

GRASS GIS – Geodatenanalyse und mehr

Markus Neteler &
GRASS Development Team

grass.osgeo.org
www.mundialis.de

INTER**GEO** 2018 – Frankfurt am Main
OSGeo-Park
Halle 12.1, Stand 101

FRANKFURT 2018
16. – 18. OKTOBER

INTERGEO®
WISSEN UND HANDELN
FÜR DIE ERDE



Was ist GRASS GIS?

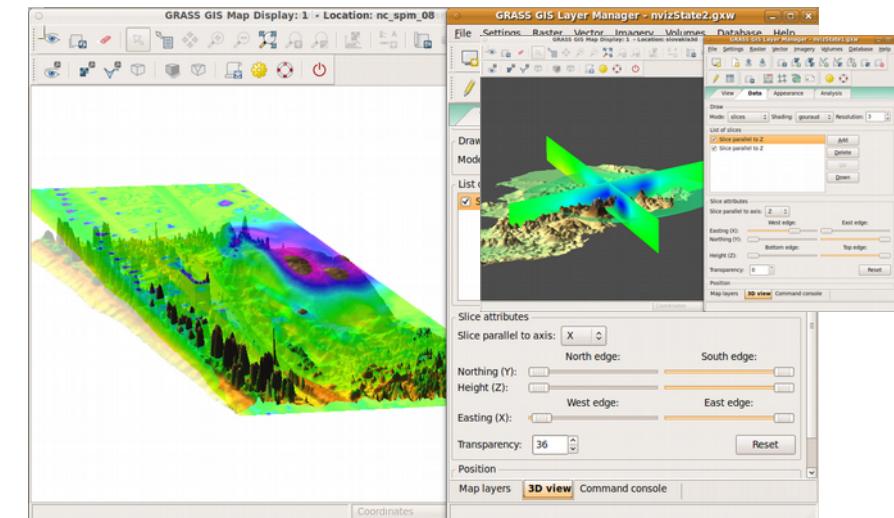
- GRASS GIS ist eine hybride, modulare GIS-Software
- GRASS = Geographic Resources Analysis Support System
- GNU General Public License – frei verfügbar

- Raster- und topologische Vektordatenfunktionalität
- 3D-Raster-Voxelbearbeitung
- Bildverarbeitung
- Visualisierungsmöglichkeiten

- Portable Software (“alle” Betriebssysteme)
- graphischen Benutzeroberfläche
- sowie Kommandozeile
- Neu: Cloud integration



Nagshead LiDAR time series: dune moving over 9 years (NC, USA) – animation

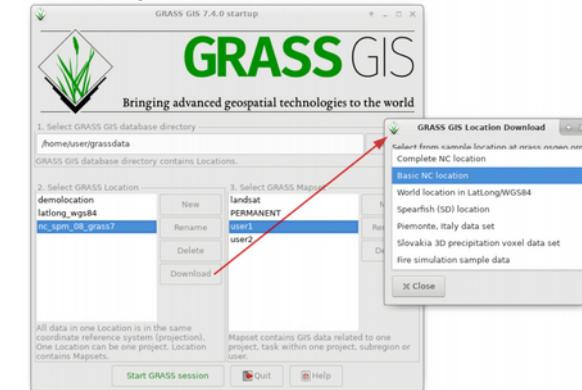




Was ist neu in GRASS GIS?

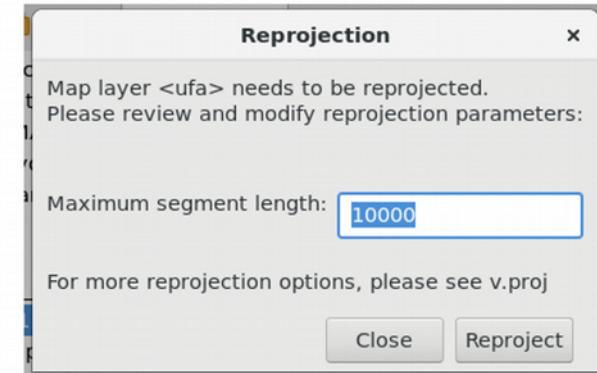
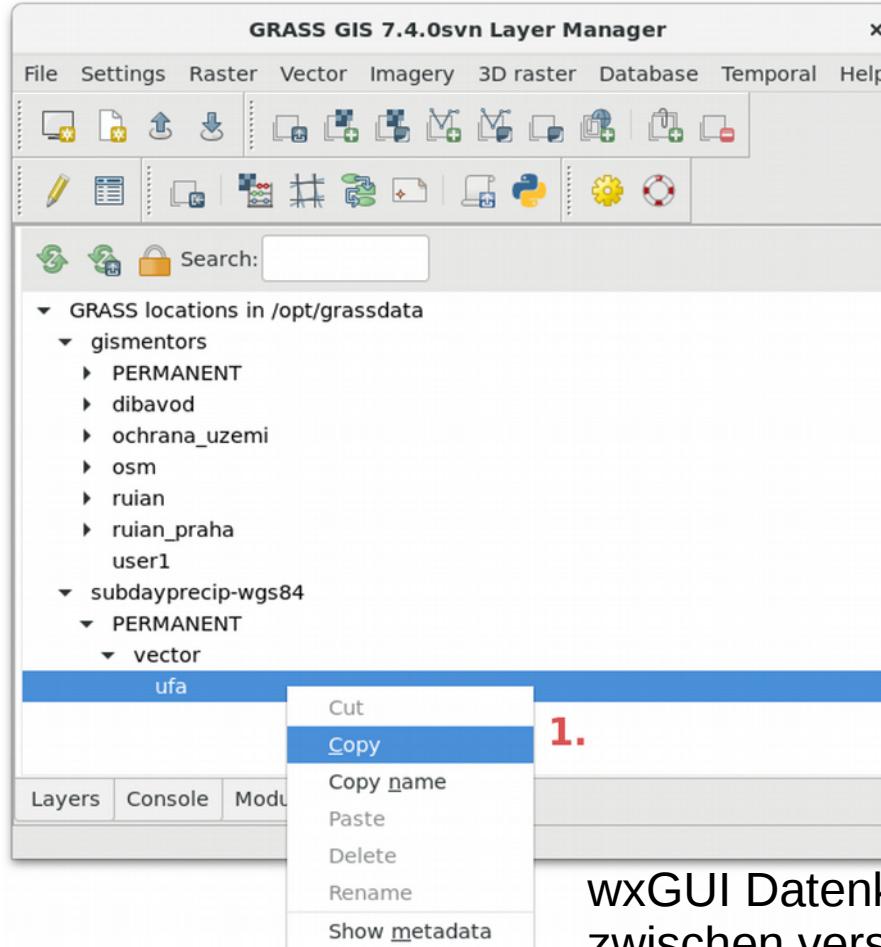
Neue stabile Version GRASS GIS 7.4 (bald neu: 7.6)

- Benutzerfreundlichkeit und grafische Benutzeroberfläche verbessert
- Datenkompression
 - neue “no data” Kompression
 - Zstandard Kompression als weitere Methode
- Unterstützung für globale Daten, die über -180/+180, -90/+90 Grad hinausreichen
- Ortho-Rektifikation mit Benutzeroberfläche neu implementiert
- Neuer Download-Link für Beispieldaten
- ... über 480 Verbesserungen seit G7.2.0





Datenkatalog: Verbesserungen



wxGUI Datenkatalog: Kopieren von Raster- und Vektorkarten zwischen verschiedenen Projekten inklusive Reprojektion



Neue Orthorectification GUI

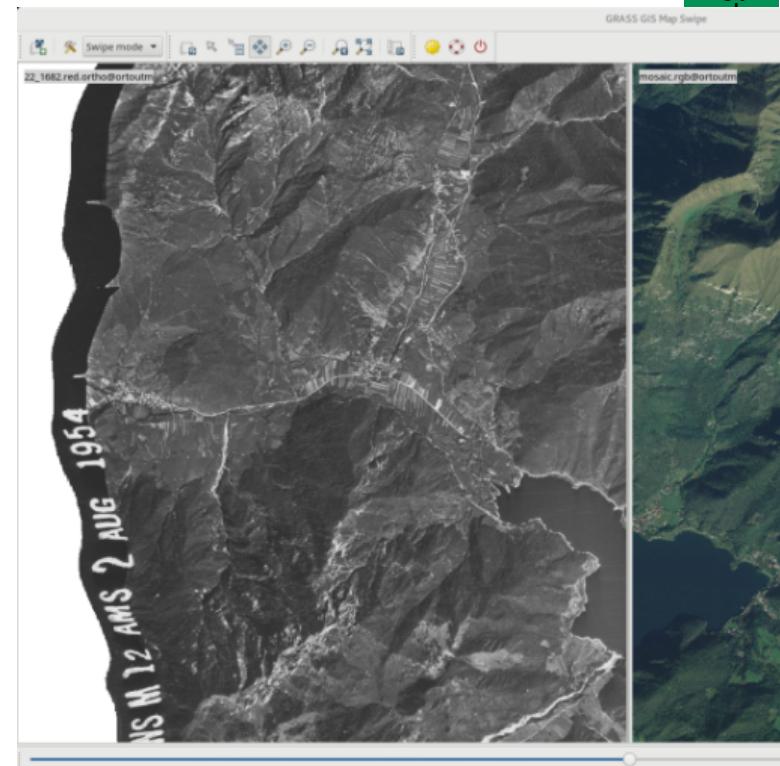
Manage Ground Control Points

GCP List

use	source E	source N	source Z	target E	target N	target Z	Forward error	Backward error
<input checked="" type="checkbox"/>	3433.76399027	4013.9244039	0.0	635890.53903	5082323.73716	700	103893.989338	1206.797055
<input checked="" type="checkbox"/>	5663.63017032	3315.20681265	0.0	630698.420894	5083666.03223	750	756671.25929	2593.658967
<input checked="" type="checkbox"/>	3484.43309002	4965.99756691	0.0	635880.958794	5080131.73067	750	118984.847243	2147.705000
<input checked="" type="checkbox"/>	3519.34793187	4907.66423358	0.0	635757.957044	5080294.5271	720.3481	115606.753734	2096.784943
<input checked="" type="checkbox"/>	1980.11435523	4745.01216545	0.0	639373.871778	5080707.57861	500	803733.994864	2206.519653
<input checked="" type="checkbox"/>	4006.45255474	2026.76399027	0.0	634353.593597	5086780.06445	850	189157.523637	1216.300242
<input checked="" type="checkbox"/>	2666.15043706	4702.67396604	0.0	637350.651294	5081740.00072	700	243470.526813	1463.770170

Source Display

Target Display



SA



Graphical Modeller

The screenshot illustrates the GRASS GIS environment. The top left shows the 'GRASS GIS 7.4.1svn Layer Manager' window with various tools and a list of layers: 'Display 1' containing 'staty@ruian' and 'dalnice5km'. The top right shows the 'GRASS GIS Map Display' window showing a map with several grey polygonal areas. The bottom left shows the 'GRASS GIS Graphical Modeler - buffer.gxm*' window, which contains a flowchart of processes:

```
graph LR; A([input silnice@osm]) --> B((1) v.extract); B --> C([output/input dalnice]); C --> D((2) v.buffer); D --> E([output data Remove]); E --> F([Display]);
```

A red arrow points from the 'Display' button in the Modeler window to the 'Display 1' tab in the Layer Manager window. Another red box highlights the 'Display' button in the Modeler window.

- anzugeigende Daten markieren
- verstrichene Rechenzeit ausgeben
- Zwischendaten löschen, wenn Berechnung abgeschlossen
- exportieren nach Python

© 2018, GRASS GIS Development Team, CC-BY-SA

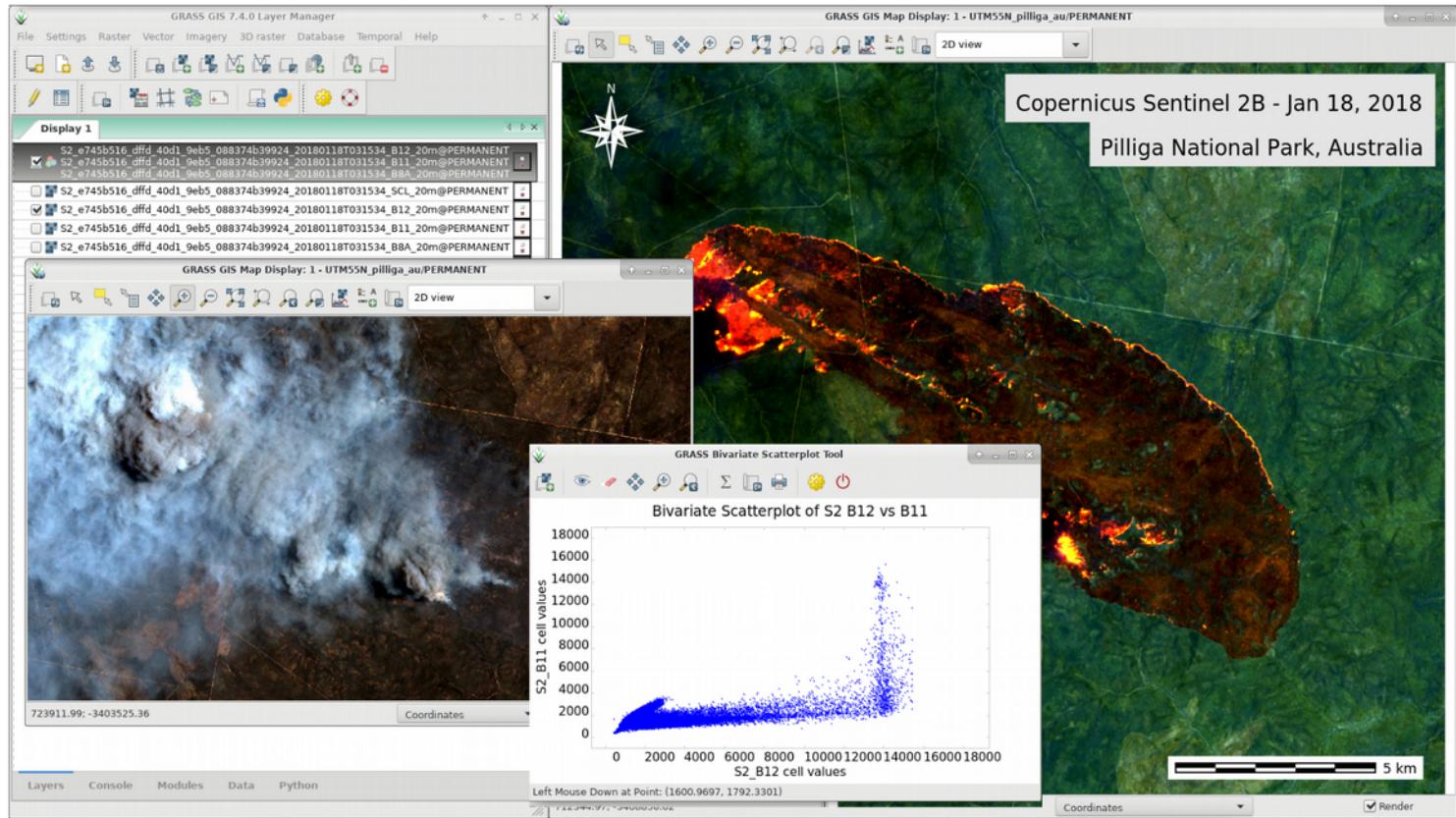


Copernicus Sentinel-2 Prozessierung

Neue Erweiterungen:

i.sentinel.download and i.sentinel.import

Beispiel:
Waldbbrand
in Australien

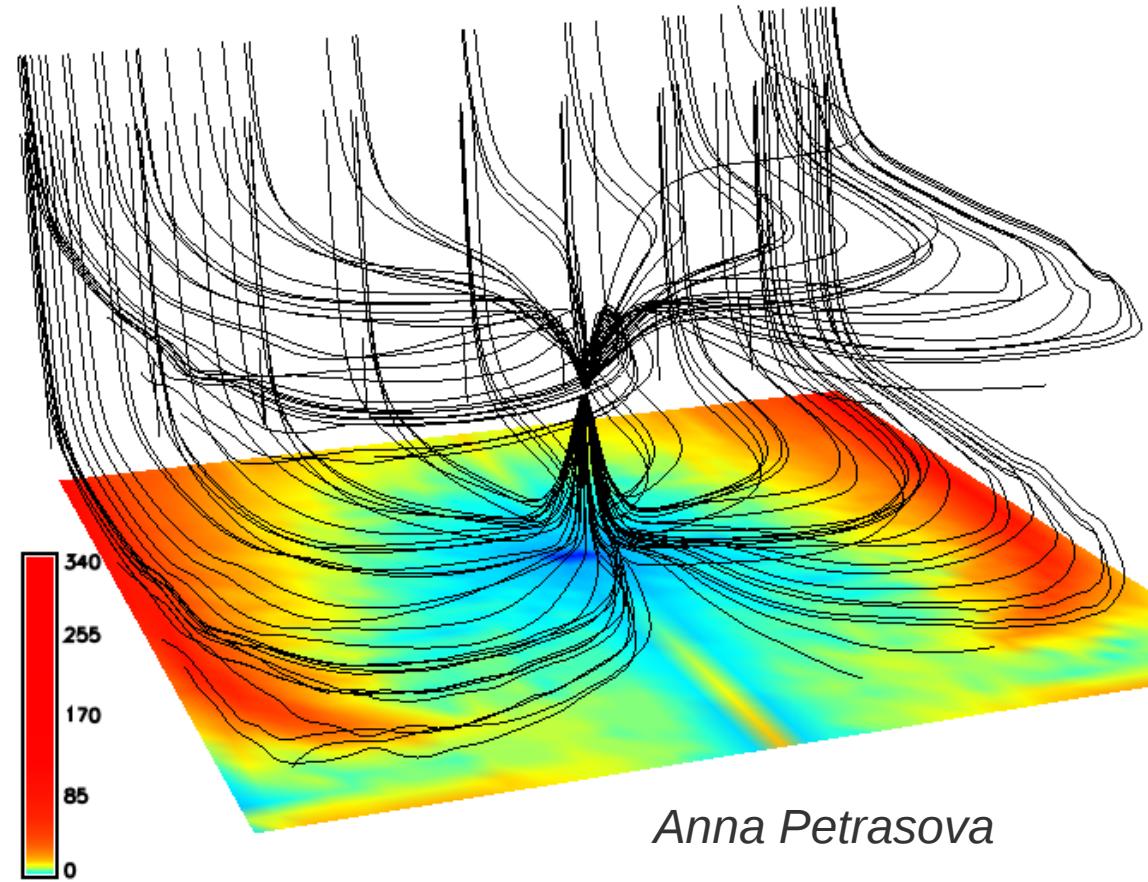




3D Rasterfließlinien

Voxel-Prozessierung:

r3.flow und r3.gradient zur Berechnung von 3D Fließlinien, 3D Fließakkumulation und zugehörige Gradienten





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

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

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

$T :=$ daily temperatures,
 $P :=$ daily precipitation

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

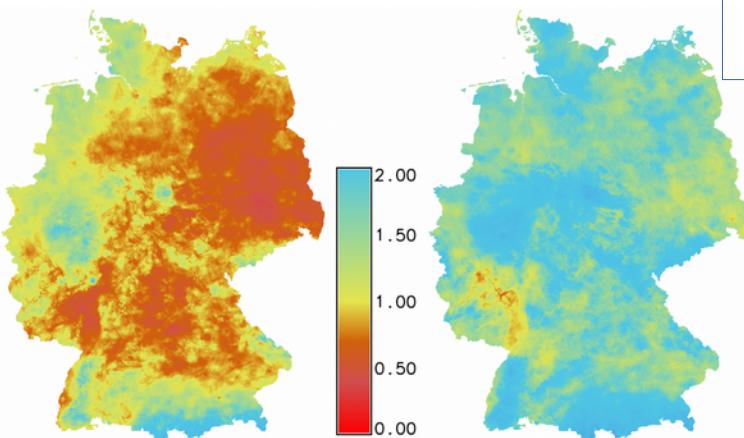


Fig. 6: HTC for 2003 and 2007
Leppelt & Gebbert, EGU
2015

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

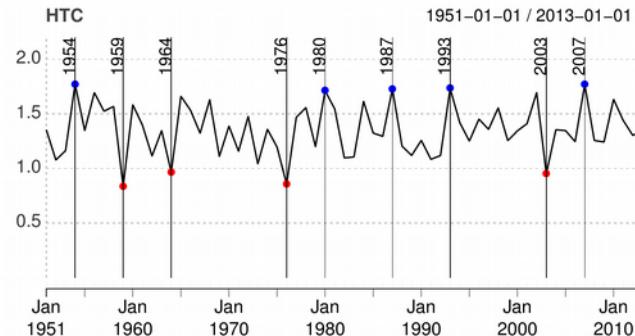


Fig. 7: HTC of extreme events for droughts (HTC < 1) in red and humid years (HTC > 1.7) in blue



Python Editor

Integrated
Python editor
for rapid
prototyping

Example:
Vector buffer

The screenshot shows the GRASS GIS 7.1 Python Editor interface. On the left, the Layer Manager window displays a map of a street network. In the center, the Simple Python Editor window contains the following code:

```
#!/usr/bin/env python

import grass.script as gscript

def main():
    streets = "streets"
    buffer = "streets_buffer"
    gscript.run_command('v.buffer', input=streets, output=buffer,
                        distance=10)

if __name__ == '__main__':
    main()
```

The bottom status bar shows coordinates: 639045.50; 229853.11.

Vaclav Petras



GRASS GIS and Python

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

```
pip install grass-session
```

Finally an easy use of GRASS GIS
as a processing backend in Python!

Combine now with GDAL, OTB, ...

```
#!/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';,}

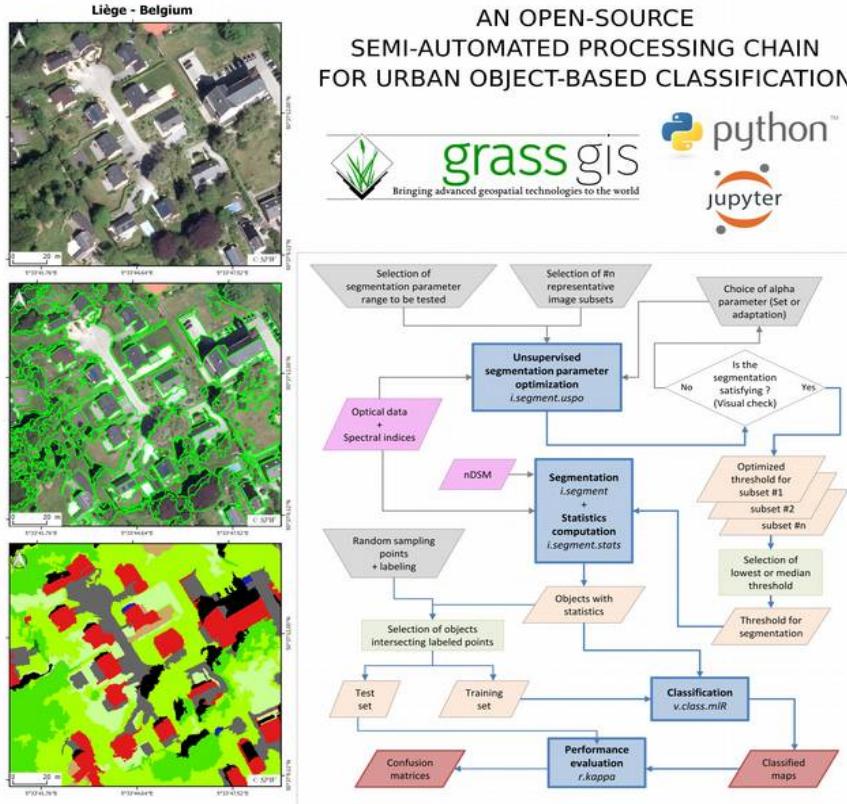
# create a new mapset in an existing location
with Session(gisdb="/tmp", location="location", mapset="test",
             create_opts=""):
    # do something in the test mapset.
    print(gcore.parse_command("g.gisenv", flags="s"))
# {u'GISDBASE': u"/tmp/",  

#  u'LOCATION_NAME': u'epsg3035',  

#  u'MAPSET': u'test';,}
```



Remote sensing in GRASS GIS: object-based image analysis



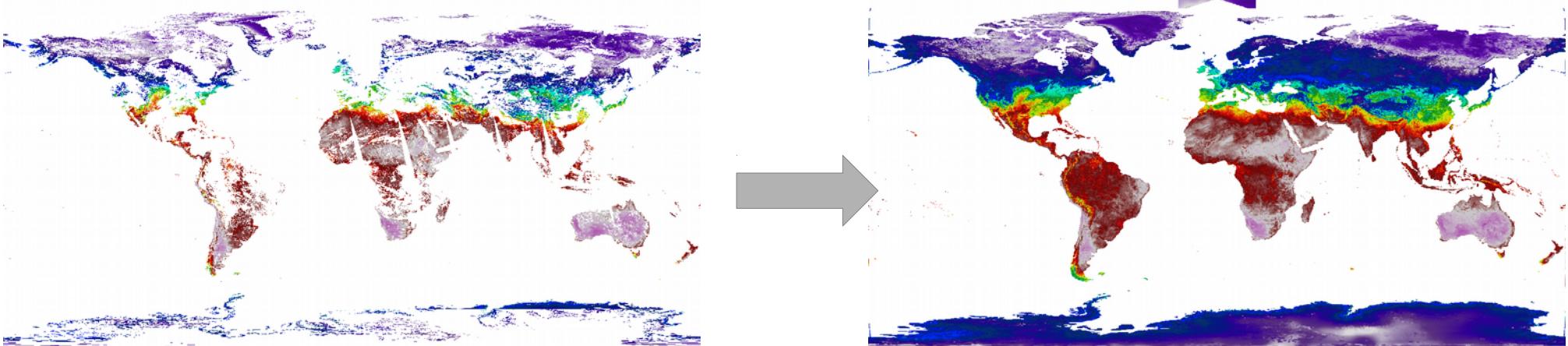
- Complete toolchain from segmentation to classification
- Including
 - unsupervised segmentation parameter optimization
 - high performance object statistics calculation
 - module-level parallelization
- Recently created module for SLIC superpixel creation

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



High-performance computing

MODIS Land Surface Temperature



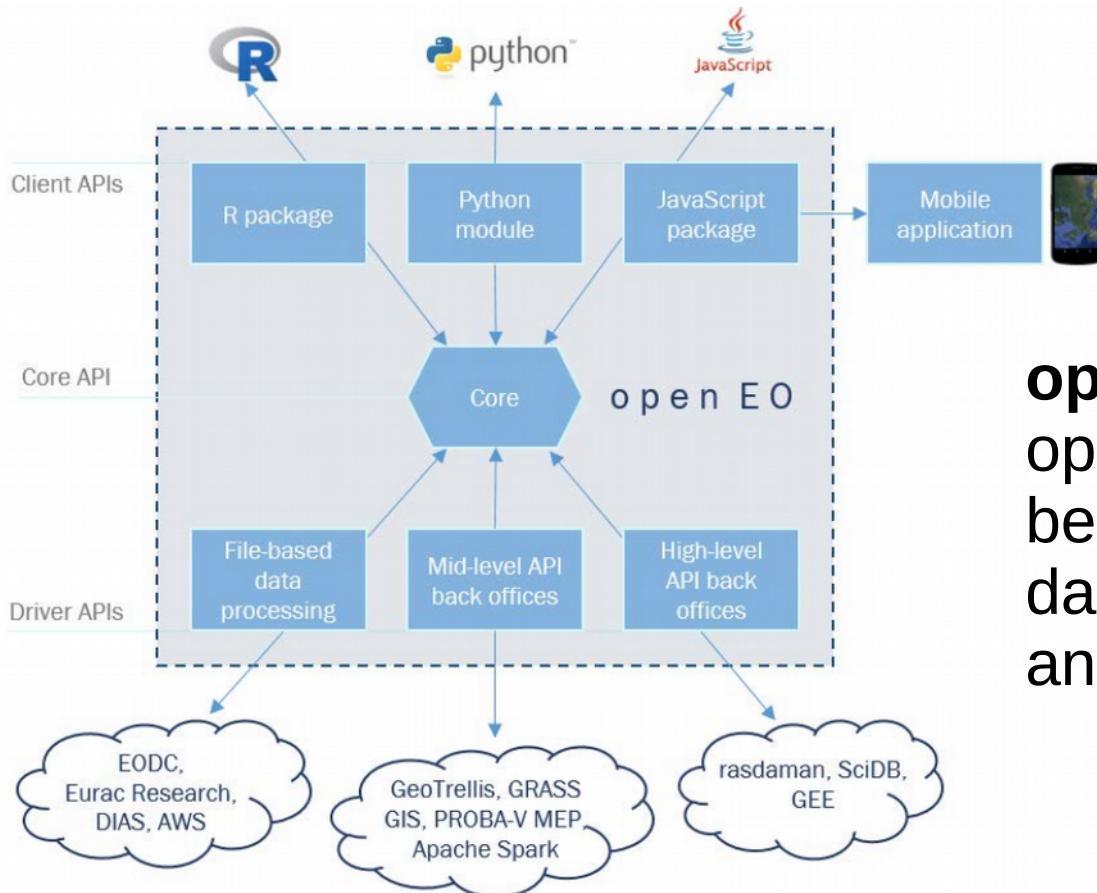
New addons for
temporal + spatial processing for
reconstruction of missing pixels

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



The openEO H2020 EU Project

2017-2020 – <http://www.openeo.org>



openEO - a common, open source interface between Earth Observation data infrastructures and front-end applications

Community activities: Code Sprint 2018 at Basecamp Bonn



20 March 2018

GRASS GIS Tutorials
@GRASSGIS

Following ▾

GRASS GIS now also supports the new
#PROJ 5 API: lists.osgeo.org/pipermail/grass/
... #osgeo #grassgis

9:53 PM - 20 Mar 2018

4 Retweets 7 Likes

• 4 7 •

Integration with QGIS 3



Vielen Dank!



GRASS GIS

Ansprechpartner:

Dr. Markus Neteler
neteler@mundialis.de
neteler@osgeo.org

grass.osgeo.org