**A Case Study: Using the InPort API as part of a process to convert multiple ArcGIS metadata records and load those records to InPort**

***General Workflow***

1. The GIS user creates a **new** ArcGIS Metadata record, using the Metadata Editor in ArcGIS Pro, for all datasets that will have metadata records loaded to InPort. The user ensures that all elements required for InPort XML metadata are present and populated
2. The user then searches the InPort website to find the project that the metadata records will be placed under. The catalog id is then recorded in the ArcGIS Metadata element (for all records): <mdParentID>gov.noaa.nmfs.inport:75873</mdParentID>
   1. Each ArcGIS Metadata record is checked for the presence of the mdParentID element using Python. If missing, then the element is added and populated
   2. Updating the InPort API search function so that it is possible to get the catalog id for a specific project (load new datasets metadata) or a dataset (updating existing datasets metadata) would be very useful
3. The Python code then reads each ArcGIS Metadata record for the datasets and then exports a new XML document to disk that was transformed using the XSLT file created by Tim
   1. The XSLT file was modified slightly to record values in the ArcGIS Metadata elements **mdFileID** and **mdParentID** elements and transfer that data to the InPort elements **catalog-item-id** and **parent-catalog-item-id**. Since this is a new record, the **mdParentID** element is populated and the value is then recorded in the InPort element **parent-catalog-item-id** when the XSLT transform is used
   2. For existing InPort records both the **mdFileID** and **mdParentID** elements will have catalog ids. The XSLT file will need to be modified to check both elements
      1. If **mdFileID** is blank and **mdParentID** has a value, then this is a new InPort record. The XSLT will populate the **parent-catalog-item-id** only
      2. If **mdFileID** and **mdParentID** have values, then this InPort record will be updated. The XSLT will populate the **catalog-item-id** only
4. The Python code then uses the /inport/api/load-inport-xml function to load the XML to InPort
5. The user then searches the InPort website to find each new InPort record and copies the catalog id value into the ArcGIS Metadata **mdFileID** element
   1. Updating the InPort API search function so that it is possible to get the catalog id for each specific dataset would be very useful in updating the ArcGIS Metadata records with both **mdFileID** and **mdParentID** element values