Deep Dive into Geoprocessing Services

Deelesh Mandloi and Monica Joseph

SEE
WHAT
OTHERS
CAN'T

Slides and Demos

- Download Slides and demos for this presentation at

http://esriurl.com/uc19ddgp

Deep dive into Geoprocessing Services

- Automating publishing through python
- Consuming Geoprocessing Services
  - Some concepts and best practices
- Administering Geoprocessing Services
  - Jobs directory, managing task messages, Large Inputs/Outputs
-

Automating Publishing through Python

Publishing Geoprocessing Service

Points to remember:

- Geoprocessing Services can be published from ArcGIS Pro and ArcMap
- Geoprocessing Service can be published to Stand-alone Server* or Federated Server*

Stand-alone Server: A Server that has it's own sharing and security model.

Federated Server: A server that is part of ArcGIS Enterprise and sharing and security is handled by ArcGIS Portal

Federated Vs Stand-alone server: https://enterprise.arcgis.com/en/server/latest/administer/windows/inside-an-arcgis-server-site.htm

Publishing Services from ArcGIS Pro through Python - Advantages

- Python/Publishing script code can be executed from any Windows or Linux machine with ArcGIS Pro-ArcPy and ArcGIS Python API modules.
- Services can be published to Stand-alone ArcGIS Server or ArcGIS Enterprise Federated Server.
- SD files can be created without server connection files* and can be published through ArcGIS Manager.

Server Connection File

A file with ags extension that stores server credentials and allows access to resources on the server and publish services to the server.

How to publish with server connection file?

1. Run geoprocessing model, scripts and tools to create results
2. Use arcpy.CreateGPSDDraft() to create a SDDraft file from results
3. Use arcpy.server.StageService to create a SD file
4. Use arcpy.server.UploadServiceDefinition geoprocessing tool to upload and publish.

How to publish with server connection file – code snippet

```
import arcpy
# Run the tool and set to a result object
result = arcpy.analysis.Buffer("c:\data.gdb\inputfc", "c:\scratch.gdb\outputfc", "1 Mile")
# Create service definition draft and returns analyzer messages
sddraft = "c:/gis/gp/drafts/Buffer.sddraft"
serverconnectionfile = "c:/gis/gp/myserver.ags"
serviceName = "MyBuffer"
analyzeMessages = arcpy.CreateGPSDDraft( result, sddraft, serviceName, server_type="ARCGIS_SERVER",
                                         connection_file_path= serverconnectionfile)
# Stage and upload the service
sd = "c:/gis/gp/sd/AnalysisReport.sd"
arcpy.StageService_server(sddraft, sd)
arcpy.UploadServiceDefinition_server(sd, serverconnectionfile)
```

How to publish without connection file?

1. Copy data to Server manually and ensure it is in the same data path as publishing machine.
2. Run geoprocessing model, scripts and tools to create results
3. Use arcpy.CreateGPSDDraft() to create a SDDraft file from results
4. Use arcpy.server.StageService to create a SD file
5. Use ArcGIS Python API to upload and publish SD file.

How to publish without connection file?

```
import arcpy
from arcgis.gis import server
# Run the tool and set to a result object
result = arcpy.analysis.Buffer("c:\data.gdb\inputfc", "c:\scratch.gdb\outputfc", "1 Mile")
# Create service definition draft and returns analyzer messages
sddraft = "c:/gis/gp/drafts/AnalysisReport.sddraft"
serviceName = "MyBuffer"
analyzeMessages = arcpy.CreateGPSDDraft( result, sddraft, serviceName, server_type="ARCGIS_SERVER",
                                         copy_data_to_server=False)
# Stage and upload the service
sd = "c:/gis/gp/sd/AnalysisReport.sd"
arcpy.StageService_server(sddraft, sd)
```

Contd….

How to publish without connection file : Contd..

```
# Publish SD through ArcGIS Python API
# server_url can be federated server or stand-alone server
server_url = https://myserver.esri.com/server
# Use username and password to publish instead of connection file
username = "adminuser"
password = "mypassword"
my_server = server.Server(server_url, username=username, password=password)
my_server.publish_sd(sd_file, service_folder)
```

CreateGPSDDraft – Special parameters

Constant_Values - Optional Parameter

- Creates a constant value for the listed parameter names for every execution of the Geoprocessing Service task. The parameter will not be exposed as task parameter in the Geoprocessing Service task.
- Example: A Buffer Geoprocessing Service task that creates 1 Mile buffer always

```
result = arcpy.analysis.Buffer("c:\data.gdb\inputfc", "c:\scratch.gdb\outputfc", "1 Mile")
arcpy.CreateGPSdDraft(result, "c:\data\Buffer.sddraft", "BufferTestParam",
                                        constantValues = [“Buffer.buffer_distance_or_field"])
```

CreateGPSDDraft: **http://esriurl.com/creategpsddraft**

CreateGPSDDraft – Special parameters

ChoiceLists - Optional Parameter

- Allows you to specify choice list values for GPString datatype.
- Example: A CreateDriveTimeArea Geoprocessing Service task with choicelist options for travelling mode

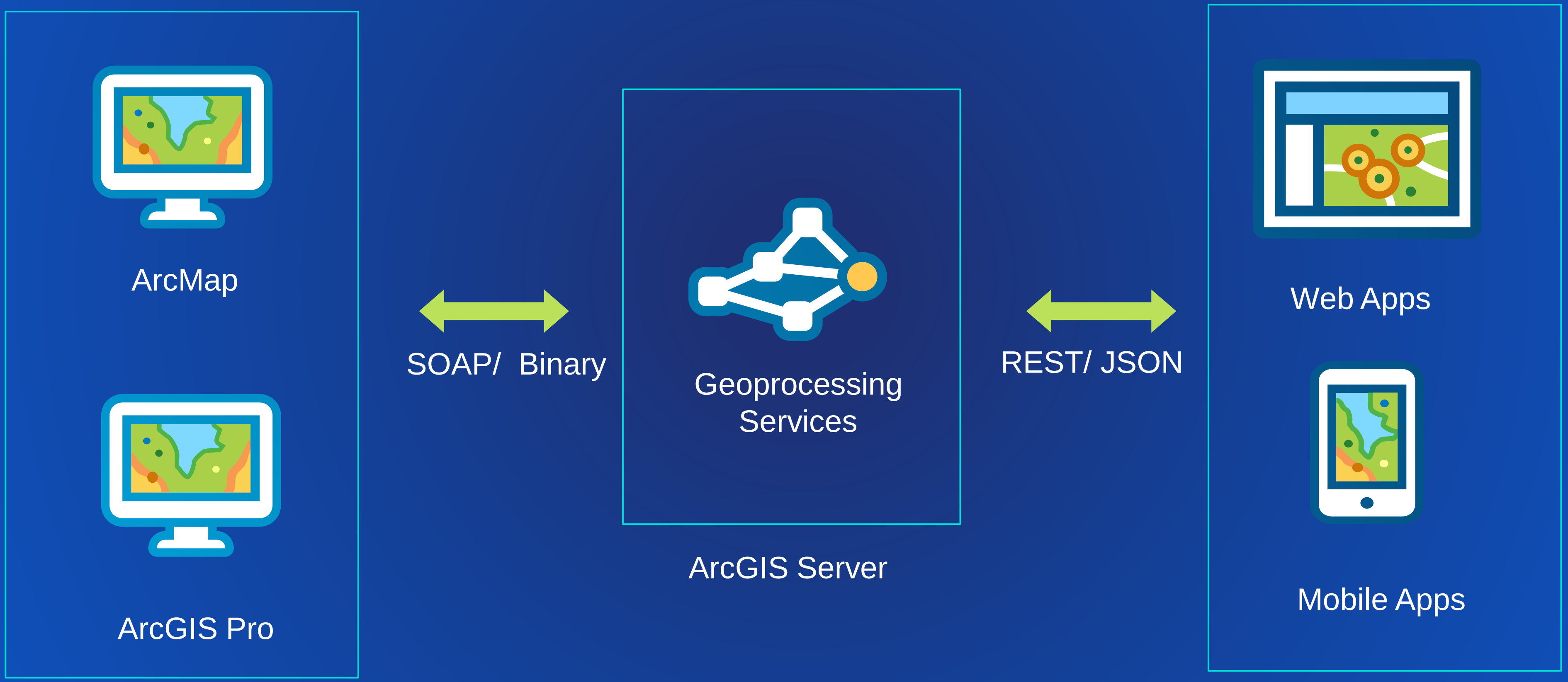
```
arcpy.CreateGPSdDraft(result, “C:\data\CDTA.sddraft", “CDTATestParam",
         “CDTA.Travel_Mode": ["Driving Time", "Walking Time"])
```

https://pro.arcgis.com/en/pro-app/arcpy/functions/creategpsddraft.htm

Demo 1 – Automating Publishing through Python

Consuming Geoprocessing Services

Geoprocessing Service and Client Communications



ArcMap

ArcGIS Pro

SOAP/ Binary

Geoprocessing Services

ArcGIS Server

REST/ JSON

Web Apps

Mobile Apps

Input data is transferred from client to server, tools are executed on the server, output data is transferred from server to client

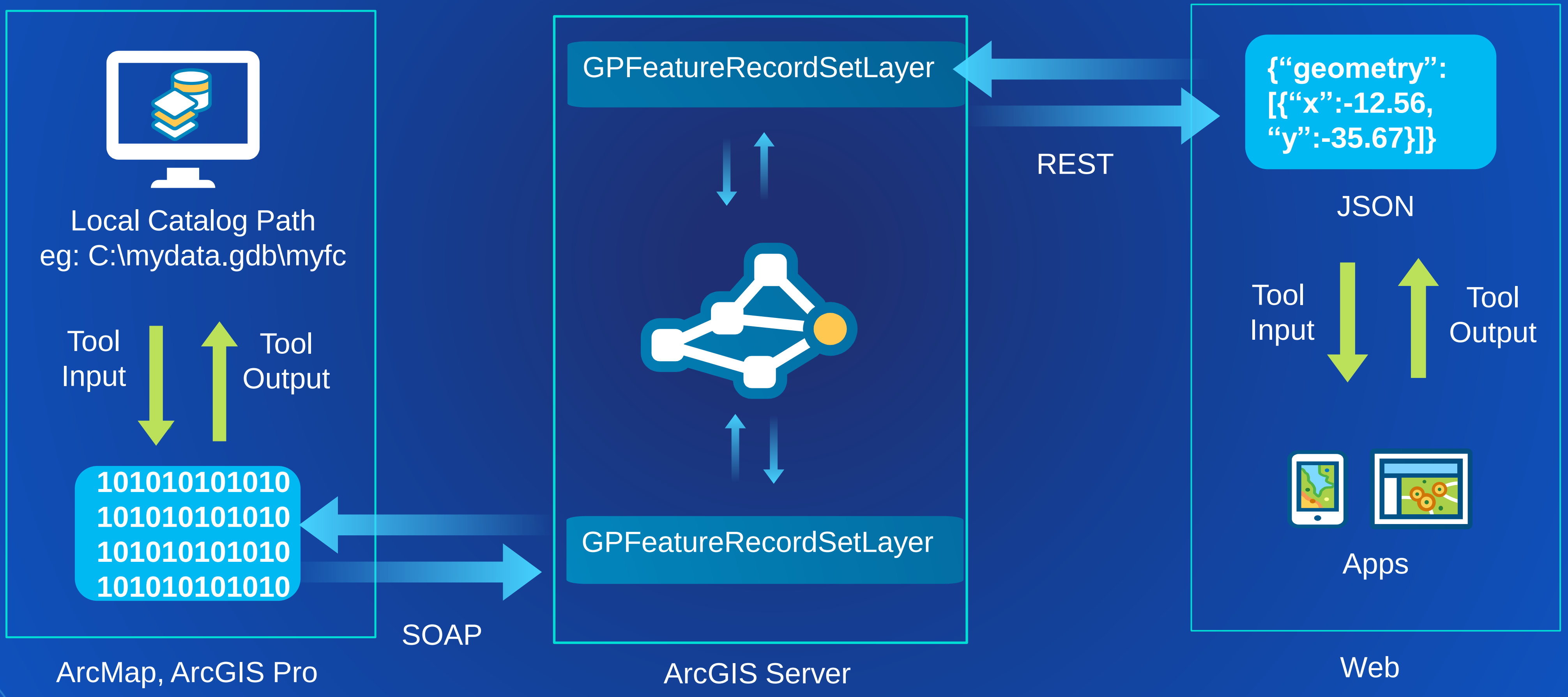
Main criteria for Input and Output Data

- Data should be transferable from client to server.
  - Should have a JSON representation for REST protocol
  - Should support a binary representation for SOAP protocol

- GPService Supported DataTypes:
  - GPFeatureRecordSetLayer/GPRecordSet (Vector and Tabular Data)
  - GPRasterDataLayer (Raster Data)
  - GPDataFile (File)
  - GPBoolean, GPDouble, GPLong, GPString, GPLinearUnit (Simple Data types)

Supported in 10.5 and higher versions

GPFeatureRecordSetLayer - Data transfer between Client and Server



GPFeatureRecordSetLayer JSON: FeatureCollection Vs URL

- Two types of popular representation
  1. FeatureCollection
  2. Feature Service or MapService URL (Supported in 10.5 and higher versions)

GPFeatureRecordSetLayer: FeatureCollection

- JSON Representation

```
{ "geometryType" : "esriGeometryPoint",
  "fields":[ {"name":"Id","type“: "esriFieldTypeOID","alias":"Id"},  {"name":"Name","type":"esriFieldTypeString","alias":"Name“}],
 "spatialReference" : {"wkid" : 4326},
 "features" : [ { "geometry" : {"x" : -104.44, "y" : 34.83,   "spatialReference" : {"wkid" : 4326}}, "attributes" : {"Id" : 43,"Name" : "Feature 1"}},
                { "geometry" : {"x" : -100.65, "y" : 33.69,  "spatialReference" : {"wkid" : 4326}},  "attributes" : {"Id" : 42,"Name" : "Feature 2"}} ]}
```

- Advantages
  - Suitable for small inputs; ArcGIS Server has a 10 MB limit on JSON payloads
- Disadvantages
  - Slow for large outputs : Causes out-of-memory exception often

GPFeatureRecordSetLayer: URL

JSON Representation:

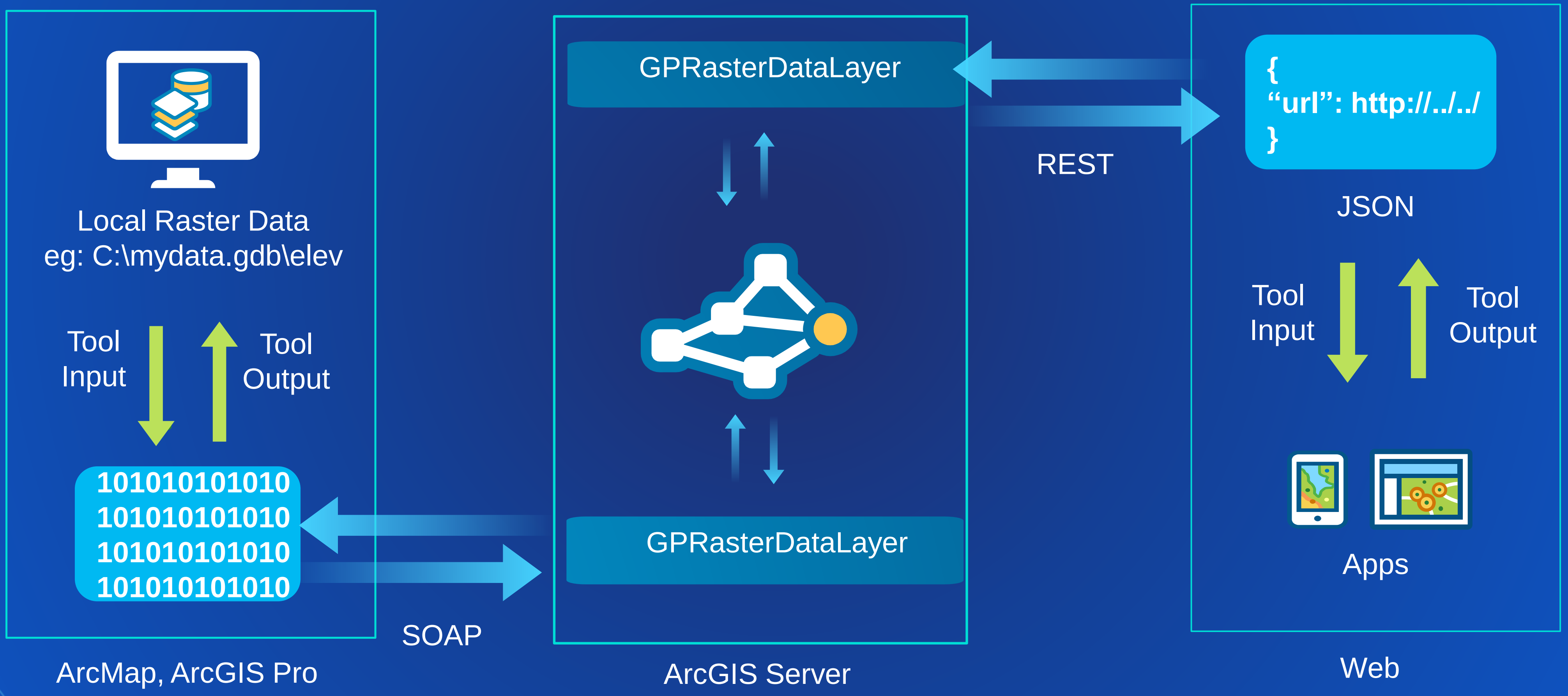
```
{
  “url”: “http://portal.esri.com/portal/rest/services/hosted/USEarthQuakes/FeatureServer/0”,
  “filter”: “STATE_ABBR = ‘CA’”,
  “time”: “1559768445, 1562360445”
}
```

Advantages:

- Access the feature data locally for feature services published on the same portal
- Downloads the data for map services and external feature services
- Recommended for large input data

Time filter supported only on services published from ArcGIS Pro from 10.7.1 onwards

GPRasterDataLayer - Data transfer between Client and Server

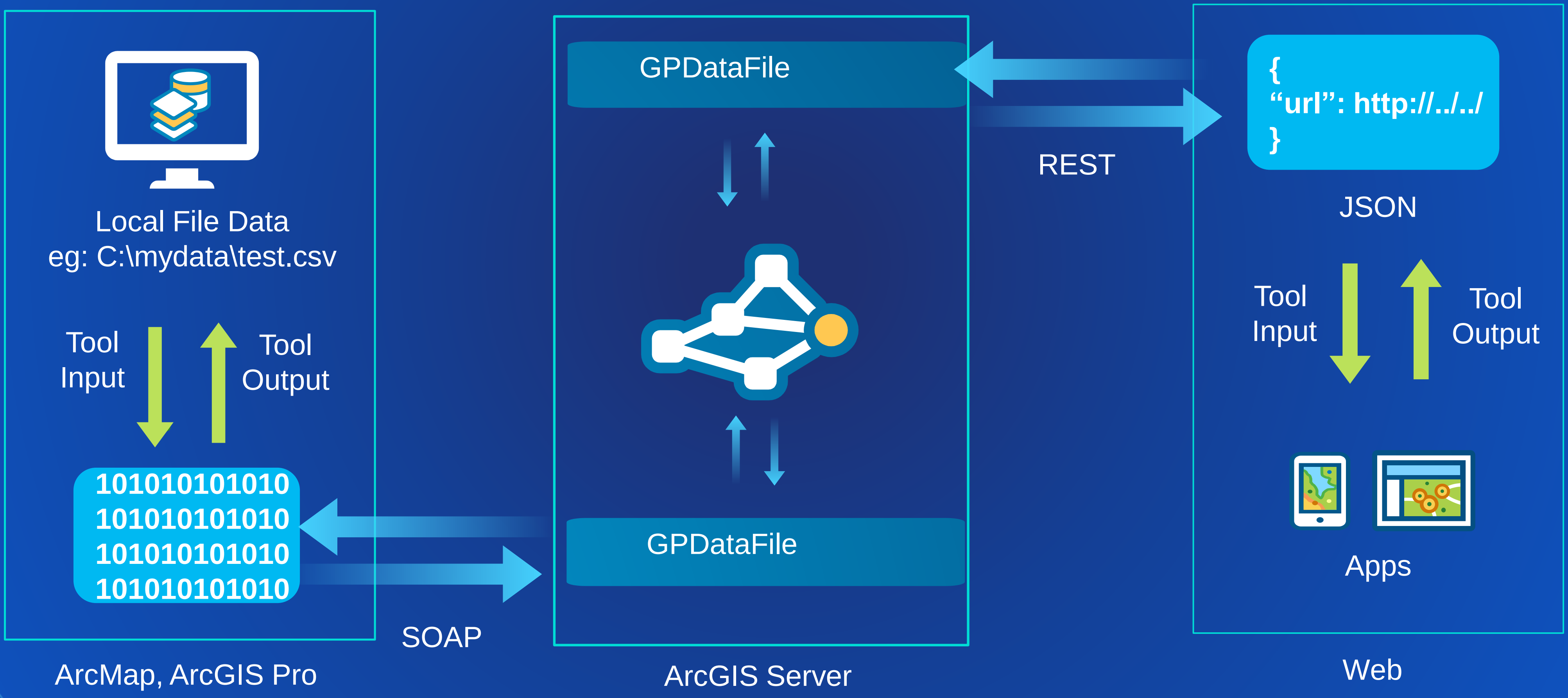


GPRasterDataLayer

- JSON Representation
  1. Images on your webserver
     - { “url”: “http://mywebserver/lake.tif”, “format”:”tif” }
  2. Uploaded image itemid to ArcGIS Server uploads directory
     - { “itemID”: <itemID> }
  3. Image Service Url (New at 10.7.1)
     - {“url”: “https://myportal.esri.com/server/rest/services/elevation/ImageServer”}

ImageService URL inputs are supported only for services published from ArcGIS Pro

GPDataFile - Data transfer between Client and Server

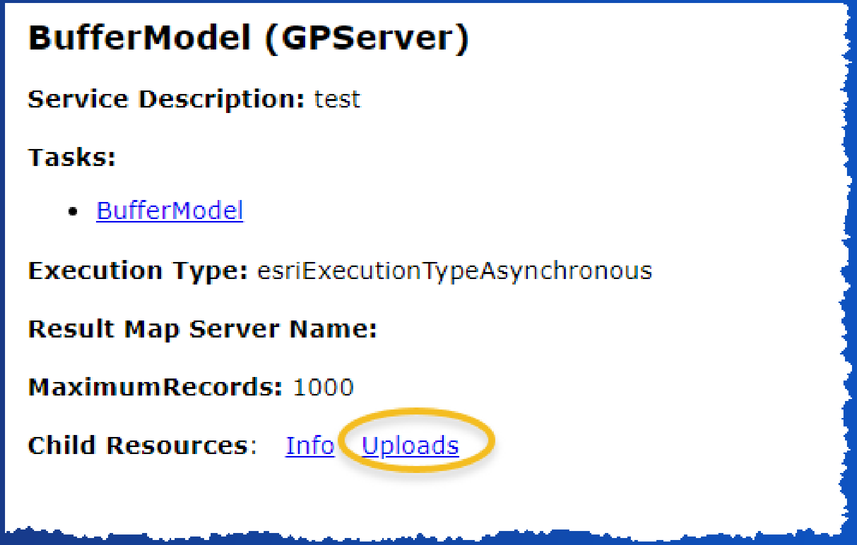


GPDataFile

- JSON Representation
  1. Files on your webserver
     - { “url”: “http://mywebserver/lake.tif” }
  2. Uploaded file itemid to ArcGIS Server uploads directory
     - { “itemID”: <itemID> }

GPRasterDataLayer/GPDataFile – Best Practice

- Best Practices:
  - Use arcgisuploads option to upload files/images to server
  - Advantages:
    - Does not require external web servers
    - Uploaded files lifespan is managed by server
  - Gotchas
    - Filter file types using allowedFileTypes to avoid unnecessary uploads.
    - Set MaxUploadSize to limit the size of the uploaded file



BufferModel (GPServer)

Service Description: test

Tasks:

- BufferModel

Execution Type: esriExecutionTypeAsynchronous

Result Map Server Name:

MaximumRecords: 1000

Child Resources: Info Uploads

Supported from 10.1 onwards

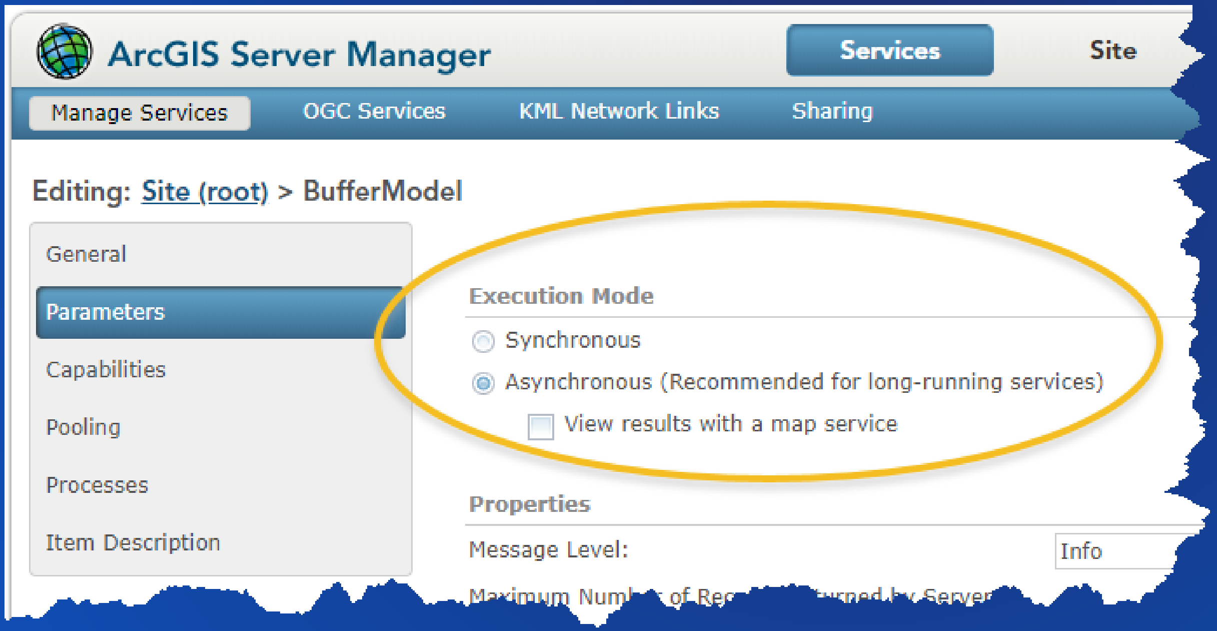
Demo 2 – Best Practices for consuming Geoprocessing Services

Administering Geoprocessing Services

Section Subhead

Types of Geoprocessing Services

- Synchronous Geoprocessing Service – Execute operation
- Asynchronous Geoprocessing Service – SubmitJob Operation



Editing: Site (root) > BufferModel

Execution Mode

○ Synchronous

● Asynchronous (Recommended for long-running services)

□ View results with a map service

Properties

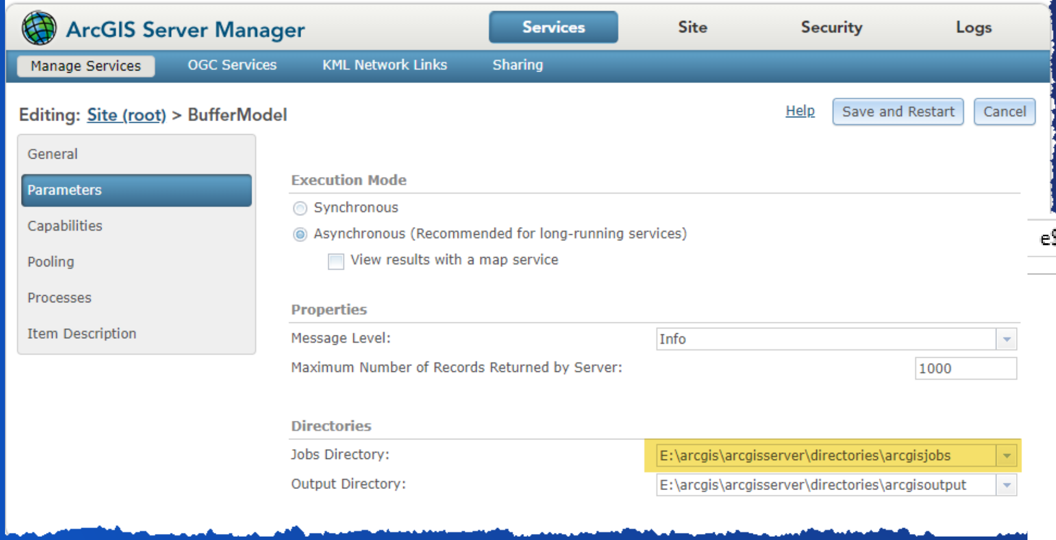
Message Level: Info

Maximum Num of Rec urned by Server

More Info: Add Help Url

Asynchronous Services – Jobs Directory

- What is Jobs directory:
  - Scratch workspace for every asynchronous job submitted to the server and also has information on status, results and message of the job.
- Where is Jobs directory?
  - A property of Geoprocessing Service, can be configured through REST Admin directory



Editing: Site (root) > BufferModel

Execution Mode

○ Synchronous

● Asynchronous (Recommended for long-running services)

□ View results with a map service

Properties

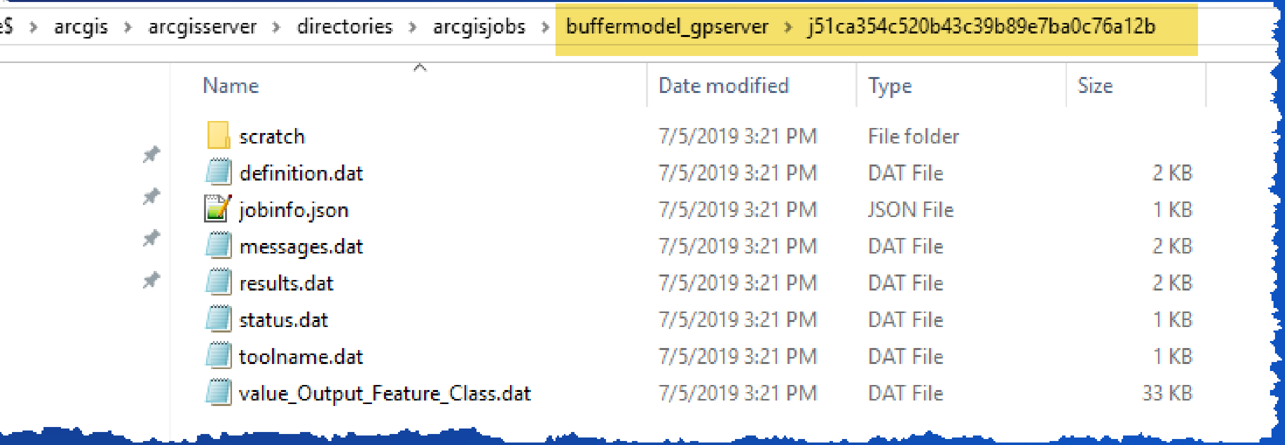
Message Level: Info

Maximum Number of Records Returned by Server: 1000

Directories

Jobs Directory: E:\arcgis\arcgisserver\directories\arcgisjobs

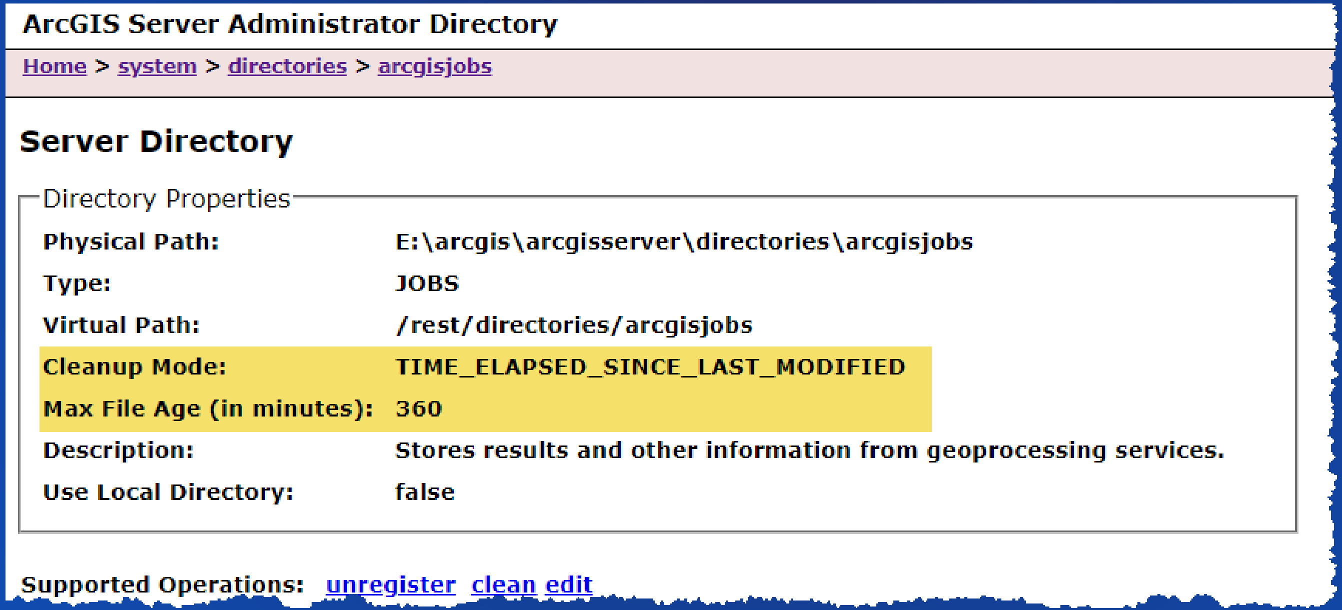
Output Directory: E:\arcgis\arcgisserver\directories\arcgisoutput


e$ › arcgis › arcgisserver › directories › arcgisjobs › buffermodel_gpserver › j51ca354c520b43c39b89e7ba0c76a12b

Name	Date modified	Type	Size
scratch	7/5/2019 3:21 PM	File folder	
definition.dat	7/5/2019 3:21 PM	DAT File	2 KB
jobinfo.json	7/5/2019 3:21 PM	JSON File	1 KB
messages.dat	7/5/2019 3:21 PM	DAT File	2 KB
results.dat	7/5/2019 3:21 PM	DAT File	2 KB
status.dat	7/5/2019 3:21 PM	DAT File	1 KB
toolname.dat	7/5/2019 3:21 PM	DAT File	1 KB
value_Output_Feature_Class.dat	7/5/2019 3:21 PM	DAT File	33 KB

Asynchronous Services – Jobs Directory

- Administering Jobs Directory
  - Set Job directory cleanup: : Default is 6 hours
  - Set independent Jobs directory for services that require long execution time



ArcGIS Server Administrator Directory

Home > system > directories > arcgisjobs

Server Directory

Directory Properties

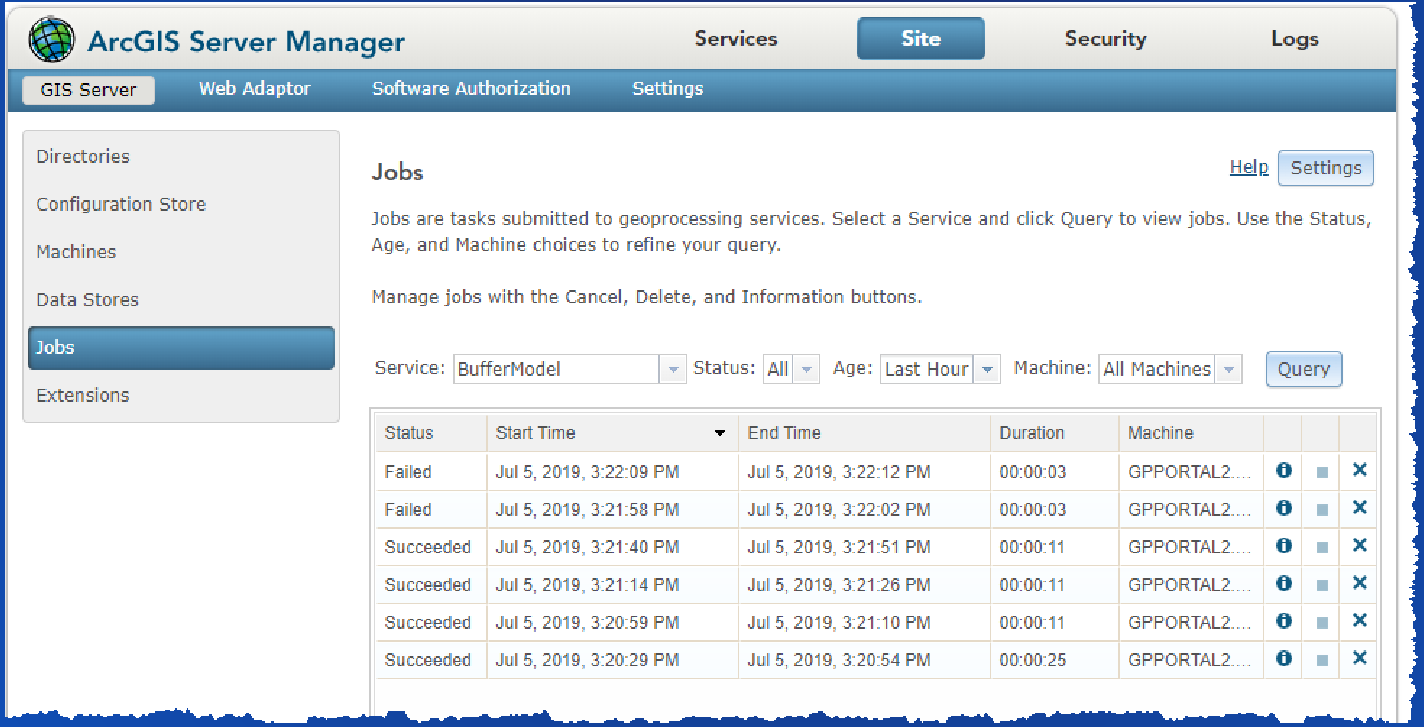
Physical Path:	E:\arcgis\arcgisserver\directories\arcgisjobs
Type:	JOBS
Virtual Path:	/rest/directories/arcgisjobs
Cleanup Mode:	TIME_ELAPSED_SINCE_LAST_MODIFIED
Max File Age (in minutes):	360
Description:	Stores results and other information from geoprocessing services.
Use Local Directory:	false

Supported Operations: unregister clean edit

Editing Jobs Directory: http://esriurl.com/editjobsdirectory

Administration – Jobs in ArcGIS Manager

- Query Jobs based on service, status, time, machine name and more
- Cancel or Delete any running jobs



Jobs

Jobs are tasks submitted to geoprocessing services. Select a Service and click Query to view jobs. Use the Status, Age, and Machine choices to refine your query.

Manage jobs with the Cancel, Delete, and Information buttons.

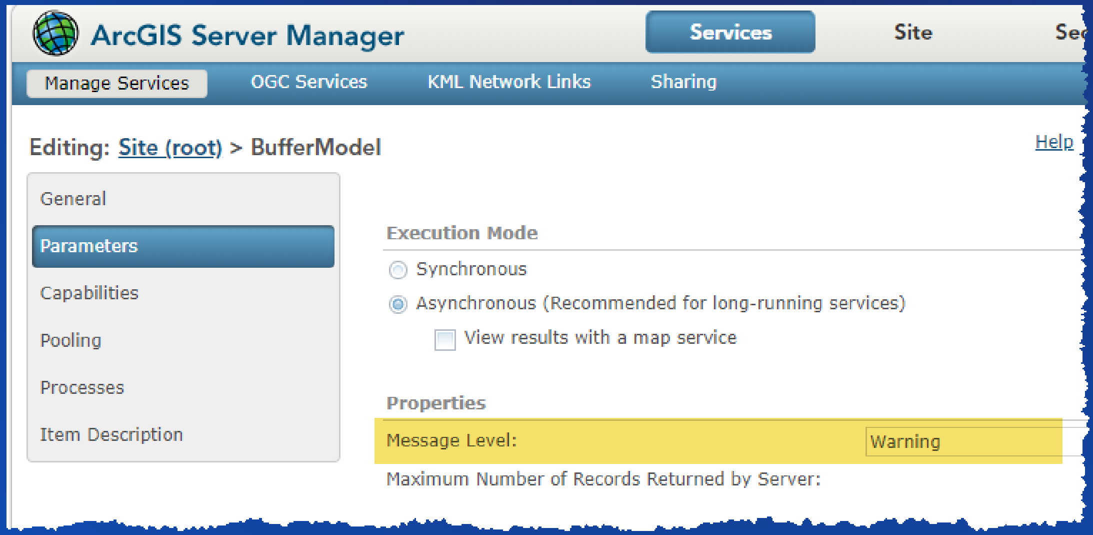
Service: BufferModel Status: All Age: Last Hour Machine: All Machines

Status	Start Time	End Time	Duration	Machine			
Failed	Jul 5, 2019, 3:22:09 PM	Jul 5, 2019, 3:22:12 PM	00:00:03	GPPORTAL2....	ℹ	■	✕
Failed	Jul 5, 2019, 3:21:58 PM	Jul 5, 2019, 3:22:02 PM	00:00:03	GPPORTAL2....	ℹ	■	✕
Succeeded	Jul 5, 2019, 3:21:40 PM	Jul 5, 2019, 3:21:51 PM	00:00:11	GPPORTAL2....	ℹ	■	✕
Succeeded	Jul 5, 2019, 3:21:14 PM	Jul 5, 2019, 3:21:26 PM	00:00:11	GPPORTAL2....	ℹ	■	✕
Succeeded	Jul 5, 2019, 3:20:59 PM	Jul 5, 2019, 3:21:10 PM	00:00:11	GPPORTAL2....	ℹ	■	✕
Succeeded	Jul 5, 2019, 3:20:29 PM	Jul 5, 2019, 3:20:54 PM	00:00:25	GPPORTAL2....	ℹ	■	✕

Jobs in ArcGIS Manager : http://esriurl.com/jobsmanager

Administration – Managing Message Level

- **Best Practices:**
  1. **Set Message level to Warning until and unless debugging**
  2. **Avoid adding too many messages in your tool, slows down performance**
  3. **For Asynchronous Services, use messages.xml in scratch folder to get more elaborate info for debugging even if the message level is set to Warning**



Editing: Site (root) > BufferModel

Execution Mode

○ Synchronous

● Asynchronous (Recommended for long-running services)

□ View results with a map service

Properties

Message Level: Warning

Maximum Number of Records Returned by Server:

Administration – SOC Heap Size

- **For large inputs and Output JSON increase SOC Heap Size – Default is 64MB**

- **When to increase heap size?**
  - **< 10.7.1 server :**
    - **Server log messages: java.lang.OutOfMemoryError: Java heap space**
  - **> =10.7.1 server:**
    - **Server log messages*: The SOC for this service may not have enough memory allocated to it. You may consider increasing the SOC heap size for this service.***

Increasing SOC Heap Size per Geoprocessing service is supported from 10.7.1 onwards

Administration – SOC Heap Size

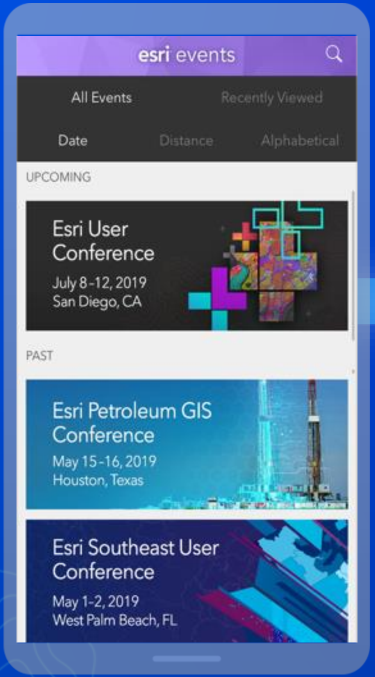
- **How to increase heap size?**
  - < 10.7.1 server :
    1. Go to Admin directory: https://<host>:<port>/<instance>/admin/machines/<machine name>
    2. Edit SOC MAX Heap Size property
    - Gotchas:
      - Increases heap size for all ArcSOC services instances on the machine.
  - > = 10.7.1 server – Increased per service
    1. Go to Admin directory: https://<host>:<port>/<instance>/admin/services/<serviceName>.GPServer
    2. Add {"frameworkProperties":{"javaHeapSize":128MB}} to service properties JSON

Geoprocessing Advanced Settings: http://esriurl.com/gpadvsettings

Demo 3 – Administering Geoprocessing Services

Please Share Your Feedback in the App

Download the Esri Events app and find your event



esri events

All Events | Recently Viewed

Date | Distance | Alphabetical

UPCOMING

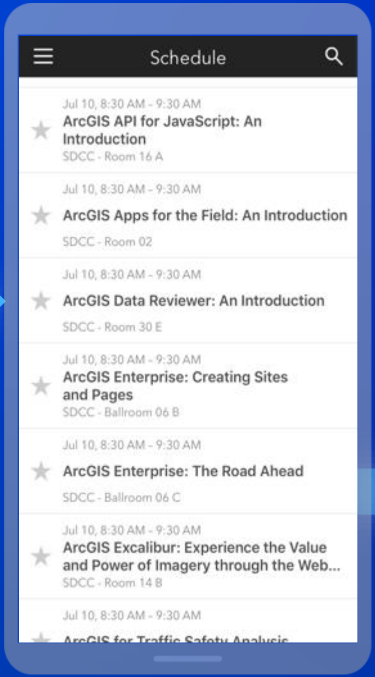
Esri User Conference
July 8-12, 2019
San Diego, CA

PAST

Esri Petroleum GIS Conference
May 15-16, 2019
Houston, Texas

Esri Southeast User Conference
May 1-2, 2019
West Palm Beach, FL

Select the session you attended



Schedule

Jul 10, 8:30 AM - 9:30 AM
ArcGIS API for JavaScript: An Introduction
SDCC - Room 16 A

Jul 10, 8:30 AM - 9:30 AM
ArcGIS Apps for the Field: An Introduction
SDCC - Room 02

Jul 10, 8:30 AM - 9:30 AM
ArcGIS Data Reviewer: An Introduction
SDCC - Room 30 E

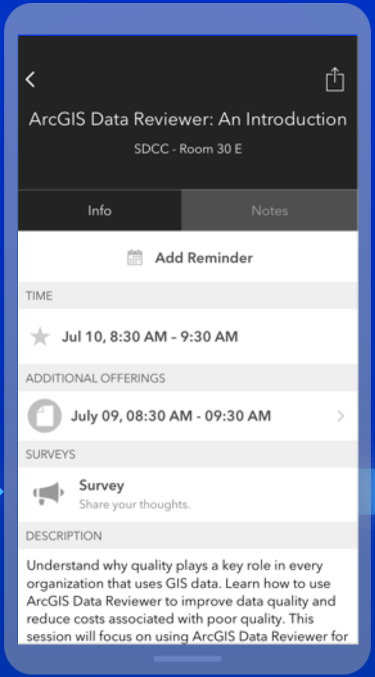
Jul 10, 8:30 AM - 9:30 AM
ArcGIS Enterprise: Creating Sites and Pages
SDCC - Ballroom 06 B

Jul 10, 8:30 AM - 9:30 AM
ArcGIS Enterprise: The Road Ahead
SDCC - Ballroom 06 C

Jul 10, 8:30 AM - 9:30 AM
ArcGIS Excalibur: Experience the Value and Power of Imagery through the Web...
SDCC - Room 14 B

Jul 10, 8:30 AM - 9:30 AM
ArcGIS for Traffic Safety Analysis

Scroll down to “Survey”



ArcGIS Data Reviewer: An Introduction
SDCC - Room 30 E

Info | Notes

Add Reminder

TIME

Jul 10, 8:30 AM - 9:30 AM

ADDITIONAL OFFERINGS

July 09, 08:30 AM - 09:30 AM

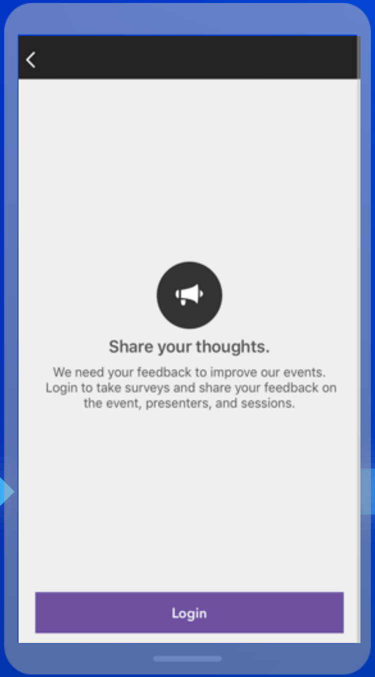
SURVEYS

Survey
Share your thoughts.

DESCRIPTION

Understand why quality plays a key role in every organization that uses GIS data. Learn how to use ArcGIS Data Reviewer to improve data quality and reduce costs associated with poor quality. This session will focus on using ArcGIS Data Reviewer for

Log in to access the survey

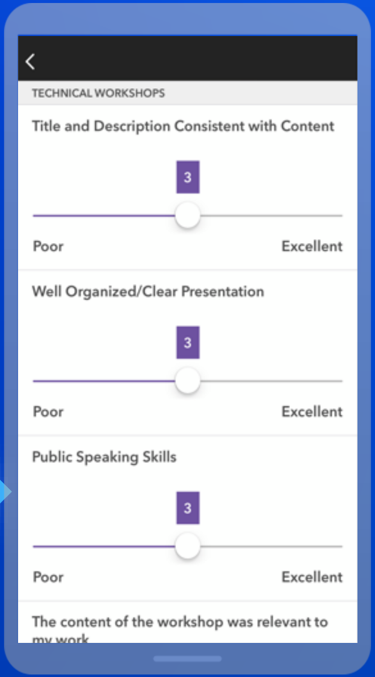


Share your thoughts.

We need your feedback to improve our events. Login to take surveys and share your feedback on the event, presenters, and sessions.

Login

Complete the survey and select “Submit”



TECHNICAL WORKSHOPS

Title and Description Consistent with Content
3
Poor | Excellent

Well Organized/Clear Presentation
3
Poor | Excellent

Public Speaking Skills
3
Poor | Excellent

The content of the workshop was relevant to my work

Questions

Deep Dive into Geoprocessing Services