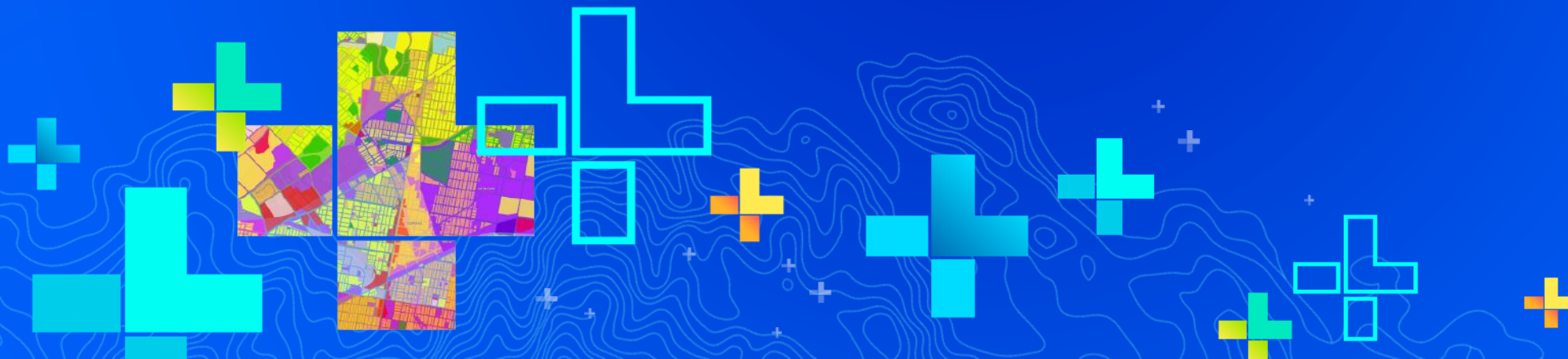




Automate Publishing of Web Layers Using ArcPy

Shilpi Jain

SEE
WHAT
OTHERS
CAN'T



Why automated?

- Faster
- Reliable
- Integrate

ArcPy Sharing Tools



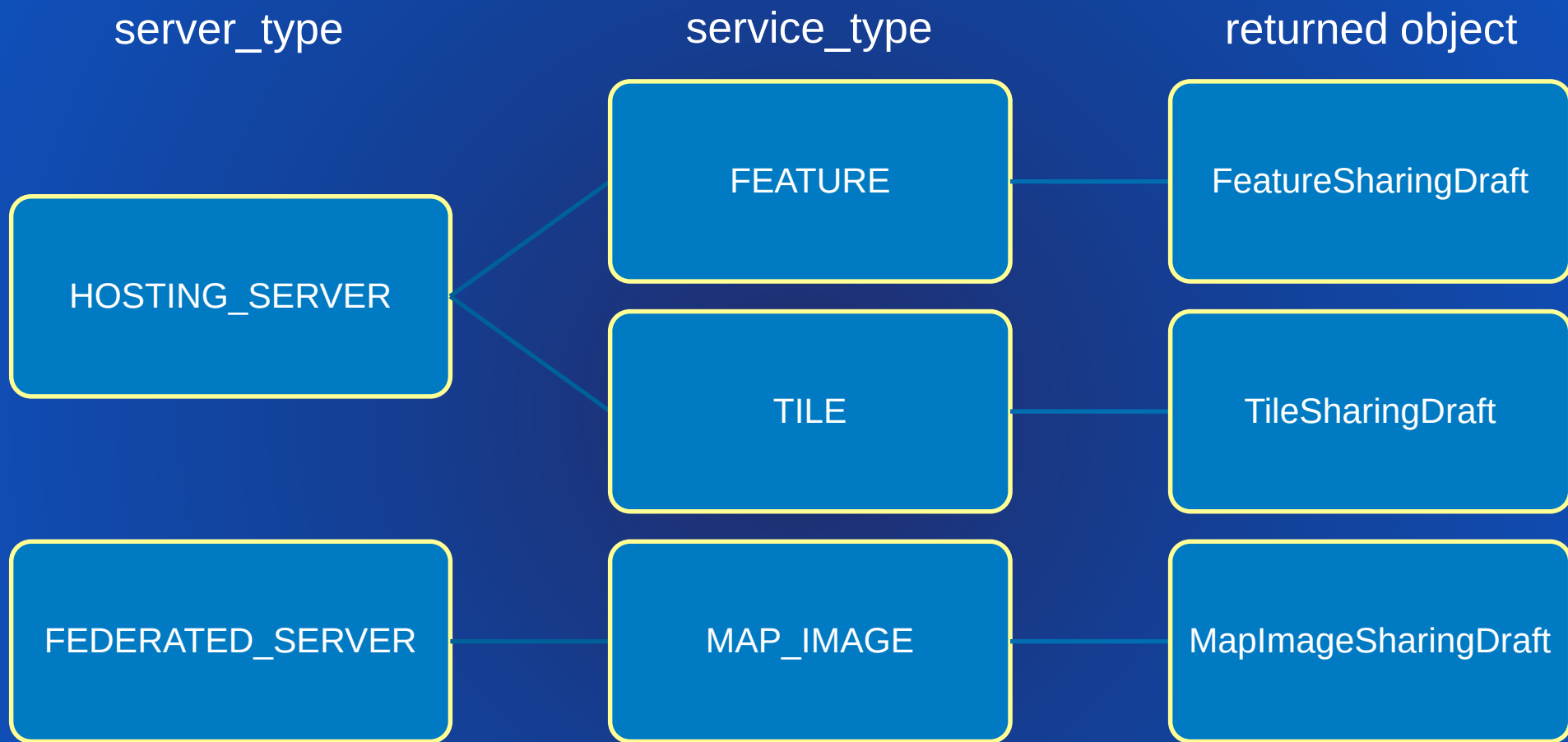
1. getWebLayerSharingDraft

Python

```
map. getWebLayerSharingDraft (server_type, service_type, service_name, {layers_and_tables})
```

Returns a class object based on service_type selected.

<http://pro.arcgis.com/en/pro-app/arcpy/mapping/map-class.htm>



Properties of returned object



Methods of returned object

`exportToSDDraft()` – returns the location and name of sddraft

Demo 1: Sharing to ArcGIS Online




```

''' the first step to automate the publishing of a map, layer, or list of layers to a hosted web layer using ArcPy, in new object-oriented approach.

    Use "getWebLayerSharingDraft" method to create a TileSharingDraft object (reference: http://pro.arcgis.com/en/pro-app/arcpy/sharing/tilesharingdraft-class.htm)
    Syntax = getWebLayerSharingDraft (server_type, service_type, service_name, {map or layers}, {...})

    Then to a Service Definition Draft (.sddraft) file with "exportToSDDraft" method.
    Syntax = exportToSDDraft (out_sddraft)
'''
# Create TileSharingDraft and set service properties
sharing_draft = m.getWebLayerSharingDraft("HOSTING_SERVER", "TILE", serviceName, lyr)
sharing_draft.portalUrl = portalURL
sharing_draft.summary = "My Summary"
sharing_draft.tags = "My Tags"
sharing_draft.description = "My Description"
sharing_draft.credits = "My Credits"
sharing_draft.useLimitations = "My Use Limitations"
sharing_draft.exportToSDDraft(sddraftPath)
print("Exported SDDraft")

''' The Service Definition Draft can then be converted to a fully consolidated Service Definition (.sd) file using the Stage Service tool.
    Staging compiles all the necessary information needed to successfully publish the GIS resource.
    Syntax = StageService_server (in_service_definition_draft, out_service_definition, staging_version)
'''
arcpy.StageService_server(sddraftPath, sdPath)
print("Created SD")

''' Finally, the Service Definition file can be uploaded and published as a GIS service to a specified online organization using the Upload Service Definition tool.
    This step takes the Service Definition file, copies it onto the server, extracts required information, and publishes the GIS resource.
    Syntax = UploadServiceDefinition_server (in_sd_file, in_server, {in_service_name}, {in_cluster}, {in_folder_type},
    {in_folder}, {in_startupType}, {in_override}, {in_my_contents}, {in_public}, {in_organization}, {in_groups})
'''
arcpy.UploadServiceDefinition_server(sdPath, 'My Hosted Services', in_override = "OVERRIDE_DEFINITION", in_public = "PUBLIC", in_organization = "SHARE_ORGANIZATION")
print("Uploaded and Shared SD")

```


2. Stage Service

Python

```
StageService_server (in_service_definition_draft, out_service_definition, {staging_version})
```

<https://pro.arcgis.com/en/pro-app/tool-reference/server/stage-service.htm>

3. Upload Service Definition

Python

```
UploadServiceDefinition_server (in_sd_file, in_server, {in_service_name}, {in_cluster}, {in_folder_type},  
{in_folder}, {in_startupType}, {in_override}, {in_my_contents}, {in_public}, {in_organization}, {in_groups})
```

<https://pro.arcgis.com/en/pro-app/tool-reference/server/upload-service-definition.htm>

server_type

- “My Hosted Services” (Hosted Web Layer)
- “HOSTING_SERVER” (Hosted Web Layer)
- ArcGIS Federated Server URL (Map Image Layer)
- ArcGIS Server Connection
- ArcGIS Server URL

4. Sign In To Portal

```
Python ? ▾ □ ×  
  
SignInToPortal (portal_url, username, password)  
  
http://pro.arcgis.com/en/pro-app/arcpy/functions/signintportal.htm
```

When to use?

If the script is to be run outside Pro.



Demo 2: Sharing to Enterprise



```
#region publish
''' the first step to automate the publishing of a map, layer, or list of layers to a hosted web layer using ArcPy, in new object-oriented approach.

    Use "getWebLayerSharingDraft" method to create a MapImageSharingDraft object (reference: http://pro.arcgis.com/en/pro-app/arcpy/sharing/mapimagesharingdraft-class.htm)
    Syntax = getWebLayerSharingDraft (server_type, service_type, service_name, {map or layers}, {...})

    Then to a Service Definition Draft (.sddraft) file with "exportToSDDraft" method.
    Syntax = exportToSDDraft (out_sddraft)
'''
# Create MapImageSharingDraft and set service properties
sharing_draft = m.getWebLayerSharingDraft("FEDERATED_SERVER", "MAP_IMAGE", serviceName, lyr) # Creates a MapImageSharingdraft class object
sharing_draft.federatedServerUrl = svrTiles
sharing_draft.portalFolder = 'sj'
sharing_draft.copyDataToServer = True # Need to register db first if set to False
sharing_draft.summary = "My Summary"
sharing_draft.tags = "My Tags"
sharing_draft.description = "My Description"
sharing_draft.credits = "My Credits"
sharing_draft.useLimitations = "My Use Limitations"
sharing_draft.exportToSDDraft(sddraftPath) # Need to crack open the sddraft to enable FS
print("Exported SDDraft")

''' The Service Definition Draft can then be converted to a fully consolidated Service Definition (.sd) file using the Stage Service tool.
    Staging compiles all the necessary information needed to successfully publish the GIS resource.
    Syntax = StageService_server (in_service_definition_draft, out_service_definition, staging_version)
'''
arcpy.StageService_server(sddraftPath, sdPath)
print("Created SD")

''' Finally, the Service Definition file can be uploaded and published as a GIS service to a specified online organization using the Upload Service Definition tool.
    This step takes the Service Definition file, copies it onto the server, extracts required information, and publishes the GIS resource.
    Syntax = UploadServiceDefinition_server (in_sd_file, in_server, {in_service_name}, {in_cluster}, {in_folder_type},
    {in_folder}, {in_startupType}, {in_override}, {in_my_contents}, {in_public}, {in_organization}, {in_groups})
'''
arcpy.UploadServiceDefinition_server(sdPath, svrTiles)
print("Uploaded and Shared SD")
#endregion
```

Conclusion

Why Automate

- Increase efficiency
- Eliminate human error
- Integrate with other GP tools

How to Automate

- `arcpy.SignInToPortal`
- `arcpy.mp.getWebLayerSharingDraft`
 - `arcpy.StageService_server`
- `arcpy.UploadServiceDefinition_server`

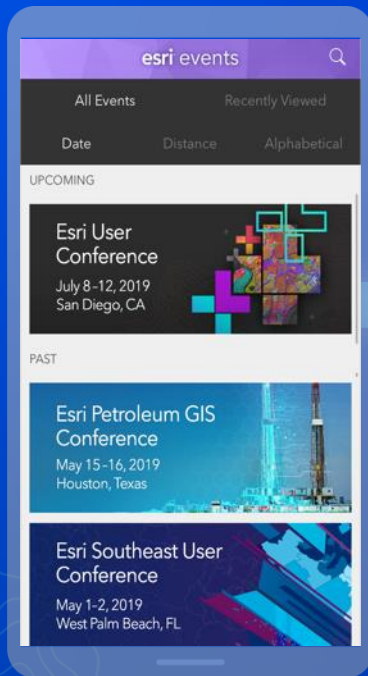


Github Link

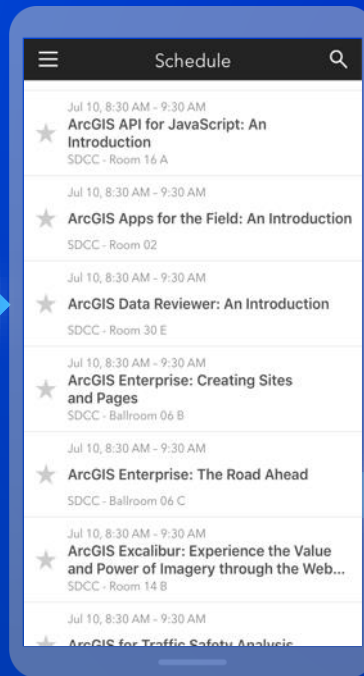
<https://github.com/jnshill/ArcPyPublishing>

Please Share Your Feedback in the App

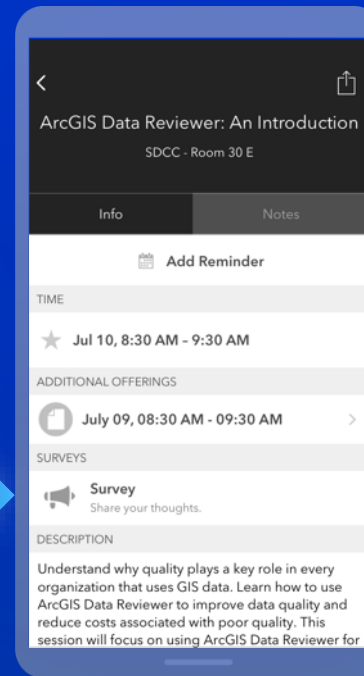
Download the Esri Events app and find your event



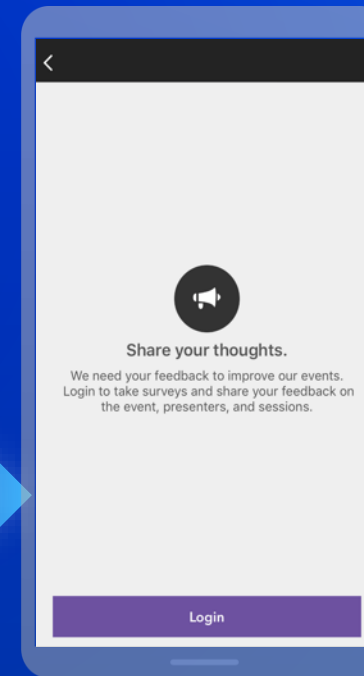
Select the session you attended



Scroll down to "Survey"



Log in to access the survey



Complete the survey and select "Submit"

