



Network Analysis with Python

Deelesh Mandloi

Slides and code samples from this demo theater

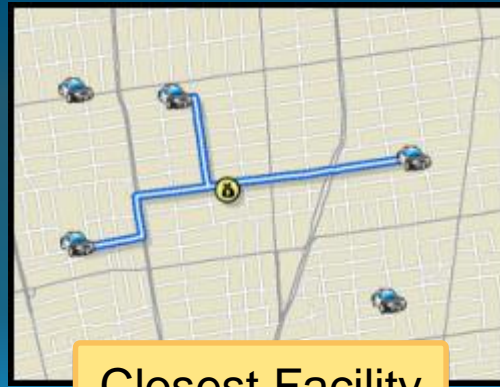
<http://esriurl.com/uc15nawpy>

Topics

- **ArcGIS Network Analyst extension and concepts**
- **Network analysis using ArcGIS Online**
- **Network analysis using ArcGIS Desktop**
 - **Network Analyst Python Module (arcpy.na)**



Route



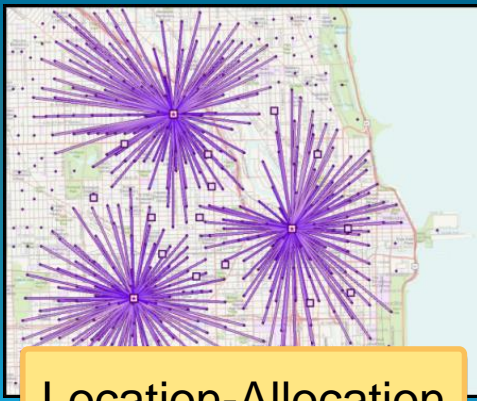
Closest Facility



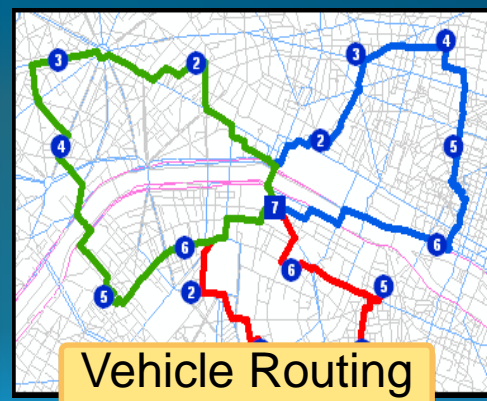
Service Area

ArcGIS Network Analyst

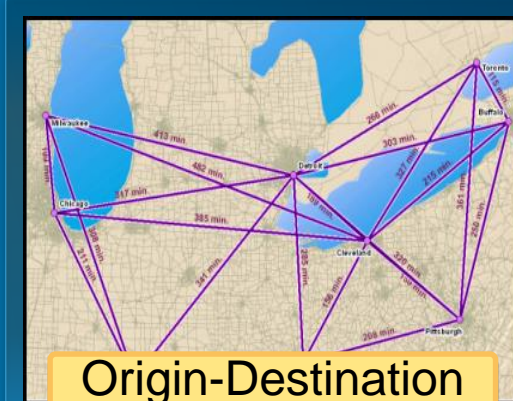
Solving transportation problems



Location-Allocation



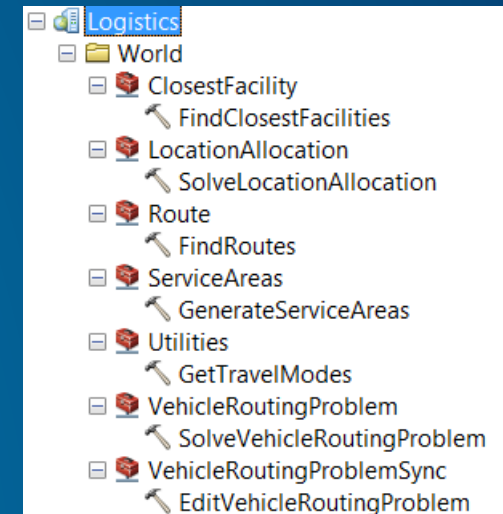
Vehicle Routing Problem



Origin-Destination Cost Matrix

Network Analysis using ArcGIS Online services

- Ready-to-use services published by Esri that are run in an Esri-administered cloud infrastructure
- Requires ArcGIS Online subscription with service credits
- No need to have an Network Analyst Extension license.
- No need for your own street data modeled as a network dataset



Using Online services from Python

- Import the appropriate toolbox passing the credentials
 - Include credentials in the script
 - Store credentials in ArcGIS Server connection file
 - Use the token of the user signed into ArcGIS Desktop
- Call the geoprocessing tool from the imported toolbox

```
import arcpy
import time
import sys

username = "<your user name>"
password = "<your password>"
sa_service = "http://logistics.arcgis.com/arcgis/services/World/ServiceAreas;{0};{1}".format(username, password)

#Add the geoprocessing service as a toolbox. Use an alias when importing
arcpy.ImportToolbox(sa_service, "agol")

#Set the variables to call the tool
facilities = r"C:/data/Inputs.gdb/Stores"
output_service_areas = "C:/data/Results.gdb/StoreServiceAreas"

#Call the tool
result = arcpy.agol.GenerateServiceAreas(facilities, "5 10 15", "Minutes")

#Check the status of the result object every 0.5 seconds
#until it has a value of 4(succeeded) or greater
while result.status < 4:
    time.sleep(0.5)

#print any warning or error messages returned from the tool
result_severity = result.maxSeverity
if result_severity == 2:
    print "An error occurred when running the tool"
    print result.getMessages(2)
    sys.exit(2)
elif result_severity == 1:
    print "Warnings were returned when running the tool"
    print result.getMessages(1)

#Get the output routes and save to a local geodatabase
result.getOutput(0).save(output_service_areas)
```

Demo

Service Areas using ArcGIS Online

Find 2, 4, 6 minutes travel
time areas around stores
using driving time and
walking time travel modes

```
'''This tool creates service areas around points us

import os
import sys
import time

import arcpy

def import_service(service_name, username="", passw
    '''Imports the service toolbox based on the spe

    #Construct the connection string to import the
    if username and password:
        tbx = "http://logistics.arcgis.com/arcgis/s
    elif ags_connection_file:
        tbx = "{0};{1}".format(ags_connection_file,
    elif token and referer:
        tbx = "http://logistics.arcgis.com/arcgis/s
    else:
        raise arcpy.ExecuteError("No valid option s

    #Import the service toolbox
    tbx_alias = "agol"
    arcpy.ImportToolbox(tbx, tbx_alias)

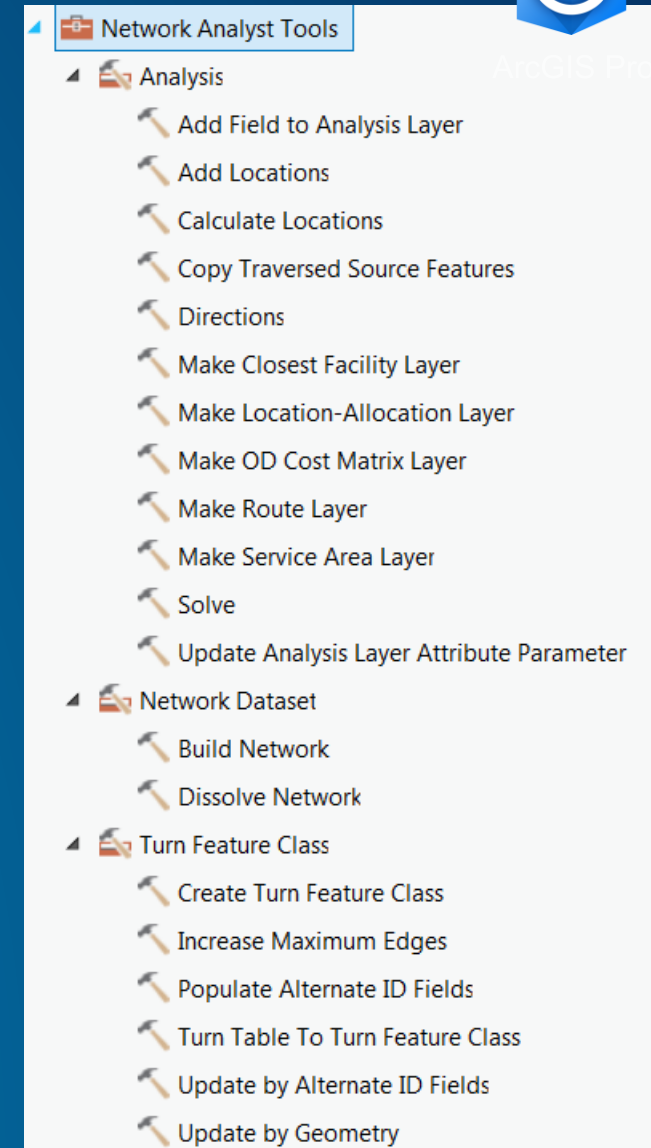
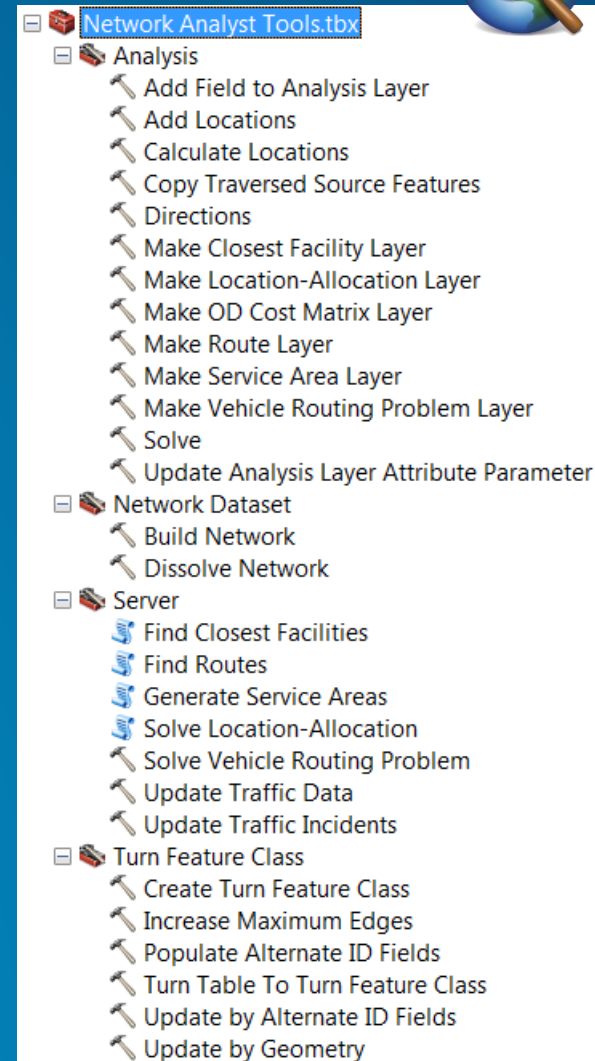
    return getattr(arcpy, tbx_alias)

def main():
    '''Program entry point'''
```

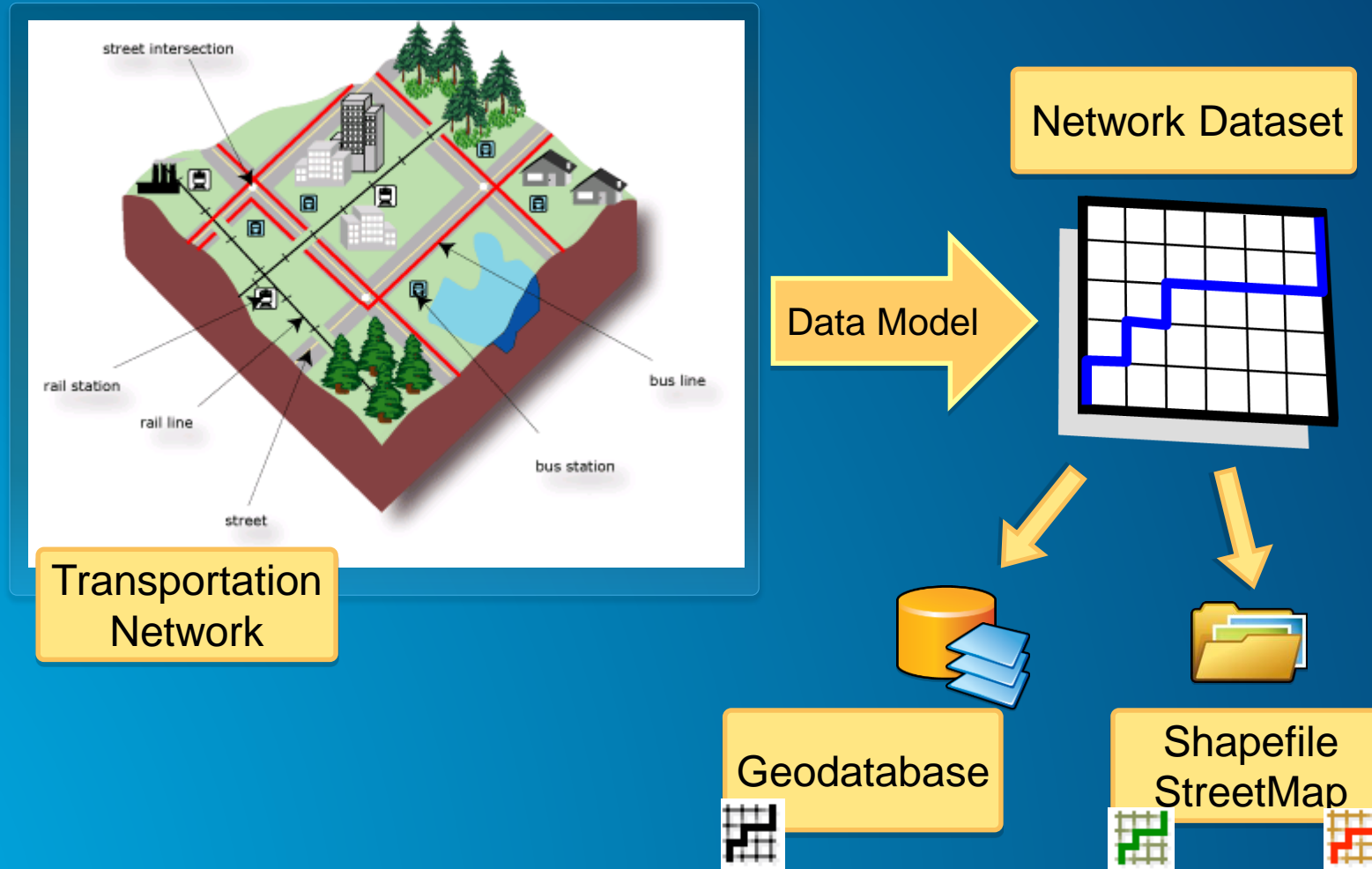
Network Analysis using ArcGIS Desktop



- Perform network analysis by running one or more geoprocessing tools from the Network Analyst Tools toolbox
- Requires ArcGIS Network Analyst Extension license.
- Need your own street data modeled as a network dataset



Network Dataset



Where do you get street data?

- Free data

- TIGER



- Community data

- OpenStreetMap

- OSM to NDS tools



- Your data

- Vendor data

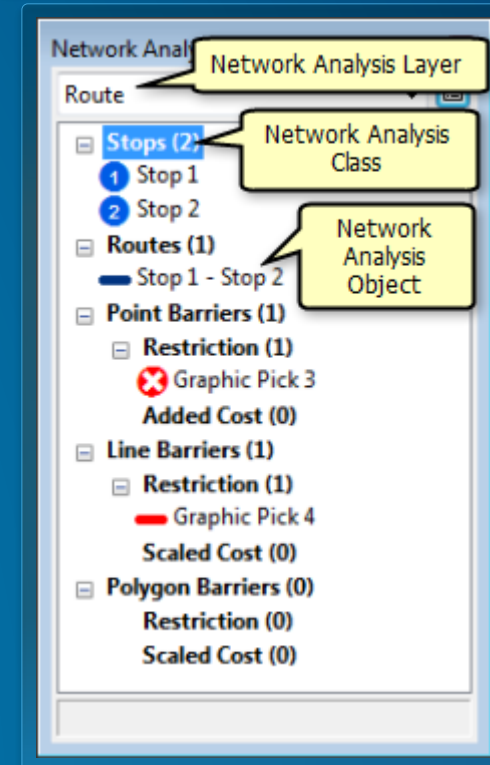
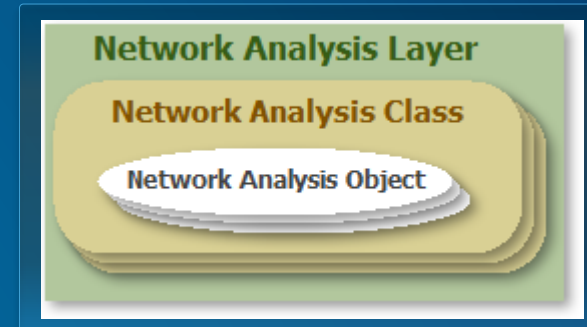
- Street data processing tools

here



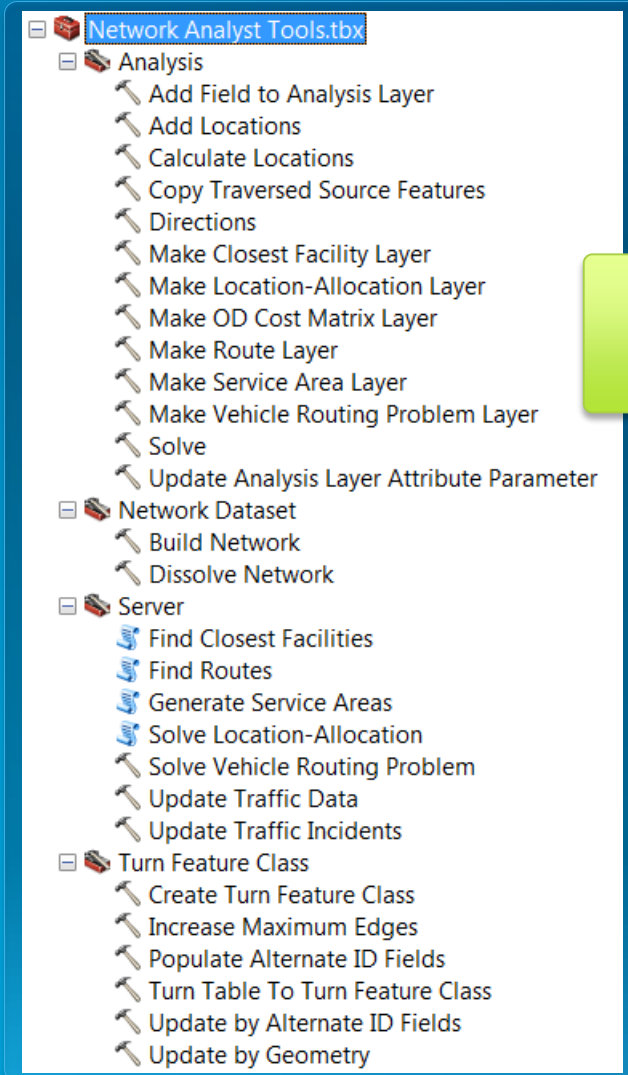
Network Analysis Layer

- Composite layer configured for a specific solver.
- Stores analysis properties, inputs, and outputs from the solver
- Contains **Network Analysis Classes** that store **Network Analysis Objects**

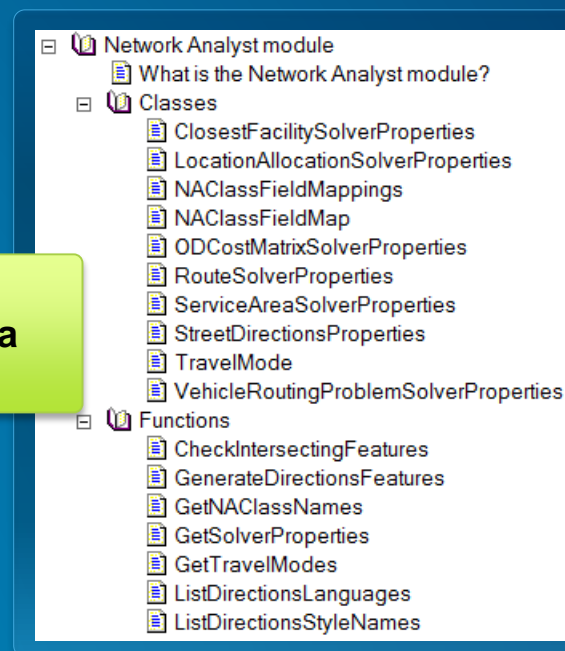


What is the Network Analyst Module?

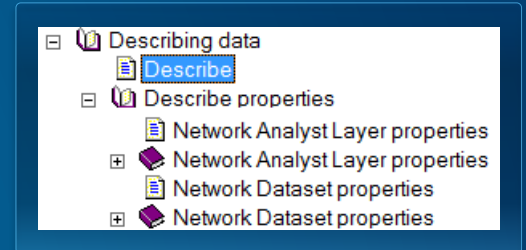
- Simplify access to Network Analyst functionality from Python



arcpy.na

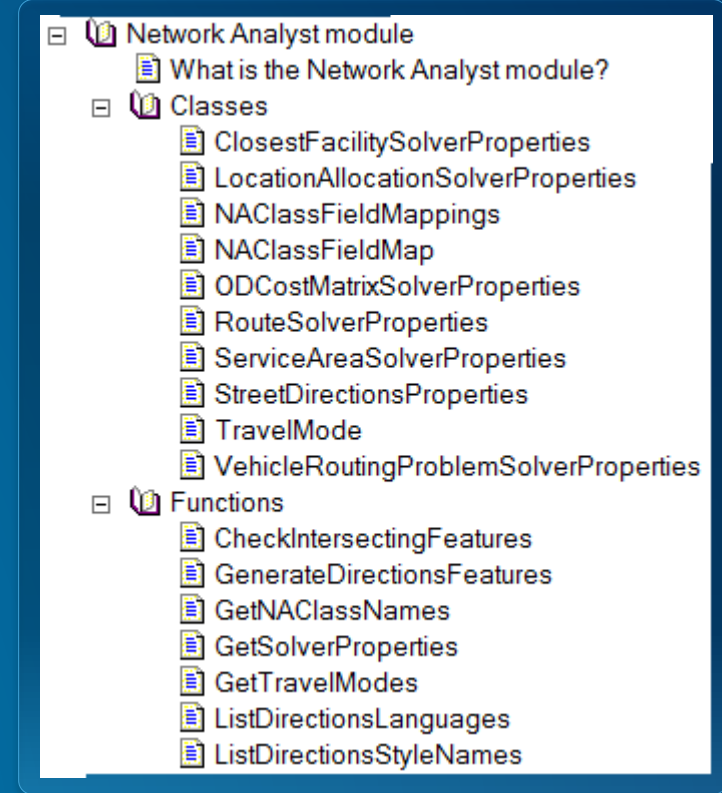


arcpy.Describe



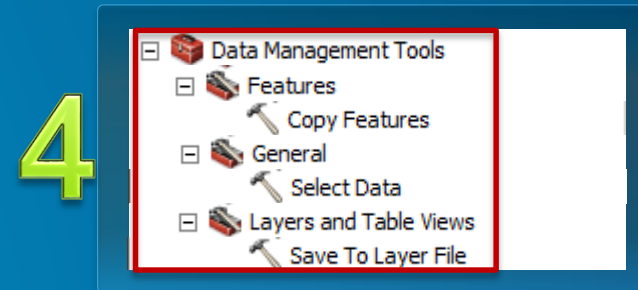
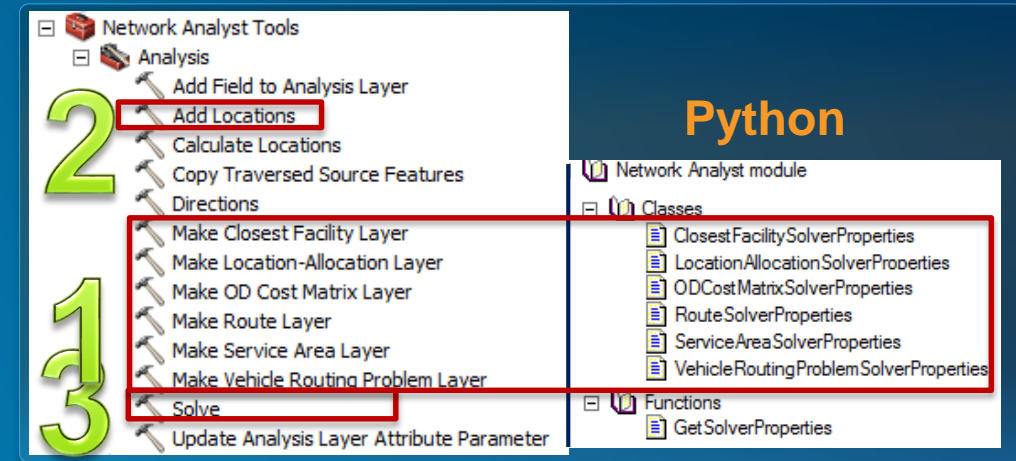
Network Analyst Module

- Support editing analysis properties of network analysis layers
 - No need to re-create layers
 - Speeds up execution
 - Simplifies script logic
 - Automate workflows from Python window
- Provide helper functions and classes to easily use Network Analyst geoprocessing tools from Python



Network Analysis Workflow

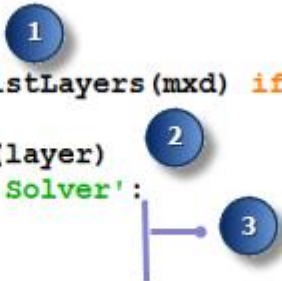
1. Make or Edit Network Analysis Layer
2. Add locations to one or more Network Analysis Classes
3. Solve
4. Use the results



Editing Network Analysis Layer Properties

- Edit analysis layers using functions and classes in `arcpy.na` module
- Common Steps
 1. Get reference to a network analysis layer object
 2. Get the solver properties object
 3. Update the properties

```
mxdoc = arcpy.mapping.MapDocument("CURRENT")  
naLayers = [layer for layer in arcpy.mapping.ListLayers(mxdoc) if layer.isNetworkAnalystLayer]  
for layer in naLayers:  
    solverProps = arcpy.na.GetSolverProperties(layer)  
    if solverProps.solverName == 'Service Area Solver':  
        solverProps.useHierarchy = True  
        solverProps.defaultBreaks = [1,2]
```



Using Network Analyst GP tools from Python

- Helper classes for complex parameter types
 - Easily specify field mappings in Add Locations tool by using `arcpy.na.NAClassFieldMappings`
- Write scripts that work across ArcGIS language versions
 - Avoid using localized strings in scripts such as sublayer names by using `arcpy.na.GetNAClassNames`

```
polygonBarriersNAClass = "Polygon Barriers"
barrierFieldMappings = "Name # Precipitation; BarrierType # 1; " + "Attr_%s %s #" % (impedance,
                                                                                       scaleFactorField)

arcpy.AddMessage("Loading weather polygons as scaled cost barriers")
arcpy.na.AddLocations(routeLayer, polygonBarriersNAClass, weatherPolygonLayer, barrierFieldMappings)
```

ArcGIS 10.0

```
naClasses = arcpy.na.GetNAClassNames(routeLayer)
polygonBarriersNAClass = naClasses['PolygonBarriers']
barrierFieldMappings = arcpy.na.NAClassFieldMappings(routeLayer, polygonBarriersNAClass, False,
                                                       arcpy.ListFields(weatherPolygonLayer))

barrierFieldMappings['Name'].defaultValue = "Precipitation"
barrierFieldMappings['BarrierType'].defaultValue = 1
barrierFieldMappings['Attr_' + defaultImpedance].mappedFieldName = scaleFactorField
arcpy.AddMessage("Loading weather polygons as scaled cost barriers")
arcpy.na.AddLocations(routeLayer, polygonBarriersNAClass, weatherPolygonLayer, barrierFieldMappings)
```

ArcGIS 10.1

ArcGIS Desktop and ArcGIS Pro differences in arcpy.na

- ArcGIS Pro uses Python 3.4 and ArcGIS Desktop uses Python 2.7
 - Can write single script that works across both versions of Python
 - Use **arcpy.AddMessage** instead of **print** statements
- Account for changes in arcpy.mapping module

```
#Get the layer object for the output polygons.
if hasattr(service_area_layer, "listLayers"):
    #When using ArcGIS Pro, get the polygons sub layer using the listLayers
    #method on the service area layer object
    polygons_layer = service_area_layer.listLayers(polygons_layer_name)[0]
else:
    #When using ArcMap, get the polygons sub layer using the ListLayers function in the mapping module
    polygons_layer = arcpy.mapping.ListLayers(service_area_layer, polygons_layer_name)[0]
```

Demo

Service Areas using ArcGIS Desktop

Find 2, 4, 6 minutes travel
time areas around stores
using driving time and
walking time travel modes

```
'''This tool creates service areas around points us

import os
import sys

import arcpy

def main():
    '''Program entry point'''

    #Check out the Network Analyst extension licens
    arcpy.CheckOutExtension("Network")

    #Get the name and version of the product used t
    install_info = arcpy.GetInstallInfo()
    product_version = "{0} {1}".format(install_info

    ##Setup inputs
    #Use the Stores feature class in a file geodataba
    cwd = os.path.dirname(os.path.abspath(__file__))
    input_gdb = os.path.join(cwd, "inputs.gdb")
    facilities = os.path.join(input_gdb, "Stores")
    #Use the Streets_ND network dataset in the Tran
    network_dataset = os.path.join(input_gdb, "Tran
    #Setup other inputs for creating service area 1
    break_values = "2 4 6"
    travel_modes = ["Driving Time", "Walking Time"]
    impedance_attribute = "TravelTime"
    search_tolerance = "20000 Meters"
    service_area_layer_name = "ServiceAreas"
```


Resources

- [ArcGIS Online services in the Network Analyst help](#)
- [Network Analyst tutorial](#)
- [Code samples in Network Analyst tools toolbox](#)

Network Analyst presentations

Tuesday

Wednesday

Thursday

8:30 am		Network Analyst: An Introduction		
9 am				
	Routing in Buildings with 3D Networks in ArcGIS Pro			
10 am		Network Analyst: Automating Workflows with Geoprocessing	Network Analysis with Python	Network Analyst: An Introduction
11 am				
	Using Navigator for ArcGIS			Using GTFS Public Transit Data in ArcGIS
12 pm		Network Analyst: Network Analysis with ArcGIS Online		
1 pm				
		Network Analyst: Creating Network Datasets		Network Analyst: Automating Workflows with Geoprocessing
2 pm	Navigator for ArcGIS: Technical Preview		Network Analyst in ArcGIS Pro	
3 pm	Performing Network Analysis	Network Analyst: Performing Network Analysis		Network Analyst: Creating Network Datasets
4 pm	Network Analysis with ArcGIS Online			

Room 15 A

Room 16 B

Demo Theater 10 Apps

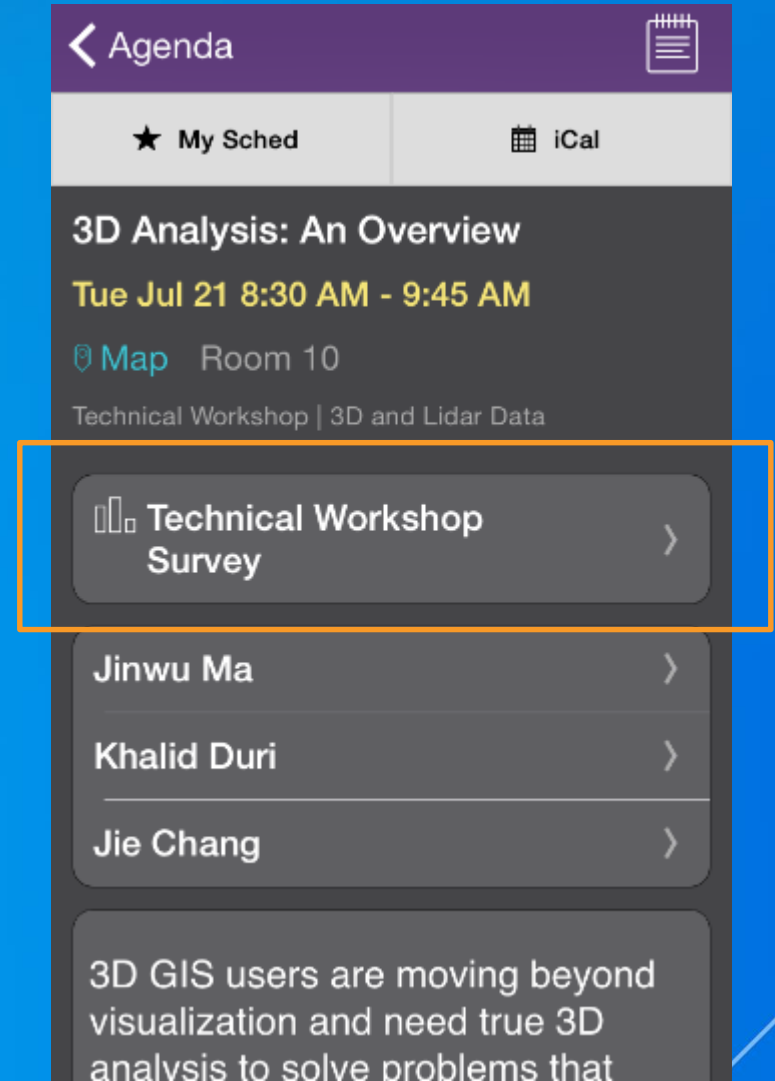
Demo Theater 13 Spatial Analysis

Tech Theater 15 Exhibit Hall A

Tech Theater 17 Exhibit Hall A

Thank you...

- Please fill out the session survey in your mobile app
- Select [enter session title here] in the Mobile App
 - Use the Search Feature to quickly find this title
- Click “Technical Workshop Survey”
- Answer a few short questions and enter any comments





Understanding our world.