



ArcGIS Network Analyst: Automating Workflows with Geoprocessing

Melinda Morang
Patrick Stevens

- **Who are we?**

- **Network Analyst Product Engineers**



- **Who are you?**

- **Current Network Analyst users?**
- **Current geoprocessing users?**
- **Have made geoprocessing models?**
- **Have made geoprocessing Python scripts?**



Topics to be covered

- **ArcGIS Network Analyst extension concepts**
- **Geoprocessing framework for network analysis**
- **ModelBuilder: Models and model tools**
- **Python: Scripts and script tools**
- **Ready-to-Use services**
- **Support and resources**

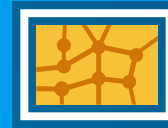


ArcGIS Network Analyst extension concepts

ArcGIS Help:
[What is the ArcGIS Network Analyst extension](#)

Topics to be covered

- **ArcGIS Network Analyst extension concepts**
- Geoprocessing framework for network analysis
- ModelBuilder: Models and model tools
- Python: Scripts and script tools
- Ready-to-Use services
- Support and resources



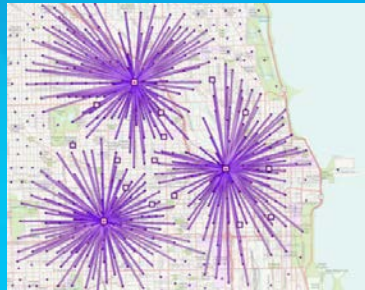
ArcGIS Network Analyst Extension does transportation analysis

Coverage

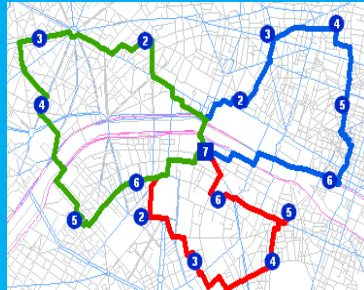


Service Area

Optimization

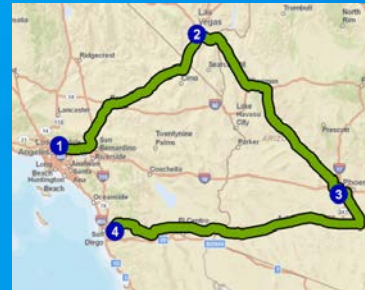


Location-Allocation

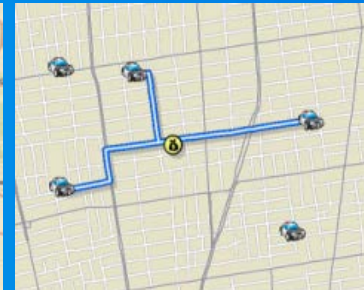


Vehicle Routing Problem

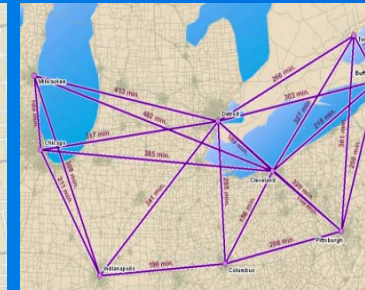
Point-to-point routing



Route

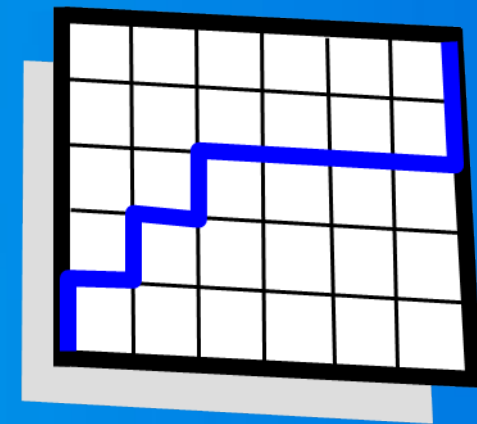
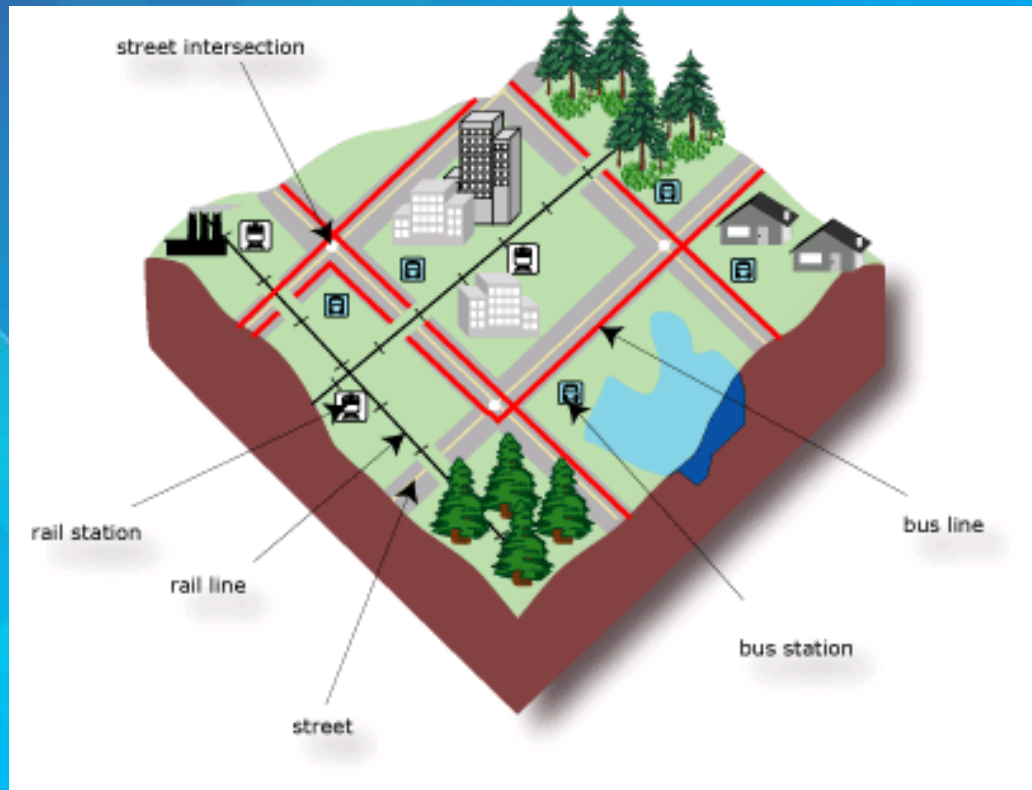


Closest Facility



Origin-Destination Cost Matrix

Network Dataset



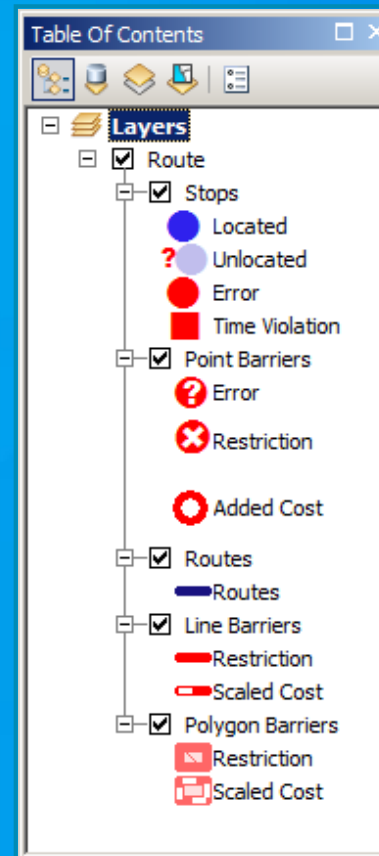


Street Data

- **Data within your organization**
- **Free data**
 - Data and maps media (prior to 10.3)
 - TIGER (Census data)
 - OpenStreetMap
 - OSM to NDS tools
 - ArcGIS Editor for OpenStreetMap
- **Pay for data**
 - HERE or TomTom
 - Vendor street data processing tools
 - StreetMap Premium for ArcGIS
- **Pay for analysis**
 - ArcGIS.com Map Viewer
 - ArcGIS.com Network Services

Network Analyst Layer

- Composite layer
- One layer type for each solver
- Analysis properties
- Inputs
- Outputs



Demo



Network Analyst workflow

Performing an analysis manually

Takeaways

Demo: Network Analyst Workflow

- Steps for network analysis:
 - **Make Layer**
 - **Add Locations**
 - **Solve**
 - **Work with results**



What is Geoprocessing?

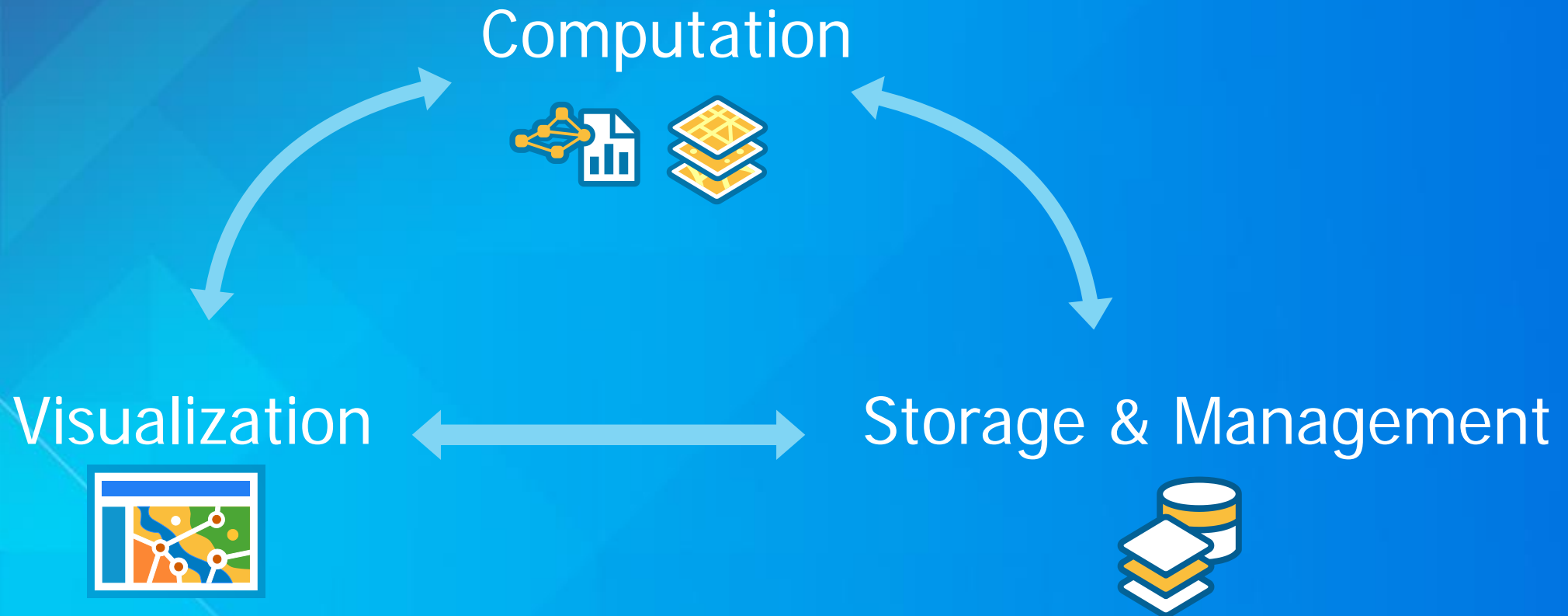
ArcGIS Help:
[The geoprocessing framework](#)

Topics to be covered

- ArcGIS Network Analyst extension concepts
- **Geoprocessing framework for network analysis**
- ModelBuilder: Models and model tools
- Python: Scripts and script tools
- Ready-to-Use services
- Support and resources



What is Geoprocessing?



What is Geoprocessing?

Computation



=

Geoprocessing

- Automating workflows



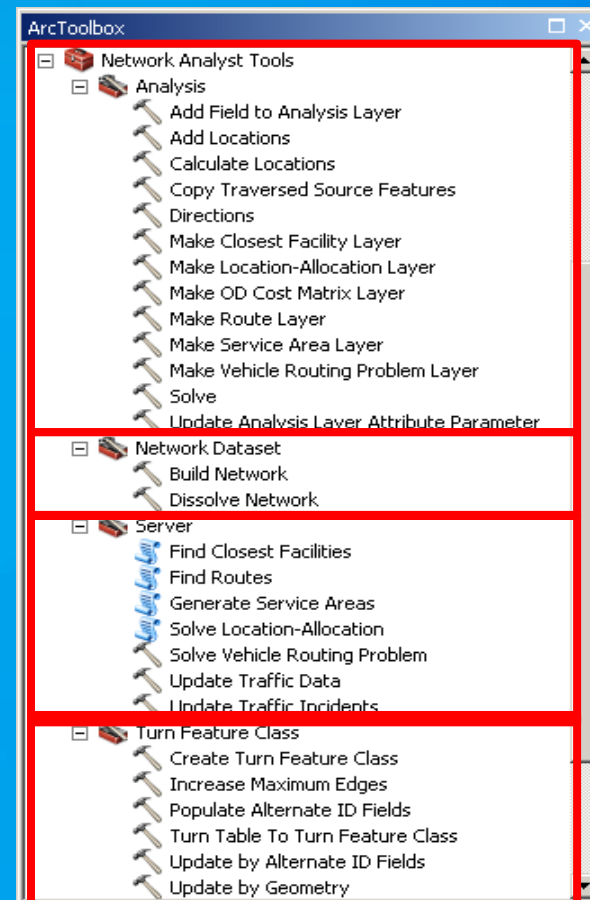
- Modeling & Analysis



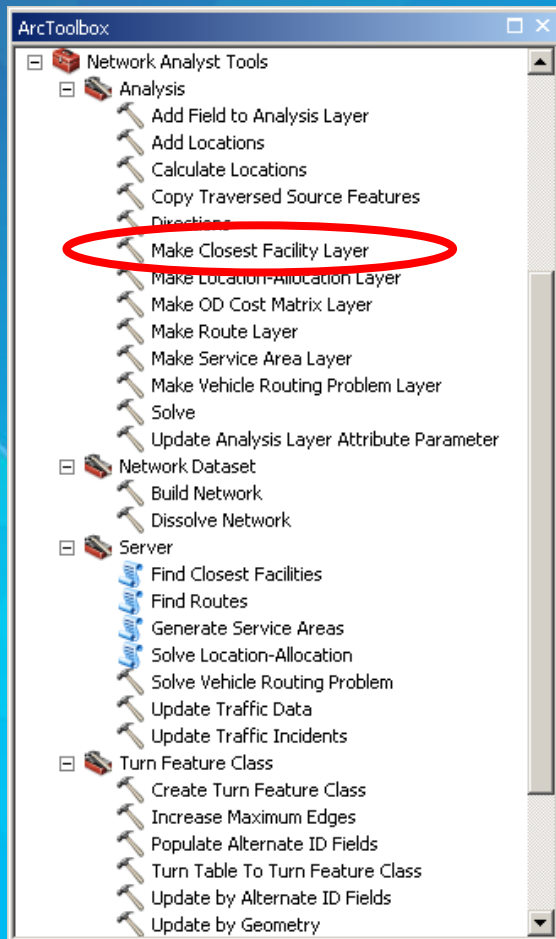
Using Geoprocessing

How?

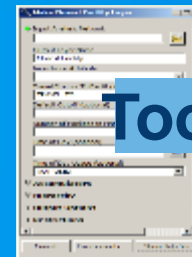
- Performing Network Analysis
- Building networks
- Publishing services
- Managing turns



Using Geoprocessing



Single tool



Tool dialog

```
Python  
>>> arcpy.na.MakeClosestFacilityLayer("Streets_ND", "Nearest ATMs", "TravelTime",  
MakeClosestFacilityLayer_na(in_network_dataset, out_network_analysis_layer_name,  
{travel_from_to}, {default_outOff}, {default_number_facilities_to_find},  
{accumulate_attribute_name;accumulate_attribute_name...}, {UTurn_policy},  
{restriction_attribute_name;restriction_attribute_name...}, {hierarchy},  
{output_path_shape}, {time_of_day}, {time_of_day_usage})
```

Python window

Chain tools

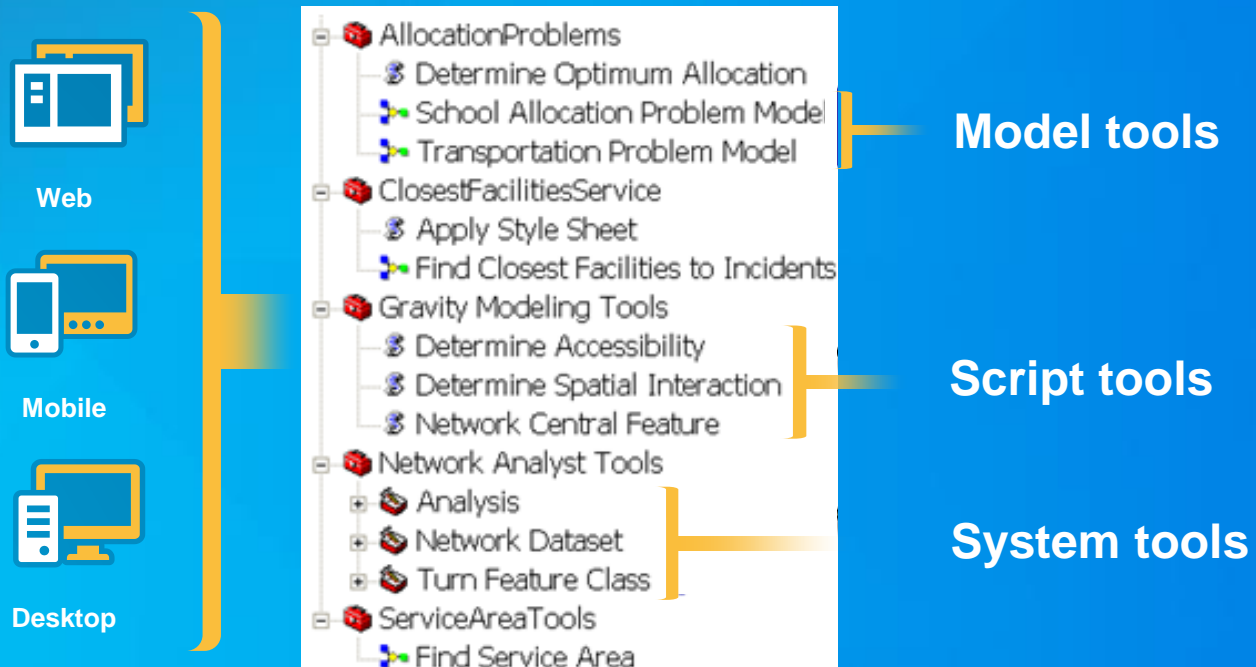
Model



```
outNALayer = arcpy.na.MakeClosestFacilityLayer(inNetworkDataset, outNALayerName,  
impedanceAttribute, "TRA",  
"1", accumulateAttribute,  
"NO_UTURNS")  
#Get the layer object from the result object. The closest facility layer can  
#now be referenced using the layer object.  
outNALayer = outNALayer.getOutput(0)
```

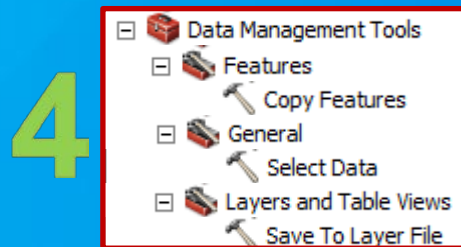
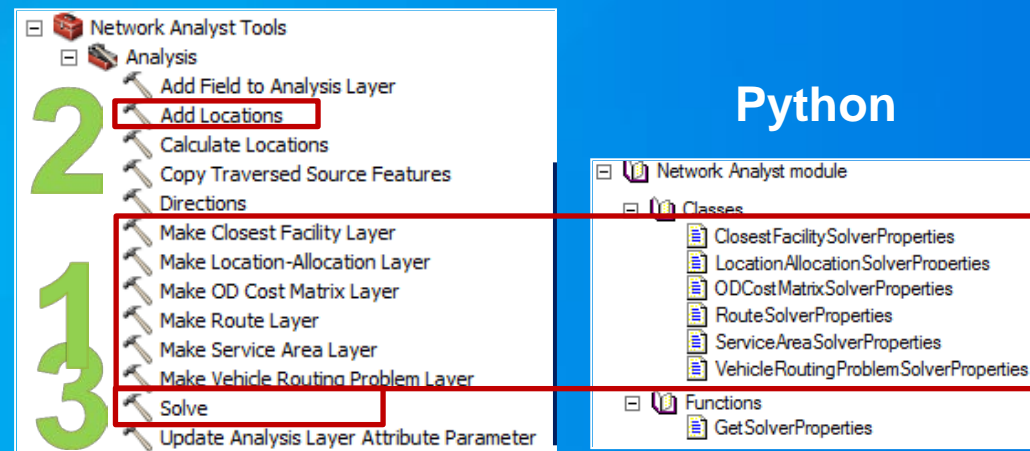
Script

Using Geoprocessing Where?



Network Analyst Workflow

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



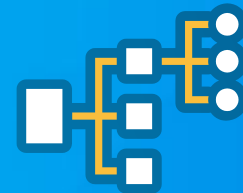


Building Geoprocessing Models

ArcGIS Help:
[What is ModelBuilder?](#)

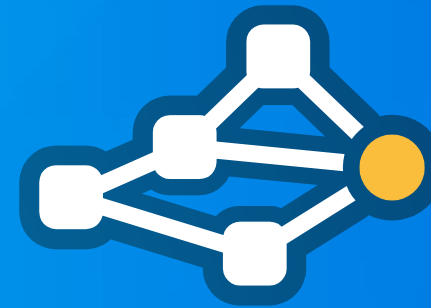
Topics to be covered

- ArcGIS Network Analyst extension concepts
- Geoprocessing framework for network analysis
- **ModelBuilder: Models and model tools**
- Python: Scripts and script tools
- Ready-to-Use services
- Support and resources

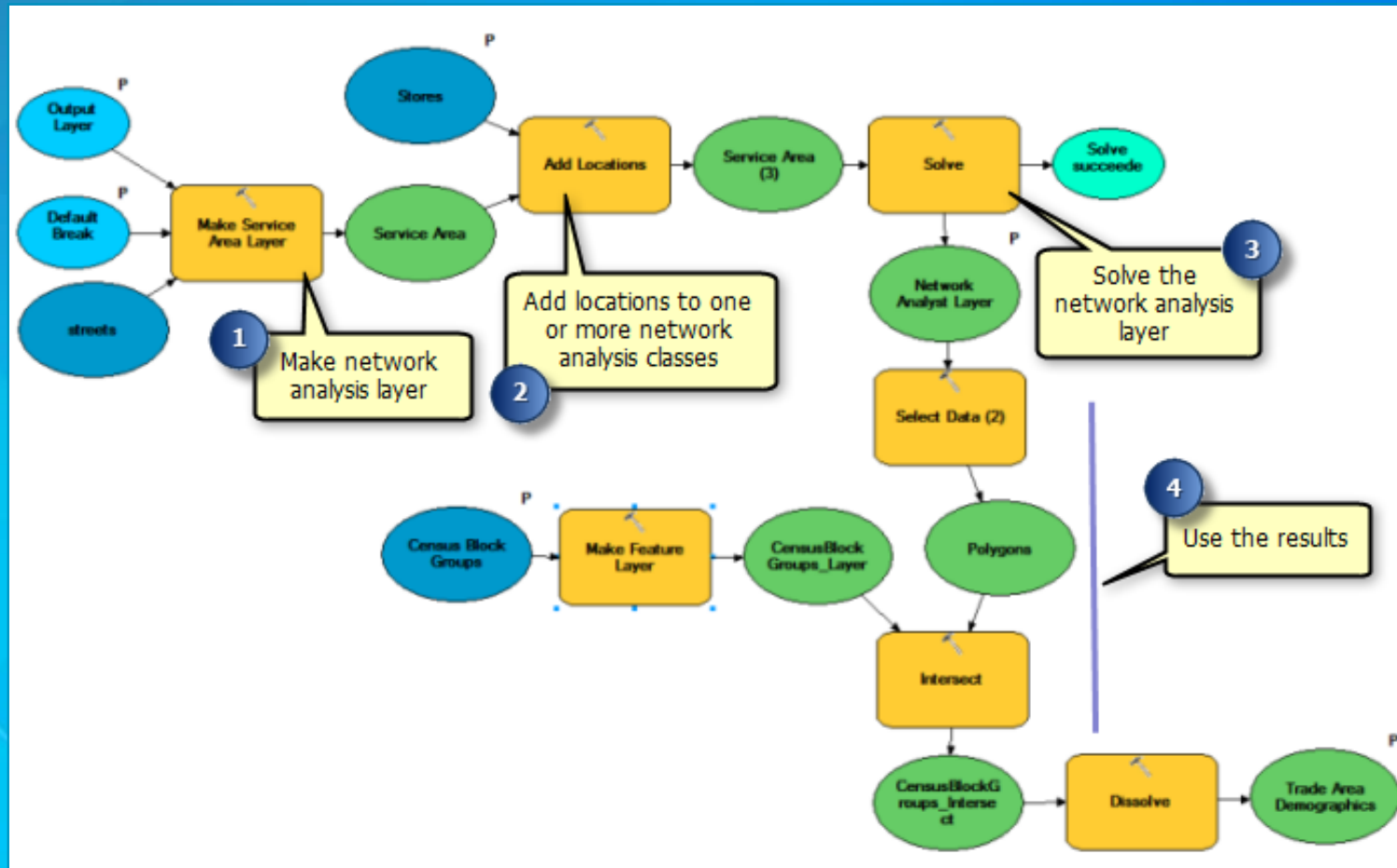


Geoprocessing Models

- Author using Model Builder
- Chain tools to perform a workflow
- Use models like ArcToolbox tools
- Use models within other models
- Apply all Model Builder techniques to network analysis models



Example: Service Area Model



Demo



Automating workflows with geoprocessing models

Performing a network analysis in Model Builder

Sharing a model as a tool

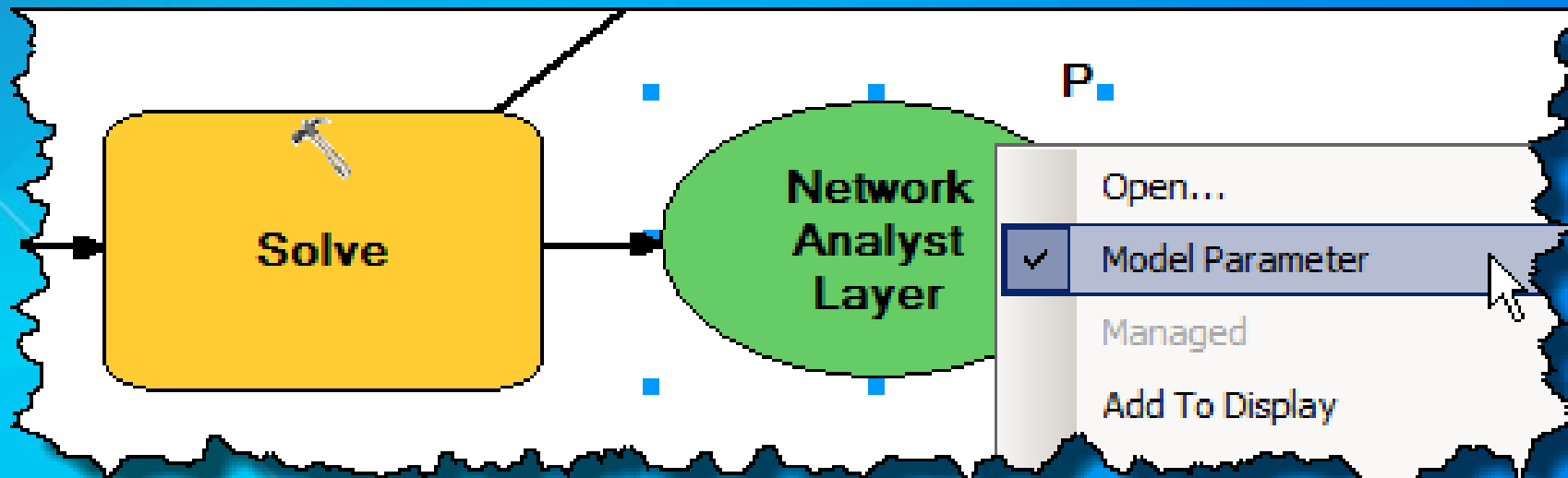
Takeaways

Demo: Automating Workflows with Geoprocessing Models

- **Automate workflows** with Model Builder
- **Share** your model as a tool
- Make inputs and outputs **model parameters**
 - Inputs can be selected by the tool's user
 - Outputs will be added to the ArcMap Table of Contents

Visualizing analysis results in ArcMap

- When running models as tools...
 - The output network analysis layer should be a **model parameter**
 - This will add the layer to the ArcMap Table of Contents



Demo



Automating workflows with geoprocessing models

Working with inputs and outputs

Make a toolbar button

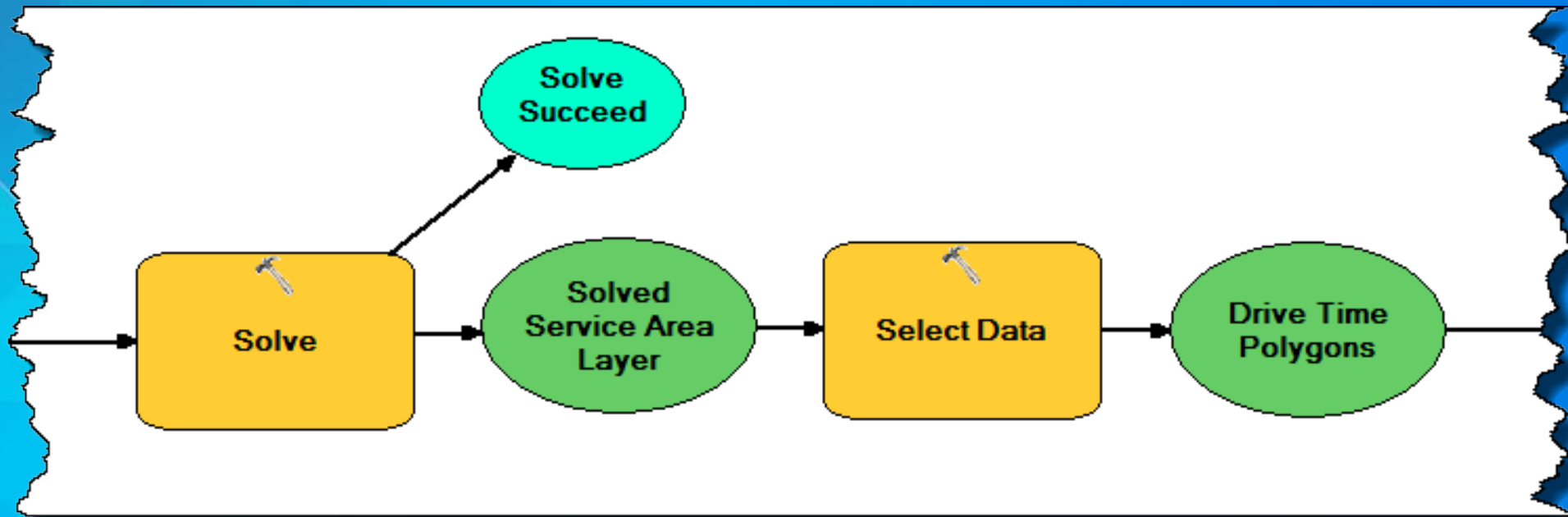
Takeaways

Demo: Automating Workflows with Geoprocessing Models

- Run a model as a **button** on a toolbar
- Use the **Select Data** tool to access NA sublayers
- Work with **external data** like CSV files

Post-processing your analysis

- Use your analysis result as an input to another tool
 - The **Select Data** tool accesses individual sublayers






Writing Python Scripts

ArcGIS Help:
[What is Python?](#)

Topics to be covered

- ArcGIS Network Analyst extension concepts
- Geoprocessing framework for network analysis
- ModelBuilder: Models and model tools
- **Python: Scripts and script tools** 
- Ready-to-Use services
- Support and resources

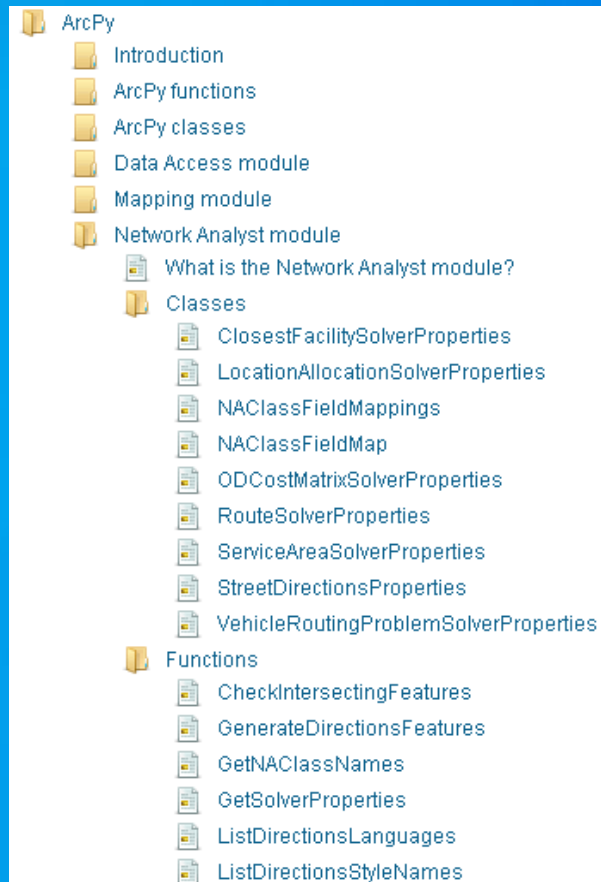
Python Scripts

- Conditional logic
- Loops
- Cursors, creating geometry
- Built-in and third party modules
- Cross-platform



Python Scripts

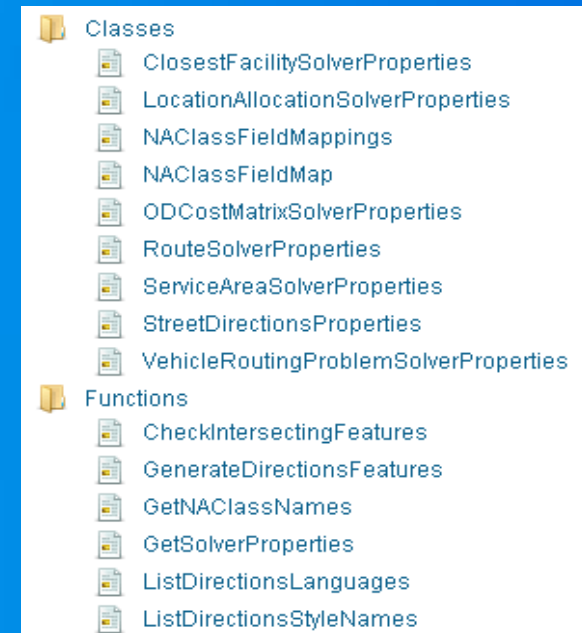
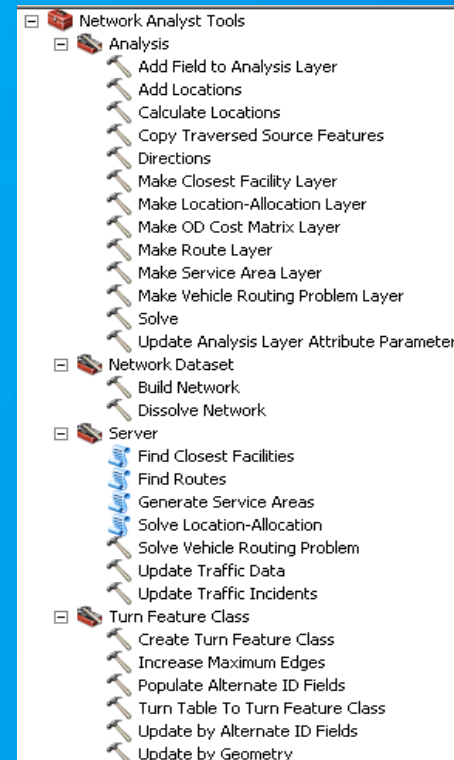
- ArcPy site package
 - Network Analyst module
 - Other geoprocessing tools
 - Other useful functions and classes
 - Describe



Network Analyst Module

arcpy.na

- Simplify access to Network Analyst functionality from Python
- Edit the analysis properties of network analysis layers
 - No need to re-create layers
 - Speed up execution
 - Simplify script logic



Python Script: Basic Building Blocks

```
#Import system modules
import arcpy
```

Import the
arcpy module

```
#!/usr/bin/env python
#Import system modules
import arcpy
from arcpy import env

try:
    #Check out the Network Analyst extension license
    arcpy.CheckOutExtension("Network")

    #Get environment settings
    env.workspace = "C:/data/Paris.gdb"
    env.overwriteOutput = True
    #Set local variables
    inNetworkDataset = "Transportation/ParisMultimodal_3D"
    outLayerName = "ClosestWarehouse"
    impedanceAttribute = "Drivetime"
    accumulateAttribute = ["Meters"]
    inFacilities = "Analysis/Warehouses"
    inIncidents = "Analysis/Stores"
    outLayerFile = "C:/data/output" + "/" + outLayerName + ".lyr"
    #Create a new closest facility analysis layer. Apart from finding the drive
    #time to the closest warehouse, we also want to find the total distance. So
```

Python Script: Basic Building Blocks

```
#!/usr/bin/env python
# Name: Solve_Workflow.py
# Description: Solve a closest facility analysis to find the closest warehouse
# Author: Esri

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

from arcpy import env

env.workspace = "C:/data/Paris.gdb"
env.createOutputFile = True

#Set local variables
inNetworkDataset = "Transportation/ParisMultiModal_3D"
outLayerName = "ClosestWarehouse"
impedanceAttribute = "Drivetime"
accumulateAttributeList = ["Meters"]
inFacilities = "Analysis/Warehouses"
inIncidents = "Analysis/Stores"
outLayerFile = "C:/data/output" + "/" + outLayerName + ".lyr"

#Create a new closest facility analysis layer. Apart from finding the drive
#time to the closest warehouse, we also want to find the total distance. To
```

Check out the
Network Analyst
Extension

Python Script: Network Analyst Workflow

```
#!/usr/bin/env python
# Name: Solve_Workflow.py
# Description: Solve a closest facility analysis to
#             from the store locations and save the
#             disk.
# Requirements: Network Analyst Extension

# Import system modules
import arcpy
from arcpy import env

resultObject = arcpy.na.MakeClosestFacilityLayer(inNetworkDataset, outNALayerName,
                                                impedanceAttribute, "TRAVEL_TO",
                                                "",1, accumulateAttributeName,
                                                "NO_UTURNS")

outNALayer = resultObject.getOutput(0)

# Set local variables
inNetworkDataset = "Transportation/ParishMultiuse01_00"
outNALayerName = "ClosestWarehouse"
impedanceAttribute = "Drivetime"
accumulateAttributeName = ["Meters"]
inFacilities = "Analysis/Warehouses"
inIncidents = "Analysis/Shops"
outLayerFile = "C:/data/output" + "/" + outNALayerName + ".lyr"
# Create a new closest facility analysis layer. Apart from finding the drive
# time to the closest warehouse, we also want to find the total distance. To
```

**Make/edit a
network analysis
layer**

Python Script: Network Analyst Workflow

Add locations to
network analysis classes

```
#!/usr/bin/env python
# Name: Solve_Workflow.py
# Description: Solve a closest facility
#             from the store location
#             task.
# Requirements: Network Analyst Extension

# Import system modules
import arcpy
from arcpy import env

subLayerNames = arcpy.na.GetNAClassNames(outNALayer)
#Stores the layer names that we will use later
facilitiesLayerName = subLayerNames["Facilities"]
arcpy.na.AddLocations(outNALayer, facilitiesLayerName, inFacilities, "", "")

env.workspace = "C:/data/paris.gdb"
env.overwriteOutput = True
#Set local variables
inNetworkDataset = "Transportation/ParisiMultimodal_ND"
outNLayerName = "ClosestWarehouse"
impedanceAttribute = "Drivetime"
accumulateAttributeNames = ["Meters"]
inFacilities = "Analysis/Warehouses"
inIncidents = "Analysis/Stores"
outLayerFile = "C:/data/output" + "/" + outNLayerName + ".lyr"
#Create a new closest facility analysis layer. Apart from finding the drive
#time to the closest warehouse, we also want to find the total distance. To
```

Python Script: Network Analyst Workflow

```
#!/usr/bin/env python
# Name: Solve_Workflow.py
# Description: Solve a closest facility analysis to find the closest warehouse
#              from the store locations and save the results to a layer file on
#              disk.
# Requirements: Network Analyst Extension

# Import system modules
import arcpy
from arcpy import env

try:
    # Check out the Network Analyst extension license
    arcpy.CheckOutExtension("Network")

    # Get environment settings
    env.workspace = "C:/data/Paris.gdb"
    env.overwriteOutput = True
    # Set local variables
    inNetworkDataset = "Transportation/ParisMultimodal_3D"

    arcpy.na.Solve(outNALayer)

    accumulateAttributesName = "Results"
    inFacilities = "Analysis/Warehouses"
    inIncidents = "Analysis/Stores"
    outLayerFile = "C:/data/output" + "/" + outNALayerName + ".lyr"
    # Create a new closest facility analysis layer. Apart from finding the drive
    # time to the closest warehouse, we also want to find the total distance. To
```

Solve the
network analysis layer

```
arcpy.na.Solve(outNALayer)
```

Python Script: Network Analyst Workflow

```
#!/usr/bin/env python
# Name: Solve_Workflow.py
# Description: Solve a closest facility analysis to find the closest warehouse
#              from the store locations and save the results to a layer file on
#              disk.
# Requirements: Network Analyst Extension

# Import system modules
import arcpy
from arcpy import env

try:
    # Check out the Network Analyst extension license
    arcpy.CheckOutExtension("Network")

    # Set environment settings
    env.workspace = "C:/data/Facilities.gdb"
    env.overwriteOutput = True
    # Set local variables
    inNetworkDataset = "Transportation/FacilitiesNetwork"
    outNALayerName = "ClosestWarehouse"
    impedanceAttribute = "DriveTime"
    accumulateAttributeList = ["None"]
    inFacilities = "Analysis/Warehouses"

    arcpy.management.SaveToLayerFile(outNALayer, outLayerFile, "RELATIVE")

    # Create a new closest facility analysis layer. Apart from finding the drive
    # time to the closest warehouse, we also want to find the total distance. To
```

Use the results

Working with analysis layers within scripts

- The network layer is retrieved as a layer object from the result object returned by Make<solver>Layer

```
resultObject = arcpy.na.MakeClosestFacilityLayer(inNetworkDataset, outNALayerName,  
                                                impedanceAttribute, "TRAVEL_TO",  
                                                "", 1, accumulateAttributeName,  
                                                "NO_UTURNS")
```

```
#Get the layer object from the result object. The closest facility layer can  
#now be referenced using the layer object.
```

```
outNALayer = resultObject.getOutput(0)
```

Working with analysis layers within scripts

- Edit the solver properties of an existing layer object

```
# Get the service area layer as an input parameter
saLayer = arcpy.GetParameter(0)

# Get the solver properties object from the service area layer
solverProps = arcpy.na.GetSolverProperties(saLayer)

#Update the properties for the service area layer using the solver properties
solverProps.defaultBreaks = [5, 10, 15]
solverProps.useHierarchy = "USE_HIERARCHY"
```

Accessing sublayers in scripts

- The Select Data tool is not meant for python scripting
- `arcpy.na.GetNAClassNames` should be used
 - Renamed or localized sublayer names will work in the script

```
#Get the names of all the sublayers within the closest facility layer.
```

```
subLayerNames = arcpy.na.GetNAClassNames(outNALayer)
```

```
#Store the layer names that we will use later
```

```
facilitiesLayerName = subLayerNames["Facilities"]
```

```
#Load the warehouses as Facilities using the default field mappings and search tolerance
```

```
arcpy.na.AddLocations(outNALayer, facilitiesLayerName, inFacilities, "", "")
```

Working with sublayers in python

- Access individual sublayers using **ListLayers** on the NA layer.

```
# Get the output Routes sublayer and save it to a feature class
routesSubLayer = arcpy.mapping.ListLayers(outNALayer, subLayerNames["Routes"])[0]
arcpy.management.CopyFeatures(routesSubLayer, outRoutesFC)
```

- Use sublayers as input to other tools (CopyFeatures, Join, Buffer, etc.).
- Use a SearchCursor to access the rows within a sublayer.

Working with analysis layers within scripts

- Easily specify field mappings in Add Locations with `arcpy.na.NAClassFieldMappings`

```
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.na.AddLocations(routeLayer, polygonBarriersNAClass, weatherPolygonLayer, barrierFieldMappings)
```

Saving analysis results

- The in-memory network analysis layer can be persisted using the **SaveToLayerFile** tool in the management module

```
arcpy.management.SaveToLayerFile(outNALayer, outLayerFile, "RELATIVE")
```

- Layer files can be dragged from disk into ArcMap

Topics to be covered

- ArcGIS Network Analyst extension concepts
- Geoprocessing framework for network analysis
- ModelBuilder: Models and model tools
- Python: Scripts and script tools
- **Ready-to-Use services**
- Support and resources



Demo



Automating workflows with python scripts

Performing a network analysis with a python script

Takeaways

Demo: Automating Workflows with Python Scripts

- Run simple python commands in ArcMap's **python window**
- **Export models** to python scripts
- Persist the in-memory network analysis layer with **SaveToLayerFile**
- Run scripts from a command prompt **outside of ArcMap**

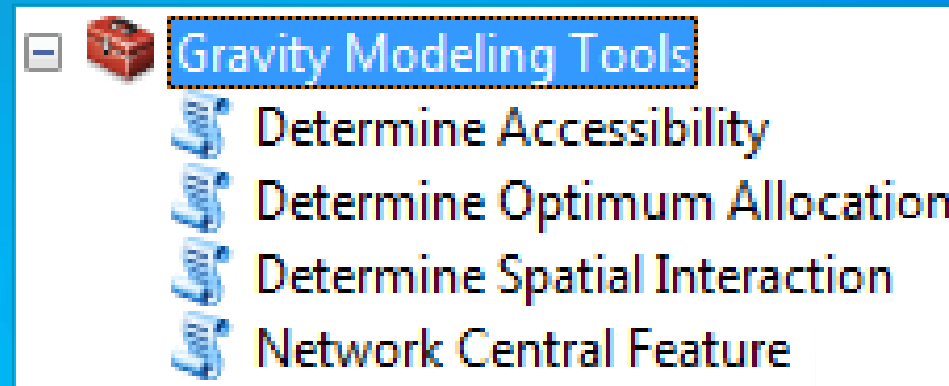


Building Script Tools

ArcGIS Help:
[What is a script tool?](#)

Script tools

- Work with your scripts through a user interface



- Use Script tools like any other tool within ArcToolbox
 - Use script tools in models and vice versa

Add outputs from script tool to ArcMap

- If a network analysis layer is the output...
 - Make an additional derived output parameter of type Network Analyst Layer
 - Use `arcpy.SetParameterAsText(...)`

```
#Do your analysis workflow  
outNALayer = arcpy.na.MakeClosestFacilityLayer(inNetworkDataset, outNALayerName,
```

```
# Set your analysis layer as an output parameter for the script tool  
arcpy.SetParameterAsText(1, outNALayerName)
```

Demo



Creating a script tool

Create a script tool to provide a UI for a Python script

Use tool validation to customize the UI.

Write a script tool to extend the capabilities of ArcGIS.



Takeaways

Demo: Creating a Script Tool

- Provide a user interface for python scripts by making a **script tool**
- Use derived output and **`arcpy.SetParameterAsText()`** to add results to the map
- Use **tool validation** to customize your script tool's UI
- Use python modules to **extend** the capabilities of ArcGIS

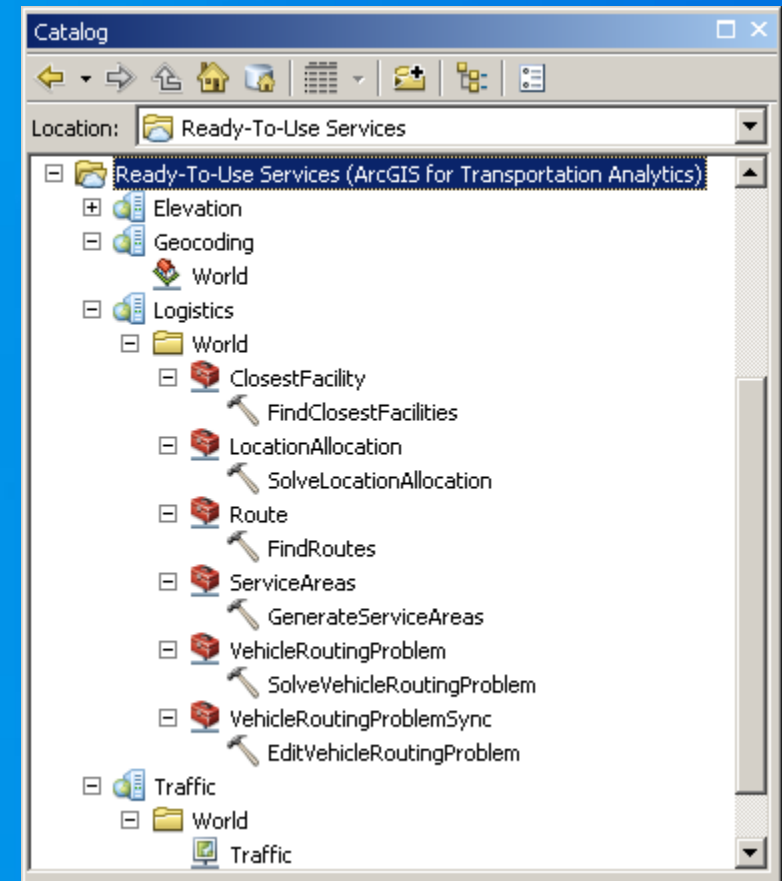
Topics to be covered

- ArcGIS Network Analyst extension concepts
- Geoprocessing framework for network analysis
- ModelBuilder: Models and model tools
- Python: Scripts and script tools
- **Ready-to-Use services**
- Support and resources



Ready-To-Use Services

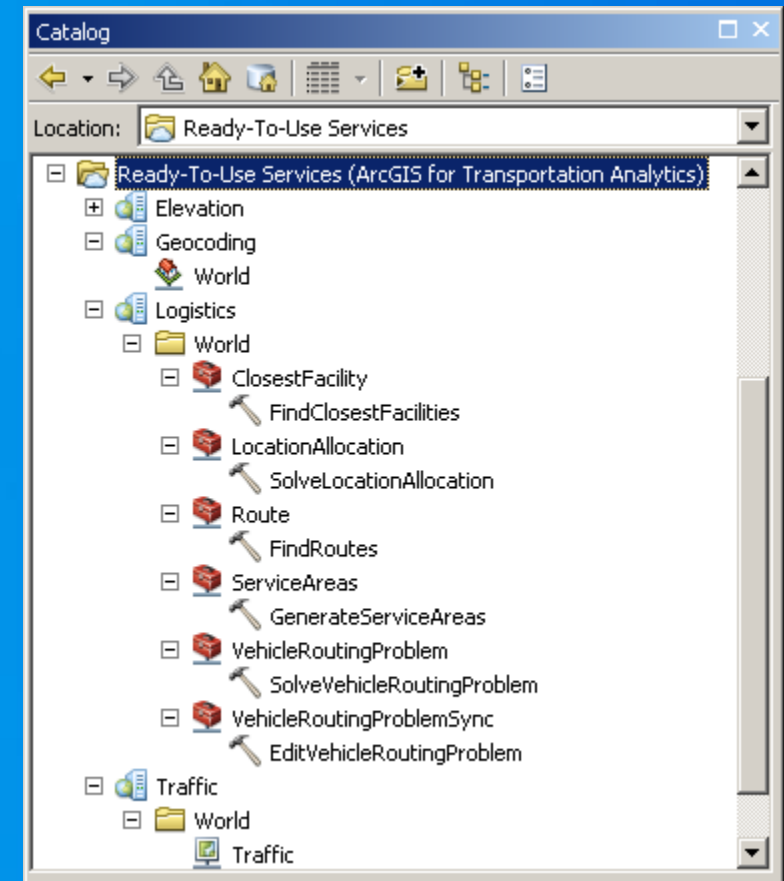
- **Published by Esri**
- **Managed by Esri**
 - **No user maintenance of servers, services, or data**
 - **Excellent uptime and reliability**



Ready-To-Use Services

Benefits

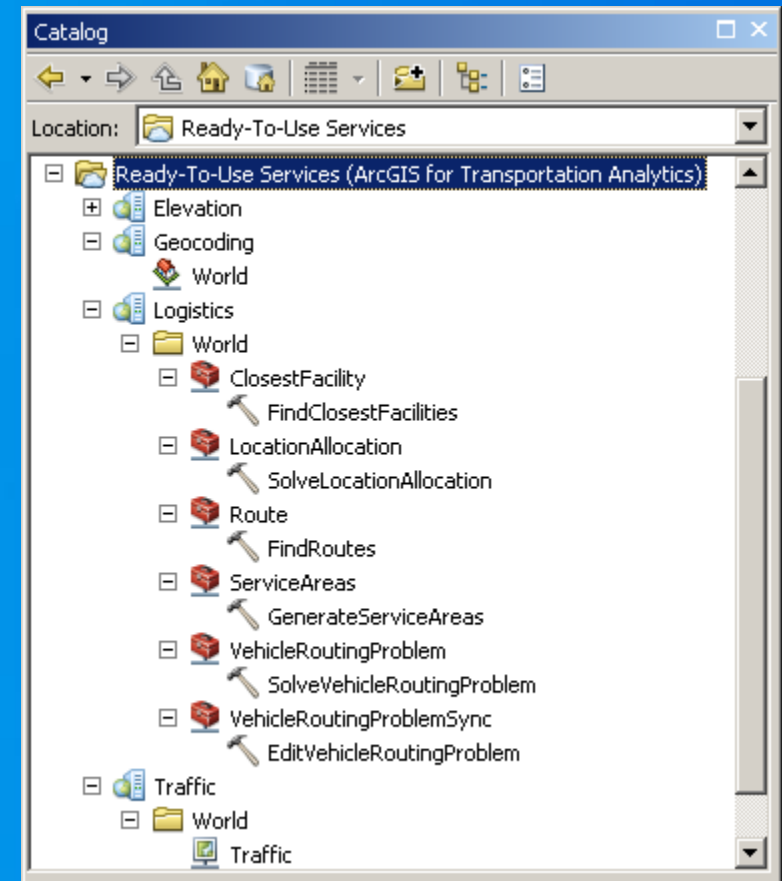
- Global data
- Live traffic
- No Network Analyst specific software required
- Create Layer, Add Locations, and Solve all in one tool



Ready-To-Use Services

Where?

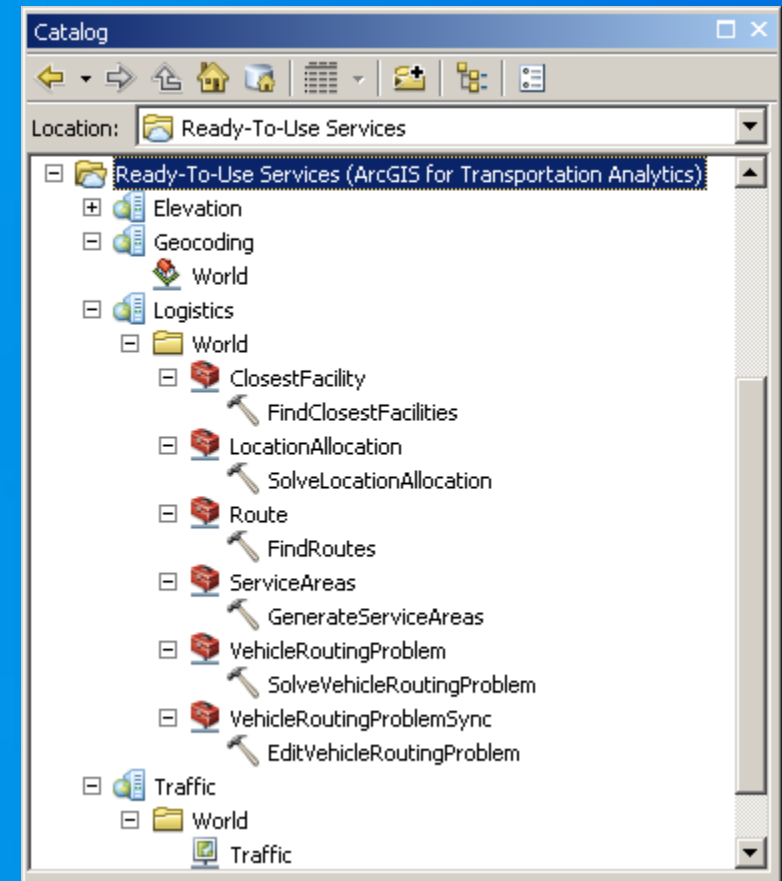
- Anywhere geoprocessing tools can be used
 - Model builder
 - Python scripts
 - ArcMap
- Like any other service that uses credits
- Supported from 10.0 SP5 and up



Ready-To-Use Services

Requirements

- ArcGIS Online subscription with credits
- Your own analysis inputs



Ready-To-Use Services

Cost

- Check [credit page](#)
- Varies per type of analysis

Network Analysis	Service Credits Used
Simple Routes	0.04 credits per route
Optimized Routes	0.5 credits per optimized route
Drive Time (Service Areas)	0.5 credits per drive time
Closest Facilities	0.5 credits per closest facility route
Multi-Vehicle Routes(VRP)	1 credits per route
Traffic	0 credits

Demo



Ready-To-Use services

Connecting to Ready-To-Use services

Using a Ready-To-Use service in a model



Takeaways

Demo: Ready-To-Use Services

- Use **Ready-To-Use services** like other geoprocessing tools
- Access high-quality **street data**

Publishing geoprocessing services

- More details about authoring, publishing and using on-premise network analysis services are available in technical workshop titled **Performing Network Analysis with ArcGIS for Server** from a previous user conference
 - [Workshop presentation](#)
 - [Workshop video](#)



Summary



Summary

- **Use the geoprocessing framework for network analyses**
 - **Network Analyst Tools**
 - **Models and Model tools**
 - **Script and Script tools**
- **Automate workflows**
- **Incorporate network analysis in larger process**

Topics to be covered

- ArcGIS Network Analyst extension concepts
- Geoprocessing framework for network analysis
- ModelBuilder: Models and model tools
- Python: Scripts and script tools
- Ready-to-Use services
- **Support and resources**





Resources

Support and resources

- **Tutorials**
 - [Network Analyst tutorial](#)
 - [Network Analyst geoprocessing service examples](#)
- [Code samples in Network Analyst tools toolbox](#)
- [ArcGIS Network Analyst Extension Discussion Forum](#)
- [ArcGIS for Transportation Analytics Group on arcgis.com](#)

Support and resources

- [Python for ArcGIS resource center](#)
- **Books**
 - [GIS Tutorial for Python Scripting](#)
 - [Python Scripting for ArcGIS](#)
 - [Getting to Know ArcGIS ModelBuilder](#)



Network Analyst at the Esri User Conference

Network Analyst presentations

Tuesday

Wednesday

Thursday

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



Understanding our world.