QGIS Application - Bug report #21781
QGis 3.6.0 crash when start on Linux OpenSuse Tumbleweed

2019-04-06 04:38 AM - christian faure

Status:	Closed		
Priority:	High		
Assignee:			
Category:	Unknown		
Affected QGIS version:	3.6.0	**Regression?:**	No
Operating System:	Linux OpenSuse Tumbleweed x86_64	**Easy fix?:**	No
Pull Request or Patch supplied:	No	**Resolution:**	up/downstream
Crashes QGIS or corrupts data:	Yes	**Copied to github as #:**	29596

Description

QGis 3.6.0 installed from repo https://download.opensuse.org/repositories/Application:/Geo
is crashing after display main window.

Any workaround?

> Application: qgis (qgis), signal: Segmentation fault
> Using host libthread_db library "/lib64/libthread_db.so.1".
> [Current thread is 1 (Thread 0x7f31559c40c0 (LWP 23300))]
>
> Thread 8 (Thread 0x7f31393ff700 (LWP 23319)):
> #0 0x00007f316387ba20 in __tls_get_addr () at /lib64/ld-linux-x86-64.so.2
> #1 0x00007f31601a26e6 in () at /usr/lib64/libQt5Core.so.5
> #2 0x00007f31603ac581 in () at /usr/lib64/libQt5Core.so.5
> #3 0x00007f3159209811 in g_main_context_check () at /usr/lib64/libglib-2.0.so.0
> #4 0x00007f3159209de0 in () at /usr/lib64/libglib-2.0.so.0
> #5 0x00007f3159209f5c in g_main_context_iteration () at /usr/lib64/libglib-2.0.so.0
> #6 0x00007f31603abe7b in QEventDispatcherGlib::processEvents(QFlags<QEventLoop::ProcessEventsFlag>) () at /usr/lib64/libQt5Core.so.5
> #7 0x00007f31603573cb in QEventLoop::exec(QFlags<QEventLoop::ProcessEventsFlag>) () at /usr/lib64/libQt5Core.so.5
> #8 0x00007f31601a2016 in QThread::exec() () at /usr/lib64/libQt5Core.so.5
> #9 0x00007f31601a3362 in () at /usr/lib64/libQt5Core.so.5
> #10 0x00007f3159b0bfab in start_thread () at /lib64/libpthread.so.0
> #11 0x00007f315fe6c6af in clone () at /lib64/libc.so.6
>
> Thread 7 (Thread 0x7f313b7fe700 (LWP 23316)):
> #0 0x00007f315fe61ae9 in poll () at /lib64/libc.so.6
> #1 0x00007f3159209e36 in () at /usr/lib64/libglib-2.0.so.0
> #2 0x00007f3159209f5c in g_main_context_iteration () at /usr/lib64/libglib-2.0.so.0
> #3 0x00007f31603abe7b in QEventDispatcherGlib::processEvents(QFlags<QEventLoop::ProcessEventsFlag>) () at /usr/lib64/libQt5Core.so.5
> #4 0x00007f31603573cb in QEventLoop::exec(QFlags<QEventLoop::ProcessEventsFlag>) () at /usr/lib64/libQt5Core.so.5
> #5 0x00007f31601a2016 in QThread::exec() () at /usr/lib64/libQt5Core.so.5
> #6 0x00007f31601a3362 in () at /usr/lib64/libQt5Core.so.5
> #7 0x00007f3159b0bfab in start_thread () at /lib64/libpthread.so.0
> #8 0x00007f315fe6c6af in clone () at /lib64/libc.so.6
>
> Thread 6 (Thread 0x7f313bfff700 (LWP 23315)):
> #0 0x00007f3159209916 in g_main_context_check () at /usr/lib64/libglib-2.0.so.0

#1 0x00007f3159209de0 in () at /usr/lib64/libglib-2.0.so.0

#2 0x00007f3159209f5c in g_main_context_iteration () at /usr/lib64/libglib-2.0.so.0

#3 0x00007f31603abe7b in QEventDispatcherGlib::processEvents(QFlags<QEventLoop::ProcessEventsFlag>) () at /usr/lib64/libQt5Core.so.5

#4 0x00007f31603573cb in QEventLoop::exec(QFlags<QEventLoop::ProcessEventsFlag>) () at /usr/lib64/libQt5Core.so.5

#5 0x00007f315a29a6e5 in QCA::SyncThread::run() () at /usr/lib64/libqca-qt5.so.2

#6 0x00007f31601a3362 in () at /usr/lib64/libQt5Core.so.5

#7 0x00007f3159b0bfab in start_thread () at /lib64/libpthread.so.0

#8 0x00007f315fe6c6af in clone () at /lib64/libc.so.6

Thread 5 (Thread 0x7f314a3c3700 (LWP 23307)):

#0 0x00007f3159b1203c in pthread_cond_wait@@GLIBC_2.3.2 () at /lib64/libpthread.so.0

#1 0x00007f314ab2fe13 in () at /usr/lib64/dri/nouveau_dri.so

#2 0x00007f314ab2fb67 in () at /usr/lib64/dri/nouveau_dri.so

#3 0x00007f3159b0bfab in start_thread () at /lib64/libpthread.so.0

#4 0x00007f315fe6c6af in clone () at /lib64/libc.so.6

Thread 4 (Thread 0x7f31514a0700 (LWP 23306)):

#0 0x00007f3159254cb0 in g_mutex_unlock () at /usr/lib64/libglib-2.0.so.0

#1 0x00007f31592097a0 in g_main_context_check () at /usr/lib64/libglib-2.0.so.0

#2 0x00007f3159209de0 in () at /usr/lib64/libglib-2.0.so.0

#3 0x00007f3159209f5c in g_main_context_iteration () at /usr/lib64/libglib-2.0.so.0

#4 0x00007f31603abe7b in QEventDispatcherGlib::processEvents(QFlags<QEventLoop::ProcessEventsFlag>) () at /usr/lib64/libQt5Core.so.5

#5 0x00007f31603573cb in QEventLoop::exec(QFlags<QEventLoop::ProcessEventsFlag>) () at /usr/lib64/libQt5Core.so.5

#6 0x00007f31601a2016 in QThread::exec() () at /usr/lib64/libQt5Core.so.5

#7 0x00007f31579c9565 in () at /usr/lib64/libQt5DBus.so.5

#8 0x00007f31601a3362 in () at /usr/lib64/libQt5Core.so.5

#9 0x00007f3159b0bfab in start_thread () at /lib64/libpthread.so.0

#10 0x00007f315fe6c6af in clone () at /lib64/libc.so.6

Thread 3 (Thread 0x7f3152ff7700 (LWP 23305)):

#0 0x00007f315fe61ae9 in poll () at /lib64/libc.so.6

#1 0x00007f3155ea4cc7 in () at /usr/lib64/libxcb.so.1

#2 0x00007f3155ea68fa in xcb_wait_for_event () at /usr/lib64/libxcb.so.1

#3 0x00007f315306c788 in () at /usr/lib64/libQt5XcbQpa.so.5

#4 0x00007f31601a3362 in () at /usr/lib64/libQt5Core.so.5

#5 0x00007f3159b0bfab in start_thread () at /lib64/libpthread.so.0

#6 0x00007f315fe6c6af in clone () at /lib64/libc.so.6

Thread 2 (Thread 0x7f3153fc3700 (LWP 23301)):

#0 0x00007f3159b1203c in pthread_cond_wait@@GLIBC_2.3.2 () at /lib64/libpthread.so.0

#1 0x00007f316000b8ac in std::condition_variable::wait(std::unique_lock<std::mutex>&) () at /usr/lib64/libstdc++.so.6

#2 0x00007f315d5708c2 in () at /usr/lib64/libQt5WebKit.so.5

#3 0x00007f315d5709db in () at /usr/lib64/libQt5WebKit.so.5

#4 0x00007f31600110ef in () at /usr/lib64/libstdc++.so.6

#5 0x00007f3159b0bfab in start_thread () at /lib64/libpthread.so.0

#6 0x00007f315fe6c6af in clone () at /lib64/libc.so.6

Thread 1 (Thread 0x7f31559c40c0 (LWP 23300)):

[KCrash Handler]

#6 0x00007f31569b875d in pj_free () at /usr/lib64/libproj.so.15

#7 0x00007f315ed72e5c in () at /usr/lib64/libgdal.so.20

#8 0x00007f315ed73e54 in OCTProj4Normalize () at /usr/lib64/libgdal.so.20

#9 0x00007f315ed6370f in OGRSpatialReference::importFromProj4(char const*) () at /usr/lib64/libgdal.so.20

#10 0x00007f3161895d22 in QgsCoordinateReferenceSystem::setProj4String(QString const&) (this=0x7ffec457eb00, proj4String=...) at /usr/include/qt5/QtCore/qarraydata.h:208

#11 0x00007f3161898fd5 in QgsCoordinateReferenceSystem::createFromProj4(QString const&) (this=0x7ffec457eb00, proj4String=...) at /usr/src/debug/qgis-3.6.0-3.3.x86_64/src/core/qgscoordinatereferencesystem.cpp:849

#12 0x00007f3161899a6c in QgsCoordinateReferenceSystem::fromProj4(QString const&) (proj4=...) at /usr/src/debug/qgis-3.6.0-3.3.x86_64/src/core/qgscoordinatereferencesystem.cpp:171

#13 0x00007f31618ed7fe in QgsEllipsoidUtils::ellipsoidParameters(QString const&) (ellipsoid=...) at /usr/src/debug/qgis-3.6.0-3.3.x86_64/src/core/qgsellipsoidutils.cpp:154

#14 0x00007f31619bf91b in QgsMapSettings::setEllipsoid(QString const&) (this=this@entry=0x558cc9b7f008, ellipsoid=...) at /usr/src/debug/qgis-3.6.0-3.3.x86_64/src/core/qgsmapsettings.cpp:311

#15 0x00007f31628f219a in QgsMapCanvas::<lambda()>::operator() (__closure=0x558cc9de97e0) at /usr/src/debug/qgis-3.6.0-3.3.x86_64/src/gui/qgsmapcanvas.cpp:144

#16 0x00007f31628f219a in QtPrivate::FunctorCall<QtPrivate::IndexesList<>, QtPrivate::List<>, void, QgsMapCanvas::QgsMapCanvas(QWidget*)::<lambda()> >::call (arg=<optimized out>, f=...) at /usr/include/qt5/QtCore/qobjectdefs_impl.h:146

#17 0x00007f31628f219a in QtPrivate::Functor<QgsMapCanvas::QgsMapCanvas(QWidget*)::<lambda()>, 0>::call<QtPrivate::List<>, void> (arg=<optimized out>, f=...) at /usr/include/qt5/QtCore/qobjectdefs_impl.h:256

#18 0x00007f31628f219a in QtPrivate::QFunctorSlotObject<QgsMapCanvas::QgsMapCanvas(QWidget*)::<lambda()>, 0, QtPrivate::List<>, void>::impl(int, QtPrivate::QSlotObjectBase *, QObject *, void **, bool *) (which=<optimized out>, this_=0x558cc9de97d0, r=<optimized out>, a=<optimized out>, ret=<optimized out>) at /usr/include/qt5/QtCore/qobjectdefs_impl.h:439

#19 0x00007f3160382343 in QMetaObject::activate(QObject*, int, int, void**) () at /usr/lib64/libQt5Core.so.5

#20 0x00007f3161eb2f55 in QgsProject::ellipsoidChanged(QString const&) (this=this@entry=0x558cc9a1a2b0, _t1=...) at /usr/src/debug/qgis-3.6.0-3.3.x86_64/build/src/core/moc_qgsproject.cpp:944

#21 0x00007f3161a4d31b in QgsProject::setEllipsoid(QString const&) (this=this@entry=0x558cc9a1a2b0, ellipsoid=...) at /usr/src/debug/qgis-3.6.0-3.3.x86_64/src/core/qgsproject.cpp:663

#22 0x00007f3162ff0457 in QgisApp::fileNew(bool, bool) (this=0x558cc9f2d410, promptToSaveFlag=<optimized out>, forceBlank=<optimized out>) at /usr/src/debug/qgis-3.6.0-3.3.x86_64/src/app/qgisapp.cpp:5498

#23 0x00007f3162ffefb5 in QgisApp::QgisApp(QSplashScreen*, bool, bool, QString const&, QString const&, QWidget*, QFlags<Qt::WindowType>) (this=<optimized out>, splash=<optimized out>, restorePlugins=<optimized out>, skipVersionCheck=<optimized out>, rootProfileLocation=..., activeProfile=..., parent=<optimized out>, fl=...) at /usr/src/debug/qgis-3.6.0-3.3.x86_64/src/app/qgisapp.cpp:1425

#24 0x0000558cc7c2c564 in main(int, char**) (argc=<optimized out>, argv=<optimized out>) at /usr/include/qt5/QtCore/qflags.h:120

[Inferior 1 (process 23300) detached]

History

#1 - 2019-04-06 09:09 AM - Jürgen Fischer

- Resolution set to up/downstream

- Status changed from Open to Feedback

Looks like it crashes in PROJ called from GDAL called from QGIS. PROJ and GDAL were apparently rebuilt in the last few days and QGIS yesterday. Are you using the latest packages? Might be a packaging issue - can you file it there?

#2 - 2019-04-06 03:30 PM - christian faure

Jürgen Fischer wrote:

> *Looks like it crashes in PROJ called from GDAL called from QGIS.   PROJ and GDAL were apparently rebuilt in the last few days and QGIS yesterday. Are you using the latest packages? Might be a packaging issue - can you file it there?*

Thanks,

From repo it seem latest version

GDAL

Version: 2.4.0-98.4-x86_64

GDAL Change log

gdal - GDAL/OGR - a translator library for raster and vector geospatial data formats

Thu Mar 7 09:00:00 2019

Bruno Friedmann <bruno@ioda-net.ch>

- Fix boo#1128276 by providing correct Obsolete

Sat Feb 23 09:00:00 2019

John Vandenberg <jayvdb@gmail.com>

- Use the Python package name 'GDAL' as used by upstream

Wed Jan 30 09:00:00 2019

Wolfgang Bauer <wbauer@tmo.at>

- Add upstream patches to fix build with the latest poppler in

  Tumbleweed:

  * Poppler-revision-version-management.patch

  * add-support-for-poppler-0.72.0-and-0.73.0.patch

Mon Dec 24 09:00:00 2018

Guillaume GARDET <guillaume.gardet@opensuse.org>

- Update to 2.4.0:

  * In a nutshell:

  - New GDAL drivers:

  ...

For **PROJ** it seem to have 2 versions

Package **PROJ5** 5.2.0-1.1 and Package **PROJ** 6.0.0-38.1

and both show mostly same file list

i've removed PROJ5, but problem remains

Change Log of PROJ

proj - Cartographic projection software

Mon Apr 1 08:00:00 2019

Kai Pastor <dg0yt@darc.de>

- Fix datumgrid packaging

Mon Mar 25 08:00:00 2019

Martin Pluskal <mpluskal@suse.com>

- Enable tests
- Fix previous changelog entry
Mon Mar 11 09:00:00 2019
Martin Pluskal <mpluskal@suse.com>
- Update to version 6.0.0 and datumgrid 1.7
  * See provided NEWS for list of all changes

The NEWS show

6.0.0 Release Notes

PROJ 6 has undergone extensive changes to increase its functional scope from a cartographic projection engine with so-called "early-binding" geodetic datum transformation capabilities to a more complete library supporting coordinate transformations and coordinate reference systems.

As a foundation for other enhancements, PROJ now includes a C++ implementation of the modelisation propopsed by the ISO-19111:2019 standard / OGC Abstract Specification Topic 2: "Referencing By Coordinates", for geodetic reference frames (datums), coordinate reference systems and coordinate operations. Construction and query of those geodetic objects is available through a new C++ API, and also accessible for the most part from bindings in the C API.

Those geodetic objects can be imported and exported from and into the OGC Well-Known Text format (WKT) in its different variants: ESRI WKT, GDAL WKT 1, WKT2:2015 (ISO 19162:2015) and WKT2:2018 (ISO 19162:2018). Import and export of CRS objects from and into PROJ strings is also supported. This functionality was previously available in the GDAL software library (except WKT2 support which is a new feature), and is now an integral part of PROJ.

A unified database of geodetic objects, coordinate reference systems and their metadata, and coordinate operations between those CRS is now available in a SQLite3 database file, proj.db. This includes definitions imported from the IOGP EPSG dataset (v9.6.0 release), the IGNF (French national mapping agency) geodetic registry and the ESRI projection engine database. PROJ is now the reference software in the "OSGeo C stack" for this CRS and coordinate operation database, whereas previously this functionality was spread over PROJ, GDAL and libgeotiff, and used CSV or other adhoc text-based formats.

Late-binding coordinate operation capabilities, that takes metadata such as area of use and accuracy into account, has been added. This can avoid in a number of situations the past requirement of using WGS84 as a pivot system, which could cause unneeded accuracy loss, or was not doable at all sometimes when transformation to WGS84 was not available. Those late-binding capabilities are now used by the proj_create_crs_to_crs() function and the cs2cs utility.

A new command line utility, projinfo, has been added to query information about a geodetic object of the database, import and export geodetic objects from/into WKT and PROJ strings, and display coordinate operations available between two CRSs.

UPDATES

----

o Removed projects.h as a public interface (#835)

o Deprecated the proj_api.h interface. The header file is still available
  but will be removed with the next major version release of PROJ. It is
  now required to define ACCEPT_USE_OF_DEPRECATED_PROJ_API_H before the
  interface can be used (#836)

o Removed support for the nmake build system (#838)

o Removed support for the proj_def.dat defaults file (#201)

o C++11 required for building PROJ (#1203)

o Added build dependency on SQLite 3.7 (#1175)

o Added projinfo command line application (#1189)

o Added many functions to proj.h for handling ISO19111 functionality (#1175)

o Added C++ API exposing ISO19111 functionality (#1175)

o Updated cs2cs to use late-binding features (#1182)

o Removed the nad2bin application. Now available in the proj-datumgrid
  git repository (#1236)

o Removed support for Chebyshev polynomials in proj (#1226)

o Removed proj_geocentric_latitude from proj.h API (#1170)

o Changed behaviour of proj: Now only allow initialization of
  projections (#1162)

o Changed behaviour of tmerc: Now default to the Extended Transverse
  Mercator algorithm (etmerc). Old implementation available by adding
  +approx (#404)

o Chaged behaviour: Default ellipsoid now set to GRS80 (was WGS84) (#1210)

o Allow multiple directories in PROJ_LIB environment variable (#1281)

o Added Lambert Conic Conformal (2SP Michigan) projection (#1142)

o Added Bertin1953 projection (#1133)

o Added Tobler-Mercator projection (#1153)

o Added Molodensky-Badekas transform (#1160)

o Added push and pop coordinate operations (#1250)

o Removed +t_obs parameter from helmert and deformation (#1264)

o Added +dt parameter to deformation as replacement for
  removed +t_obs (#1264)
...

QGIS version 3.6.0-3.3-x86_64

QGIS ChangeLog:

qgis - QGIS is a Geographic Information System (GIS)

Thu Apr 4 08:00:00 2019
Christophe Giboudeaux <christophe@krop.fr>
- Fix the build requirements for the Tumbleweed repo, pg_config
  is now in postgresqlXX-server-devel (bsc#1131494)
Fri Mar 22 09:00:00 2019
Guillaume GARDET <guillaume.gardet@opensuse.org>
- Force usage of proj5 as proj (version 6) is not supported yet
Wed Feb 27 09:00:00 2019
Bruno Friedmann <bruno@ioda-net.ch>
- Update to 3.6.0 version
- Cleanup of spec and history
- Added libexiv2-devel as new BuildRequires
- Fix python3-GDAL new name
Mon Jan 21 09:00:00 2019
Nicolas Rochard <nrochard@gmail.com>
- Update to 3.4.4
...

#3 - 2019-04-06 04:10 PM - Jürgen Fischer

christian faure wrote:

> *Jürgen Fischer wrote:*
>
> > *… Might be a packaging issue - can you file it there?*

…

> For **PROJ** it seem to have 2 versions
> Package **PROJ5** 5.2.0-1.1 and Package **PROJ** 6.0.0-38.1
> and both show mostly same file list

> i've removed PROJ5, but problem remains

…

> *QGIS ChangeLog:*
> *…*
> *Thu Apr 4 08:00:00 2019*
> *Christophe Giboudeaux <christophe@krop.fr>*
> *- Fix the build requirements for the Tumbleweed repo, pg_config*
> *is now in postgresqlXX-server-devel (bsc#1131494)*
> *Fri Mar 22 09:00:00 2019*
> *Guillaume GARDET <guillaume.gardet@opensuse.org>*
> *- Force usage of proj5 as proj (version 6) is not supported yet*

From the changelog it looks like it's still using proj 5 - so removing proj 5 is a bad idea.

Anyway, it still looks like a opensuse packaging problem and not an actual QGIS issue.

#4 - 2019-04-07 03:53 AM - christian faure

Jürgen Fischer wrote:

> *christian faure wrote:*
>
> > *Jürgen Fischer wrote:*
> >
> > > *… Might be a packaging issue - can you file it there?*
> >
> > *…*
> >
> > *For* ***PROJ*** *it seem to have 2 versions*
> > *Package* ***PROJ5*** *5.2.0-1.1 and Package* ***PROJ*** *6.0.0-38.1*
> > *and both show mostly same file list*
> >
> > *i've removed PROJ5, but problem remains*
> >
> > *…*
> >
> > *QGIS ChangeLog:*
> > *…*
> > *Thu Apr 4 08:00:00 2019*
> > *Christophe Giboudeaux <christophe@krop.fr>*
> > *- Fix the build requirements for the Tumbleweed repo, pg_config*
> > *is now in postgresqlXX-server-devel (bsc#1131494)*
> > *Fri Mar 22 09:00:00 2019*
> > *Guillaume GARDET <guillaume.gardet@opensuse.org>*
> > *- Force usage of proj5 as proj (version 6) is not supported yet*
> >
> > *From the changelog it looks like it's still using proj 5 - so removing proj 5 is a bad idea.*

> *Anyway, it still looks like a opensuse packaging problem and not an actual QGIS issue.*

Thanks for this!
i've removed Proj v6, reinstalled v5 and created symlink libproj.so.15 pointing to libproj.so.13 and QGis is working!
Solved!

#5 - 2019-04-07 11:37 AM - Giovanni Manghi

- Status changed from Feedback to Closed