

## QGIS Application - Bug report #14192

### Multi-threaded usage of QgsNetworkAccessManager

2016-01-29 12:00 AM - Patrick Valsecchi

<b>Status:</b> Closed	
<b>Priority:</b> Normal	
<b>Assignee:</b>	
<b>Category:</b> Network	
<b>Affected QGIS version:</b> master	<b>Regression?:</b> No
<b>Operating System:</b>	<b>Easy fix?:</b> No
<b>Pull Request or Patch supplied:</b> No	<b>Resolution:</b>
<b>Crashes QGIS or corrupts data:</b> Yes	<b>Copied to github as #:</b> 22194

#### Description

QgsNetworkAccessManager has a singleton that is used all over the place. The problem is that "all over the place" means from different threads and according to the Qt documentation, QNetworkAccessManager is only reentrant (thread safe only if threads are using different instances).

This problem can easily be proven by adding a check in this method:

```
QgsNetworkAccessManager* QgsNetworkAccessManager::instance()
{
    static QgsNetworkAccessManager* sInstance( new QgsNetworkAccessManager( QApplication::instance() ) );
    if ( sInstance->thread() != QThread::currentThread() ) {
        abort();
    }
    return sInstance;
}
```

QGIS will crash quickly when you start it:

```
#0 0x00007ffff43ee267 in __GI_raise (sig=sig@entry=6) at ../sysdeps/unix/sysv/linux/raise.c:55
#1 0x00007ffff43efeca in __GI_abort () at abort.c:89
#2 0x00007ffff63970a5 in QgsNetworkAccessManager::instance () at
/home/pvalsecchi/src/QGIS/src/core/qgsnetworkaccessmanager.cpp:104
#3 0x00007ffff6bba8c50 in QgsWmsCapabilitiesDownload::downloadCapabilities (this=0x3218810)
at /home/pvalsecchi/src/QGIS/src/providers/wms/qgswmscapabilities.cpp:1936
#4 0x00007ffff6bba881e in QgsWmsCapabilitiesDownload::downloadCapabilities (this=0x3218810, baseUrl=..., auth=...)
at /home/pvalsecchi/src/QGIS/src/providers/wms/qgswmscapabilities.cpp:1907
#5 0x00007ffff6bc0ca5f in QgsWMSConnectionItem::createChildren (this=0x3218c20)
at /home/pvalsecchi/src/QGIS/src/providers/wms/qgswmsdataitems.cpp:69
#6 0x00007ffff624474c in QgsDataItem::runCreateChildren (item=0x3218c20) at
/home/pvalsecchi/src/QGIS/src/core/qgsdataitem.cpp:385
#7 0x00007ffff6250026 in QtConcurrent::StoredFunctorCall1<QVector<QgsDataItem*>, QVector<QgsDataItem*>
*>(QgsDataItem*), QgsDataItem*>::runFunctor (this=0x3216eb0) at /usr/include/qt4/QtCore/qtconcurrentstoredfunctioncall.h:267
#8 0x00007ffff624fef5 in QtConcurrent::RunFunctionTask<QVector<QgsDataItem*> >::run (this=0x3216eb0)
at /usr/include/qt4/QtCore/qtconcurrentrunbase.h:106
#9 0x00007ffff5a7adba in QThreadPoolThread::run (this=0x1203390) at concurrent/qthreadpool.cpp:108
#10 0x00007ffff5a87d1c in QThreadPrivate::start (arg=0x1203390) at thread/qthread_unix.cpp:349
#11 0x00007fffd1a6aa in start_thread (arg=0x7fff5f5b7700) at pthread_create.c:333
#12 0x00007ffff44bfeed in clone () at ../sysdeps/unix/sysv/linux/x86_64/clone.S:109
```

I propose we fix it using a QThreadStorage like that:

```
QgsNetworkAccessManager* QgsNetworkAccessManager::instance()
{
    static QThreadStorage<QgsNetworkAccessManager> sInstances;
    return &sInstances.localData();
}
```

And we make the constructor private to force the code using it to go through instance().

---

**Related issues:**

Related to QGIS Application - Bug report # 14401: QGIS Crashes if WFS credent...	<b>Closed</b>	<b>2016-03-01</b>
--	---------------	-------------------

---

**Associated revisions****Revision 2eb82430 - 2016-03-05 12:28 PM - Jürgen Fischer**

use individual network managers for threads (fixes #13721, fixes #14401, implements #14192)

**Revision c128a552 - 2016-03-23 09:38 PM - Jürgen Fischer**

use individual network managers for threads (fixes #13721, fixes #14401, implements #14192)

(cherry picked from commit 2eb82430bbdb02b0789b3ffe80d5b6fd747fa8cc)

---

**History****#1 - 2016-03-05 04:23 AM - Jürgen Fischer**

- *Status changed from Open to Closed*

Fixed in changeset commit:"2eb82430bbdb02b0789b3ffe80d5b6fd747fa8cc".