

QGIS Application - Feature request #5742

Preserve GRASS raster labels

2012-06-04 11:58 PM - Paolo Cavallini

Status:	Open	
Priority:	Normal	
Assignee:		
Category:	Processing/GRASS	
Pull Request or Patch supplied:		Resolution:
Easy fix?:	No	Copied to github as #: 15249
Description		
<ul style="list-style-type: none">- GRASS raster can have labels (very convenient, e.g. r.param.scale saves feature names along with categories)- when saving to TIFF, these are stripped- as a consequence, the flow DTM>Features raster>Feature vector strips the categories, and data are lost. <p>Of course we can Use raster values as categories, but keeping the labels would be much better. One way would be to pipe directly the two GRASS commands, without exporting in between, but this is not implemented yet.</p> <p>Notes from Etienne Tourigny:</p> <p>labels could be added as GTIFF metadata, which GDAL supports but I don't know how/if this could be done in GRASS</p> <p>If the export is using r.out.gdal there must be a way to put/get metadata in a gtiff?</p> <p>from r.out.gdal:</p> <pre>metaopt=string[,string,...]</pre> <p>Metadata key(s) and value(s) to include</p> <p>In the form of "META-TAG=VALUE", separate multiple entries with a comma. Not supported by all output format drivers.</p> <p>Notes from Benjamin Duche:</p> <p>Theoretically, it wouldn't be very hard to read the labels from the mapset in which SEXTANTE is storing the (temporary) GRASS datasets; they are saved as plain ASCII there.</p> <p>However, the host GIS for SEXTANTE must then somehow deal with this additional data in some useful way (e.g. via a virtual raster attribute table).</p>		

History

#1 - 2012-06-07 08:56 AM - Paolo Cavallini

From Markus Metz, GRASS mailing list:

An alternative would be to make use of the GDAL raster attribute table in both GRASS and QGIS, with the benefit that reading labels for other raster maps imported with GDAL would also be supported.

IIUC, QGIS is currently using GDALGetRasterCategoryNames() which has very limited functionality and supports only integer maps. In a GDAL raster attribute table, as with GRASS-internal categories, floating point values and data ranges are supported, e.g.

min=358.5, max=359.5, name=359 degrees ccw from east

A GDAL raster attribute table would also solve the problem of importing/exporting color rules for floating point data types.

#2 - 2013-11-17 07:48 AM - Giovanni Manghi

- Status changed from Open to Feedback

Paolo Cavallini wrote:

From Markus Metz, GRASS mailing list:===

An alternative would be to make use of the GDAL raster attribute table in both GRASS and QGIS, with the benefit that reading labels for other raster maps imported with GDAL would also be supported.

IIUC, QGIS is currently using GDALGetRasterCategoryNames() which has very limited functionality and supports only integer maps. In a GDAL raster attribute table, as with GRASS-internal categories, floating point values and data ranges are supported, e.g.

min=358.5, max=359.5, name=359 degrees ccw from east

A GDAL raster attribute table would also solve the problem of importing/exporting color rules for floating point data types.

may be this has been implemented in qgis?

see

#8bdaf42dbaec240151eacaa9a28e02f49a53fd66

#3 - 2013-11-17 10:36 PM - Paolo Cavallini

I think more work has to be done on the Processing side, to add these options to the export command.

#4 - 2014-10-04 11:52 AM - Giovanni Manghi

- Category deleted (59)

- Project changed from 78 to QGIS Application

#5 - 2014-10-04 11:53 AM - Giovanni Manghi

- Category set to Processing/GRASS

#6 - 2016-02-19 11:29 AM - Médéric RIBREUX

Hello,

after some work on r.category implementation, I've found that in GRASS7 (and only GRASS7), you can use the '-t' parameter for r.out.gdal (command that

is internally used by Processing GRASS) to export raster labels. Labels are exported in GDAL Raster Attribute Table (RAT). The RAT takes the form of an xml file named after the raster filename concatenated with '.aux.xml'.

For the moment, QGIS raster provider can't use the values that are in the RAT. The only thing that QGIS uses for rasters are GDAL categories (and only for integer rasters). But we can make Processing GRASS algorithms to export raster labels just by adding the '-t' parameter to every r.out.gdal output (for the default case).

For importing labels and keeping labels, this is a little bit more complex: it seems that only r.in.gdal (which duplicates the raster) can import labels into GRASS7 (but the import is complete from the RAT and very well integrated into GRASS database). Processing uses r.external for raster imports (because it just makes a link to the raster file, there is no data duplication). We have different choices (as far as I can see):

- replace r.external to r.in.gdal by default (everywhere) with the drawback to always duplicate data (but some of GRASS7 algorithms need to use r.in.gdal instead of r.External).
- use only r.in.gdal for some algorithms (r.category, r.reclass): implementation can be made with the 'ext' mechanism so no need to change main Grass7Algorithm.py.
- add an (virtual) option as a checkbox in every algorithm dialog box which could be entitled like: 'preserve raster labels (warning: duplicates raster data in temporary GRASS database)'. If the user checks the box, use r.in.gdal instead of r.external (just some lines to add in Grass7Algorithm.py and for the GUI). It would be like the region extent or the cell size options.
- add a new general option for GRASS7 for 'preserve raster labels' which define if user want to import labels by default.
- both of the two last points.
- do nothing and open a GRASS7 ticket to add label preservation for r.external !

What do you think about this ? Feel free to add any other choice or just your opinion...

#7 - 2016-02-19 11:49 AM - Paolo Cavallini

IMHO the last option is the best one.

#8 - 2016-03-13 08:53 AM - Médéric RIBREUX

- Status changed from Feedback to Open

Ok, I have opened [a ticket into GRASS bugtracker for r.external...](#)

On QGIS side, we need to add '-t' option for r.out.gdal command in Grass7Algorithm.py. I'll do it once [my global r.* Pull-Request](#) will be merged.

#9 - 2017-05-01 12:48 AM - Giovanni Manghi

- Easy fix? set to No

#10 - 2018-02-23 07:11 PM - Paolo Cavallini

- Assignee deleted (Victor Olaya)

- Description updated

Still true in Q3