

QGIS Application - Bug report #21195

processing.run('native:clip') creates empty shapefile when called externally (via Python bindings)

2019-02-07 03:31 AM - Christoph Nolte

Status:	Closed	
Priority:	Normal	
Assignee:		
Category:	Processing/QGIS	
Affected QGIS version:	3.4.4	Regression?: No
Operating System:	OSX	Easy fix?: No
Pull Request or Patch submitted:	No	Resolution: worksforme
Crashes QGIS or corrupts data:	No	Copied to github as #: 29013

Description

The QGIS 3.4.4. Python bindings don't seem to work as expected in my Anaconda environment on OSX.

Specifically 'native:clip' creates an empty shapefile when it shouldn't.

Reproducible Example:

I want to clip one of the attached files with another (MA_rivers.shp with 25017_county_2016.shp).

There doesn't seem anything wrong with the files. In the QGIS Application GUI, "Vector > Geoprocessing Tools > Clip" returns the expected correctly clipped shapefile (5 files, total 430KB), and the following expected output:

Processing algorithm...

Algorithm 'Clip' starting...

Input parameters:

```
{ 'INPUT' : '/Users/production/Dropbox/Data/GIS/USA/GDB/- temp/MA_rivers.shp', 'OUTPUT' :  
'/Users/production/Dropbox/Data/GIS/USA/GDB/- temp/clipped.shp', 'OVERLAY' :  
'/Users/production/Dropbox/Data/GIS/USA/GDB/- temp/25017_county_2016.shp' }
```

Execution completed in 0.25 seconds

Results:

```
{'OUTPUT': '/Users/production/Dropbox/Data/GIS/USA/GDB/- temp/clipped.shp'}
```

Loading resulting layers

Algorithm 'Clip' finished

However, when I attempt to run exactly the same command through an external Python script with qgis and processing (see below), the output is an empty shapefile (3 files, total 234 bytes).

Note: this is in spite of the algorithm indicating that it finished successfully (it also returns "{ 'OUTPUT': '/Users/production/Dropbox/Data/GIS/USA/GDB/- temp/clipped.shp' }")

```
import os, sys  
QGIS_ENV = os.path.expanduser('~/.Anaconda/envs/gis/')  
sys.path.append(QGIS_ENV + 'QGIS.app/Contents/Resources/python')  
sys.path.append(QGIS_ENV + 'QGIS.app/Contents/Resources/python/plugins')  
  
from qgis.core import *  
  
QgsApplication.setPrefixPath(os.path.join(QGIS_ENV, 'Contents'))  
qgs = QgsApplication([], False)  
qgs.initQgis()
```

```

import processing
from processing.core.Processing import Processing
Processing.initialize()

from qgis.analysis import QgsNativeAlgorithms
QgsApplication.processingRegistry().addProvider(QgsNativeAlgorithms())

1. Shortcut to working folder
wf = lambda x: os.path.expanduser('~/.Dropbox/Data/GIS/USA/GDB/- temp/' + x)
2. Shortcut for QgsVectorLayer
QVL = lambda filepath: QgsVectorLayer(filepath, "", 'ogr')

processing.run('native:clip', {'INPUT': QVL,
                                'OUTPUT': wf('clipped.shp'),
                                'OVERLAY': QVL})

```

Returns: {'OUTPUT': '/Users/production/Dropbox/Data/GIS/USA/GDB/- temp/clipped.shp'}

My system is OSX 10.14.3, QGIS 3.4.4, and I run this through my Jupyter notebook using Anaconda 3 (with a QGIS recipe from Chris Holden's Anaconda channel (ceholden)).

History

#1 - 2019-02-07 03:41 AM - Christoph Nolte

Apologies - seems I didn't get the formatting right the first time. Here's how my code looks like:

```

import os, sys
QGIS_ENV = os.path.expanduser('~/.Anaconda/envs/gis/')
sys.path.append(QGIS_ENV + 'QGIS.app/Contents/Resources/python')
sys.path.append(QGIS_ENV + 'QGIS.app/Contents/Resources/python/plugins')

from qgis.core import *

QgsApplication.setPrefixPath(os.path.join(QGIS_ENV, 'Contents'))
qgs = QgsApplication([], False)
qgs.initQgis()

import processing
from processing.core.Processing import Processing
Processing.initialize()

from qgis.analysis import QgsNativeAlgorithms
QgsApplication.processingRegistry().addProvider(QgsNativeAlgorithms())

# Shortcut to working folder
wf = lambda x: os.path.expanduser('~/.Dropbox/Data/GIS/USA/GDB/- temp/' + x)
# Shortcut for QgsVectorLayer
QVL = lambda filepath: QgsVectorLayer(filepath, "", 'ogr')

processing.run('native:clip',

```

```
{'INPUT': QVL(wf('MA_rivers.shp')),
'OUTPUT': wf('clipped.shp'),
'OVERLAY': QVL(wf('25017_county_2016.shp'))})
```

#2 - 2019-02-07 04:56 AM - Nyal Dawson

- Status changed from Open to Feedback

Do you get any console output when running your script?

#3 - 2019-02-07 05:17 PM - Christoph Nolte

I usually run this in Jupyter, but I just ran it in the Python shell. The only console outputs I get are "True" after telling QGIS to load the native algorithms, and then the output dictionary returned from 'native:clip'. The resulting shapefile is still empty.

```
>>> import os, sys
>>> QGIS_ENV = os.path.expanduser('~/.Anaconda/envs/gis/')
>>> sys.path.append(QGIS_ENV + 'QGIS.app/Contents/Resources/python')
>>> sys.path.append(QGIS_ENV + 'QGIS.app/Contents/Resources/python/plugins')
>>>
>>> from qgis.core import *
>>>
>>> QgsApplication.setPrefixPath(os.path.join(QGIS_ENV, 'Contents'))
>>> qgs = QgsApplication([], False)
>>> qgs.initQgis()
>>>
>>> import processing
>>> from processing.core.Processing import Processing
>>> Processing.initialize()
>>>
>>> from qgis.analysis import QgsNativeAlgorithms
>>> QgsApplication.processingRegistry().addProvider(QgsNativeAlgorithms())
True
>>>
>>> # Shortcut to working folder
... wf = lambda x: os.path.expanduser('~/.Dropbox/Data/GIS/USA/GDB/- temp/' + x)
>>> # Shortcut for QgsVectorLayer
... QVL = lambda filepath: QgsVectorLayer(filepath, '', 'ogr')
>>>
>>> processing.run('native:clip',
...               {'INPUT': QVL(wf('MA_rivers.shp')),
...               'OUTPUT': wf('clipped.shp'),
...               'OVERLAY': QVL(wf('25017_county_2016.shp'))})
{'OUTPUT': '/Users/production/Dropbox/Data/GIS/USA/GDB/- temp/clipped.shp'}
>>>
```

#4 - 2019-02-07 11:36 PM - Nyal Dawson

I would double check that layers can be read correctly - you may be missing environment setups.

Try adding:

```
vl = QgsVectorLayer('/path/to/some/existing/shapefile.shp', '', 'ogr')
assert vl.isValid()
```

and make sure that layers can indeed be read

#5 - 2019-02-08 12:53 AM - Christoph Nolte

This is going into the right direction. The QgsVectorLayer isn't valid. However, the filepath is valid (exists). The shapefile is valid (clipping works if done in the QGIS GUI). What would be the next step to troubleshoot this?

```
>>>os.path.exists(wf('25017_county_2016.shp'))
True
>>>QgsVectorLayer(wf('25017_county_2016.shp'), '', 'ogr').isValid()
False
```

#6 - 2019-02-08 01:10 AM - Christoph Nolte

Thank you for pointing me in the right direction. I actually did set the PrefixPath incorrectly. I guess I anticipated that I would get an error if the PrefixPath wasn't correct. In the end, replacing the above "setPrefixPath" line with this one solved the issue:

```
QgsApplication.setPrefixPath(QGIS_ENV + 'QGIS.app/Contents/MacOS', True)
```

Would it make sense to create a warning for the user when the PrefixPath is wrong, instead of running the clipping, and make it appear as if everything worked? Some clipping does actually result in legitimate empty files, so it would be good to be warned if there is an issue with loading the layers.

Either way, thanks a lot for your time! I truly appreciate it and can't wait to port my spatial work to standalone scripts.

#7 - 2019-02-08 12:14 PM - Giovanni Manghi

Christoph Nolte wrote:

line with this one solved the issue:

closing?

#8 - 2019-02-08 01:52 PM - Christoph Nolte

Yes, this particular question has been solved.

I do think it makes sense to have processing/qgis warn users when the loaded shapefile is not valid, instead of just running the algorithm and then creating an empty output. But that's a different feature request.

#9 - 2019-02-08 02:01 PM - Giovanni Manghi

- Resolution set to worksforme

- Status changed from Feedback to Closed

Files

Archive.zip	961 KB	2019-02-07	Christoph Nolte
-------------	--------	------------	-----------------