CE 2017

Agenda

Introduction

During the second day of EC2017 I attended the developers session and the last technical talk, an introduction to ArcGIS Python API.

The developers session was a light introduction to the geodevelopers.org community resources, as well as some of the key components of ArcGIS platform.

WebGL, Three.js and ArcGIS JavaScript API were briefly introduced to the attendees.

ArcGIS R-bindings, which gives access to ArcGIS users to the vast array of data analysis and statistical anylisis of R environment, was succinctly introduced to audience.

Eventually, the last technical talk gave a general overview of the ArcGIS Python API modules and functionalities.

A bit on developers

- ESRI is funding geodevelopers.org, a comunity for geodevelopers working mainly on ArcGIS technologies, APIs and apps.
- Community members can interact and access the resources via:

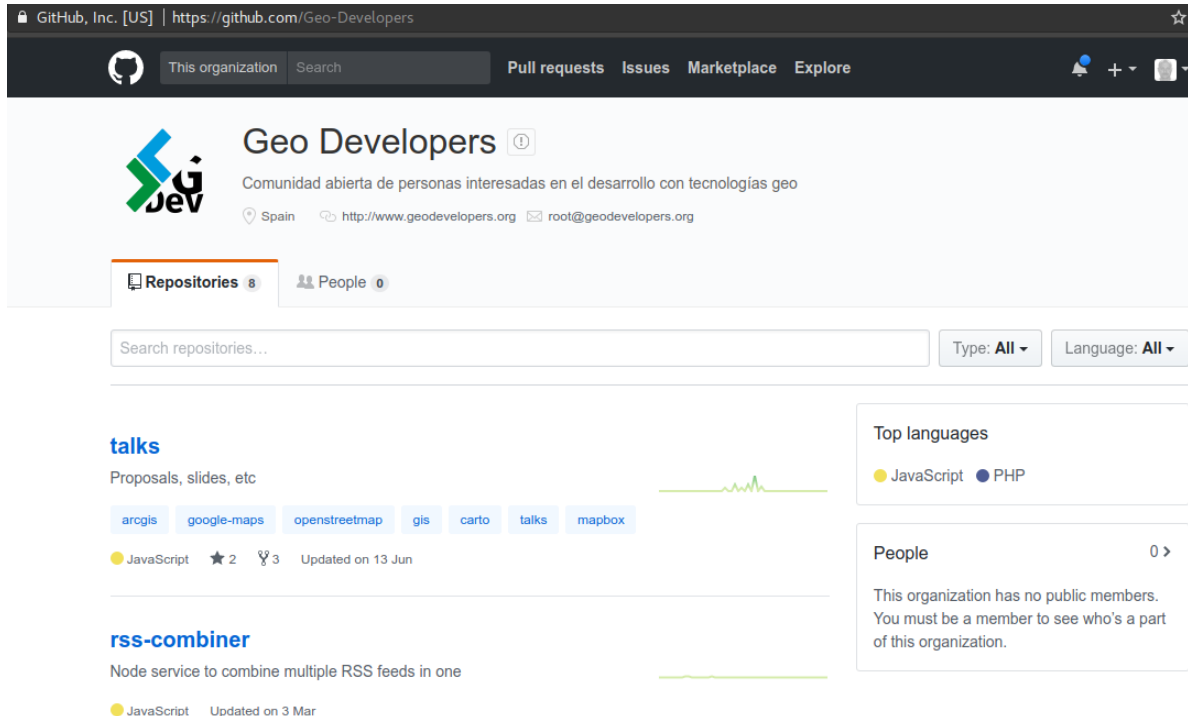Meetup: meetup.com/Geo-Developers

Twitter: twitter.com/Geo_Developers

Github: github.com/Geo-Developers

YouTube: youtube.com/GeoDevelopers

Slack: geodevelopers.org/community

Geo-Developers on GitHub

A bit of R bindings for ArcGIS

```r
tool_exec <- function(in_params, out_params)
{
  #### Load Library for Analysis ####
  if (!requireNamespace("SemiPar", quietly = TRUE))
    install.packages("SemiPar")
  require(SemiPar)

  #### Get Input Parameters ####
  input_features <- in_params[[1]]
  input_predictions <- in_params[[2]]
  dep_variable <- in_params[[3]]
  # ...
  output_features <- out_params[[1]]
  output_graph_pdf <- out_params[[2]]


  #### Import Dataset to Dataframe ####
  fc <- arc.open(input_features)
  df <- arc.select(fc, c(dep_variable, nonlin_variables, lin_variables))
  df['x'] <- arc.shape(df)$x
  df['y'] <- arc.shape(df)$y

  ###
  # script functionality
  ###

  arc.write(output_features, pred_df)
  return(out_params)
}
```
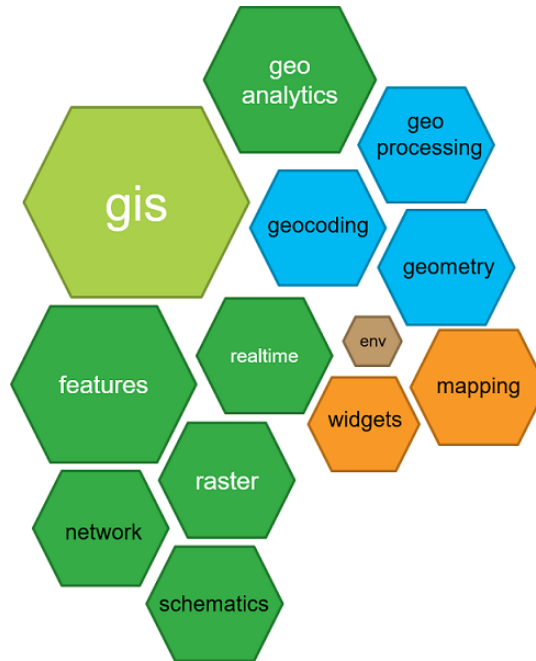
Introduction to ArcGIS Python API

- The ArcGIS API for Python provides a **pythonic representation of a GIS**.
- The API has Python *modules, classes, functions,* and *types* for managing and working with elements of the ArcGIS platform information model.

ArcGIS modules

ArcGIS API contains 13 modules which allow advanced users to perform and automatize administration, analysis and visualization tasks.

- **arcgis.gis** module
- **arcgis.env** module
- **arcgis.features** module
- **arcgis.raster** module
- **arcgis.network** module
- **arcgis.schematics** module
- **arcgis.geoanalytics** module
- **arcgis.geocoding** module
- **arcgis.geometry** module
- **arcgis.geoprocessing** module
- **arcgis.realtime** module
- **arcgis.mapping** module
- **arcgis.widgets** module

Python raster analysis

```
import arcgis
from arcgis.gis import GIS
portal = GIS("portal url", "username","password")
montana_ndvi = portal.content.search("GeneratedRasterProduct_JRA9W9")[0]
ndvi_map = portal.map(location, 12)
ndvi_map
```
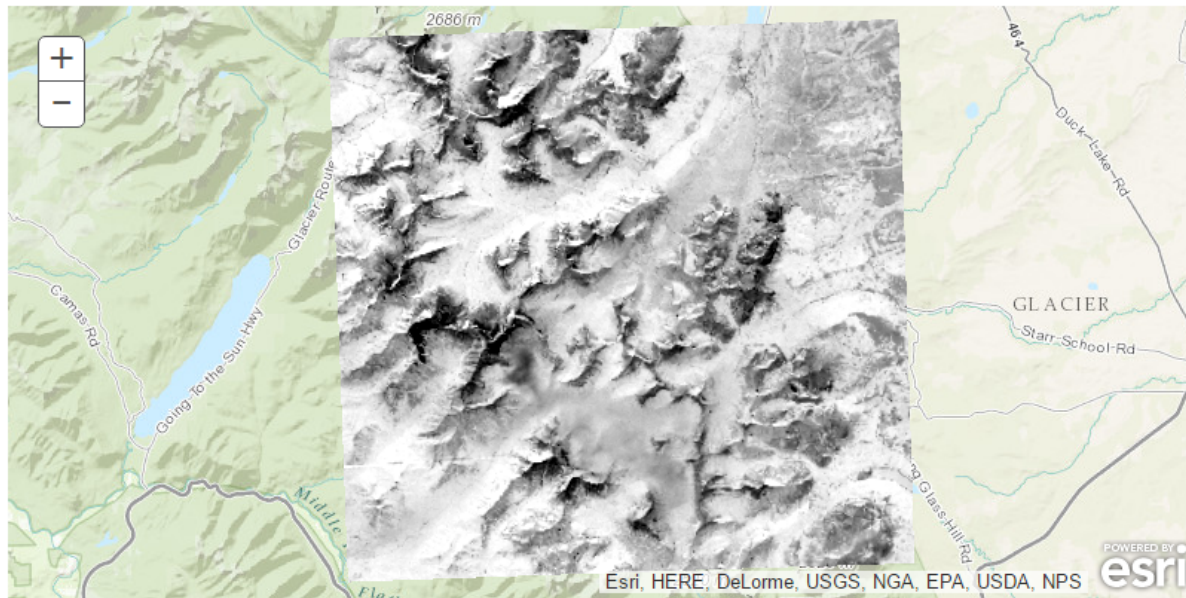
Python Mapping

```python
from arcgis.gis import *
gis = GIS()
map1 = gis.map('USA',3)
map1
```