

QGIS Application - Bug report #19595

"Clip Raster by Mask Layer" is actually "Resize to mask layer"

2018-08-10 08:56 PM - Loren Amelang

Status: Closed	
Priority: Normal	
Assignee:	
Category: Processing/GDAL	
Affected QGIS version: 3.2.1	Regression?: No
Operating System: Windows 10 Creators with 3000x2000 screen resolution	Easy fix?: No
Pull Request or Patch supplied: No	Resolution: invalid
Crashes QGIS or corrupts data: No	Copied to github as #: 27422
Description	
<p>[To see this problem and others with more context, check #19584.]</p> <p>I had three 400MB 1-meter DEM .tif files, and RiverGIS only lets you query one file name per project, so I wanted to consolidate them. I tried Raster -> Miscellaneous -> Merge, but got a Memory Error every time.</p> <p>I thought I'd cut the file size, so I made a vector layer with the rectangular boundary of my project. It included the corner where the three files met, and a portion of each file, plus some blank space in the fourth quadrant.</p> <p>I used Raster -> Extraction -> Clip Raster by Mask Layer, and it clipped the parts of each file outside the boundary. BUT it made each output file the size of the full mask boundary! Is that expected? So they didn't get much smaller.</p> <p>I'd expect a clipped file to retain its original boundaries, except where it is actually clipped further. The function I used should be called "Resize to mask layer"!</p>	
Related issues:	
Related to QGIS Application - Bug report # 19584: Raster Clip and Merge issue...	Closed 2018-08-10

History

#1 - 2018-08-10 09:43 PM - Jürgen Fischer

- Related to Bug report #19584: Raster Clip and Merge issues (QGIS 3.2.1) added

#2 - 2018-08-10 09:45 PM - Jürgen Fischer

- Description updated

#3 - 2018-08-13 12:59 PM - Giovanni Manghi

- Category changed from Processing/QGIS to Processing/GDAL

- Status changed from Open to Feedback

I had three 400MB 1-meter DEM .tif files, and RiverGIS only lets you query one file name per project, so I wanted to consolidate them. I tried Raster -> Miscellaneous -> Merge, but got a Memory Error every time.

add the project/data, thanks.

I thought I'd cut the file size, so I made a vector layer with the rectangular boundary of my project. It included the corner where the three files met, and a portion of each file, plus some blank space in the fourth quadrant. I used Raster -> Extraction -> Clip Raster by Mask Layer, and it clipped the parts of each file outside the boundary. BUT it made each output file the size of the full mask boundary! Is that expected? So they didn't get much smaller. I'd expect a clipped file to retain its original boundaries, except where it is actually clipped further. The function I used should be called

| *"Resize to mask layer"!*

The QGIS Processing/GDAL tools are just a GUI for the... GDAL tools command line utilities. In this case the tool used for the clip is gdalwarp

<https://www.gdal.org/gdalwarp.html>

can you see an option that would prevent that?

#4 - 2018-08-13 11:29 PM - Loren Amelang

As I said in #19594, the DEM files are 1200 MB, and the project folder is 2.73 GB. Upload from here would take days...

The QGIS Processing/GDAL tools are just a GUI for the... GDAL tools command line utilities. In this case the tool used for the clip is gdalwarp

<https://www.gdal.org/gdalwarp.html>

can you see an option that would prevent that?

The command you see before you run it doesn't show the options. Log afterward does:

GDAL command:

```
gdalwarp -ot Float32 -of GTiff -cutline "C:/Users/loren/Giant Files/HEC-RAS Dam Break/QGIS Projects/FEMA808/CenterlineLayerExtent.gpkg"
-crop_to_cutline "C:/Users/loren/Giant Files/HEC-RAS Dam Break/QGIS Projects/FEMA808/DEM_46_436_to2226.tif"
C:/Users/loren/AppData/Local/Temp/processing_6d811e1bb3544572b1fa6f5c07aa1ad7/d0ee9a39826c4270ad52573bf81db808/OUTPUT.tif
```

<https://www.gdal.org/gdalwarp.html>

-crop_to_cutline:

(GDAL >= 1.8.0) Crop the extent of the target dataset to the extent of the cutline.

Polygon cutlines may be used as a mask to restrict the area of the destination file that may be updated, including blending.

If the OGR layer containing the cutline features has no explicit SRS, the cutline features must be in the SRS of the destination file.

When writing to a not yet existing target dataset, its extent will be the one of the original raster unless -te or -crop_to_cutline are specified.

My output extent seems to be the extent of the cutline mask, NOT the original raster. I expected it to be the original raster, but clipped way smaller... But I guess they could argue that "Crop the extent of the target dataset to the extent of the cutline" means to make the output file as big as the mask - which they did. But I don't consider "crop" an appropriate name for a function that makes your file about 8X larger!

#5 - 2018-08-14 11:51 AM - Giovanni Manghi

Loren Amelang wrote:

| *As I said in #19594, the DEM files are 1200 MB, and the project folder is 2.73 GB. Upload from here would take days...*

can you post a link where we can download the datasets you are using?

| *But I don't consider "crop" an appropriate name for a function that makes your file about 8X larger!*

have you checked in the advanced parameters what TYPE of raster are you outputting? If you are outputting FLOAT and your inputs are integer then is likely explained why you are getting large outputs. Anyway, even admitting that this was a bug (which I don't think it is) on this specific case you should report it to GDAL, not QGIS (here QGIS works are a GUI for GDAL).

#6 - 2018-08-14 09:51 PM - Loren Amelang

- File *CenterlineLayerExtentBufSqr.gpkg* added

@Giovanni Manghi,

The huge DEM files are at:

https://prd-tnm.s3.amazonaws.com/StagedProducts/Elevation/1m/IMG/USGS_NED_one_meter_x46y436_CA_FEMA_R9_Russian_2017_IMG_2018.zip

https://prd-tnm.s3.amazonaws.com/StagedProducts/Elevation/1m/IMG/USGS_NED_one_meter_x47y435_CA_FEMA_R9_Russian_2017_IMG_2018.zip

https://prd-tnm.s3.amazonaws.com/StagedProducts/Elevation/1m/IMG/USGS_NED_one_meter_x47y436_CA_FEMA_R9_Russian_2017_IMG_2018.zip

Cropping boundary is attached.

The DEMs begin as 400 MB each, float32, and they form a triangle so the uncropped output file is about 1500 MB. I guess if some bit of the process was still 32-bit, that would be dangerous.

Do you have a link handy for GDAL reports?

#7 - 2018-08-15 11:42 AM - Giovanni Manghi

- File *Screenshot_20180815_104311.png* added

Loren Amelang wrote:

@Giovanni Manghi,

The huge DEM files are at:

https://prd-tnm.s3.amazonaws.com/StagedProducts/Elevation/1m/IMG/USGS_NED_one_meter_x46y436_CA_FEMA_R9_Russian_2017_IMG_2018.zip

https://prd-tnm.s3.amazonaws.com/StagedProducts/Elevation/1m/IMG/USGS_NED_one_meter_x47y435_CA_FEMA_R9_Russian_2017_IMG_2018.zip

https://prd-tnm.s3.amazonaws.com/StagedProducts/Elevation/1m/IMG/USGS_NED_one_meter_x47y436_CA_FEMA_R9_Russian_2017_IMG_2018.zip

Cropping boundary is attached.

The DEMs begin as 400 MB each, float32, and they form a triangle so the uncropped output file is about 1500 MB. I guess if some bit of the process was still 32-bit, that would be dangerous.

Do you have a link handy for GDAL reports?

So let's make a short resume here:

- 1) The 3 dem files are 385MB each, in IMG format https://www.gdal.org/frmt_hfa.html and are FLOAT
- 2) merging the 3 dem files with `gdal_merge` has worked without any issue here. The command created by QGIS was

```
gdal_merge.py -ot Float32 -of GTiff -o /tmp/processing_40aa293201da415a9c0d3c70702e9c26/123ac210bb0048c8b6f4247d1ec4696e/OUTPUT.tif
--optfile /tmp/processing_40aa293201da415a9c0d3c70702e9c26/mergeInputFiles.txt
```

and I can re-use it to do the operation from the command line, again without any issue.

3) The merged DEM file is TIFF/FLOAT and is 1.5GB, which is compatible with the change of format (by default the GeoTIFF generated by QGIS have no compression whatsoever)

4) I used your polygon to clip the merged DEM (with the `-crop_to_outline` option) and had no issues at all. And the result is what is expected: a clip of your (merged) dem that has exactly the extent (bbox) of the clipping polygon

```
gdalwarp -ot Float32 -of GTiff -cutline /home/giovanni/Downloads/CenterlineLayerExtentBufSqr.gpkg -crop_to_outline
/home/giovanni/Downloads/merged.tif /tmp/processing_40aa293201da415a9c0d3c70702e9c26/e13f5bf4cbbf4bee81e3e55dd923606a/OUTPUT.tif
```

5) if you need a smaller result (the result of the above clip is a 653MB Tiff) you can use the `gdal_translate` tool, it has a list of predefined profiles, one of them is "high compression" that will return a 160mb clipped dem

Result: I don't see any issues at all. I used QGIS 3.2 on Linux and also tested on a Windows 10 VM

Attached image.

#8 - 2018-08-15 10:57 PM - Loren Amelang

- File *GDAL Merge Log.txt* added
- File *Before Merge.JPG* added
- File *After Merge (hit 58 pct once).JPG* added
- File *Second Try Memory.JPG* added

So I'm following your resume...

```
gdal_merge.py -ot Float32 -of GTiff -o
/tmp/processing_40aa293201da415a9c0d3c70702e9c26/123ac210bb0048c8b6f4247d1ec4696e/OUTPUT.tif --optfile
/tmp/processing_40aa293201da415a9c0d3c70702e9c26/mergeInputFiles.txt
```

and I can re-use it to do the operation from the command line, again without any issue.

My command is different:

```
cmd.exe /C gdal_merge.bat -ot Float32 -of GTiff -o
C:/Users/loren/AppData/Local/Temp/processing_544af2b511d3484ab90a79f29ca1f08b/a380246e366848acbc7f8f9db32b56fc/OUTPUT.tif
--optfile C:/Users/loren/AppData/Local/Temp/processing_544af2b511d3484ab90a79f29ca1f08b\mergeInputFiles.txt
```

I always see `cmd.exe` and `*.bat`, never `*.py`. See "GDAL Merge Log.txt".

Except for the two tries I reported yesterday in #19596, all of the other twenty or so times I've tried this have shown the "memory error". Every try today fails.

QGIS remained open last night while the system hibernated... No, restarting it fresh does not help, it just failed again.

"Before Merge.JPG" shows the command, and memory usage before the merge.

"After Merge (hit 58 pct once).JPG" shows the result of the first (memory error) run. Total memory usage never went above 58%.

"Second Try Memory.JPG" shows the max memory used by Python jumped way up on the second failed try. (Yes, I can re-use the dialog, if I know to un-select the automatically selected files and re-select the original ones - #19628.)

The system has 8 GB of RAM and never shows using more than about 60% of it.

4) I used your polygon to clip the merged DEM (with the `-crop_to_outline` option) and had no issues at all. And the result is what is expected: a clip of your (merged) dem that has exactly the extent (bbox) of the clipping polygon

Yes, clipping the merged DEM to that polygon gives the result I'd expect.

What failed for me was using that function to clip each of the individual DEM files to their useful fragments *before* merging them (in the hope of avoiding the memory error).

Even if only a tiny corner of the source file was inside the clip polygon, the result was the size of the full polygon.

Which was still too big to merge here.

After I manually clipped each file to the small segment needed, merge was able to handle them - 327 MB instead of 1.6 GB.

So size really is a factor...

But yesterday it was able to merge the 1.6 GB version - twice!

So the problem is that the problem appears randomly, like almost all of my QGIS problems.

As in #19596. I guess you're saying I really do need to take this all up with an astrologer or shaman...

Can you think of anything that would inject such randomness into QGIS?

Is there some debug tool that would show me more of what is happening internally when these problems appear?

(Short of downloading the whole source and learning my way around it all...)

#9 - 2018-08-31 05:22 PM - Giovanni Manghi

- Resolution set to invalid

- Status changed from Feedback to Closed

Loren Amelang wrote:

| So I'm following your resume...
| [...]

you sure you can use and re-use the command I posted? It has unix type paths, I don't think that they will work on Windows? or have you adapted them? Assuming that you adapted them, do you confirm that you can use them from the Osgeo4W shell without any issue?

| My command is different:
| [...]

no, actually is identical.

| I always see `cmd.exe` and `*.bat`, never `*.py`. See "GDAL Merge Log.txt".

the fact that from within QGIS it starts with "cmd.exe" and launches a .bat file instead of a .py one is not important. It is like that because QGIS needs to

launch an external command on the (windows) command line (cmd.exe) and on Windows the gdal python scripts are wrapped into .bat files.

*Except for the two tries I reported yesterday in #19596, all of the other twenty or so times I've tried this have shown the "memory error". Every try today fails.
QGIS remained open last night while the system hibernated... No, restarting it fresh does not help, it just failed again.*

I still think that is a resources problem of your environment, not a qgis one at all.

*"Before Merge.JPG" shows the command, and memory usage before the merge.
"After Merge (hit 58 pct once).JPG" shows the result of the first (memory error) run. Total memory usage never went above 58%.*

The process you should monitor is gdal_merge, not qgis or else. I don't think that if it fails for memory shortage it will leave any trace after crashing.

*"Second Try Memory.JPG" shows the max memory used by Python jumped way up on the second failed try. (Yes, I can re-use the dialog, if I know to un-select the automatically selected files and re-select the original ones - #19628.)
The system has 8 GB of RAM and never shows using more than about 60% of it.*

Bottom line, and we need a straight answer here: do you have any evidence that running a gdal_merge operation directly from the OSGeo4W shell works while launching the very same command (same options and inputs) from within QGIS does fail?

Anyway: <https://issues.qgis.org/issues/19594#note-14>

*Yes, clipping the merged DEM to that polygon gives the result I'd expect.
What failed for me was using that function to clip each of the individual DEM files to their useful fragments before merging them (in the hope of avoiding the memory error).
Even if only a tiny corner of the source file was inside the clip polygon, the result was the size of the full polygon.
Which was still too big to merge here.*

if you are using the GDAL based tool, the clip option of the gdalwarp tool says "Crop the extent of the target dataset to the extent of the outline", so the result you are seeing is expected.

You may want to try other tools in the Processing toolbox that do the same, I'm almost sure there is a native QGIS one as also a SAGA one. Possibly they work the way you expect.

*After I manually clipped each file to the small segment needed, merge was able to handle them - 327 MB instead of 1.6 GB.
So size really is a factor...
But yesterday it was able to merge the 1.6 GB version - twice!
So the problem is that the problem appears randomly, like almost all of my QGIS problems.*

if you refer to problems in qgis while running gdal algorithms then they are not qgis problems. And the "randomness" you are seeing is likely a result of the availability of memory resources at a given moment (when you try to run an operation that make use of large inputs).

As in #19596. I guess you're saying I really do need to take this all up with an astrologer or shaman...

don't think so.

Files

CenterlineLayerExtentBufSqr.gpkg	116 KB	2018-08-14	Loren Amelang
Screenshot_20180815_104311.png	351 KB	2018-08-15	Giovanni Manghi
GDAL Merge Log.txt	2.33 KB	2018-08-15	Loren Amelang
Before Merge.JPG	277 KB	2018-08-15	Loren Amelang
After Merge (hit 58 pct once).JPG	465 KB	2018-08-15	Loren Amelang
Second Try Memory.JPG	37.7 KB	2018-08-15	Loren Amelang