GRASS GIS 7.4: What's new in a nutshell

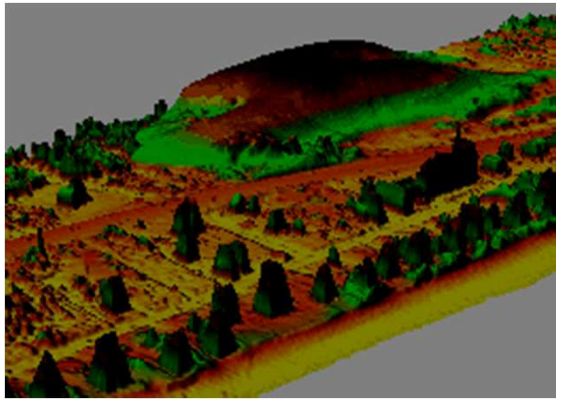
Markus Neteler, Veronica Andreo, Luca Delucchi, Martin Landa, Moritz Lennert, Vaclav P
GRASS Development Team

grass.osgeo.org

FOSS4G 2018 – Dar es Salaam

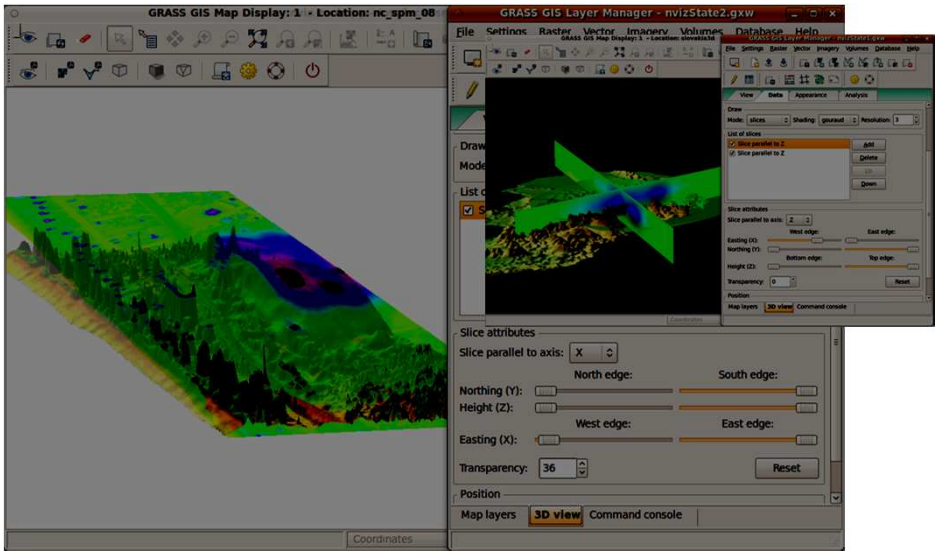
What is GRASS GIS?

- GRASS GIS is a hybrid, modular GIS software
- GRASS = Geographic Resources Analysis Support System
- GNU General Public License - freely available



Nags Head LiDAR time series: dune moving c

- Raster and topological vector data functionality
- 3D raster (voxel) processing
- Image processing
- Visualization options
- Time series analysis

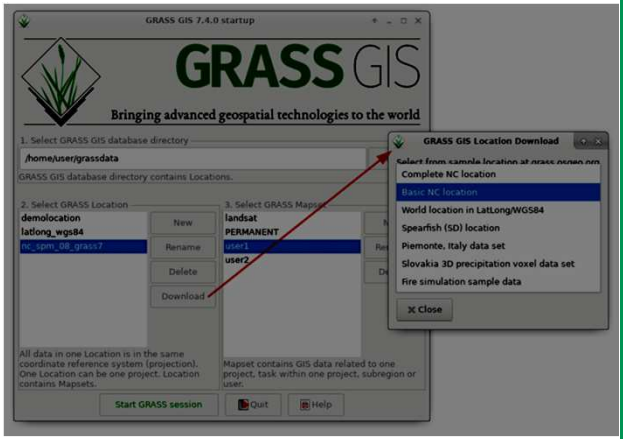


- Portable software ("all" operating systems)
- Graphical user interface and command line

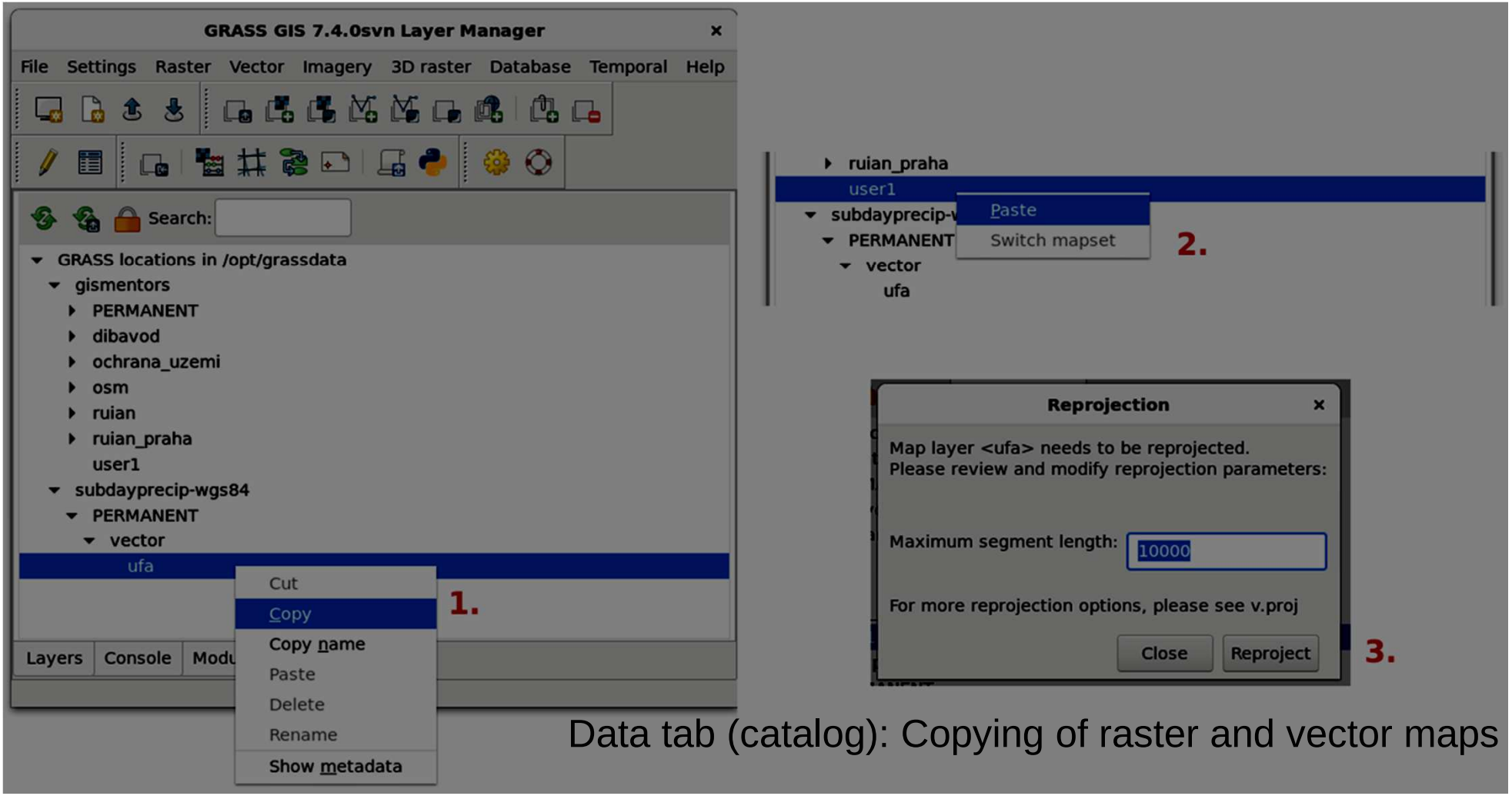
What is new in GRASS GIS 7.4?

New stable version GRASS GIS 7.4

- Improved usability and graphical user interface
- New compression of internal "no data" file which can be huge
- Support for global data extending beyond -180/+180, -90/+90
- Orthorectification with user interface was newly implemented in GRASS GIS 7
- GUI: New *Download* button for sample data
- ... over 480 improvements since G7.2.0

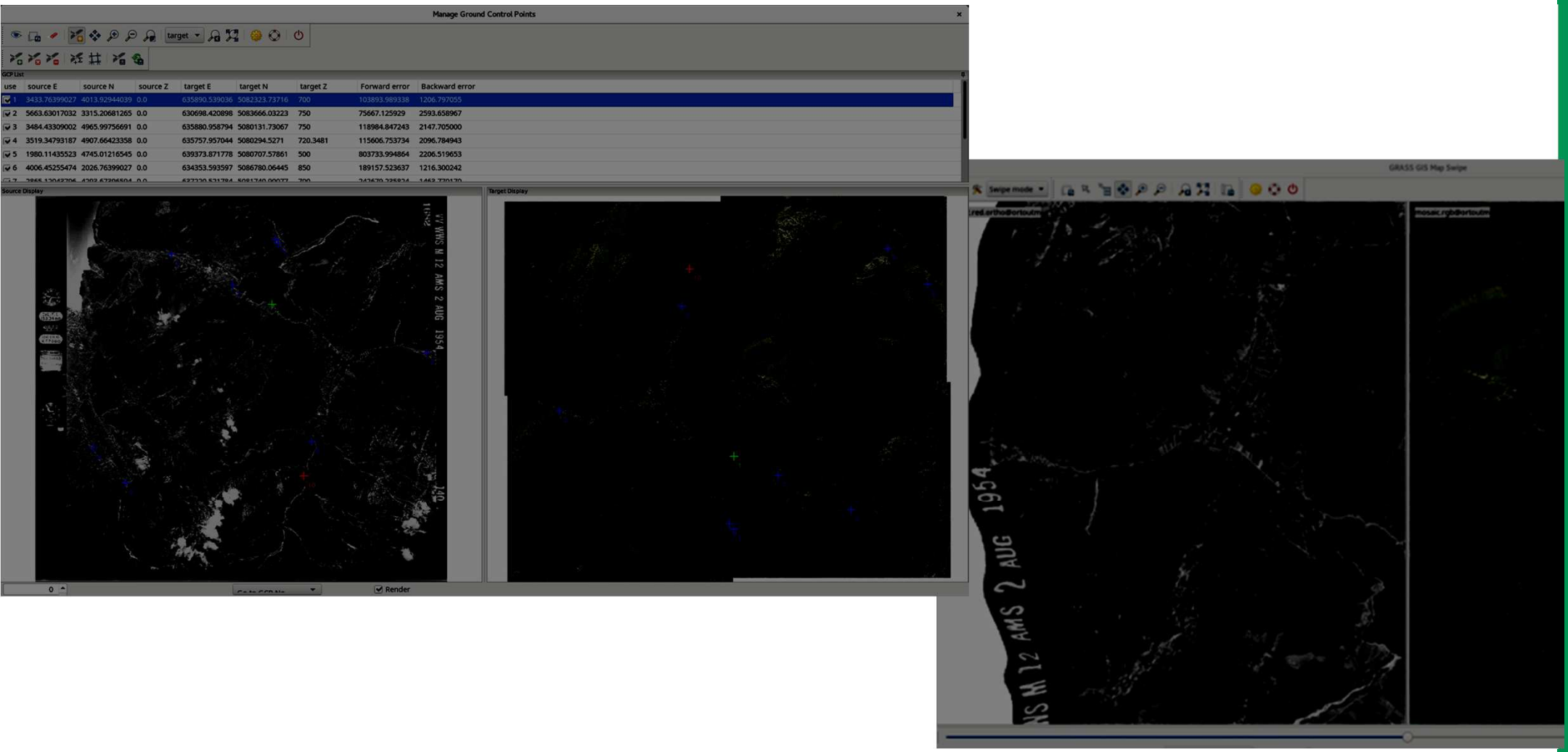


Data catalog improvements

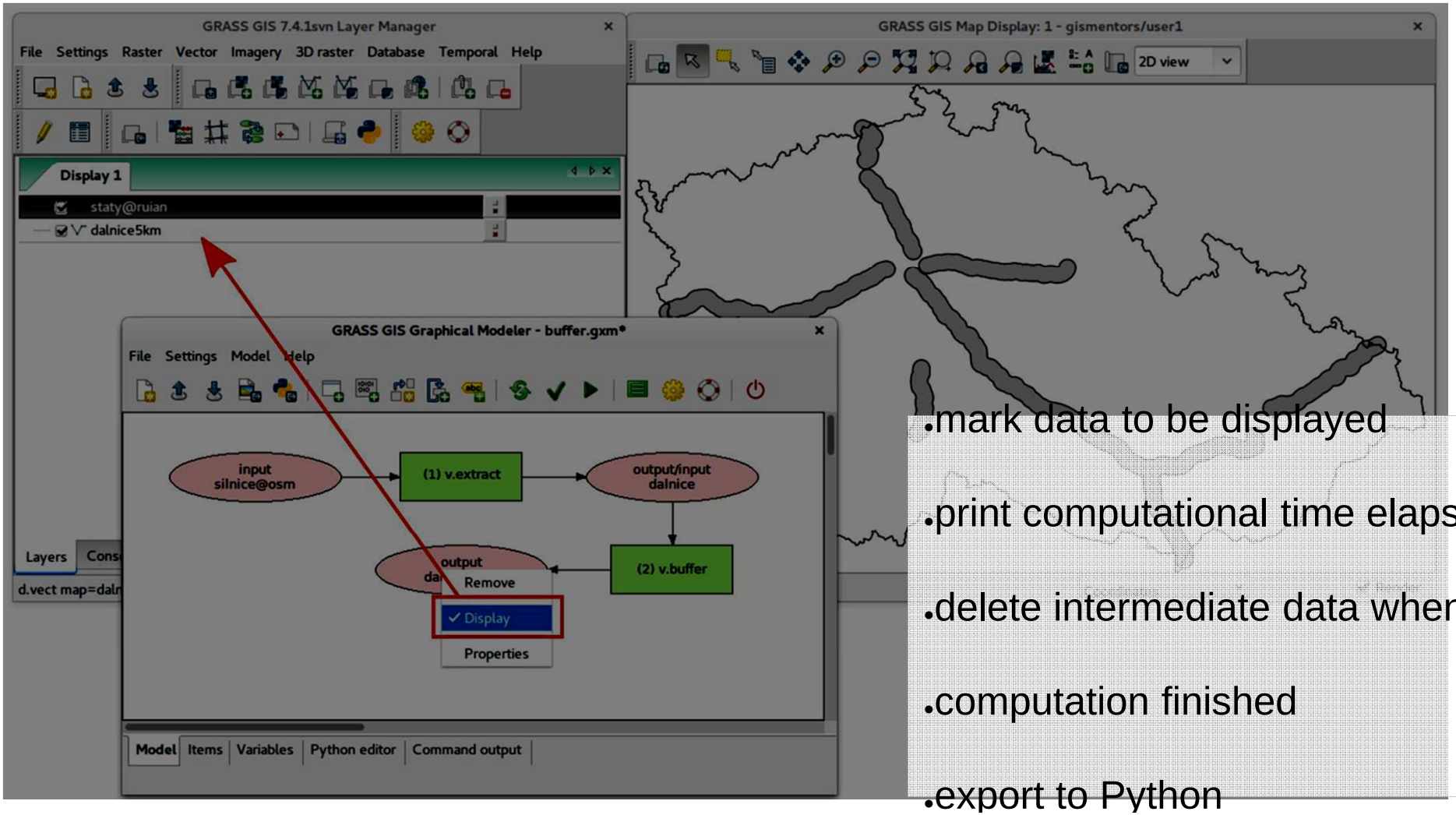


Data tab (catalog): Copying of raster and vector maps betwee

New Orthorectification GUI



Graphical Modeller



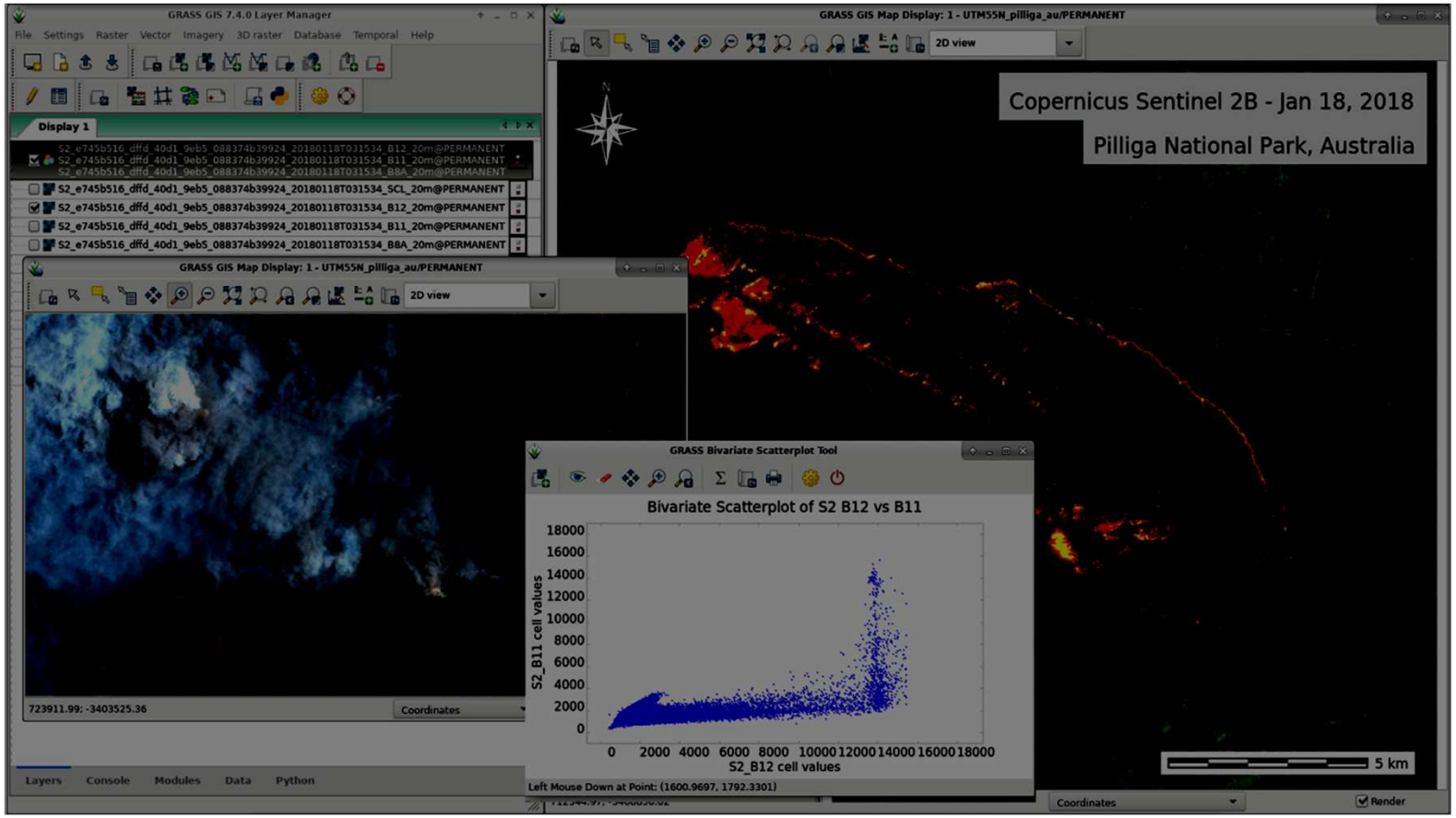
- mark data to be displayed
- print computational time elapsed
- delete intermediate data when
- computation finished
- export to Python

Copernicus Sentinel-2 processing

New addons: *i.sentinel.download*, *i.sentinel.import*, *i.sentinel.preproc* and *i.se*

Example:

Wildfire in Australia

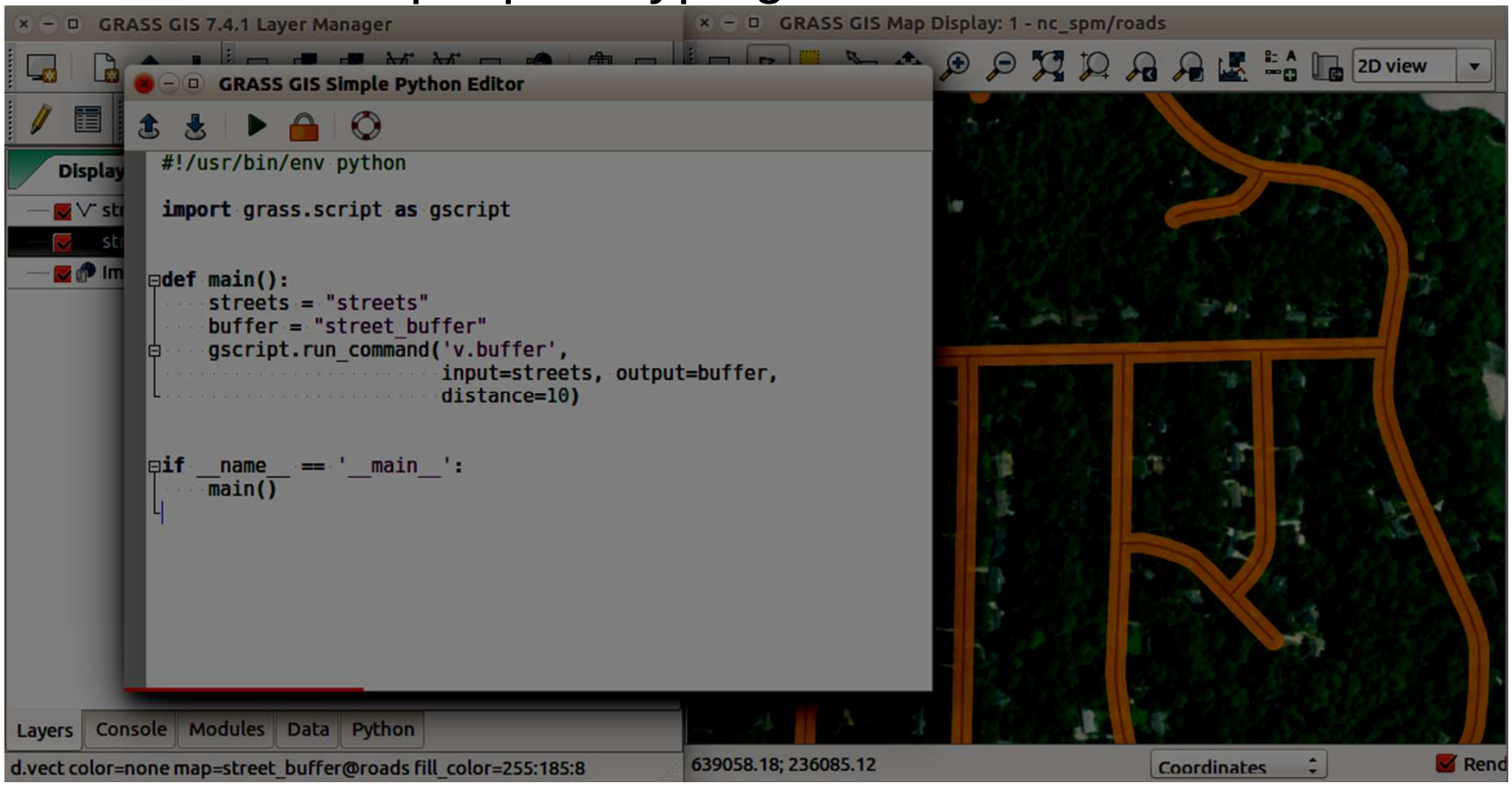


Python Editor

Integrated Python editor for rapid prototyping

Example:

Vector buffer

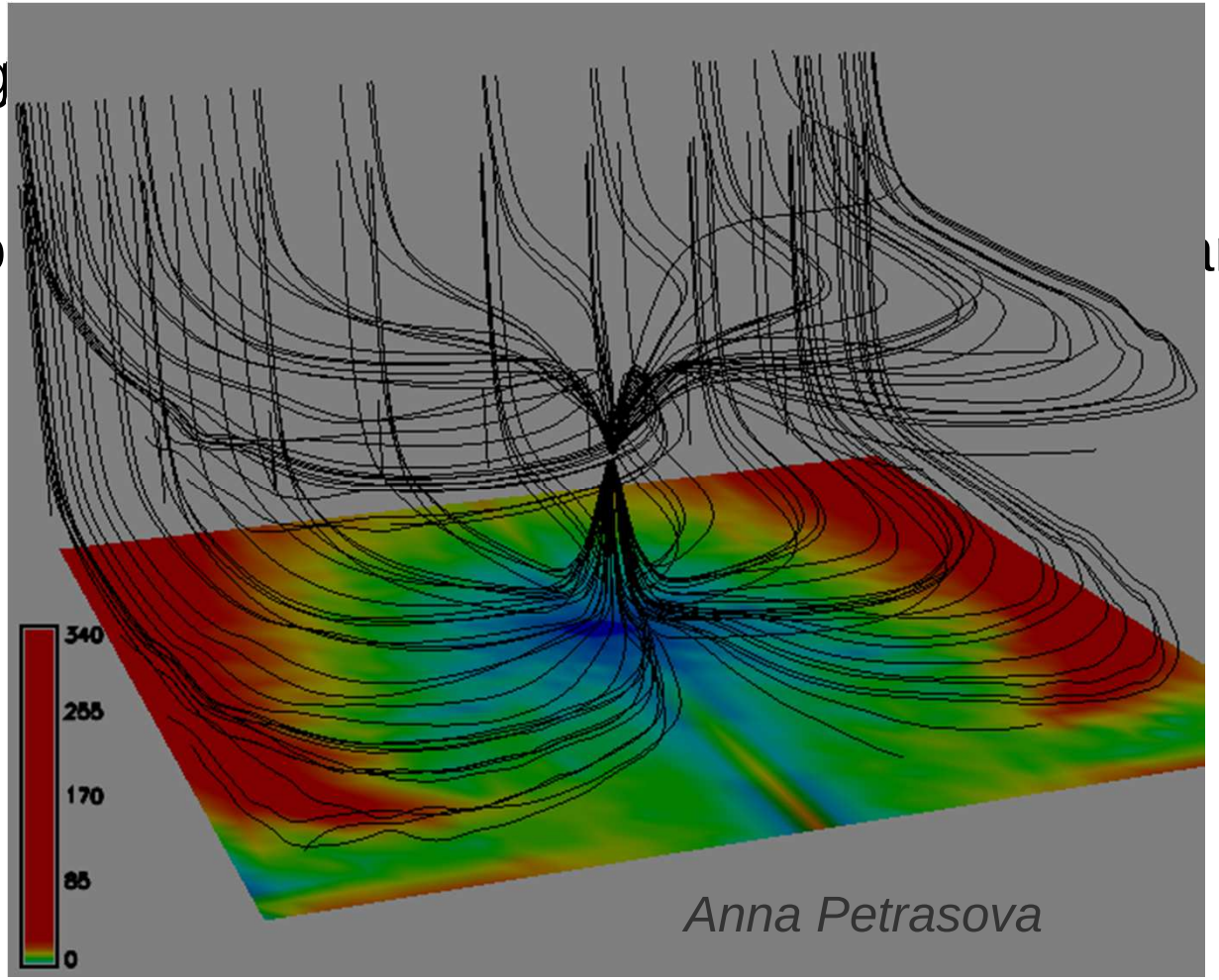


Vaclav Petras

3D raster gradients and flowlines

3D raster (voxel) processing

r3.flow and *r3.gradient* to co ınd re



TGRASS: *t.rast.algebra* and *t.rast3d.algebra*: temporal algebra

Compute annual hydro-thermal coefficients (HTC) from daily climate data

$$HTC = \frac{\sum P_{(T>10°C)}}{\sum T_{(T>10°C)} \cdot \frac{1}{10}}$$

T := daily temperatures,
P := daily precipitation

~ **60 years of daily data**, each pixel in time = virtual meteo station

```
t.rast.algebra "HTC = (D {+,contains,1} if(T >= 10, P, 0)) /
                      (D {+,contains,1} if(T >= 10, T / 10, 0))"
```

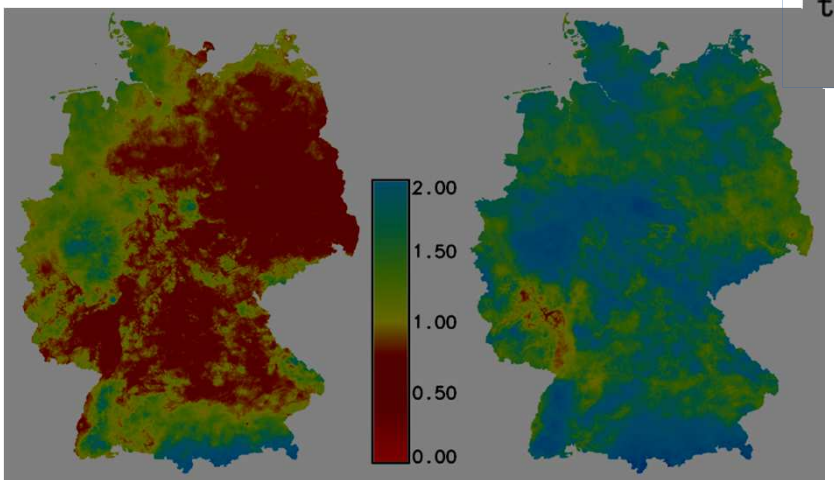


Fig. 6: HTC for 2003 and 2007

Leppelt & Gebbert, EGU 2015

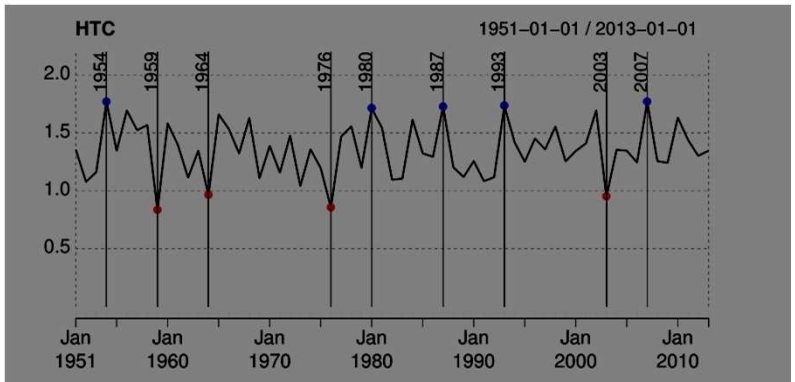


Fig. 7: HTC of extreme events for droughts (HTC < 1) in red and humid years (HTC > 1.7) in blue

GRASS GIS and Python

Using GRASS GIS from “outside” through “grass-session”

```
pip install grass-session
```

Now it’s easy to use GRASS GIS
as a processing backend in Python!

Combine with GDAL, OTB, ...

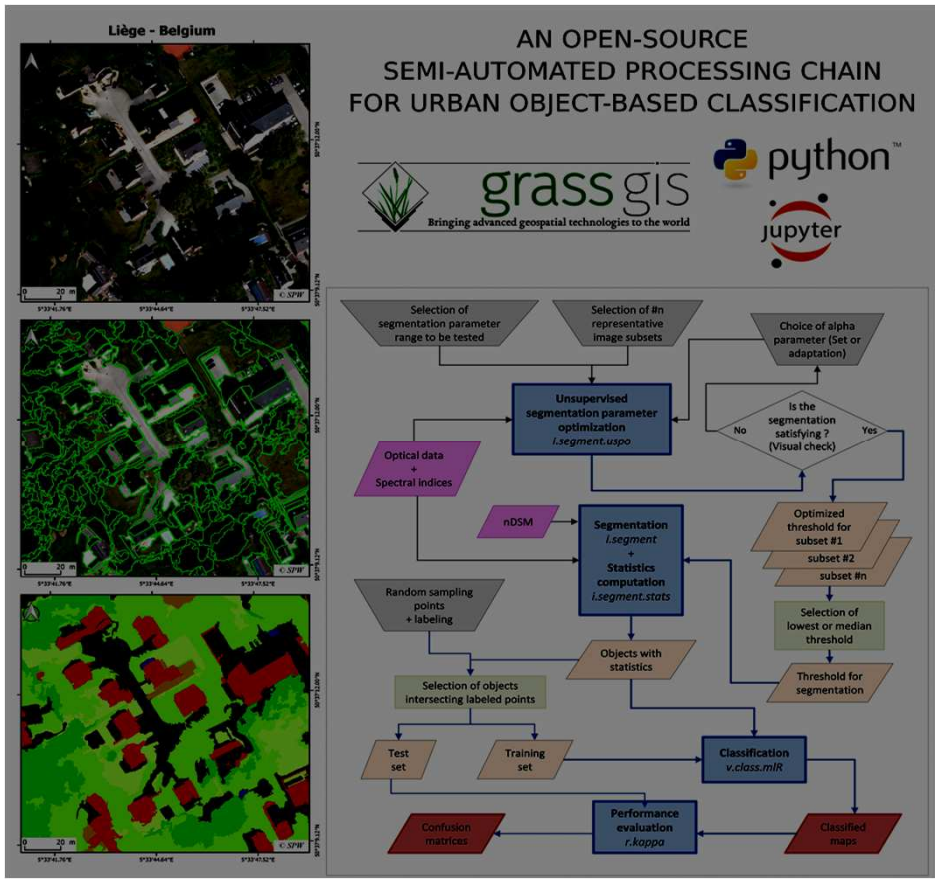
```python
#!/usr/bin/env python
# filename: test_session.py

from grass_session import Session
from grass.script import core as gcore

# create a new location from EPSG code (can also be a GeoTIFF or SHP or ... file)
with Session(gisdb="/tmp", location="location",
             create_opts="EPSG:4326"):
   # do something in permanent
   print(gcore.parse_command("g.gisenv", flags="s"))
# {u'GISDBASE': u"'/tmp/';",
#  u'LOCATION_NAME': u"'epsg3035';",
#  u'MAPSET': u"'PERMANENT';",}

# create a new mapset in an existing location
with Session(gisdb="/tmp", location="location", mapset="test",
             create_opts=""):
    # do something in the test mapset.
    print(gcore.parse_command("g.gisenv", flags="s"))
# {u'GISDBASE': u"'/tmp/';",
#  u'LOCATION_NAME': u"'epsg3035';",
#  u'MAPSET': u"'test';",}
```

Remote sensing in GRASS GIS: object-based image analysis

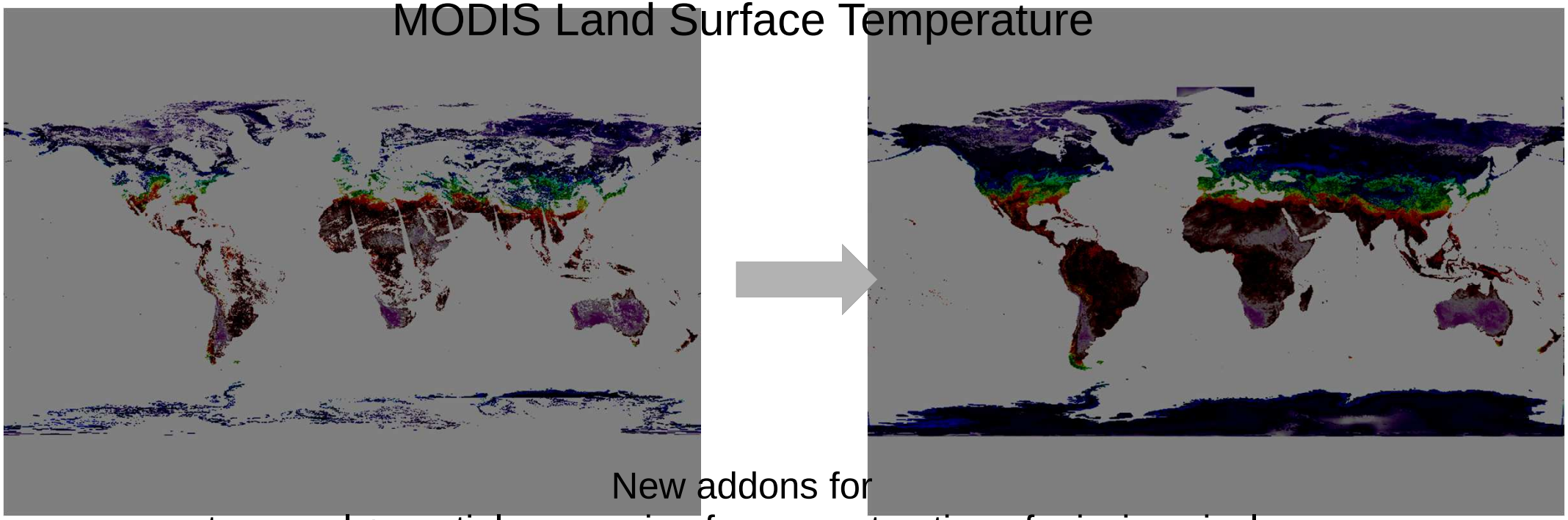


Source : http://dx.doi.org/10.3390/rs9040358

- Complete toolchain from segmentation
- Including
  - unsupervised segmentation parameter
  - high performance object statistics calc
  - module-level parallelization
- Recently created module for SLIC sup

High-performance computing

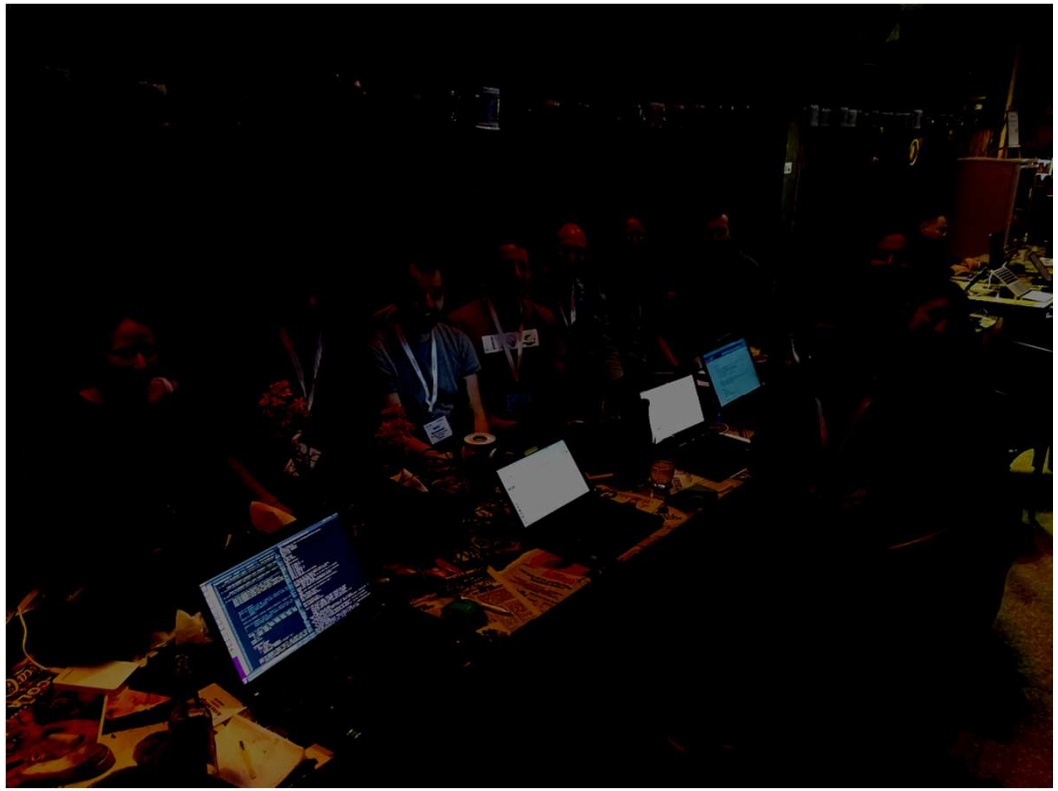
MODIS Land Surface Temperature

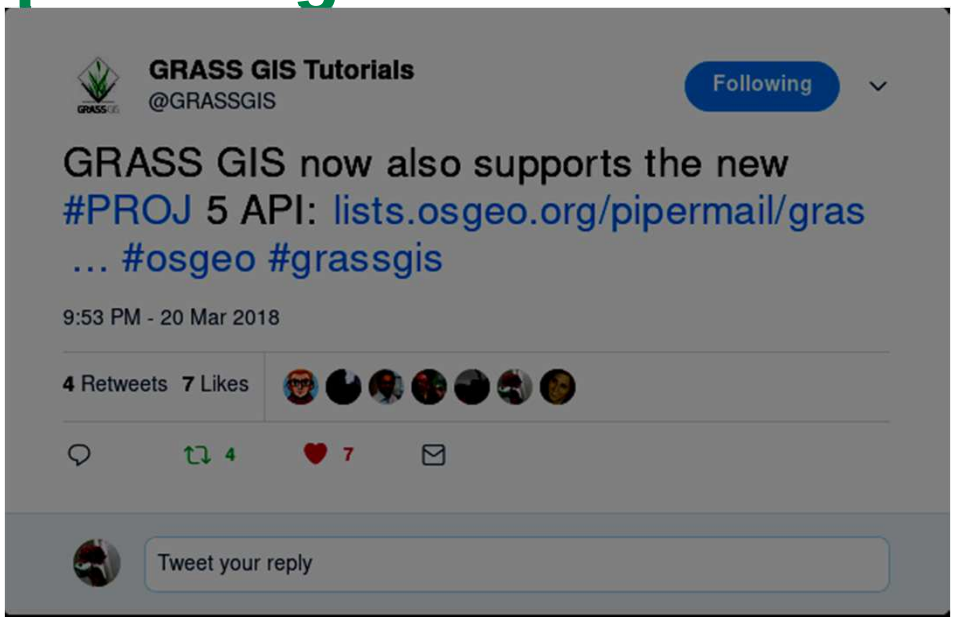


New addons for
temporal + spatial processing for reconstruction of missing pixels

Data: https://zenodo.org/record/1135230

Community activities: Code Sprint 2018 at FOSSGIS Bonn Basecamp – Integration





20 March 2018

Community activities: Google Code-IN for 13-17 year old pre-university students

https://grasswiki.osgeo.org/wiki/GRASS_GCI_Ideas_2017

3.1 **Install** GRASS GIS on your computer and download North Carolina dataset

3.2 **Compile** GRASS GIS

3.3 Add examples and/or screenshots to different **manual** pages

3.4 Add **test suites** to different modules

3.5 **Designs**

- 3.5.1 Splash screen for GRASS GIS GUI start-up
- 3.5.2 T-shirt for 2018 Code Sprint
- 3.5.3 Banner for location wizard

3.6 **Blog** entry about GRASS GIS

3.7 **Videos**

- 3.7.1 How to create a location

Around 120 students

Community activities: GSoC 2018

Google Summer of Code 2018

https://trac.osgeo.org/grass/wiki/GSoC/2018

OSS-Fuzz - Continuous **Fuzzing** for Open Source Software for GRASS GIS
Implement a series of **image fusion** algorithms in GRASS GIS
Enhance 3D **rendering** capabilities in GRASS GIS
Additional functionality for running GRASS GIS modules in **Jupyter** Notebook
Integration of **PDAL** into GRASS GIS
Benchmarking framework for GRASS GIS
GRASS GIS as a post-processing part of **WebODM**
Additional **GUI** tools for image analysis
Module to create quadtree **tiling**
Tools for generating **unit tests** from examples in the manual
Mapnik rendering engine for GRASS GIS
Generalized GUI code for **Qt-based GUI**
GRASS GIS **3D viewer** NVIZ module independent of the main GUI
Integration of v.profile into **GUI** profiling tool
Add **CMake** build system for GRASS GIS
Add a cloud masking module for **Sentinel** data in GRASS GIS
Full support of **Python 3** in GRASS GIS
Improve GRASS integration in **QGIS 3**
New easy-to-use CLI and **API** for GRASS GIS

Thanks for your attention!

grass.osgeo.org