

2008 ESRI User Conference
Technical Workshops
August 4–8, 2008

Geoprocessing Services

Nathan Warmerdam

Outline

- **Overview of Geoprocessing Services**
- **Why Geoprocessing Services**
- **Publishing services**
- **Authoring services**

Geoprocessing Services

- **The geoprocessing service allows you to publish custom tools to be used via ArcGIS Server.**
- **Geoprocessing services can be used by many different client applications**
 - ArcGIS Desktop
 - ArcGIS Engine
 - ArcGIS Explorer
 - Web ADF
 - WSDL
 - Rest
 - JavaScript
 - FLEX

Geoprocessing Services

- **You create models or scripts to perform desired tasks on a server.**
- **Endless array of tasks can be created**
 - Spatial analysis (vector, raster, network...)
 - Data Management (geodatabase, file based data)
 - Conversion (ETL and data loading)
 - etc
- **You need to be knowledgeable about using geoprocessing tools to create a good geoprocessing service.**

Why use Geoprocessing Services?

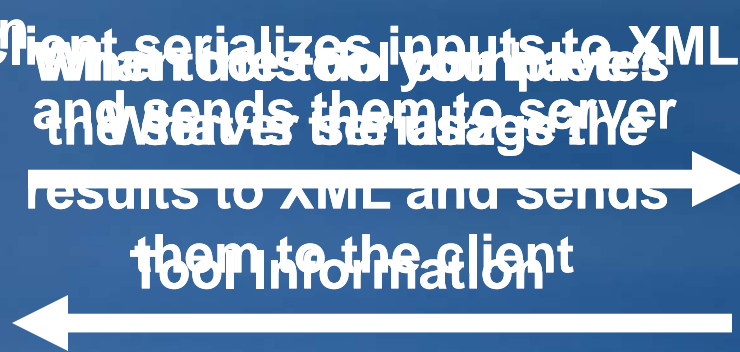
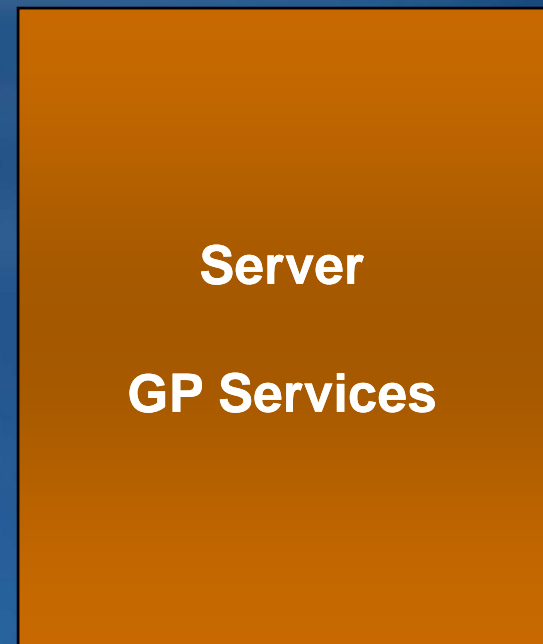
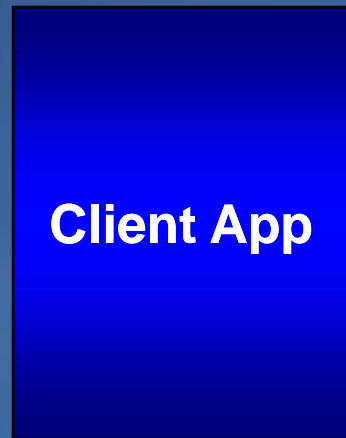
- Allows organizations to centralize both data and processing on the server
- Allows geoprocessing operations to be run on the server from light weight clients across the web
- Analyst expertise stored in models can be exposed to wider audiences and skill levels
- **No programming necessary**

- **Demo: Using a geoprocessing service in ArcGIS Explorer**

How a Geoprocessing Service Works - *a high level peek behind the scenes*

Client receives results,
guides serializes them,
user hits input dialogs
and displays them
and hits ok
tool information

Server receive request,
deserializes the inputs
and runs the tool



Client serializes inputs to XML
and sends them to server
Server deserializes the inputs
and runs the tool
Server serializes results to XML
and sends them to the client
Tool information

Geoprocessing Services – Authoring, Publishing, Consuming

- **Authoring – designing the service**
 - A GIS Analyst authors a model containing the geoprocessing functionality he/she wishes to provide to clients.
- **Publishing – creating the service**
 - Publish a toolbox that contains a model
 - Publish a map document with a Tool Layer
- **Consuming – using the service**
 - Use the geoprocessing service in out of the box clients or custom clients

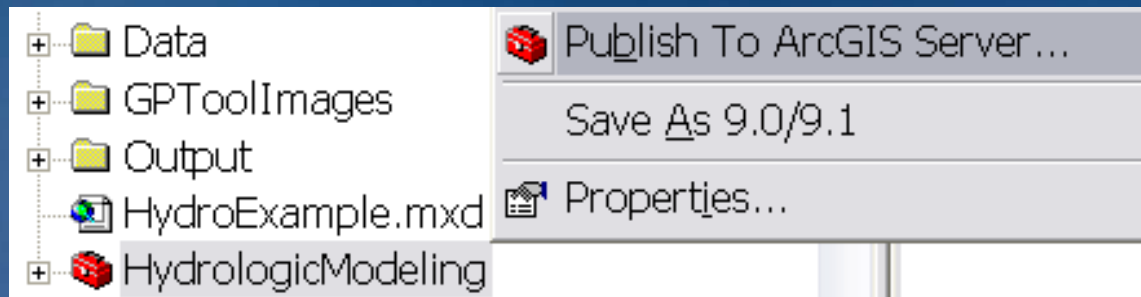
Geoprocessing Service Behavior

- **Geoprocessing Services are very flexible and allow many different behaviors and optimizations**
- **Before Authoring and Publishing, identify what you want your service to do and how you want it to behave with clients.**

Publishing

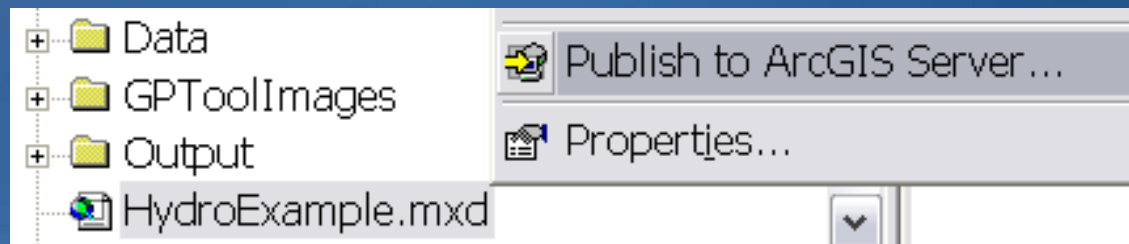
- **Publish a Toolbox**

- The tasks in the geoprocessing service are determined by the tools in the toolbox



- **Publish a Map Document**

- The tasks in the geoprocessing service are determined by the “**tool layers**” in the map document
- Inputs may be map layers



Publishing a Map Document: two configurations

- **Geoprocessing Service WITH an associated Map Service**
 - Output available as a map service draw or data.
 - Better cartographic rendering of results
 - Render large results (>1000's of features)
 - Prevents output feature geometry from being sent to the client
- **Geoprocessing Service WITHOUT an associated Map Service**
 - Output only available as data
 - Inputs may be map layers

ArcGIS Server - Geoprocessing Service Properties

General Parameters Capabilities Pooling Processes

Execution Type: Synchronous Asynchronous

The tools exposed by the Geoprocessing Service are stored in:

A toolbox

Toolbox

A map

Map Document:

C:\Demos\UC2008\ServerTW\toolayer\Hurricane1

Data Frame:

Layers

Change...

Result Map Service:

HurricaneBufferService

Jobs Directory:

c:\arcgisserver\arcgisjobs

Virtual Jobs Directory:

http://vailima/arcgisjobs

Properties

Execution Type: Synchronous Asynchronous

The tools exposed by the Geoprocessing Service are stored in:

A toolbox

Toolbox

A map

Map Document:

C:\Demos\UC2008\ServerTW\toolayer\Hurricane1

Data Frame:

Layers

Change...

Result Map Service:

None

Jobs Directory:

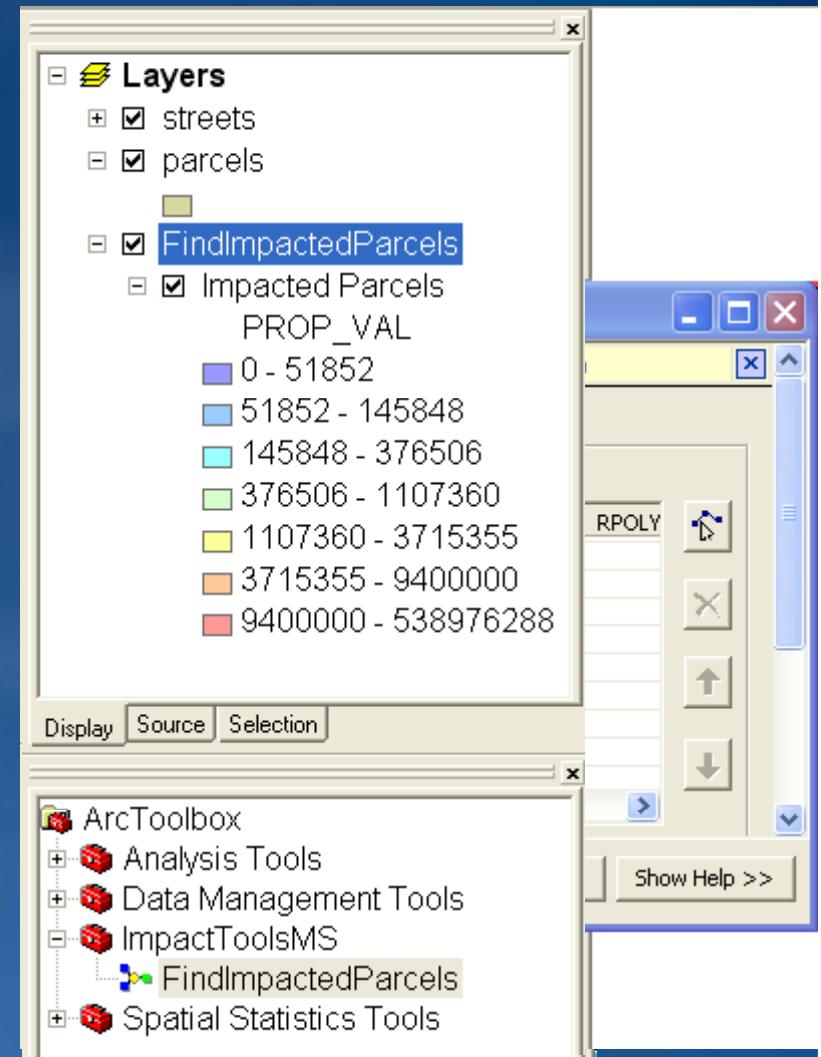
c:\arcgisserver\arcgisjobs

Virtual Jobs Directory:

http://vailima/arcgisjobs

Tool Layer

- **What is a tool layer?**
 - A special group layer containing outputs of a tool
 - Defines parameter symbology
- **How to create?**
 1. Drag and drop a tool into a map
 - Tool outputs are added as sub-layers
 2. Open tool dialog and run



- **Demo – Creating a Tool Layer and Publishing a Map Document**

Publishing Considerations – Execution Mode

- **Asynchronous (Submit Job)**

- Results are saved on the server
- Results can be drawn on the server
- Clients free to do other tasks
 - e.g. in ArcMap you can pan/zoom, run other tools while the job is running
- Appropriate for longer processing jobs.

- **Synchronous (Execute)**

- Appropriate for faster processing jobs. (<10 seconds)
- Client always receives and draws data.
- Desktop Client waits until job is completed and results are returned

Publishing Considerations

- **Maximum Number of Records**

- This property limits the number of features returned from the server. The default is 500.
- Prevents large amounts of data from being transported across the internet.
- **Can be a gotcha. If your results don't display because of this reason, there should be a message in the tool messages**

- **Number of Instances**

- How many concurrent requests can run

- **Timeout**

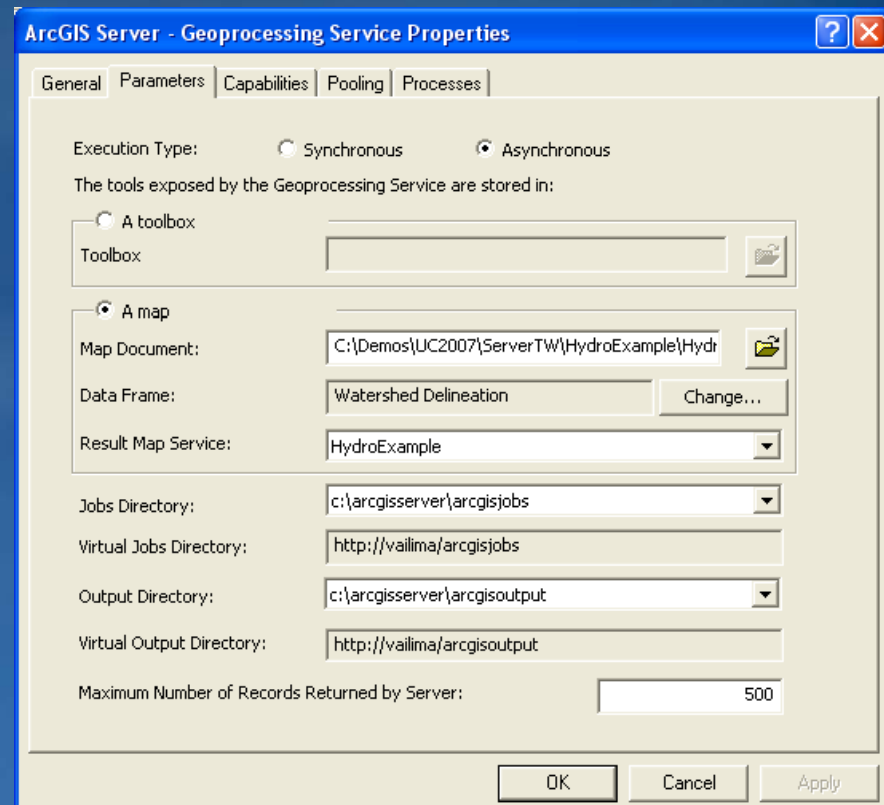
- How long before the service automatically kills itself

- **Messages**

Modifying Service Properties

- The properties of a service can be modified by stopping the service and opening its properties page

- Can view and change
 - Execution mode
 - Source toolbox or map
 - Associated map service
 - Maximum number of records
 - Jobs directory location
 - Messages (9.3)

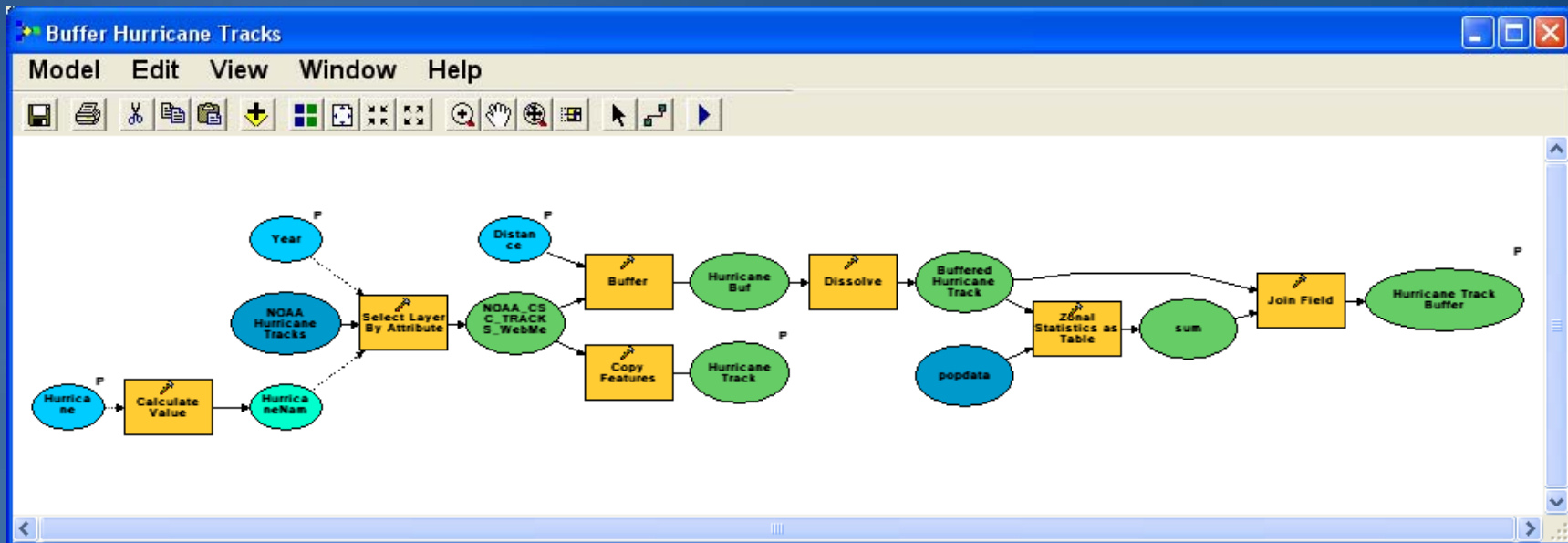


Publishing – Assumptions

- **Defaults of Toolbox “right-click” publish**
 - Service is Asynchronous
 - No messages returned
 - Maximum of 500 features returned
- **Defaults of Map “right-click” publish**
 - Service is Asynchronous
 - Has an associated map service
 - No messages returned
 - Maximum 500 features returned

Authoring – overview

- **Model tools or Script tools** contain the geoprocessing functionality run by geoprocessing services
 - We do not recommend that you publish system tools directly.
- A geoprocessing service is a tool plus its associated data



Authoring a Model suitable for publishing

- Models and scripts that are run by services need to have certain characteristics to run correctly.
- Changes will likely be required to allow existing toolboxes with models and scripts to be published.
- Things to consider when creating a model for use as a Geoprocessing Service
 - Data Type of Parameters
 - Data Management (Source, Intermediate, Output)
 - Symbology
 - Optimization

Authoring a Model suitable for publishing

- **The tool must be portable**
 - A new job workspace is created on the server each time the tool is executed.
 - The tool needs to be constructed so that it will run in the job workspace created by the server.
- **The tool will be run by the ArcGISSOC account**
 - Make sure the ArcGISSOC account has access to all the input data.

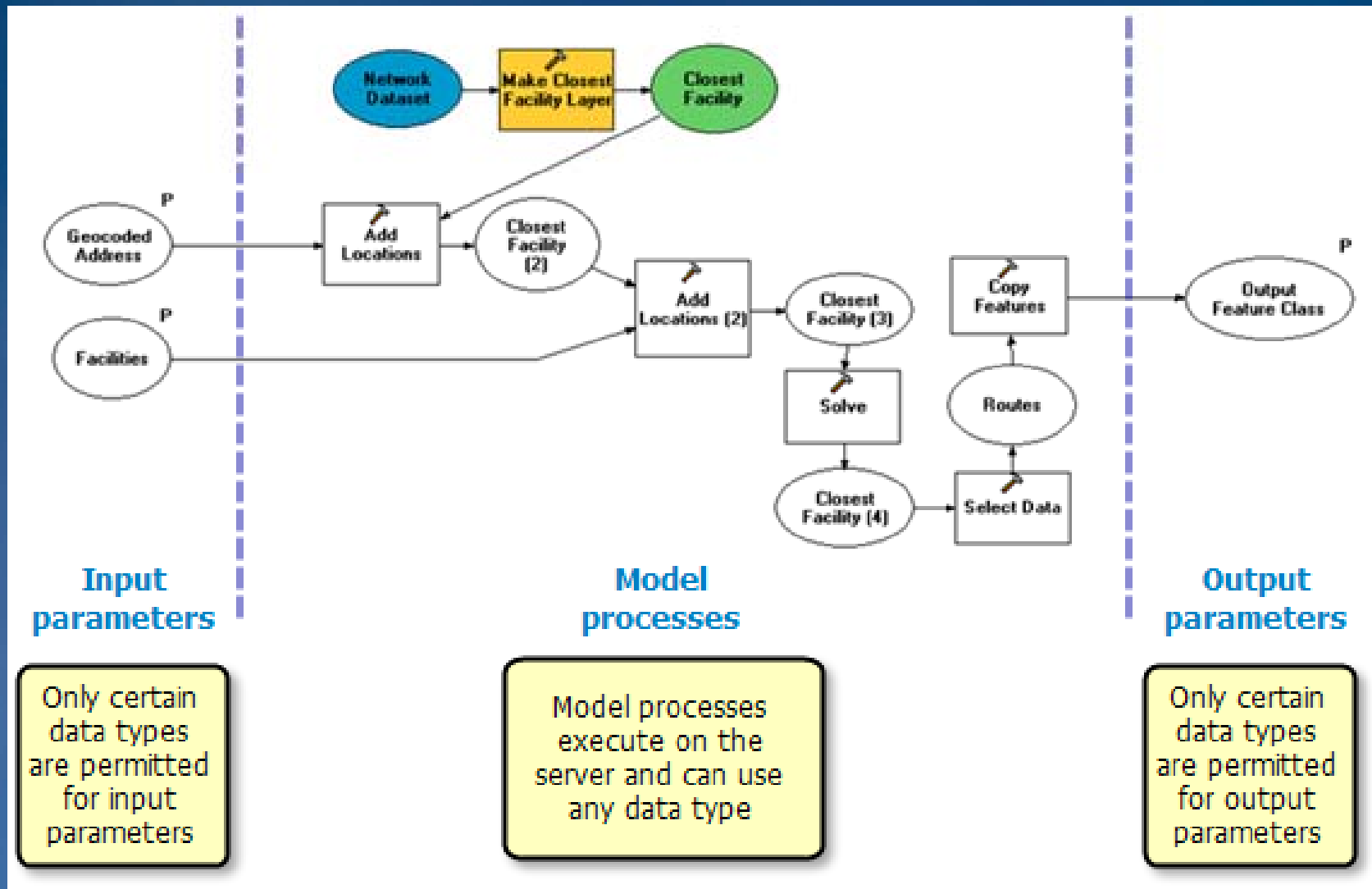
Parameter Types

- A subset of desktops data types are supported as valid parameters to a geoprocessing service
 - Subset determined by ESRI's "out of the box" light weight clients:
 - ArcGIS Explorer
 - Web Mapping Applications
- Publishable tools need to be built accordingly.

- String
- Long
- Double
- Boolean
- Date
- Linear Unit
- Feature Set
- Feature Class***
- Record Set
- Table***
- Table View
- Raster Dataset
- File
- Feature Layer
- Raster Layer
- Layer

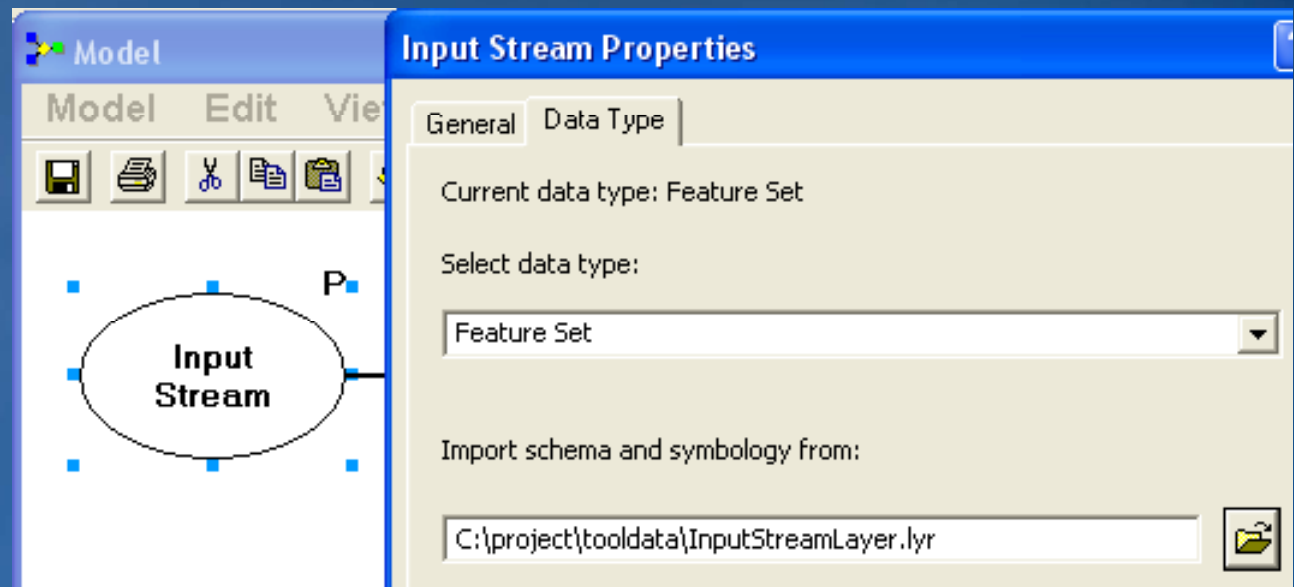
***output only**

Parameter Types



Parameter Types – Features and Records

- Feature Class and Table variables
 - Publish only as output parameters.
- Feature Set and Record Set variables
 - Use Feature/Record Set for interactive input of features or rows
 - Schema defined in properties from existing layer, feature class or table
 - Fields
 - Field domains
 - Feature type
 - Symbology

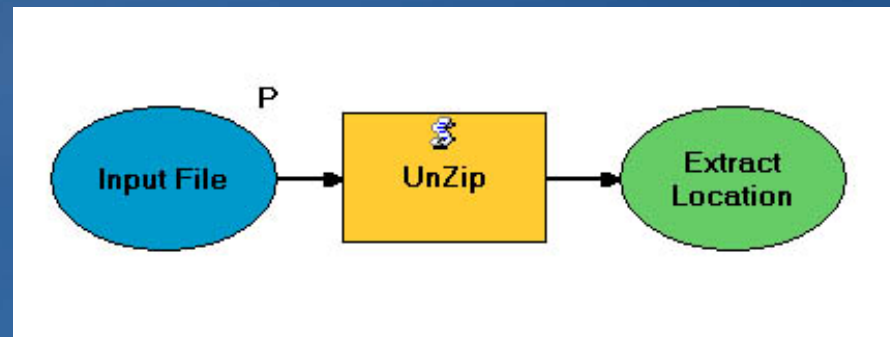


Parameter Types – Layers

- **Layer parameter type allows clients to select from layers in a map on the server.**
 - Enables the use of datasets on the server.
 - Gives ability to work with “non-publishable” data types.
- **Models that use layers must be published as Tool Layers in Map Documents**

Parameter Types - File

- Publishes as input or output parameter
- Can be used to upload zip files to a server.
 - Samples in the help:
http://webhelp.esri.com/arcgisdesktop/9.3/index.cfm?id=907&pid=899&topicname=Python_scripts_to_zip_and_unzip_data
 - Can send anything up to the server in a zip file and unzip server side.



Data Management – Source Data

- **Source data, maps and models are the information the service uses to run.**
- **Data / Tools / Scripts must be accessible by this account by the ArcGISSOC account**
- **Troubleshooting technique.**
 - **Login as the ArcGISSOC account and run the tool you want to publish. If there is an accessibility problem, the tool will fail.**

Data Management – Source Data

- If the data is not large or centralized, it can be useful to package everything in a folder and use relative paths
 - [http://webhelp.esri.com/arcgisdesktop/9.3/index.cfm?TopicName=Methods for distributing tools](http://webhelp.esri.com/arcgisdesktop/9.3/index.cfm?TopicName=Methods%20for%20distributing%20tools)
- If the data is distributed around the network, use UNC paths when building the model/script.
- Copy SDE Connection files relative to tbx.
 - Will not find “database connections” node.

Data Management – Data Generated by Job Submissions

- When ArcGIS Server is installed, a directory called “arcgisjobs” is created to hold data created by the jobs.
- Each time a new job is submitted to the server, a directory is created in the arcgisjobs folder to hold the data generated by each new job.

```
[-] arcgisserver
  [-] arcgiscache
  [-] arcgisjobs
    [-] HydrologicModelingTbx_GPServer
      [+] J9711F162B991423BA38DE2628EFB7A5E
      [+] JA2BBE914F7144214817CDD905E095397
      [+] JCACFAEF428004CD99731BE344CC9657D
      [+] JCF00CA5933E74647A23CDCF3D89C9793
```

Data Management – Intermediate and Output

- When the server runs a model, the output and intermediate data should be written to the job directory
- Use the `%ScratchWorkspace%` inline variable in your paths.
 - `%scratchworkspace%\output.shp`
- A file geodatabase named “scratch.gdb” is guaranteed to be in the jobs folder created for each job submission
 - `%scratchworkspace%\scratch.gdb\output`

- **Demo: Making an existing model service worthy**

Working with UNIX

- Models and/or Map Documents are composed on Windows.
- To make the paths portable to UNIX, use **relative paths** or use **UNC paths**.
- UNC paths are converted to UNIX paths on the server
 - `\\<host>\<folder>` -> `/<host>/<folder>`
 - Do not use a samba server on Windows with a different name than the `<host>` on the UNIX network.

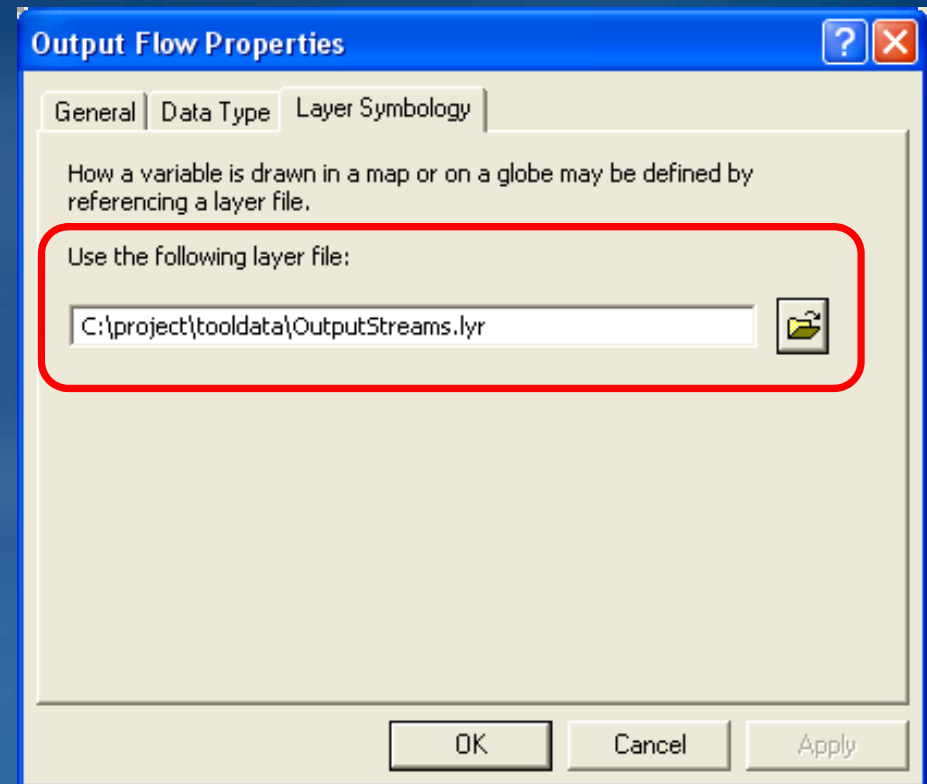
Symbology – Overview

- As the author of a service you may want to set the symbology you want the client to use.
- ESRI's out of the box clients (ArcGIS Desktop and Explorer) will respect symbology set on services.



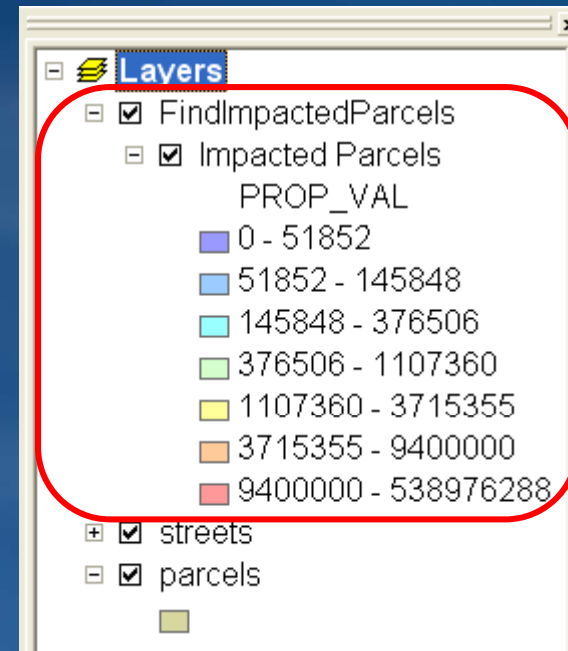
Symbology – when publishing a toolbox

- Symbology is stored as a property of variables in a model.
- Symbology is stored as a property of script tool parameters
- Use a layer file to define the symbology
- Not all symbology is supported on light weight clients
 - Simple symbols only



Symbology – when publishing a Map Document

- Symbology stored in a **Tool Layer**
- A Tool Layer is a group layer in the table of contents representing a tool and its parameters.
- The map service that draws the result will use the symbology of the tool layer



Client Symbology Properties

- **Feature symbology**
 - Single Symbol
 - Unique Value
 - Graduated Colors
 - **Transparency**
- **Raster symbology**
 - Unique Value
 - Classified
 - Stretched
 - Transparency
- **Update symbology by job**
 - Feature Unique Value
 - Feature Graduated Colors
 - Raster Classified

Yellow = New at 9.3

Documentation

- **Demo: Review of Documentation**
- **Entire New Book in 9.3**
 - Desktop Help → Geoprocessing → Geoprocessing with ArcGIS Server
 - http://webhelp.esri.com/arcgisdesktop/9.3/index.cfm?TopicName=An_overview_of_geoprocessing_with_ArcGIS_Server
- **Examples**
 - A dozen examples
 - Data for examples included with Tutorial Data.

Troubleshooting

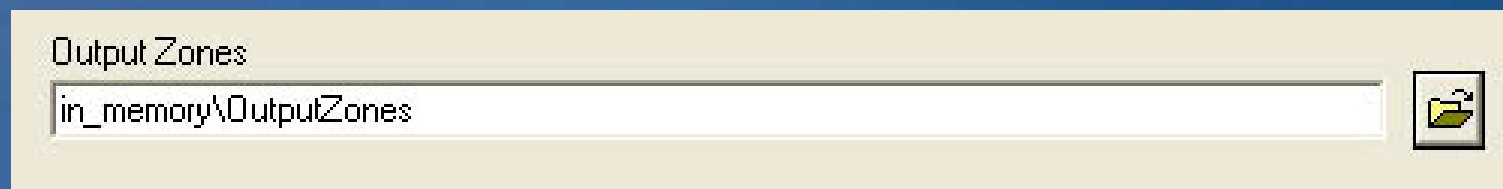
- Check the log files if you are getting errors
 - Via manager or on disk (**ArcGIS\server\user\log**)
- Debugging
 - Change to Asynchronous
 - Turn Messages On
 - Run and check Job Directory
 - See “Messages.xml”
 - See intermediate and output data

Tool Optimization – Pre-Processing

- It is smart to pre-process any geoprocessing operations that you can and remove them from your model.
 - EX: A suitability model may use slope and aspect as criteria. It is not necessary to run slope every time the model is executed. Pre-process slope and aspect.

Tool Optimization – In Memory

- **Data can be written out to the “in_memory” workspace.**
 - Only appropriate when overhead of writing to disk is significant portion of the total time it takes to run the model
 - If output is “in_memory” the client must draw the result
- **Use the “in_memory” keyword to indicate that a dataset will be stored in memory.**



Server Tuning: Large Datasets

- Large input record set or raster
 - Increase the web server maximum size setting
 - **C:\inetpub\wwwroot\ArcGIS\Services\web.config file**

```
<?xml version="1.0" encoding="utf-8"?>  
<configuration  
  xmlns="http://schemas.microsoft.com/.NetConfiguration/v2.0">  
  <system.web>  
    <httpRuntime maxRequestLength="20000"></httpRuntime>  
  </system.web>
```

Server Tuning: Timeouts

There are 3 Timeouts to be aware of:

- Client Wait Timeout

C:\inetpub\wwwroot\ArcGIS

```
<system.web>
```

```
  <httpRuntime executionTimeout="600" />
```

```
</system.web>
```

- ArcGIS Server Wait Timeout
 - ArcGIS Server Execution Timeout
- If you are using synchronous execution with lots of users, you may need to increase the wait timeouts.

Tuning: UNC Paths

- Reading and writing data to UNC paths is slower
- If using one server machine avoid UNC where possible
 - Use local path for jobs directory.
 - Use local path to source data if possible.
- If using a distributed server (many machines)
 - Jobs directory must be a UNC path.
 - Can use the **in_memory** workspace for feature classes and tables.
 - Make a copy of the input data on each SOC machine.
 - Use LocalJobsDirectory setting.

Local Jobs Directory

- New “LocalJobsDirectory” property added to reduce the use of UNC paths.
- When specified all intermediate and output data are written to a local job directory and then copied to the main server jobs directory
- Only relevant if server is distributed or jobs directory is a UNC location.
- Synchronous services only until 9.3 sp1

Setting the LocalJobsDirectory

- Stop the ArcGIS Server Object Manager (SOM) service
- Manually edit the geoprocessing service's .cfg file and add the <LocalJobsDirectory> tag.
 - .cfg for each service found in <install location>\Server\cfg
- Restart the SOM
- Restart the geoprocessing service.

```
<Properties>  
  <Toolbox>C:\project\HydroTools.tbx</Toolbox>  
  <MaximumRecords>500</MaximumRecords>  
  <LocalJobsDirectory>C:\localjobs</LocalJobsDirectory>  
  <JobsDirectory>\\savaii\arcgisserver\arcgisjobs</JobsDirectory>  
  <JobsVirtualDirectory>http://vailima/arcgisjobs</JobsVirtualDirectory>  
  <ExecutionType>Asynchronous</ExecutionType>  
  <OutputDir>\\savaii\arcgisserver\arcgisoutput</OutputDir>  
  <VirtualOutputDir>http://vailima/arcgisoutput</VirtualOutputDir>  
  <ShowMessages>FALSE</ShowMessages>  
</Properties>
```

Improvements in 9.3

- **9.3 goal was not to add functionality but to improve quality**
- **Speed**
 - Large record set save much faster
 - 100k features server went from 20 min to 1 min
 - Tools faster with feature sets/record sets
 - Tools faster with SDE **(9.2 SP4)**
 - In-memory Python script tools execute much faster
 - Faster Synchronous execute

Improvements in 9.3

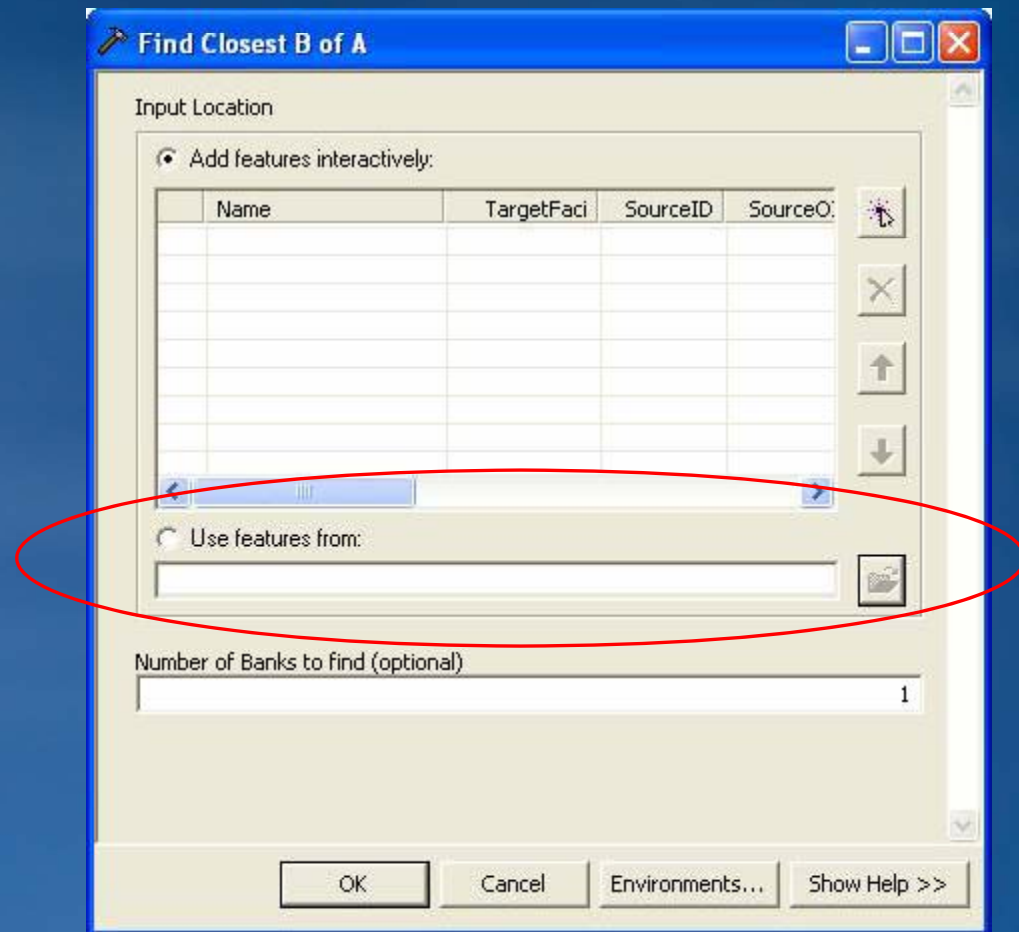
- **Publishing Validation**
 - Immediate feedback given to the user
 - Check for invalid sub-layers added.
 - Publishing of invalid maps and toolbox disallowed
- **Identify with map server geoprocessing results**
- **Zoom to layer with map server geoprocessing results**
 - Desktop / Server 9.2 SP5
 - ArcGIS Explorer build 440 or higher.
- **Show Messages Option**
 - May not want to show tool execution messages for security reasons.
 - The default is to **not** show messages.

Improvements in 9.3

- “I want to send a feature class to my service”

- Modified the Feature Set control in Desktop to be able to load a feature class.

- Can edit attributes in main panel.



Improvements in 9.3

Performance Logging

- Info:Standard (Info2): Record elapsed time for all service methods.
- Info:Detailed (Info3): Record elapsed time for key sub processes.

2008-02-08T13:40:45	INFO2	<u>Population.GPServer</u>	<u>GPServer.Execute</u>	Binary request received. Request size is 39434 bytes.	mb8	2240		100003
2008-02-08T13:40:45	INFO3	<u>Population.GPServerSync</u>	<u>GPServerSync.Load.summarizePopulation</u>	Load job j5c26f564c33640ffb6adae2df3cba23e message type Binary message size 39435	mb8	2324	0.10153	20023
2008-02-08T13:40:46	INFO3	<u>Population.GPServerSync</u>	<u>GPServerSync.Execute.summarizePopulation</u>	Execute job j5c26f564c33640ffb6adae2df3cba23e	mb8	2324	0.93816	20022
2008-02-08T13:40:46	INFO3	<u>Population.GPServerSync</u>	<u>GPServerSync.Save.summarizePopulation</u>	Save job j5c26f564c33640ffb6adae2df3cba23e message type Binary message size 1964	mb8	2324	0.00549	20024
2008-02-08T13:40:46	INFO2	<u>Population.GPServer</u>	<u>GPServer.Execute</u>	Binary request successfully processed. Response size is 1963 bytes.	mb8	2240	1.06580	100004

Programmatic Access

- Engine
- .NET ADF
- Java ADF
- WSDL
- REST (9.3)
- JavaScript APIs (9.3)

New API Options

- **Result Options**

- Users can tell the server how they want the results
 - ESRI or KML formats
 - Spatial Reference

- **Get Job Input Values**

- GetJobToolName
- GetJobResultOptions
- GetJobEnvironmentValues
- GetJobResultMapExtent

Summary

- **Geoprocessing and Server**
- **Publishing**
 - Publish the container of the model tool
 - Toolbox
 - Map
- **Authoring**
 - Model / Script tools contains substance
 - Model / Script tools need to have certain characteristics to be published

Questions?

Geoprocessing Sessions of Note

- **General Geoprocessing**

- Wednesday

- 3:15 to 4:30 – **Effective Analysis & Data Management** – Rm 1A/B

- **Model Builder**

- Wednesday

- 1:30 to 2:45 – **Model Builder An Introduction** – Rm 5A/B

- 3:15 to 4:30 – **Model Builder Advanced Techniques** – Rm 5A/B

- Friday

- 9:00 to 10:15 – **Model Builder Advanced Techniques** – Rm 5A/B

- **Python Scripting**

- Tuesday

- 3:15 to 4:30 – **Python Scripting Advanced Techniques** – Rm 5A/B

- Wednesday

- 8:30 to 9:45 – **Building Geoprocessing Tools with Python** – Rm 7A/B

- Thursday

- 8:30 to 9:45 – **Python Scripting An Introduction** – Rm 5A/B

- 10:15 to 11:30 – **Python Scripting Advanced Techniques** – Rm 5A/B

- 1:30 to 2:45 – **Building Geoprocessing Tools with Python** – Rm 7A/B