QGIS Application - Bug report #17364
svg marker symbol: relative path broken

2017-10-31 04:45 AM - Mathieu Pellerin - nIRV

Status:	Closed		
Priority:	High		
Assignee:	Nyall Dawson		
Category:	Symbology		
Affected QGIS version:	master	**Regression?:**	Yes
Operating System:		**Easy fix?:**	No
Pull Request or Patch supplied:	No	**Resolution:**	fixed/implemented
Crashes QGIS or corrupts data:	No	**Copied to github as #:**	25261

Description

For quite a long time now under QGIS master (what will become 3.0), providing a relative path for an SVG marker (through an expression) is broken.

Steps to reproduce

1. Decompress the attached relative_svg_broken.zip file
2. Notice the presence of an SVG image in the decompressed folder (ca.svg)
3. Open the relative_svg_broken.qgs project file
4. Notice the "?" symbol on the canvas, indicating a failure to read the SVG image
5. Open the style dock for the project's point layer, go to the SVG marker layer
6. Notice the image path is set via the following expression: 'ca.svg'

This is a regression, used to work just fine under 2.1X.

Associated revisions

Revision ec224117 - 2018-01-29 10:51 AM - Nyall Dawson

Expose project QgsPathResolver to render context

So that it can be used when resolving paths to data defined
file-based settings, e.g. svg marker paths.

Fixes #17364

History

#1 - 2017-10-31 05:02 AM - Nyall Dawson

- Assignee set to Martin Dobias

Martin - I had a quick look at this, and I don't believe it's resolvable without either adding the project instance as a member to QgsRenderContext, OR adding the project's path resolver to QgsRenderContext.

Does that sound correct to you?

My preference would be to add QgsProject to QgsRenderContext. I believe it's the correct thing to do, since the project IS part of the context under which a render operation occurs. I also can't see how we would ever remove QgsProject.instance() from QgsExpression without doing this (maybe via expression context handling of project related members).

(assigned to you just to ping you with this question)

#2 - 2017-10-31 08:35 AM - Martin Dobias

Yeah you are right, we will need access to QgsPathResolver or QgsProject in rendering. The other question is how do we get it there - we would probably need QgsMapSettings to also have access to path resolver or project - and this would propagate also to QgsMapCanvas and other classes...

I was wondering if we could use the fact that if layers belong to a project, they are parented to it in QObject hierarchy - shall we maybe use that to implicitly find out project of a layer?

Finally it is the question of whether to use just the path resolver or project. If we use QgsProject, we get more power at the expense of allowing rendering code to do stuff it should not do (project modification, threading issues). I do not have a clear answer yet :-)

#3 - 2017-10-31 11:58 PM - Nyall Dawson

> *Finally it is the question of whether to use just the path resolver or project. If we use QgsProject, we get more power at the expense of allowing rendering code to do stuff it should not do (project modification, threading issues). I do not have a clear answer yet :-)*

My gut feeling is QgsProject is the way to go, and we'll just need to be vigilant about blocking incorrect use of it.

There's a lot of expression functions now which rely on access to the project in order to retrieve map layers, relations, and edit widget configuration. I think having access to the whole project is going to be the only way we'll eventually be able to pull QgsProject::instance out of expressions.

#4 - 2018-01-27 12:32 AM - Nyall Dawson

- Assignee changed from Martin Dobias to Nyall Dawson

#5 - 2018-01-29 06:33 AM - Nyall Dawson

- Status changed from Open to In Progress

PR https://github.com/qgis/QGIS/pull/6203

#6 - 2018-01-29 10:51 AM - Nyall Dawson

- Status changed from In Progress to Closed

- % Done changed from 0 to 100

Applied in changeset commit:qgis|ec224117debaa31ac76bdbc73c3158edb8175e27.

#7 - 2018-02-22 11:14 AM - Giovanni Manghi

- Resolution set to fixed/implemented

Files

relative_svg_broken.zip	9.04 KB	2017-10-31	Mathieu Pellerin - nIRV