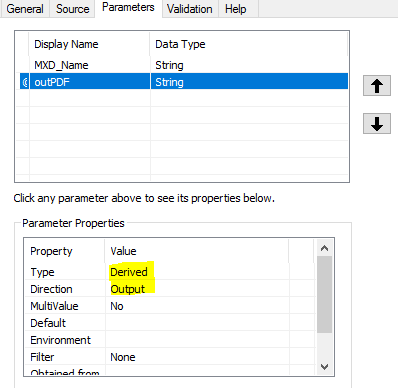
**ArcGIS JS API and Geoprocessing Services**

1. **Geoprocessing Service**

Create a python script that takes any required input parameters, using arcpy.GetParameterAsText(index\_number). In order to reference results, include an output parameter as well, arcpy.SetParameterAsText(index number, output\_variable or string). This may be the name of a file generated, or any other string that may be needed.

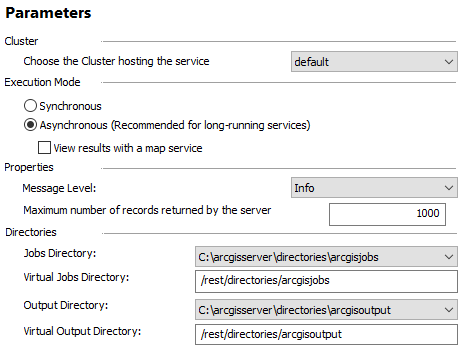
Once the python file has been tested, import it into a toolbox using ArcCatalog. Right-click on a toolbox and choose Add>Script. After filling in the script name and description, choose the python file just created.

On the parameters page, specify all of the input parameters, as well as the output parameter, in the proper order. Use the index numbers specified in the python code. For the output parameter, be sure to set it to “Derived” and “Output” in the properties.



Once the script has been set up in the toolbox, try running it. Upon a successful run, go to Geoprocessing>Results to pull up the Results window. Find the task just completed, and right-click. Choose Share As>Geoprocessing Service to publish a new service.

On the service parameters, be sure that the Execution Mode is set to Asynchronous, unless the script executes very fast (i.e. less than 5 sec). The mode that is chosen will influence the job call through JavaScript later on.



After publishing, navigate to the REST service that was just created. Under Supported Operations, click Submit Job (or Execute of Synchronous was chosen).

