QGIS Application - Bug report #3807
QgsRasterLayer.identify() is always True

2011-05-08 04:19 AM - Ricardo Silva

Status:	Closed		
Priority:	Low		
Assignee:			
Category:	Rasters		
Affected QGIS version:	master	**Regression?:**	No
Operating System:		**Easy fix?:**	No
Pull Request or Patch supplied:	No	**Resolution:**	end of life
Crashes QGIS or corrupts data:	No	**Copied to github as #:**	13865

Description

Using the python bindings, when trying to identify a point outside of current raster layer, The [[QgsRasterLayer]].identify() method is returning:

(True, {PyQt4.QtCore.QString(u'Band 1'): [[PyQt]]4.QtCore.QString(u'out of extent')})

which doesn't make much sense IMO. If the result is the "out of extent" string, then the identifying has failed, so the first entry of the tuple should be False, instead of True.

History

#1 - 2011-05-08 04:31 AM - Ricardo Silva

This is always true if the identified point is inside the extents of the raster layer, but in an area that is not part of the layer itself, in which case [[QgsRasterLayer]].identify() returns:

(True, {PyQt4.QtCore.QString(u'Band 1'): [[PyQt]]4.QtCore.QString(u'null (no data)')})

I guess this is not intended behavior as well, because the tuple's first value is always True.

#2 - 2012-09-16 03:44 PM - Borys Jurgiel

- Affected QGIS version set to master

- Crashes QGIS or corrupts data set to No

- Pull Request or Patch supplied set to No

- Assignee deleted (Borys Jurgiel)

#3 - 2014-06-28 07:40 AM - Jürgen Fischer

- Target version changed from Version 2.0.0 to Future Release - Lower Priority

#4 - 2015-12-03 02:38 AM - Médéric RIBREUX

- Category changed from Python plugins to Rasters

- Tag set to easy

- Operating System deleted (Debian)

Hello, bug triage...

this is still true in QGIS 2.13 even if the identification mechanism have changed. Today, we use identify method on the raster layer provider:

```
ident = rlayer.dataProvider().identify(QgsPoint(15.30, 40.98), QgsRaster.IdentifyFormatValue)
if ident.isValid()
  ident.results()


# When you are out of the raster extent, ident is valid but the results are: {1:None}
```

We need to make identify method returns a non valid QgsRasterIdentifyResult when identification is outside the extent of the raster. Currently, when you look at the code, when we are outside of the extent, QGIS still returns a valid QgsRasterIdentifyResult with a QVariant for each band of the raster. I think this is easy to fix as long as this does not break something else.

#5 - 2017-05-01 01:10 AM - Giovanni Manghi

- Regression? set to No

- Easy fix? set to No

#6 - 2019-03-09 04:04 PM - Giovanni Manghi

- Resolution set to end of life

- Status changed from Open to Closed

End of life notice: QGIS 2.18 LTR

Source:

http://blog.qgis.org/2019/03/09/end-of-life-notice-qgis-2-18-ltr/