QGIS Application - Feature request #1878
remove setjmp/longjmp in grass plugin&provider and use exceptions instead

2009-08-15 06:11 AM - Jürgen Fischer

Status:	Closed		
Priority:	Low		
Assignee:			
Category:	GRASS		
Pull Request or Patch supplied:		**Resolution:**	fixed
Easy fix?:	No	**Copied to github as #:**	11938

Description

remove setjmp/longjmp in grass plugin&provider and use exceptions instead

History

#1 - 2009-08-19 07:53 AM - Paolo Cavallini

A side effect of this patch seems to be that GRASS vectors are not visible on the canvas, unless the editing is activated

#2 - 2009-08-26 04:20 PM - Jürgen Fischer

Replying to [comment:3 pcav]:

> *A side effect of this patch seems to be that GRASS vectors are not visible on the canvas, unless the editing is activated*

The updated patch should fix this and #1900.

#3 - 2009-08-27 05:27 AM - Giovanni Manghi

Hi,

if I apply the patch I get the following error while compiling.

By the way I cannot compile under windows, so it would be hard in any case to say if the crash is gone or not.

```
In file included from /home/gio/Desktop/qgis_unstable/src/providers/grass/qgsgrassprovider.cpp:19:
/home/gio/Desktop/qgis_unstable/src/providers/grass/qgsgrass.h: In constructor ‘QgsGrass::Exception::Exception(const char*)’:
/home/gio/Desktop/qgis_unstable/src/providers/grass/qgsgrass.h:37: error: no matching function for call to ‘std::exception::exception(const char*&)’
/usr/include/c++/4.3/exception:59: note: candidates are: std::exception::exception()
/usr/include/c++/4.3/exception:57: note:                 std::exception::exception(const std::exception&)

maker2: *** [src/providers/grass/CMakeFiles/qgisgrass.dir/qgsgrassprovider.o] Error 1
maker1: *** [src/providers/grass/CMakeFiles/qgisgrass.dir/all] Error 2
make: *** [all] Error 2
```

#4 - 2009-09-03 09:18 AM - Giovanni Manghi

Replying to [comment:4 jef]:

> *The updated patch should fix this and #1900.*

Hi Jurgen,
is there problem applying this patch? I'm asking because I have a few colleagues using qgis-dev and they are hitting the #1900 problem. Thanks.

#5 - 2009-09-03 11:21 AM - Jürgen Fischer

Replying to [comment:6 lutra]:

> *is there problem applying this patch? I'm asking because I have a few colleagues using qgis-dev and they are hitting the #1900 problem. Thanks.*

Probably not, but I don't know. I'm no GRASS user and therefore didn't do real testing. Did you test it?

#6 - 2009-09-04 04:27 AM - Giovanni Manghi

Replying to [comment:7 jef]:

> *Replying to [comment:6 lutra]:*
>
> > *is there problem applying this patch? I'm asking because I have a few colleagues using qgis-dev and they are hitting the #1900 problem. Thanks.*
>
> *Probably not, but I don't know. I'm no GRASS user and therefore didn't do real testing. Did you test it?*

I have applied the patch and I see no problem compiling and also using qgis/grass doing my normal tasks. #1900 is definitely gone (at least under linux).

#7 - 2009-09-05 05:01 AM - Jürgen Fischer

- Status changed from Open to Closed
- Resolution set to fixed

applied in commit:43408110 (SVN r11561)

#8 - 2010-01-12 09:58 AM - Redmine Admin

- Status changed from Closed to Feedback
- Resolution deleted (fixed)

Replying to [comment:9 jef]:

> *applied in commit:43408110 (SVN r11561)*

Are you sure it works after http://trac.osgeo.org/grass/changeset?old_path=grass%2Ftrunk%2Flib%2Fgis%2Ferror.c&old=23438&new_path=grass%2Ftrunk%2Flib%2Fgis%2Ferror.c&new=23439? For me exceptions cannot be caught after that change. The installed routine in QGIS is called, but then it terminates. It seems, that a function in a library in C cannot be interrupted by calling a function in C++ which throws an exception. Wasn't used setjmp/longjmp to solve it?

```
#3909  0x0066f452 in std::terminate() () from /usr/lib/libstdc++.so.6
#3910  0x0066f591 in +cxa_throw () from /usr/lib/libstdc++.so.6
#3911  0x05d6eed7 in [[QgsGrass]]::openMapset(QString, QString, QString) ()
   from /home/radim/apps/lib/libqgisgrass.so.1.5.0
#3912  0x07c842e4 in ?? () from /usr/lib/grass64/lib/libgrass_gis.so
#3913  0x07c848f2 in G_fatal_error () from /usr/lib/grass64/lib/libgrass_gis.so
#10 0x01fea6d5 in ?? () from /usr/lib/grass64/lib/libgrass_vect.so
#3914 0x01feb225 in Vect+open_old () from /usr/lib/grass64/lib/libgrass_vect.so
#3915 0x01feb77c in Vect_open_old_head () from /usr/lib/grass64/lib/libgrass_vect.so
#3916 0x037ea4a1 in [[QgsGrassSelect]]::vectorLayers(QString, QString, QString, QString) ()
   from /home/radim/apps/lib/qgis/libgrassplugin.so
```

I dont understand how it can come to [[QgsGrass]]::openMapset however.

#9 - 2010-01-12 10:53 AM - Jürgen Fischer

Replying to [comment:10 rblazek]:

> *Replying to [comment:9 jef]:*
>
> > *applied in commit:43408110 (SVN r11561)*
>
> *Are you sure it works after http://trac.osgeo.org/grass/changeset?old_path=grass%2Ftrunk%2Flib%2Fgis%2Ferror.c&old=23438&new_path=grass%2Ftrunk%2Flib%2Fgis%2Ferror.c&new=23439? For me exceptions cannot be caught after that change. The installed routine in QGIS is called, but then it terminates. It seems, that a function in a library in C cannot be interrupted by calling a function in C++ which throws an exception. Wasn't used setjmp/longjmp to solve it?*

I just verified that it works as expected on Windows. I'll try on linux in the evening.

#10 - 2010-01-12 11:50 AM - Redmine Admin

Previous backtrace is probably invalid, maybe old core. The following backtrace makes sense ,QgsGrass::error_routine is called, exception is thrown but it is not caught and program is terminated by std::terminate().

```
#3909  0x09859452 in std::terminate() () from /usr/lib/libstdc++.so.6
#3910  0x09859591 in +cxa_throw () from /usr/lib/libstdc++.so.6
#3911  0x02dd2042 in [[QgsGrass]]::error_routine (
    msg=0xbfb46078 "Unable to open vector map <fiumi_buf@pok> on level 2. Try to rebuild vector topology by v.build.", fatal=1) at
/home/radim/devel/qgis/src/providers/grass/qgsgrass.cpp:401
#3912  0x07b412e4 in ?? () from /usr/lib/grass64/lib/libgrass_gis.so
#3913  0x07b418f2 in G_fatal_error () from /usr/lib/grass64/lib/libgrass_gis.so
```

#10 0x058956d5 in ?? () from /usr/lib/grass64/lib/libgrass_vect.so
#3914 0x05896225 in Vect+open_old () from /usr/lib/grass64/lib/libgrass_vect.so
#3915 0x0589677c in Vect_open_old_head () from /usr/lib/grass64/lib/libgrass_vect.so
#3916 0x044f3a43 in [[QgsGrassSelect]]::vectorLayers (gisdbase=..., location=..., mapset=..., mapName=...)
at /home/radim/devel/qgis/src/plugins/grass/qgsgrassselect.cpp:416

#11 - 2010-01-12 01:28 PM - Jürgen Fischer

Replying to [comment:13 rblazek]:

> *Previous backtrace is probably invalid, maybe old core. The following backtrace makes sense ,QgsGrass::error_routine is called, exception is thrown but it is not caught and program is terminated by std::terminate().*

> #3909 0x09859452 in std::terminate() () from /usr/lib/libstdc++.so.6
> #3910 0x09859591 in +cxa_throw () from /usr/lib/libstdc++.so.6
> #3911 0x02dd2042 in [[QgsGrass]]::error_routine (
> msg=0xbfb46078 "Unable to open vector map <fiumi_buf@pok> on level 2. Try to rebuild vector topology by v.build.", fatal=1) at /home/radim/devel/qgis/src/providers/grass/qgsgrass.cpp:401
> #3912 0x07b412e4 in ?? () from /usr/lib/grass64/lib/libgrass_gis.so
> #3913 0x07b418f2 in G_fatal_error () from /usr/lib/grass64/lib/libgrass_gis.so
> #10 0x058956d5 in ?? () from /usr/lib/grass64/lib/libgrass_vect.so
> #3914 0x05896225 in Vect+open_old () from /usr/lib/grass64/lib/libgrass_vect.so
> #3915 0x0589677c in Vect_open_old_head () from /usr/lib/grass64/lib/libgrass_vect.so
> #3916 0x044f3a43 in [[QgsGrassSelect]]::vectorLayers (gisdbase=..., location=..., mapset=..., mapName=...)
> at /home/radim/devel/qgis/src/plugins/grass/qgsgrassselect.cpp:416

.

I get:

Debug: /home/fischer/src/qgis/qgis_unstable/src/providers/grass/qgsgrass.cpp: 734: (vectors) mapsetPath = /home/fischer/test/grass/spearfish60/user1
Debug: /home/fischer/src/qgis/qgis_unstable/src/plugins/grass/qgsgrassselect.cpp: 340: (setLayers) setLayers()
Debug: /home/fischer/src/qgis/qgis_unstable/src/providers/grass/qgsgrass.cpp: 335: (setLocation) gisdbase = /home/fischer/test/grass location = spearfish60
Debug: /home/fischer/src/qgis/qgis_unstable/src/plugins/grass/qgsgrassselect.cpp: 432: (vectorLayers) GRASS vector successfully opened
Debug: /home/fischer/src/qgis/qgis_unstable/src/plugins/grass/qgsgrassselect.cpp: 340: (setLayers) setLayers()
Debug: /home/fischer/src/qgis/qgis_unstable/src/providers/grass/qgsgrass.cpp: 335: (setLocation) gisdbase = /home/fischer/test/grass location = spearfish60
Debug: /home/fischer/src/qgis/qgis_unstable/src/providers/grass/qgsgrass.cpp: 394: (error_routine) error_routine (fatal = 1): Unable to open vector map <fiumi_buf@user1> on level 2. Try to rebuild vector topology by v.build.
Debug: /home/fischer/src/qgis/qgis_unstable/src/plugins/grass/qgsgrassselect.cpp: 421: (vectorLayers) Cannot open GRASS vector: Unable to open vector map <fiumi_buf@user1> on level 2. Try to rebuild vector topology by v.build.

#12 - 2010-01-13 03:25 AM - Redmine Admin

For me it is still crashing, the exception is not caught. I tried to call a GRASS function and catch exception in libqgisgrass and also both moved to plugin, but it does not work. I have also recompiled GRASS 6.4 from source but nothing helped.

It does not work even with reverted
http://trac.osgeo.org/grass/changeset?old_path=grass%2Ftrunk%2Flib%2Fgis%2Ferror.c&old=23438&new_path=grass%2Ftrunk%2Flib%2Fgi%2Ferror.c&new=23439

The change in GRASS above probably is not significant, G_fatal_error does not continue in execution after the installed error_routine was called, but the exception cannot be caught.

Kbuntu 9.10, gcc (Ubuntu 4.4.1-4ubuntu8) 4.4.1, grass-6.4.0RC5, GDAL svn trunk, QGIS svn trunk, libstdc++6-4.4-dbg

#13 - 2010-01-13 04:47 AM - Redmine Admin

jef, I have created very simple test, attached (except.tar). Could you please try to just run 'make' and './test'? For me it does not work:

```
./test
set_error_routine() start
set_error_routine() end
error_cpp called
caught int from error_cpp
error() start
error_routine called
terminate called after throwing an instance of 'int'
Aborted (core dumped)
```

#14 - 2010-01-13 05:42 AM - Jürgen Fischer

Replying to [comment:16 rblazek]:

> *jef, I have created very simple test, attached (except.tar). Could you please try to just run 'make' and './test'? For me it does not work:*

Odd. Using GCC 4.4.2 it works on 64bit, but fails on 32bit.

#15 - 2010-01-13 07:27 AM - Redmine Admin

The problem is that C code must be compiled with -fexceptions, -fno-exceptions is probably default for gcc on some platforms.

#16 - 2010-01-13 10:36 AM - Redmine Admin

Request in GRASS http://trac.osgeo.org/grass/ticket/869

#17 - 2010-01-14 08:26 AM - Redmine Admin

It seems that we are back at the beginning of this ticket. GRASS developer seem to be reluctant to compile with -fexception and using setjmp/longjmp was suggested http://lists.osgeo.org/pipermail/grass-dev/2010-January/048086.html

For curiosity, were there particular problems with setjmp/longjmp?

In any case, I think that using exceptions is much better, I don't want to return to setjmp/longjmp.

#18 - 2010-01-14 11:13 PM - Jürgen Fischer

Replying to [comment:20 rblazek]:

> *It seems that we are back at the beginning of this ticket. GRASS developer seem to be reluctant to compile with -fexception and using setjmp/longjmp was suggested http://lists.osgeo.org/pipermail/grass-dev/2010-January/048086.html*

I think we shouldn't try to intercept fatal GRASS errors in the GRASS library, we shouldn't be using GRASS libraries in the first place.

> And I think that's also what the other GUI frontends do. But that probably is a major rewrite of the plugin.
>
> > For curiosity, were there particular problems with setjmp/longjmp?
>
> > In any case, I think that using exceptions is much better, I don't want to return to setjmp/longjmp.
>
> None that I know of. Just uglyness.

#19 - 2010-01-16 01:28 AM - Jürgen Fischer

- Status changed from Feedback to Open

#20 - 2010-01-25 02:43 AM - Jürgen Fischer

Replying to [comment:21 jef]:

> *?*
> *I think we shouldn't try to intercept fatal GRASS errors in the GRASS library, we shouldn't be using GRASS libraries in the first place.*

IIRC the grass libraries also check for the grass version. So using the libraries directly also introduces a dependency to the exact GRASS version the plugin was build with.

#21 - 2010-01-26 12:16 AM - Redmine Admin

Replying to [comment:23 jef]:

> *IIRC the grass libraries also check for the grass version. So using the libraries directly also introduces a dependency to the exact GRASS version the plugin was build with.*

I am writing GRASS raster provider which is an experiment using 'GRASS' modules instead of libs. I decided however to use new modules compiled within QGIS for more reasons:

 - There is no great discipline of GRASS modules options/output stability

 - The output from GRASS modules is not always suitable

 - If something in GRASS changes, it is better IMO to get compilation error instead of silently wrong results because an output format has changed

 - Changes in library are much less probable than those in modules because usually involve a lot of work because hundreds of modules depend on them. Anybody can however change a module in few minutes.

We will see which problems it brings us.

#22 - 2010-03-15 12:51 PM - Redmine Admin

- Status changed from Open to Closed

- Resolution set to fixed

I have added exceptions support as requirement for GRASS libs.

Files

1878_grassexception.diff	21.7 KB	2009-08-27	Jürgen Fischer
except.tar	10 KB	2010-01-13	Redmine Admin