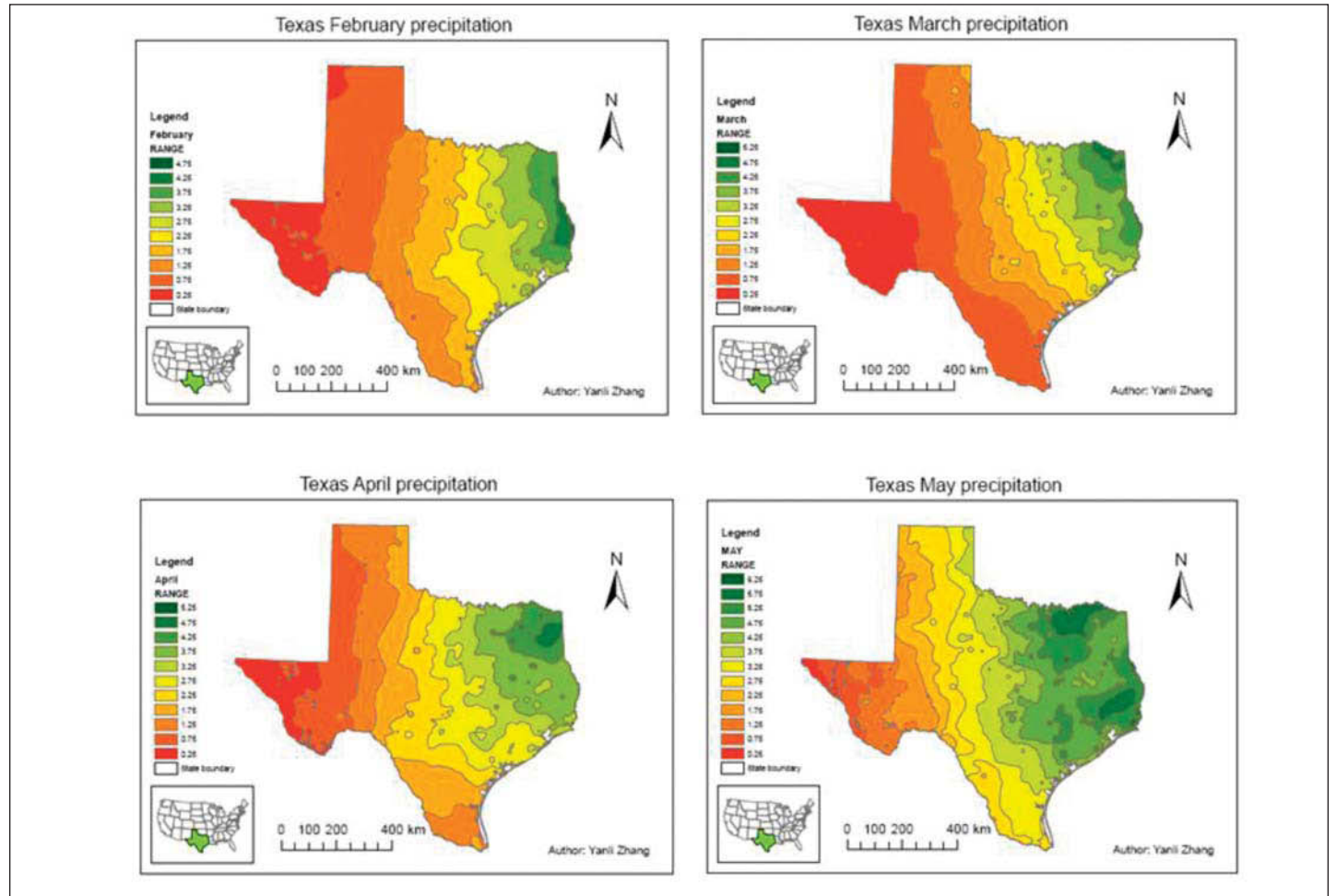
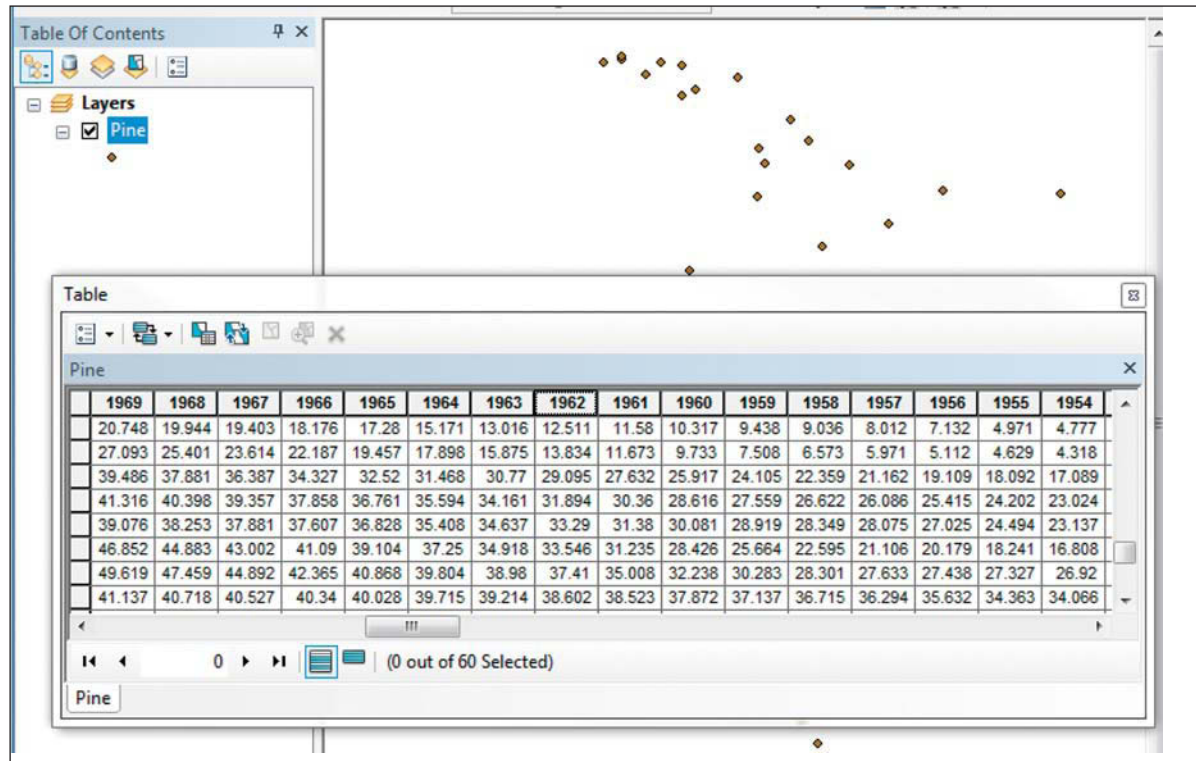


Figure 4 (right): A demonstration of the output of the layer-driven thematic map book.

Figure 5 (below): An attribute field-driven thematic map.

Figure 6 (below, right): ArcPy code to create an attribute field-driven thematic map book.



```
import arcpy, os

#Specify the map document and the data frame
mxd = arcpy.mapping.MapDocument(r"C:\temp\pine.mxd")
df = arcpy.mapping.ListDataFrames(mxd, "Layers")[0]

outPath = r"C:\temp\output\\" #output file folder

#Year (2004, 2003, ...) is used as attribute field name
#Also it is used to control the loop
year = 2004

while year >= 1960: # 1960 is the last year in the attribute table
    for maplayer in arcpy.mapping.ListLayers(mxd):
        if maplayer.name == "Pine":
            #change the symbology
            if maplayer.symbologyType == "GRADUATED_SYMBOLS":
                maplayer.symbology.valueField = str(year)
                maplayer.symbology.classBreakValues = [0, 20.0, 30.0, 40.0, \
50.0, 60.0, 70.0, 80.0, \
90.0, 100.0, 110.0, 120.0, \
130.0, 140.0, 150.0]

                maplayer.symbology.classBreakLabels = ["0 to 19.99", "20 to 29.99", "30 to 39.99", \
"40 to 49.99", "50 to 59.99", "60 to 69.99", \
"70 to 79.99", "80 to 89.99", "90 to 99.99", \
"100 to 109.99", "110 to 119.99", \
"120 to 129.99", "130 to 139.99", \
"140 to 150"]

            # Change title for each map
            for elm in arcpy.mapping.ListLayoutElements(mxd, "TEXT_ELEMENT"):
                if elm.text.find("Plot 4 Year") >= 0:
                    elm.text = "Plot 4 Year " + str(year)

            #export each map as a jpeg file. Also we can use ExportToPDF method.
            nameJPEG = outPath + str(year) + ".jpg" #specify each output file name
            arcpy.mapping.ExportToJPEG(mxd, nameJPEG)
            year = year - 1

del mxd, df
```

## SAF Policy Team's 2012 Election Analysis

After more than two years of campaigning, the result of the 2012 election was essentially a draw. The Republicans maintain a majority in the House of Representatives, the Democrats continue to control the Senate, and President Barack Obama returns to the White House for a second term. As of Thursday, November 8, the Senate reported a net gain of two seats by the Democrats (North Dakota and Indiana) and one Independent (Maine). The Democrats also captured two seats in the House previously held by Republicans, but the final tally in the House is still in flux, with seven races still undecided at the time this issue went to press.

What do these results mean for SAF?

The leadership in the House and Senate will remain intact, and SAF's policy team will encounter many of the same challenges faced over the last two years,

including the difficulties involved with moving policy initiatives forward, particularly stand-alone bills. However, the turnover also provides an opportunity for outreach, and in January and February the SAF policy team will be busy educating new legislators about SAF and its policy priorities. The 80 new members of the House and 13 new senators will arrive in Washington in January. The policy team will identify new members with a stake in forestry and natural resources, and will work in conjunction with state and local chapters to conduct targeted outreach to those new members and their staffs.

We also expect the President to continue pursuit of a natural-resources agenda similar to that of the last four years, with an emphasis placed on domestic energy development and production. However, a similar set of goals does

not guarantee that the same individuals will be implementing the strategy. Secretary of State Hillary Clinton's announcement that she will step aside in January reminds us that leadership changes are not uncommon during a second term. The policy team is not aware of pending leadership changes at the US Department of Agriculture, the Department of the Interior, or the Environmental Protection Agency, but we are prepared to engage new agency contacts and political appointees when necessary.

Most important, the conclusion of the election season means that the policy team will have to be ready to act quickly in the next two months of this "lame duck" Congress. We already are aware of preliminary, bipartisan discussions about how to resolve the looming budget sequestration, also known as the "fiscal cliff," which would result in mandatory

cuts to federal spending across the board, coupled with the expiration of a number of tax-code exemptions. We also are aware that the building momentum in the House to pass a one-year extension of the 2008 Farm Bill has met intense opposition in the Senate.

The policy team will be prepared to act based on information we receive from our contacts in Congress and coalition partners. Meetings scheduled with congressional and committee staff after they return from the election recess will allow the policy team to remind our contacts of SAF policy priorities, discuss how we can work together to achieve these goals, and keep SAF members and Council abreast of the latest developments on Capitol Hill.

For more information contact John Barnwell, [barnwellj@safnet.org](mailto:barnwellj@safnet.org), or Kelsey Delaney, [delaneyk@safnet.org](mailto:delaneyk@safnet.org).