Subject: IDL-Python bridge with GDAL/OGR
Posted by Gordon Farquharson on Fri, 30 Sep 2016 17:51:41 GMT
View Forum Message <> Reply to Message

Hi All

What am I doing wrong in the following code:

```
IDL> osr = python.import('osgeo.osr')
% Loaded DLM: PYTHON27.
IDL> srs = osr.SpatialReference()
IDL> srs
<osgeo.osr.SpatialReference; proxy of <Swig Object of type 'OSRSpatialReferenceShadow *' at
0x7f60567008d0> >
IDL> srs.__doc__
Proxy of C++ OSRSpatialReferenceShadow class
IDL> srs.ImportFromProj4('+proj=utm +zone=10 +ellps=WGS84 +datum=WGS84 +units=m
+no_defs')
% Python: Unknown method: "IMPORTFROMPROJ4"
% Execution halted at: $MAIN$
IDL> >>>srs.ImportFromProj4('+proj=utm +zone=10 +ellps=WGS84 +datum=WGS84 +units=m
+no_defs')
% PYTHON_RUN: Exception: name 'srs' is not defined.
% Execution halted at: $MAIN$
     IDL> python.run("srs.ImportFromProj4('+proj=utm +zone=10 +ellps=WGS84 +datum=WGS84
+units=m +no_defs')")
```

According to the class definition of of SpatialReference [1], the method ImportFromProj4 does exist. The code works in Python on my system:

```
Python 2.7.9 (default, Jun 29 2016, 13:08:31)
[GCC 4.9.2] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>  import osgeo.osr
>>>  srs = osgeo.osr.SpatialReference()
>>>  srs.ImportFromProj4('+proj=utm +zone=10 +ellps=WGS84 +datum=WGS84 +units=m
+no_defs')
0
>>>  srs.ExportToWkt()
'PROJCS["UTM Zone 10, Northern Hemisphere",GEOGCS["WGS
84",DATUM["WGS_1984",SPHEROID["WGS
84",6378137,298.257223563,AUTHORITY["EPSG","7030"]],TOWGS84[0,0,0,0,0,0,0],AUTHORIT
Y[ "EPSG","6326"]],PRIMEM["Greenwich",0,AUTHORITY["EPSG",
"8901"]],UNIT["degree",0.0174532925199433,AUTHORITY["EPSG
","9108"]],AUTHORITY["EPSG","4326"]],PROJECTION["Transverse_Mercator
"],PARAMETER["latitude_of_origin",0],PARAMETER["central_meridian
",-123],PARAMETER["scale_factor",0.9996],PARAMETER["false_easting
",500000],PARAMETER["false_northing",0],UNIT["Meter",1]]'
```

My feeling is that I don't understand how to use the IDL-Python bridge correctly yet.

Gordon

[1] http://gdal.org/python/osgeo.osr.SpatialReference-class.html

---

Subject: Re: IDL-Python bridge with GDAL/OGR
Posted by Gordon Farquharson on Fri, 30 Sep 2016 18:21:08 GMT
View Forum Message <> Reply to Message

OK. I think I figured it out.

```
IDL> osr.SpatialReference.ImportFromProj4(srs, '+proj=utm +zone=10 +ellps=WGS84
+datum=WGS84 +units=m +no_defs')
          0
IDL> osr.SpatialReference.ExportToWkt(srs)
PROJCS["UTM Zone 10, Northern Hemisphere",GEOGCS["WGS
84",DATUM["WGS_1984",SPHEROID["WGS
84",6378137,298.257223563,AUTHORITY["EPSG","7030"]],TOWGS84[0,0,0,0,0,0,0],AUTHORIT
Y[ "EPSG","6326"]],PRIMEM["Greenwich",0,AUTHORITY["EPSG",
"8901"]],UNIT["degree",0.0174532925199433,AUTHORITY["EPSG
","9108"]],AUTHORITY["EPSG","4326"]],PROJECTION["Transverse_Mercator
"],PARAMETER["latitude_of_origin",0],PARAMETER["central_meridian
",-123],PARAMETER["scale_factor",0.9996],PARAMETER["false_easting
",500000],PARAMETER["false_northing",0],UNIT["Meter",1]]
```

But, I'm confused. Why can't I call methods of the srs object, like one does on Python?

Gordon

---

Subject: Re: IDL-Python bridge with GDAL/OGR
Posted by Jim Pendleton on Fri, 30 Sep 2016 18:52:16 GMT
View Forum Message <> Reply to Message

On Friday, September 30, 2016 at 12:21:16 PM UTC-6, Gordon Farquharson wrote:
> OK. I think I figured it out.
>
> IDL> osr.SpatialReference.ImportFromProj4(srs, '+proj=utm +zone=10 +ellps=WGS84
+datum=WGS84 +units=m +no_defs')
>           0
> IDL> osr.SpatialReference.ExportToWkt(srs)
> PROJCS["UTM Zone 10, Northern Hemisphere",GEOGCS["WGS
84",DATUM["WGS_1984",SPHEROID["WGS
84",6378137,298.257223563,AUTHORITY["EPSG","7030"]],TOWGS84[0,0,0,0,0,0,0],AUTHORIT

Y[ "EPSG","6326"]],PRIMEM["Greenwich",0,AUTHORITY["EPSG",
"8901"]],UNIT["degree",0.0174532925199433,AUTHORITY["EPSG
","9108"]],AUTHORITY["EPSG","4326"]],PROJECTION["Transverse_Mercator
"],PARAMETER["latitude_of_origin",0],PARAMETER["central_meridian
",-123],PARAMETER["scale_factor",0.9996],PARAMETER["false_easting
",500000],PARAMETER["false_northing",0],UNIT["Meter",1]]
>
> But, I'm confused. Why can't I call methods of the srs object, like one does on Python?
>
> Gordon

Instead of the syntax
IDL> srs = osr.SpatialReference()
what if you try the following?
IDL> srs = osr.SpatialReference

The error message indicates that by calling the object creation, you're getting a reference to the proxy SWIG object instead of a reference the underlying python object.

Just a guess.

Jim P.

---

Subject: Re: IDL-Python bridge with GDAL/OGR
Posted by Gordon Farquharson on Fri, 30 Sep 2016 19:02:22 GMT
View Forum Message <> Reply to Message

Hi Jim

Thanks for the suggestion.

On Friday, September 30, 2016 at 11:52:30 AM UTC-7, Jim P wrote:
> Instead of the syntax
> IDL> srs = osr.SpatialReference()
> what if you try the following?
> IDL> srs = osr.SpatialReference
>
> The error message indicates that by calling the object creation, you're getting a reference to the proxy SWIG object instead of a reference the underlying python object.

Your idea makes a difference, but results in a different error:

IDL> srs = osr.SpatialReference
IDL> srs
<class 'osgeo.osr.SpatialReference'>
IDL> srs.ImportFromProj4('+proj=utm +zone=10 +ellps=WGS84 +datum=WGS84 +units=m
+no_defs')

```
% PYTHON_CALLMETHOD: Exception: unbound method ImportFromProj4() must be
                    called with SpatialReference instance as first
                    argument (got str instance instead).
IDL> srs.ImportFromProj4(srs, '+proj=utm +zone=10 +ellps=WGS84 +datum=WGS84 +units=m
+no_defs')
% PYTHON_CALLMETHOD: Exception: unbound method ImportFromProj4() must be
                    called with SpatialReference instance as first
                    argument (got type instance instead).
```

There seem to be subtleties of the IDL-Python bridge that I do not understand.

Gordon

---