QGIS Application - Bug report #17985
Huge memory leak with QgsFeature.attributes() in Python

2018-01-28 10:55 PM - Nyall Dawson

Status:	Closed			
Priority:	High			
Assignee:	Denis Rouzaud			
Category:	Python bindings / sipify			
Affected QGIS version:	master	**Regression?:**	Yes
Operating System:		**Easy fix?:**	No
Pull Request or Patch supplied:		**Resolution:**	fixed/implemented
Crashes QGIS or corrupts data:	Yes	**Copied to github as #:**	25881

Description

See discussion in https://github.com/qgis/QGIS/pull/6186 and https://github.com/qgis/QGIS/pull/6190

Steps to reproduce:

Load a big layer with many NULL values

Run the python code:

```
for f in iface.activeLayer().getFeature():
    f.attributes()
```

And watch your memory go!

It's caused by the ConvertFromTypeCode in QgsAttributes.h/sip - commenting out the code which sets list items in the loop avoids the leak. Martin has also deduced that it's only triggered by the presence of NULL values in the list.

History

#1 - 2018-01-30 12:21 AM - Nyall Dawson

Nathan reports that the leak is also present in the QgsFeature python code - e.g. calling feature[0] or feature['my_attr'] will leak if the attribute is NULL.

To be precise - it's not technically a memory leak, it's just that Python is not garbage collecting these values.

#2 - 2018-01-30 01:00 AM - Nathan Woodrow

The following code also seems to show the issue

```
ids = set()
def leakme():
    count = 0
    for f in iface.activeLayer().getFeatures():
        a = f[4]
        ids.add(id(a))
        if count == 1000:
            break
        count += 1

leakme()
```

print(ids)

#3 - 2018-02-19 12:23 AM - Nathan Woodrow

@Nayll This one is fixed now isn't it?

#4 - 2018-02-19 04:58 AM - Nyall Dawson

Let me verify a few other things first - I did see evidence of another similar leak at one stage.

#5 - 2018-02-22 07:54 PM - Giovanni Manghi

- Status changed from Open to Feedback

#6 - 2018-06-08 01:58 PM - Denis Rouzaud

Nyall, can this be closed?

#7 - 2018-08-25 04:46 PM - Giovanni Manghi

Hi, what is the status of this issue?

#8 - 2018-08-25 10:18 PM - Nyall Dawson

- Resolution set to fixed/implemented

- Status changed from Feedback to Closed

Fixed