

Script para a geração de GeoSpatialPDF para todo o continente português, em ArcPY

1. Após vários ensaios, chegou-se à conclusão que o ArcGIS (ESRI) é o software que melhores resultados oferece na exportação para GeoSpatialPDF. Foi também testado o QGIS, que apesar de também permitir criar GeoSpatialPDF, é mais fechado em termos de opções e os resultados finais parecem não ter tanta qualidade.
2. No ArcGIS, após várias tentativas para encontrar o equilíbrio entre peso em MB do ficheiro produzido, resolução (qualidade final da imagem) e área geográfica abrangida, foram tomadas as seguintes opções:
 - a) Tamanho da folha de impressão: Formato A1 – Landscape;
 - b) A área de impressão é total, ou seja fica sem margens no papel; Isto para evitar que apareçam zonas brancas no GeoSpatialPDF
 - c) Resolução do PDF: 300dpi
 - d) Advanced – Exportar a georreferenciação
3. Era importante conseguir um zoom de ecrã que permitisse visualizar a cobertura de ortofotos com a melhor resolução e maior escala;
4. No arcGIS foi criado um documento com os seguintes temas:
 - a) Quadricula_Operacional_5km_ETRS86 – Serve para limitar as áreas de 5kmX5km;
 - b) Limites administrativos de Portugal Continental (CAOP 2016 – DGT) – representados a linha tracejada;
 - c) ESRI BaseMap WorldImagery - Mapa base com as imagens da ESRI, disponibilizadas em Abril e Maio de 2017;
5. A escala de “impressão” é de aprox. 1:9111; não se procurou um valor “standard” pois considerou-se que o mais importante seria abranger a maior área geográfica possível, por cada quadrícula de 5kmX5Km; E em termos de futuras utilizações, com o AvenzaMaps, essa diferença será irrelevante;
6. Script de exportação
 - a) Define documento de trabalho, dataframe activo e layer que contém as quadrículas 5kmX5km

- b) Para cada FID (que é sequencial, sem saltos, entre 0 e 3923)
- Selecciona a quadricula com esse FID
 - Faz Zoom para a selecção
 - Procura o ID correspondente
 - Exporta para pdf , com o nome GeoPDF+ID
- c) Para se poder utilizar a funcionalidade “ZoomToSelected” e exportar a área correspondente a essa quadricula, o script funciona com o respectivo documento ArcGIS aberto.

7. Código

```
import arcpy
mxdoc=arcpy.mapping.MapDocument("CURRENT")
df = arcpy.mapping.ListDataFrames(mxdoc, "Layers") [0]
#print mxdoc.title
lyr_Qdr = arcpy.mapping.ListLayers(mxdoc)[0]
print lyr_Qdr.name
#numFeat=arcpy.GetCount_management(lyr_Qdr)
#print numFeat
for i in range(50,100):
    print i
    arcpy.SelectLayerByAttribute_management(lyr_Qdr, "NEW_SELECTION", ' "FID" = '+str(i))
    df.zoomToSelectedFeatures()
    df.extent = lyr_Qdr.getSelectedExtent()
    arcpy.RefreshActiveView()
    valueID = [row[0] for row in arcpy.da.SearchCursor(lyr_Qdr, "ID")]
    print valueID[0]
    arcpy.mapping.ExportToPDF(mxdoc,r"C:\00_UserMLurdes\GeoPDF_"+str(valueID[0])+".pdf")
```

Versão nº2

```
>>> import arcpy
... from arcpy import env
... from arcpy import mapping
... mxdoc=arcpy.mapping.MapDocument("CURRENT")
... df = arcpy.mapping.ListDataFrames(mxdoc, "Layers") [0]
... i=0
... lyr_Qdr = arcpy.mapping.ListLayers(mxdoc)[0]
... print lyr_Qdr.name
... numFeat=str(arcpy.GetCount_management(lyr_Qdr))
... print numFeat
... while i < int(numFeat):
...     print i
...     rows=arcpy.SearchCursor(lyr_Qdr, "FID = " + str(i))
...     for row in rows:
...         xmin,ymin,xmax,ymax=row.shape.extent.XMin,
row.shape.extent.YMin, row.shape.extent.XMax,
row.shape.extent.YMax
...         print str(xmin)
...         print str(xmax)
...         print str(ymin)
```

```

...         print str(ymax)
...         df = arcpy.mapping.ListDataFrames(mxdoc)[0]
...         newExtent=df.extent
...         newExtent.XMin, newExtent.YMin, newExtent.XMax,
newExtent.YMax = xmin, ymin, xmax, ymax
...         df.extent=newExtent
...         arcpy.RefreshActiveView()
...         valueID = [row2[0] for row2 in
arcpy.da.SearchCursor(lyr_Qdr, "ID", ' "FID" = '+str(i))]
...         print valueID[0]
...
arcpy.mapping.ExportToPDF(mxdoc,r"C:\00_UserMLurdes\GeoMapas/
2\GeoPDF_"+str(valueID[0])+".pdf")
...         print('Exported image' )
...         print(i, 'of', numFeat)
...         i=i+1
...     print ("Done")

```

Versão nº3 (final ?)

```

>>> import arcpy
... from arcpy import env
... from arcpy import mapping
...
mxdoc=arcpy.mapping.MapDocument("C:/00_UserMLurdes/GeoMapas/
Teste.mxd")
... df = arcpy.mapping.ListDataFrames(mxdoc, "Layers") [0]
... i=2051
... lyr_Qdr = arcpy.mapping.ListLayers(mxdoc)[0]
... print lyr_Qdr.name
... numFeat=str(arcpy.GetCount_management(lyr_Qdr))
... print numFeat
... while i < 3000:
...     print i
...     rows=arcpy.SearchCursor(lyr_Qdr, "FID = " + str(i))
...     for row in rows:
...         xmin,ymin,xmax,ymax=row.shape.extent.XMin,
row.shape.extent.YMin, row.shape.extent.XMax,
row.shape.extent.YMax
...         print str(xmin)
...         #print str(xmax)
...         #print str(ymin)
...         #print str(ymax)
...         df = arcpy.mapping.ListDataFrames(mxdoc)[0]
...         newExtent=df.extent
...         newExtent.XMin, newExtent.YMin, newExtent.XMax,
newExtent.YMax = xmin, ymin, xmax, ymax
...         df.extent=newExtent
...         #arcpy.RefreshActiveView()
...         valueID = [row2[0] for row2 in
arcpy.da.SearchCursor(lyr_Qdr, "ID", ' "FID" = '+str(i))]

```

Cria GeoSpatial PDF usando ArcPY

```
...  
arcpy.mapping.ExportToPDF(mxdoc,r"C:\00_UserMLurdes\GeoMapas  
2\GeoPDF_"+str(valueID[0])+".pdf")  
...         #print('Exported image' )  
...         #print(i, 'of', numFeat)  
...         i=i+1  
... print ("Done")
```

Milú Caridade Gaspar
ICNF, Abril.2017