

QGIS Application - Feature request #2433

Fix relative paths from absolute-with-symlinks

2010-02-11 04:00 PM - Sandro Santilli

Status:	Closed	
Priority:	Low	
Assignee:	nobody -	
Category:	Project Loading/Saving	
Pull Request or Patch supplied:		Resolution: invalid
Easy fix?:	No	Copied to github as #: 12493
Description If your project file refers to resources by absolute path but the absolute path is formed by symlinks, the current code resolve them to a suboptimal <code>../../../../looking/path</code> . This patch uses QT library to "canonicalize" the absolute path prior to resolve against project dir. It is not tested under WIN. It would likely need canonicalization of the project filename too to be complete.		

History

#1 - 2010-02-11 04:44 PM - Jürgen Fischer

What's the point in using two different paths to the same directory in one project?

If you use the same path for both the datasources and the project, you already get what you want - whether or not the path includes a symlink.

If different paths are used, it might be even to get exactly this behavior.

Moreover canonicalPath() only works on existing files and (some?) GRASS data sources have paths that only exist partly - and the patch would clear those. This one wouldn't:

```
Index: src/core/qgsproject.cpp
=====
--- src/core/qgsproject.cpp    (revision 12928)
+++ src/core/qgsproject.cpp    (working copy)
@@ -1409,6 +1409,9 @@
 }

    QString srcPath = src;
+   QFileInfo fi( srcPath );
+   if ( fi.exists() )
+       srcPath = fi.canonicalFilePath();
    QString projPath = fileName();

    #if defined( Q_OS_WIN )
```

Still IMHO that patch is unnecessary and might even be counter productive.

Replying to [comment:1 jef]:

What's the point in using two different paths to the same directory in one project?

No point. It was qgis doing that, I just used the "browse" dialog to select the files, and maybe I started it from my home directory so had to go find the files somewhere. Didn't know I would have been somewhere else when re-opening the project and "saving as relative".

If you use the same path for both the datasources and the project, you already get what you want - whether or not the path includes a symlink.

If different paths are used, it might be even to get exactly this behavior.

I'm trying hard but I can't figure a reason why a user would ever want to get that behaviour. I would really expect it to "just work".

Of course an **additional** "canonicalization" of paths could be performed at "load time" but you never know what the user want. He may have added a symlink for the precise reason to find a predictable **absolute** path in the project file for example...
... but, when you want to go "relative" you really want the relative path to be as small as possible and that's what you get if you canonicalize both paths in the project file and pathname of the project file itself. My patch only canonicalize the paths in the project file.

Moreover canonicalPath() only works on existing files and (some?) GRASS data sources have paths that only exist partly - and the patch would clear those. This one wouldn't:

```
> Index: src/core/qgsproject.cpp
> =====
> --- src/core/qgsproject.cpp    (revision 12928)
> +++ src/core/qgsproject.cpp    (working copy)
> @@ -1409,6 +1409,9 @@
>  }
>
>
>  QString srcPath = src;
>  QFileInfo fi( srcPath );
>  if ( fi.exists() )
>  {
>      srcPath = fi.canonicalFilePath();
>      QString projPath = fileName();
>
>  }
>  #if defined( Q_OS_WIN )
```

Still IMHO that patch is unnecessary and might even be counter productive.

I would rather get a warning popup if a filename in the project file can't be found.
After all you can't guarantee you'll produce a project file that will work if you

can't find the files.

#3 - 2010-02-12 04:44 AM - Jürgen Fischer

Replying to [comment:2 strk]:

Replying to [comment:1 jef]:

What's the point in using two different paths to the same directory in one project?

No point. It was qgis doing that, I just used the "browse" dialog to select the files, and maybe I started it from my home directory so had to go find the files somewhere.

I don't understand. Didn't you use two different paths to the same directory and QGIS merely didn't notice or care, that there was a symlink involved?

I'm trying hard but I can't figure a reason why a user would ever want to get that behaviour. I would really expect it to "just work".

I does just work, doesn't it? It's just that the paths in the project file aren't the one you expected.

I would rather get a warning popup if a filename in the project file can't be found. After all you can't guarantee you'll produce a project file that will work if you can't find the files.

I was not implying that anything can't be found. For GRASS rasters the "filename" isn't actually a path to a file. The referred information is in multiple files in the directories within the GRASS mapset. The path to the mapset is still valid, existing and can be expressed relatively to the project file.

#4 - 2010-02-12 06:25 AM - Sandro Santilli

Replying to [comment:3 jef]:

Replying to [comment:2 strk]:

Replying to [comment:1 jef]:

What's the point in using two different paths to the same directory in one project?

No point. It was qgis doing that, I just used the "browse" dialog to select the files, and maybe I started it from my home directory so had to go find the files somewhere.

I don't understand. Didn't you use two different paths to the same directory and QGIS merely didn't notice or care, that there was a symlink involved?

I never used paths **explicitly**. What I did was requesting the load of two shapefiles.

Most likely the path to the directories containing the two shapefiles were the same as

IIRC the file browser remembers last location used. In any case the paths found in the .qgs file shared the path up to (but not including) the filename, so that part was fine.

Now, the "problem" (qgis problem, IMHO), is that these paths were reached using symlinks.

Next, I wanted to convert my "project file" into something I could send to my client.

A zip file containing the shapefiles AND the qgis project file. Easy thing shouldn't it be ?

At this time BOTH the shapefiles AND the project file were already in the same directory.

So I went to "project options" and specified "save paths as relative".

I expected qgis being able to find them, as BOTH the shapefiles AND the project file were in the same **canonical** directory.

| *I does just work, doesn't it? It's just that the paths in the project file aren't the one you expected.*

Yes, if you want. It still works for me, but still can't put in the zip file :)

| *For GRASS rasters the "filename" isn't actually a path to a file.
The referred information is in multiple files in the directories within the GRASS mapset.
The path to the mapset is still valid, existing and can be expresses relatively to
the project file.*

Uhm... ok, what about passing the dirname to the canonicalizator function then ?

Is the dirname expected to exist ? Or, if other kind of "source-specification" exist, is there a way to tell what refer to a filesystem path and what not ?

Mind you: I didn't introduce this "relative paths" thing myself in qgis, so please don't take the chance to have me rewrite it all :P

#5 - 2010-02-12 06:34 AM - Sandro Santilli

How about adding this:

```
if ( srcPath.empty() ) srcPath = src;
```

Between lines 1413 and 1414 of my patch ?

QDir::canonicalPath() is documented to return the empty string

if the canonical path doesn't exist

#6 - 2010-02-12 07:30 AM - Sandro Santilli

I add an updated patch.

Checks if the path does have a "canonical" representation (it does if it exists on the filesystem as such) and falls-back to using the original representation.

Also, canonicalizes the project filename too (this one might be unneeded if fileName() returns it already canonicalized, which I didn't check).

#7 - 2010-02-12 07:48 AM - Sandro Santilli

Argh, forget the second patch. It would abort. I was assuming too much.

I think I'm off on this, isn't that much important to justify the time spent.

#8 - 2010-04-07 04:57 AM - Marco Hugentobler

- *Resolution set to invalid*
- *Status changed from Open to Closed*

So we can close this ticket, right?

Files			
relative_path_from_symlink.diff	533 Bytes	2010-02-11	Sandro Santilli
relative_path_from_symlink_2.diff	2.83 KB	2010-02-12	Sandro Santilli