QGIS Application - Bug report #16823

If multi threaded rendering is enabled adding a WCS Layer makes multiple GetCoverage calls

2017-07-10 03:49 AM - William Valencia

Status:	Closed			
Priority:	Normal			
Assignee:				
Category:	Web Services clients/WCS			
Affected QGIS version:2.18.9		Regression?:	No	
Operating System:		Easy fix?:	No	
Pull Request or Patch supplied:		Resolution:	end of life	
Crashes QGIS or corru pits data:		Copied to github a	Copied to github as #: 24722	

Description

I have a WCS Server and I use QGIS to add a WCS Layer from my WCS server. I noticed there is 1 GetCapabilities call, 1 DescribeCoverage call, and 4 GetCoverage calls that QGIS requests from my server for adding a single layer. This seems excessive. Shouldn't there be only 1 GetCoverage call?

This is running 2.18.9 QGIS on OSX.

QGIS version 2.18.9 QGIS code branch Release 2.18 Compiled against Qt 4.8.6 Running against Qt 4.8.6 Compiled against GDAL/OGR 2.1.3 Running against GDAL/OGR 2.1.2 Compiled against GEOS 3.6.1-CAPI-1.10.1 Running against GEOS 3.5.1-CAPI-1.9.1 r4246 PostgreSQL Client Version 9.3.4 SpatiaLite Version 4.3.0a QWT Version 6.1.3 **PROJ.4 Version** 492

History

#1 - 2017-07-10 04:06 AM - William Valencia

I noticed something weird with QGIS. So when making an "Add WCS Layer" request with QGIS, I noticed that (After initial GetCapabilities and DescribeCoverage), multiple redundant calls to my WCS Server are made. The actual "Add" Button makes 12 calls to EGI. It calls 1 GetCapabilities, 1 DescribeCoverage, 4 GetCoverage and then repeats the last 6 calls for a total of 12 calls. I am unsure why QGIS makes 4 GetCoverage calls in the first place, then it repeats the last 6 calls.

#2 - 2017-07-10 01:48 PM - Giovanni Manghi

- Priority changed from High to Normal
- Category changed from Actions to Web Services clients/WCS
- Status changed from Open to Feedback

Would you mind asking in the developers mailing list of this is expected or not? thanks.

#3 - 2017-07-19 07:08 PM - William Valencia

I asked the developers mailing list and included their response below. Looks like the extra 6 repeated calls are definitely a bug with multi threaded rendering. The multiple (4) GetCoverage calls may or may not be needed any more as well.

Hi William

On Wed, Jul 12, 2017 at 8:29 AM, William Valencia <<u>william.m.valencia@nasa.gov</u>> wrote:

I noticed something weird with QGIS. So when making an "Add WCS Layer" request with QGIS, I noticed that (After initial GetCapabilities and DescribeCoverage), multiple redundant calls to my WCS Server are made. The actual "Add" Button makes 12 calls to my WCS Server. It calls 1 GetCapabilities, 1 DescribeCoverage, 4 GetCoverage and then repeats the last 6 calls for a total of 12 calls. I am unsure why QGIS makes 4 GetCoverage calls in the first place, let alone the repeated last 6 calls for the total of 12 calls. Anyone know why QGIS is doing this?

I am not the author of the WCS provider, but I have some leads... the extra GetCoverage calls are used for detection of some server behaviors, from the code comments it looks like at some point there were incompatibilities among WCS server implementations that lead to addition of these checks to work around the incompatibilities. Not sure if these checks are still necessary with the current versions of map servers.

For the repeated calls, this a side-effect of introduction of multi-threaded rendering since QGIS 2.4. WCS provider get cloned for use in worker thread, but the cloning is sub-optimal - it does not pass the downloaded capabilities, so cloned provider has to fetch that again. It should not be hard to fix (I have fixed that for WMS in the initial work, but WCS provider had low priority at the time).

Regards Martin

#4 - 2017-09-25 04:52 PM - Giovanni Manghi

- Subject changed from Adding a WCS Layer makes multiple (4) GetCoverage calls. to If multi threaded rendering is enabled adding a WCS Layer makes multiple GetCoverage calls

- Status changed from Feedback to Open

- Operating System deleted (MacOS Sierra 10.12.5)

#5 - 2019-03-09 03:10 PM - Giovanni Manghi

- Resolution set to end of life
- Status changed from Open to Closed

End of life notice: QGIS 2.18 LTR

Source:

http://blog.qgis.org/2019/03/09/end-of-life-notice-qgis-2-18-ltr/

QGIS 3.4 has recently become our new Long Term Release (LTR) version. This is a major step in our history – a long term release version based on the massive updates, library upgrades and improvements that we carried out in the course of the 2.x to 3x upgrade cycle.

We strongly encourage all users who are currently using QGIS 2.18 LTR as their preferred QGIS release to migrate to QGIS 3.4. This new LTR version will receive regular bugfixes for at least one year. It also includes hundreds of new functions, usability improvements, bugfixes, and other goodies. See the relevant changelogs for a good sampling of all the new features that have gone into version 3.4

Most plugins have been either migrated or incorporated into the core QGIS code base.

We strongly discourage the continued use of QGIS 2.18 LTR as it is now officially unsupported, which means we'll not provide any bug fix releases for it.

You should also note that we intend to close all bug tickets referring to the now obsolete LTR version. Original reporters will receive a notification of the ticket closure and are encouraged to check whether the issue persists in the new LTR, **in which case they should reopen the ticket**.

If you would like to better understand the QGIS release roadmap, check out our roadmap page! It outlines the schedule for upcoming releases and will help you plan your deployment of QGIS into an operational environment.

The development of QGIS 3.4 LTR has been made possible by the work of hundreds of volunteers, by the investments of companies, professionals, and administrations, and by continuous donations and financial support from many of you. We sincerely thank you all and encourage you to collaborate and support the project even more, for the long term improvement and sustainability of the QGIS project.