

# Using Free and Open Source Solutions in Geospatial Science Education

How to make education better for science and future

Vaclav Petras (Vashek)

Anna Petrasova, Keren Cepero Perez, Markus Neteler,  
Luca Delucchi, Martin Landa, Helena Mitasova

North Carolina State University  
Fondazione Edmund Mach  
Czech Technical University in Prague and GISMentors

July 15, 2015  
FOSS4G Europe



# Free and open source software

*Open Source Software Is Now a Norm in Businesses*

Katherine Noyes, PCWorld, May 18, 2011

*Open Source has Become Mainstream but Still Drives Innovation*

Talend Yves de Montcheuil, ZDNet, May 2, 2012

*10 of Europe's 15 largest banks are now running [...] Postgres*

Sandor Klein said for ZDNet (Toby Wolpe), November 19, 2013

*Redmond top man Satya Nadella: 'Microsoft loves Linux'*

Neil McAllister, The Register, October 20, 2014

*Survey indicates four out of five developers now use open source*

Steven J. Vaughan-Nichols, ZDNet, October 29, 2014

*64% of internet exchange points are now using [...] an open source solution*

Gijs Hillenius, Joinup Open source observatory, June 8, 2015

*Open Sourcing Is No Longer Optional, Not Even for Apple*

Klnt Finley, WIRED, June 9, 2015

# Free and open source software

*Software [...] developed as part of novel methods is as important for the method's implementation [...] Such software [...] must be made available to readers upon publication.*

Nature Methods 4, 189, 2007

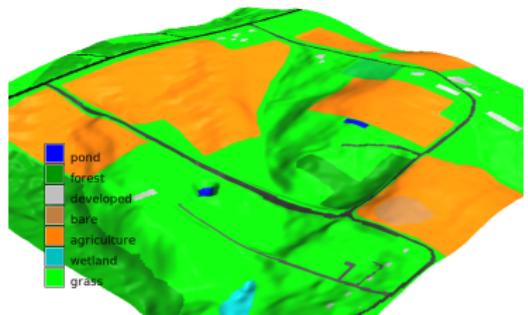
*The opposite of 'open' isn't closed. The opposite of open is 'broken.'*

Cable Green (quoting John Wilbanks) at Open Scotland Summit 2013

# Courses at North Carolina State University

## Geospatial Analysis and Modeling

- ▶ started in 2008
- ▶ on-campus and distance education
- ▶ every semester 30-60 students
- ▶ software:
  - ▶ GRASS GIS
  - ▶ ArcGIS

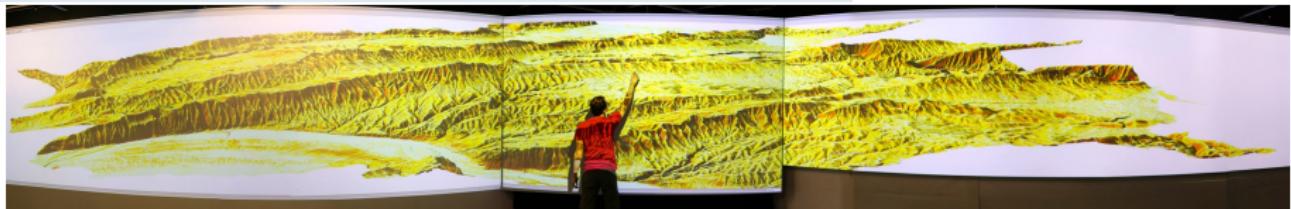
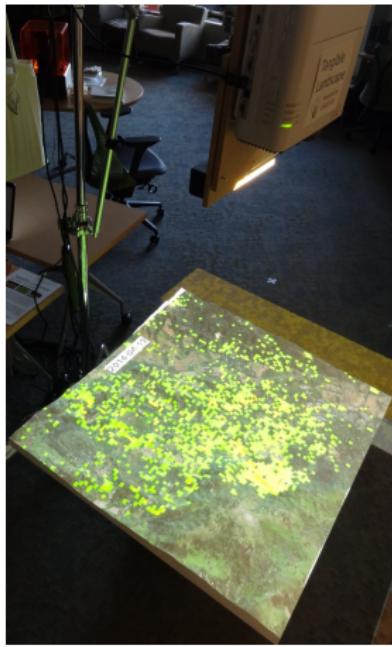


Listing only geospatial courses where presentation authors are involved.

# Courses at North Carolina State University

## Multidimensional Geospatial Modeling

- ▶ software:
  - ▶ GRASS GIS often with new features such as Temporal Framework (GRASS GIS 7)
  - ▶ + whatever the students need,  
e.g. XBeach, libLAS or LAStools
- ▶ curriculum depends on students projects
- ▶ new technologies:
  - ▶ eye tracking
  - ▶ Tangible Landscape
  - ▶ NCSU Hunt Lib Teaching and Vis Lab



## GIS for Designers

- ▶ software in class:
  - ▶ ArcGIS
  - ▶ GRASS GIS
  - ▶ Rhino (Rhinoceros)
- ▶ for projects architects and designers combine a lot of tools
  - ▶ Tangible Landscape (powered by GRASS GIS) was one of them



## UAV/lidar Data Analytics

- ▶ under development for this fall semester
- ▶ Agisoft PhotoScan in class, OpenDroneMap in projects

Related talk: Flow analysis using sUAS and lidar data (Helena Mitasova)



# NCSU OSGeoREL workshops and tutorials

## Introduction to GRASS GIS

Delivered at NCSU

## Spatio-temporal data handling and visualization in GRASS GIS

FOSS4G 2014 (Portland) workshop, also delivered at NCSU

## Soil erosion and deposition modeling

Part of a broader project

## How to write a Python GRASS GIS 7 addon

FOSS4G Europe 2015 (Como) workshop, also delivered at NCSU

Workshops are a way how to experiment with what to teach and how.

# The idea

- ▶ lectures:
  - ▶ theory, concepts
  - ▶ software-independent
- ▶ labs and assignments:
  - ▶ relate to given lecture
  - ▶ hands-on, practical
  - ▶ students use software

# The problem

- ▶ students are becoming software users instead of scientists
- ▶ students mix software details and general concepts
  - ▶ Shapefile or Feature class instead of Vector data
  - ▶ Generic (genericized) trademarks
- ▶ software promotes software/vendor-specific formats/technologies
  - ▶ teachers end up using them
  - ▶ and students end up knowing just them
- ▶ single software choice limits explored algorithms
  - ▶ *[Part of the assignment was already in] Surface Water Hydrology [class]. In [that class] we used the D8 algorithm exclusively and focused on [...] ArcHydro. In this assignment it was interesting to see the use the Dinf algorithm which considers more direction choices [...]*

# The solution

- ▶ lectures:
  - ▶ theory, concepts
  - ▶ software-independent
- ▶ labs and assignments:
  - ▶ relate to given lecture
  - ▶ hands-on, practical
  - ▶ **students use two different software packages**, in our case:
    - ▶ GRASS GIS (free and open source)
    - ▶ ArcGIS (proprietary)
- ▶ similar task in both
- ▶ students have always at least two different examples to see what is a general concept and what is specific to a particular software
- ▶ they gain flexibility to choose optimal solutions for their future work

# The solution

- ▶ lectures:
  - ▶ theory, concepts
  - ▶ software-independent
- ▶ labs and assignments:
  - ▶ relate to given lecture
  - ▶ hands-on, practical
  - ▶ students use two different software packages, in our case:
    - ▶ GRASS GIS (free and open source)
    - ▶ ArcGIS (proprietary)
- ▶ similar task in both
- ▶ students have always at least two different examples to see what is a general concept and what is specific to a particular software
- ▶ they gain flexibility to choose optimal solutions for their future work

# The solution

- ▶ lectures:
  - ▶ theory, concepts
  - ▶ software-independent
- ▶ labs and assignments:
  - ▶ relate to given lecture
  - ▶ hands-on, practical
  - ▶ students use two different software packages, in our case:
    - ▶ GRASS GIS (free and open source)
    - ▶ ArcGIS (proprietary)
- ▶ similar task in both
- ▶ students have always at least two different examples to see what is a general concept and what is specific to a particular software
- ▶ they gain flexibility to choose optimal solutions for their future work

# Can I use two open source packages?

Yes, just select different-enough packages.

No:

- ▶ GRASS GIS + QGIS with GRASS plugin

Yes:

- ▶ GRASS GIS + QGIS with SAGA algorithms in Processing

Maybe:

- ▶ QGIS with GRASS plugin + R with spgrass6/rgrass7

# Can I use two proprietary packages?

No, at least one of the used packages should be open source.

- ▶ teaching for future industry, not past
- ▶ teaching for science

# Teaching materials

- ▶ file format
  - ▶ originally HTML
  - ▶ selecting new one
    - ▶ Markdown, missing general standard
    - ▶ reStructuredText, hot candidate
  - ▶ result: HTML (same as delivery format)
  - ▶ presentation slides in HTML5 (Reveal.js)
- ▶ license: CC BY-SA
- ▶ Git (GitHub hosted) for revision control, collaboration and sharing source code



[geospatial.ncsu.edu/  
osgeorel/courses.html](http://geospatial.ncsu.edu/osgeorel/courses.html)

The screenshot shows the homepage of the NCSU GIS/MEA582 course. At the top, there's a banner with a colorful elevation map background and the text "NCSU GIS/MEA582: Geospatial Modeling and Analysis". Below the banner are navigation links: Syllabus, Schedule, Course logistics, Lectures, Assignments, and Projects. A section titled "Geospatial data models" follows, with a "Resources" list containing links to "GRASS GIS overview and manual" and "Recommendations and tutorial how to use wxGUI from the first assignment". A note for Windows users about displaying legend numbers is present. At the bottom, there's a list of steps for setting up the software environment.

- GRASS GIS overview and manual
- Recommendations and tutorial how to use wxGUI from the first assignment

For Windows users: When showing legend, make sure the numbers are displayed. If not, please go through the following steps:

1. In Layer Manager toolbar find Settings -> Map Display tab
2. Set font -> select font (e.g. arial)
3. Click on Save to save settings
4. Click on Render map (second button in Map Display toolbar) and legend numbers should appear.

## Resampling to higher resolution

Resample the given raster map to higher and lower resolution (30m->10m, 30m->100m) and compare resampling by nearest neighbor with bilinear and bicubic method.

First, set the region to 30m resolution and display the 30m resolution elevation raster.

```
g.region swwake_30m -p  
d.rast elev_ned_30m
```

# GRASS GIS advantage for teaching materials maintenance

- ▶ Screenshots are hard to update while text is easy to update.
- ▶ GRASS GIS workflow recorded as commands.
- ▶ GUI dialog filled according to the command.
- ▶ Commands can be automatically extracted and tested.

The elevation map "elev\_ned10m\_nn" looks the same as the original one, so no changes were made.

The screenshot shows the GRASS GIS command line interface with the command `r.slope.aspect elevation=elev_ned10m_nn aspect=aspect_ned10m_nn`. The parameters `elevation` and `aspect` are circled in green. The input raster name `elev_ned10m_nn` is circled in red. The dialog box for `r.slope.aspect` is shown below, with its description and parameter settings. The input field also has `elev_ned10m_nn` circled in red.

`r.slope.aspect [raster, terrain, aspect, slope, curvature] - + ×`

Generates raster maps of slope, aspect, curvatures and partial derivatives from an elevation raster map. Aspect is calculated counterclockwise from east.

Required Outputs Settings Optional Command output Manual

Name of input elevation raster map: \*  
`elev_ned10m_nn`

(elevation=name)

For ArcGIS we also use just text, but, unlike in GRASS GIS, the names in dialogs are not part of the API, so they change more often. (Course running since 2008.)

## *Integrating Free and Open Source Solutions into Geospatial Science Education*

Open Access

Vaclav Petras<sup>1, 4</sup>, Anna Petrasova<sup>1, 4</sup>, Brendan Harmon<sup>2, 4</sup>,  
Ross K. Meentemeyer<sup>3, 4</sup>, and Helena Mitasova<sup>1, 4</sup>

<sup>1</sup>Department of Marine, Earth, and Atmospheric Sciences

<sup>2</sup>Department of Landscape Architecture

<sup>3</sup>Department of Forestry and Environmental Resources

<sup>4</sup>Center for Geospatial Analytics and NCSU OSGeoREL – part of ICA-OSGeo-ISPRS Network (aka **Geo for All**)

North Carolina State University, Raleigh, USA

In: *ISPRS International Journal of Geo-Information*. 2015.



doi:10.3390/ijgi4020942

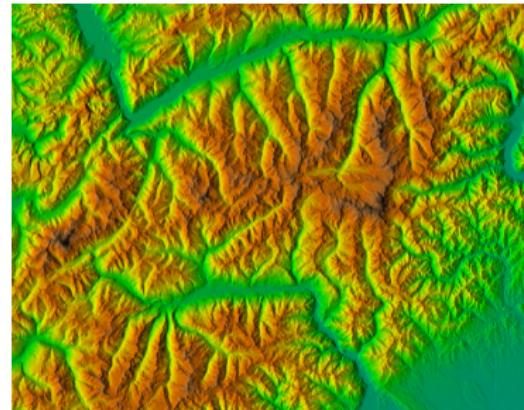


# Standardized Sample Dataset

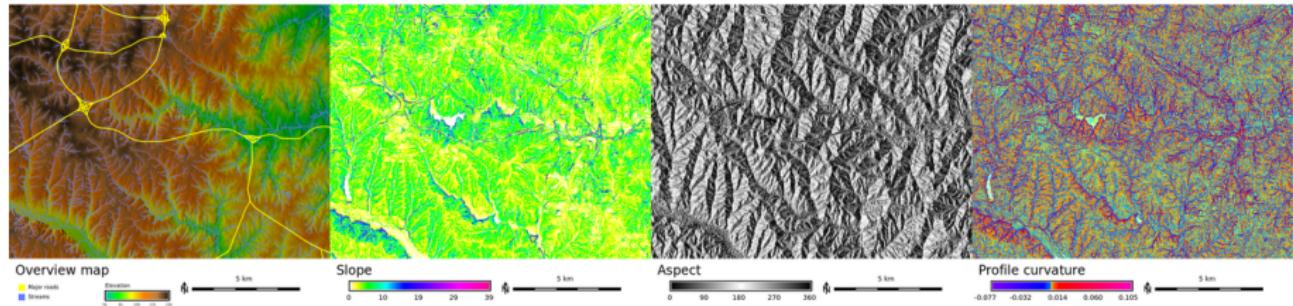
- ▶ other regions local datasets
- ▶ limits sharing of hands-on teaching material
- ▶ new version of North Carolina
  - ▶ commonly available data, frequently used in examples
  - ▶ standardized names such as *elevation*, *streets*, or *lakes*
    - ▶ rather than *srtm*, *dem\_10m*, *streets\_como*
- ▶ challenges:
  - ▶ attributes, coordinates, values, extents, resolutions

```
g.region raster=elevation  
r.relief input=elevation output=shade  
  
d.shade shade=shade color=elevation
```

▶ wiki page



# Standardized Sample Dataset: North Carolina, USA

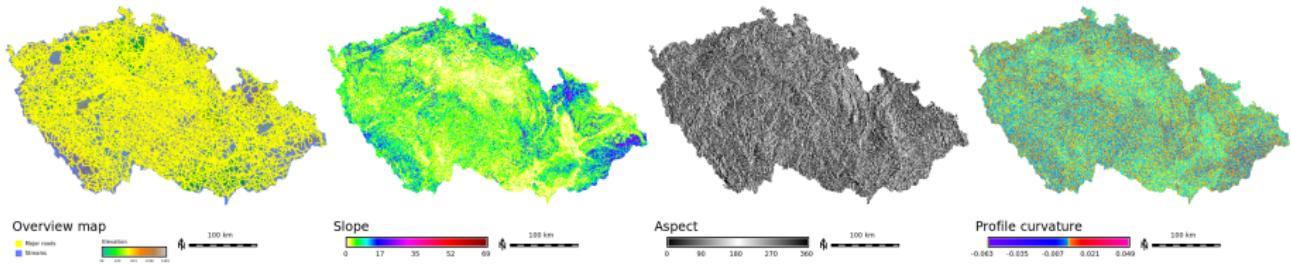


Helena Mitasova<sup>1</sup> and Markus Neteler<sup>2</sup>, authors of  
*Open Source GIS: A GRASS GIS Approach* (fourth edition in preparation)

<sup>1</sup>Department of Marine, Earth, and Atmospheric Sciences, North Carolina State University, USA

<sup>2</sup>Research and Innovation Centre, Fondazione Edmund Mach, Italy

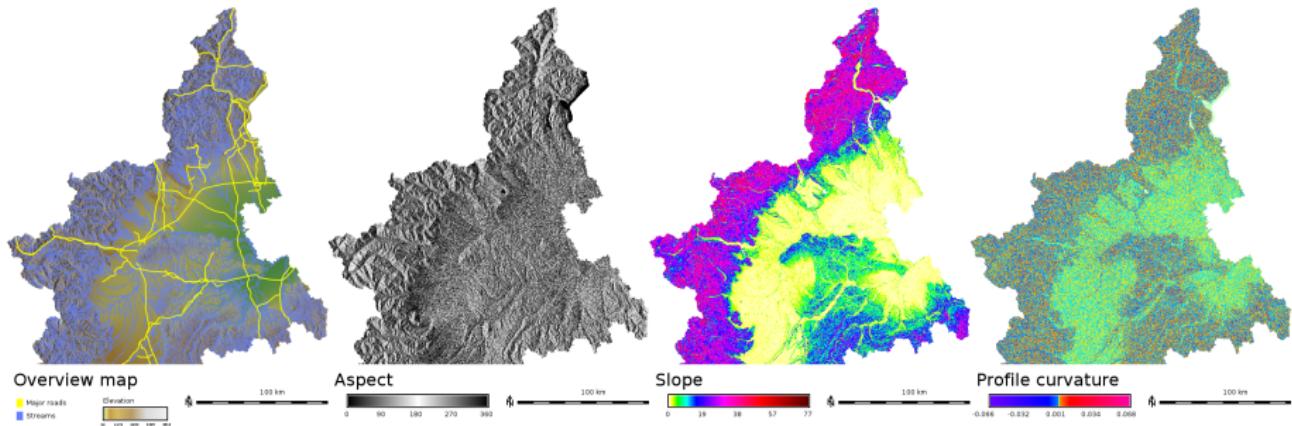
# Standardized Sample Dataset: Czech Republic



Martin Landa\* and Jachym Cepicky from GISMentors

\*OSGeoREL at Czech Technical University in Prague, Faculty of Civil Engineering

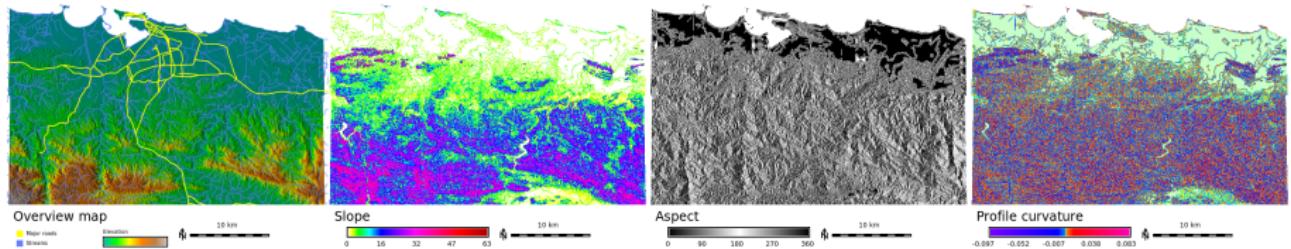
# Standardized Sample Dataset: Piedmont, Italy



Markus Neteler and Luca Delucchi

Research and Innovation Centre, Fondazione Edmund Mach, Italy

# Standardized Sample Dataset: Puerto Rico



Keren Cepero Perez

# IPython Notebook

For workshop:

- ▶ <https://github.com/wenzeslaus/python-grass-addon>
- ▶ IPython Notebook
- ▶ Docker
- ▶ GRASS GIS
  - ▶ <https://github.com/wenzeslaus/grass-gis-docker>

# Future directions: Tools for open science course

- ▶ Course dedicated to
  - ▶ exploring important role FOSS plays in science
  - ▶ overview of tools and methods common in FOSS and desperately needed in science
  - ▶ open access, open data, open standards, open...
  - ▶ *teach next generation of scientists*
- ▶ Possible topics:
  - ▶ combining SW, e.g. QGIS + GRASS GIS + R + Inkscape
  - ▶ Markdown, L<sup>A</sup>T<sub>E</sub>X, Git, IPython Notebook, command line, plain text
  - ▶ general features and common misconceptions
    - ▶ vendor neutrality
    - ▶ using open source versus writing code
    - ▶ community and support
    - ▶ gratis versus libre
    - ▶ counteract FUD against open source from past
  - ▶ availability of tools even when not used by the university
  - ▶ reusability, reproducability (std in FOSS)

# Future directions

- ▶ OpenStreetMap, TeachOSM, LearnOSM
  - ▶ for introduction to geography or GIS
  - ▶ as an example of community-based project
  - ▶ as data source
- ▶ MapStory
- ▶ GIS.lab for easy lab setup
- ▶ IPython/Jupyter, JupyterHub, tmpnb
- ▶ desktop to browser: GTK+ Broadway, noVNC, rollApp
- ▶ web-based tool to explore algorithm behavior in teaching materials
- ▶ link teaching materials, standard user manual and the source code

Thank you for your attention.