GRASS GIS 7:
A theoretical and practical course

Session 1 – Course Introduction
GRASS GIS 7 and OSGeo Overview

Markus Neteler

NINA 2016, Oslo, Norway

mundialis GmbH & Co. KG
http://www.mundialis.de

About the trainer

Markus Neteler: Germany – Italy – Germany

GRASS GIS since 1993, FOSSGIS.de, GFOSS.it, OSGeo.org, etc.

Worked in research from 1999-2016 (mainly in Italy)

Since 2016: partner and general manager at mundialis, Bonn (DE)

mundialis GmbH & Co. KG

- founded in 2015 in Bonn by T. Adams, H. Paulsen and M. Neteler
- 6 staff
- massive GIS data processing and Earth Observation
- offers decades of experience in Open Source GIS (especially GRASS GIS development)
- gained HPC experience through processing of
- MODIS Land Surface Temperature : “EuroLST” – 15 years of gap free daily data at 250m resolution

Dr. Markus Neteler
mundialis GmbH & Co. KG
Kölnstraße 99
53111 Bonn, Germany

Email: neteler@mundialis.de
Web: http://www.mundialis.de

Course overview

1) Introduction
Who is the presenter
Course overview: structure
What are OSGeo, GFOSS, GRASS GIS, OSGeolive

2) GRASS GIS Software first steps
- Intro QGIS-Processing-GRASS GIS
- Software installation notes
- Data: course data: North Carolina
- Using GRASS GIS in QGIS through "Processing"

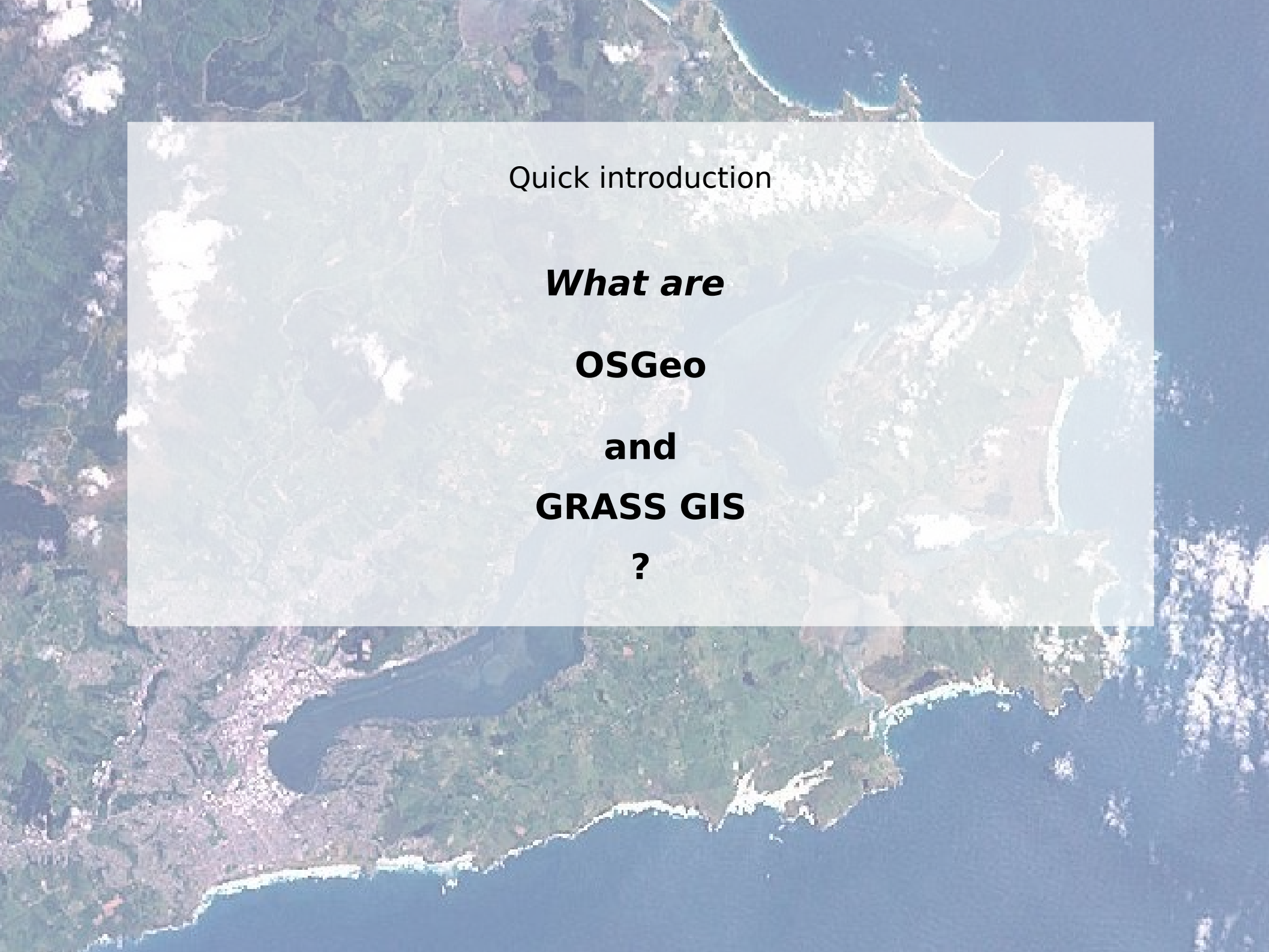
3) GRASS GIS general introduction
- Database structure of GRASS GIS
- About the course data set
- First steps in using GRASS GIS 7
  - Graphical user interface (GUI)
  - GRASS command structure
  - Command line or GUI?
  - Creating a perspective view

4) GRASS GIS raster introduction
- raster processing concepts
- import of a GeoTIFF (DEM)
- Color tables, NULL masks etc
- hydrological modelling
- raster capabilities in GRASS GIS

5) GRASS GIS vector introduction
- Why a topological vector data model
- Vector feature extraction
- Vector geometry dissolving
- Geometry editing/digitizing
- Import/export
- Capabilities of the vector engine

6) GRASS GIS sampling
- Random sampling
- Stratified sampling

7) GRASS GIS Temporal introduction
- Temporal GRASS framework
- Working with climate data
- Starting with temporal framework
- Aggregating , querying, visualizing temporal data



Quick introduction

What are

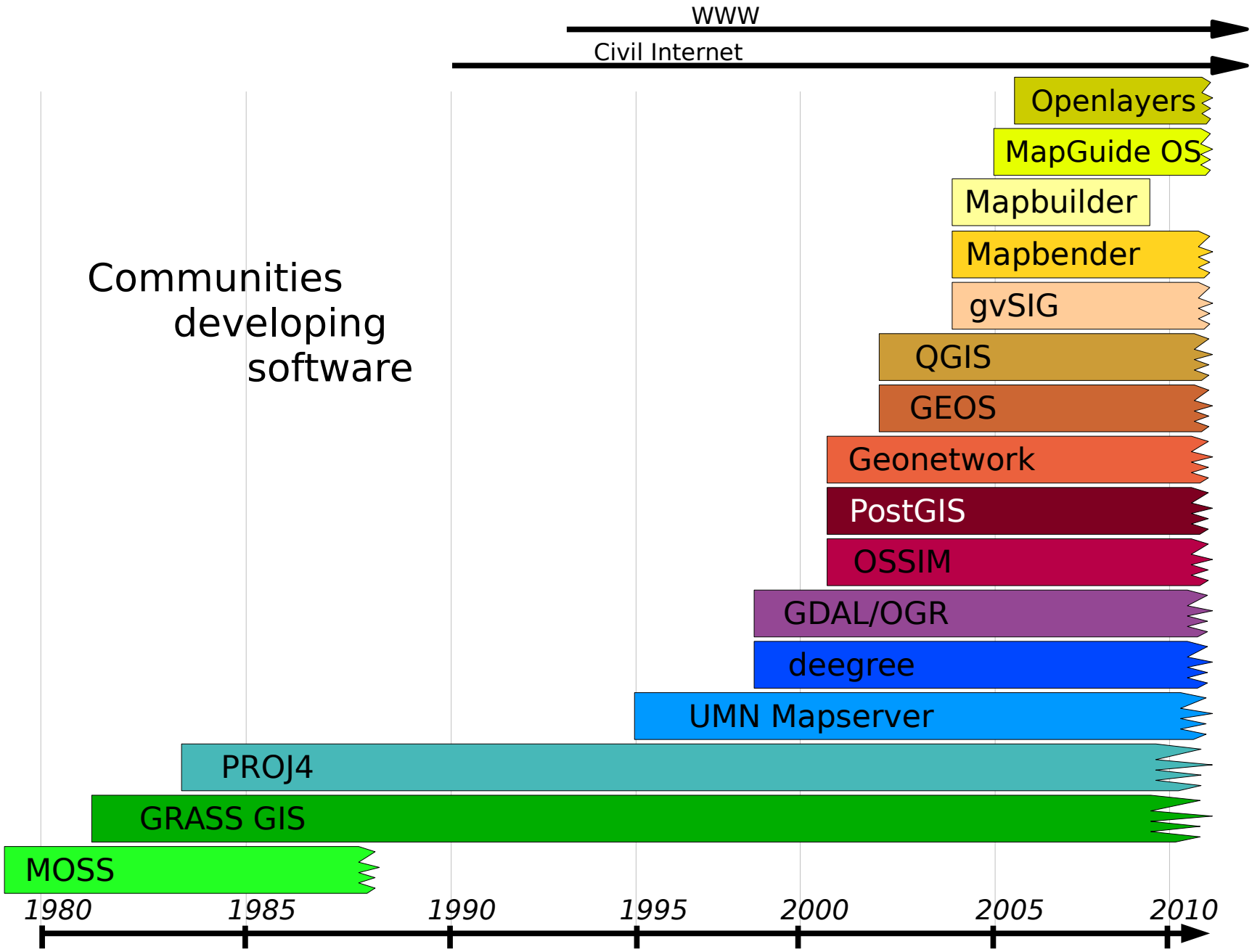
OSGeo

and

GRASS GIS

?

Open Source GIS Timeline



http://wiki.osgeo.org/wiki/Open_Source_GIS_History#Timeline

Open Source GIS brought to you by...

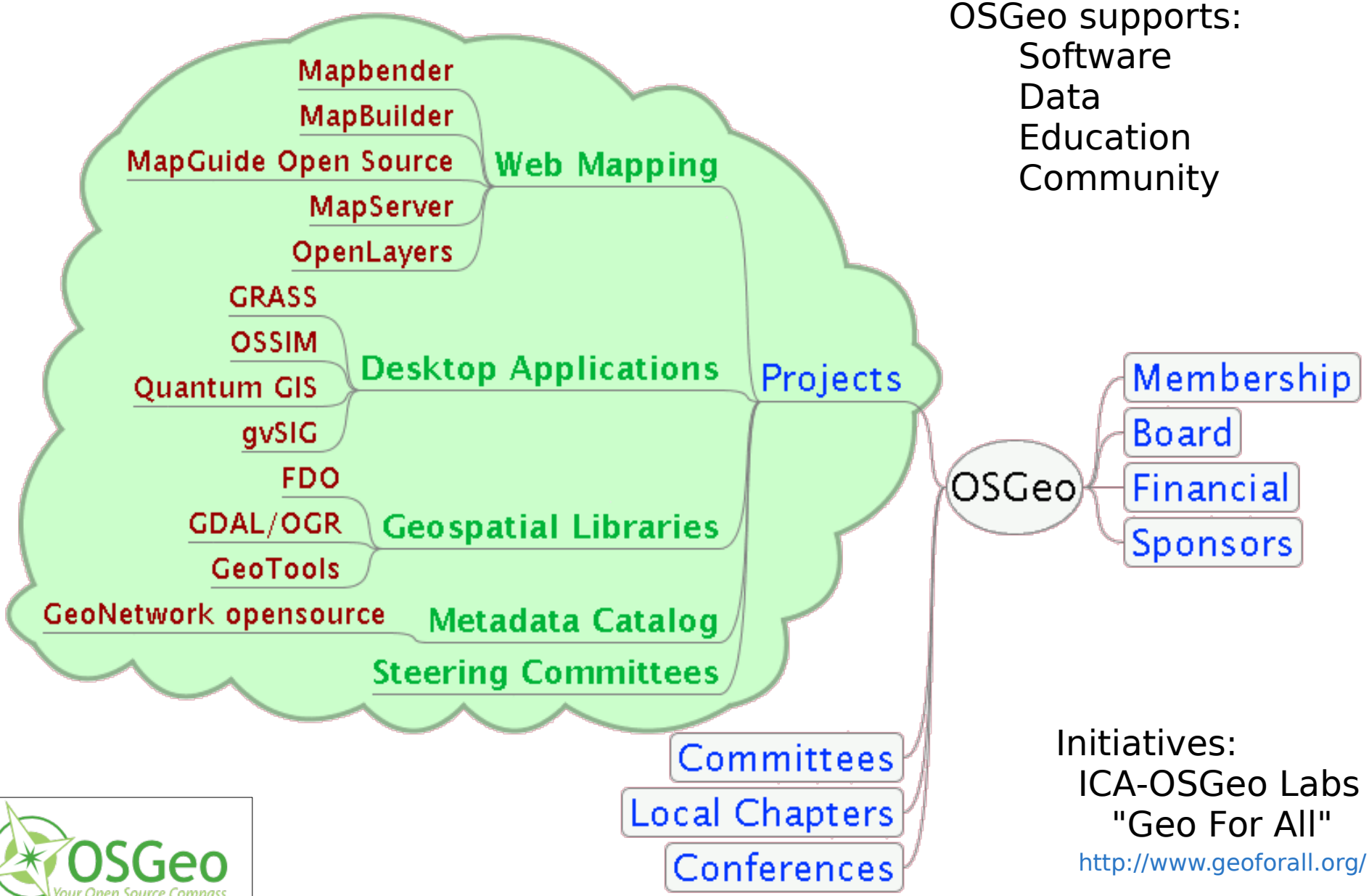
... the OSGeo community: at time more than 28,000 unique subscribers in over 240 topic oriented mailing lists, since 2006

http://www.osgeo.org

Open Source Geospatial Foundation

http://www.osgeo.org

OSGeo supports:
- Software
- Data
- Education
- Community



Initiatives:
- ICA-OSGeo Labs
- "Geo For All"

http://www.geoforall.org/

Geo for All!

http://www.geoforall.org/

- Home
- About
- Advisory board
- How to join
- Locations
- News
- Newsletters
- Resources and references
- Training
- Webinars

Home

Mission - "Making geospatial education and opportunities accessible to all"

"Geo for All" is the Open Source Geospatial Foundation's Open Educational outreach with our partners worldwide with the mission for making geospatial education and opportunities accessible to all.

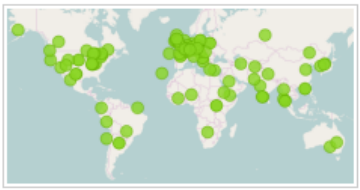
In September 2011, the Open Source Geospatial Foundation (OSGeo) and the International Cartographic Association (ICA) signed a Memorandum of Understanding with the aim of developing on a global basis collaboration opportunities for academia, industry and government organizations in open source GIS software and data. The MoU aims to provide expertise and support for the establishment of Open Source Geospatial Laboratories and Research Centres across the world for supporting development of open-source geospatial software technologies, training and expertise. The International Society for Photogrammetry and Remote Sensing (ISPRS) joined the Geo for All initiative in July 2014. The goal of the "Geo for All" initiative is to promote and enhance education, research and service activities carried out by these stakeholders in the area of **Open Geospatial Science & Applications** all over the world. By combining the potential of free and open GI software, open data, open standards, open access to research publications, open education resources in Geospatial education and research will enable creation of sustainable innovation ecosystem to advance the discipline. This is key for widening geospatial education opportunities, accelerating new discoveries and helping solving global cross disciplinary societal challenges from Climate change mitigation to sustainable cities.

OSGeo

ICA ACI

ICA-OSGeo MoU

isprs Information from Imagery



110 labs worldwide
as of 20th August, 2016

» GeoforAll overview presentation

The new Geo for All newsletter, April 2016 is available...

The April 2016 Geo for All newsletter is available! Check the most recent news and information about events to come. The Lab of the Month is the Regional Centre for Mapping of Resources for Development (RCMRD), Nairobi, Kenya!...

Geo for All Educator Awards 2015...

On behalf of the GeoForAll Educator Award Selection Committee, we are pleased to inform all that the Individual and Team Awards for the "GeoForAll - Global Educator of the Year Award 2015" has been announced at the FOSS4G 2015 - Europe "Open Innovation for Europe" conference at Como, Italy on Friday...

ISPRS joins ...

The "Geo for

The FOSSGIS “ecosystem”: Try it without hassle – OSGeo LiveDVD and LiveUSB stick

OSGeo Live

http://live.osgeo.org/

Home | Contents | Standards | Download | Metrics | Sponsors | Contact Us

English | Español | Català | Français | Deutsch | Italiano | Polski | Ελληνικά | Русский | 中文 | 한국어 | 日本語

Welcome to OSGeo-Live 10.0

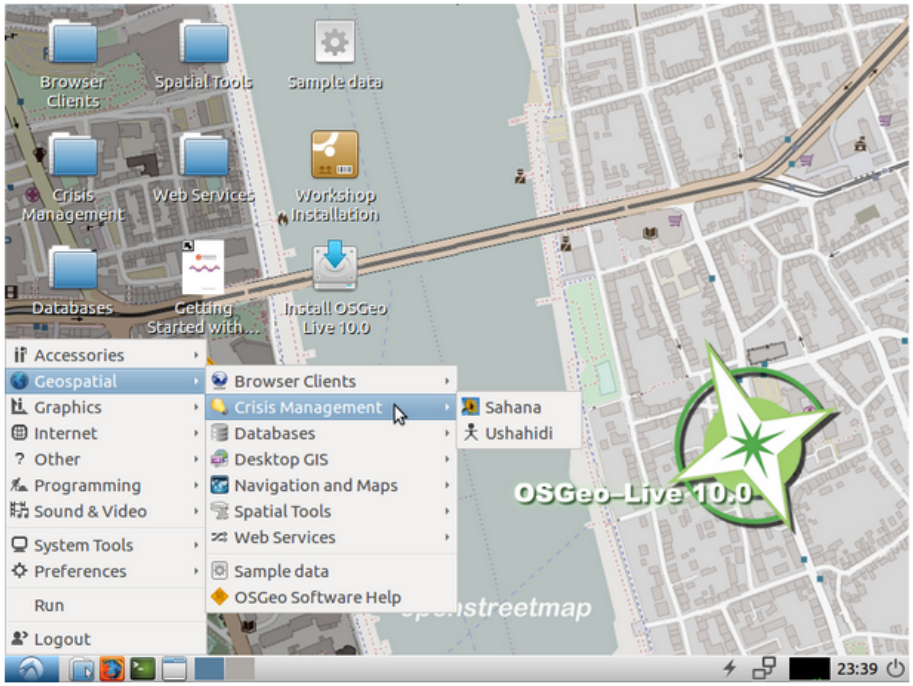
OSGeo-Live is a self-contained bootable DVD, USB thumb drive or Virtual Machine based on Lubuntu, that allows you to try a wide variety of open source geospatial software without installing anything. It is composed entirely of free software, allowing it to be freely distributed, duplicated and passed around.

It provides pre-configured applications for a range of geospatial use cases, including storage, publishing, viewing, analysis and manipulation of data. It also contains sample datasets and documentation.

To try out the applications, simply:

1. Insert DVD or USB thumb drive in computer or virtual machine.
2. Reboot computer. (verify boot device order if necessary)
3. Press “Enter” to startup & login.
4. Select and run applications from the “Geospatial” menu.

OSGeo-Live is an OSGeo Foundation project. The OSGeo Foundation is a not-for-profit supporting Geospatial Open Source Software development, promotion and education.



Quick Starts

- Getting started with the OSGeo-Live DVD
- Change language or keyboard type
- Install OSGeo-Live on your hard disk
- Run OSGeo-Live in a Virtual Machine
- Create an OSGeo-Live bootable USB thumb drive

Presentation

A half hour presentation, highlighting all OSGeo-Live applications, is available with slides, script, and abstract.

English | Español | Català | Français | Deutsch | Italiano | Polski | Ελληνικά | Русский | 中文 | 한국어 | 日本語

Copyright & Disclaimer

Development meetings: Community sprints

Short guide to

GRASS GIS

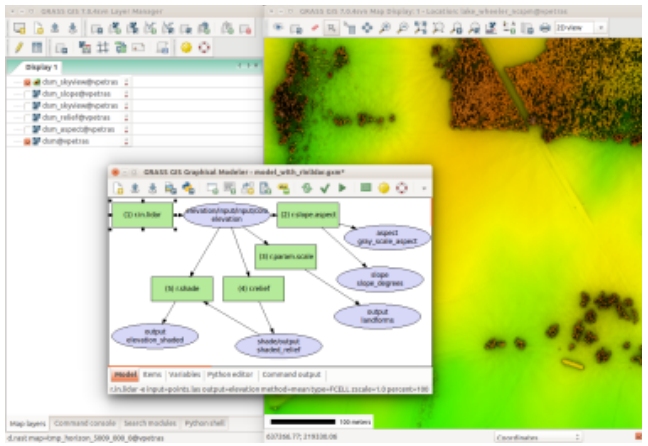
What's GRASS GIS?

http://grass.osgeo.org

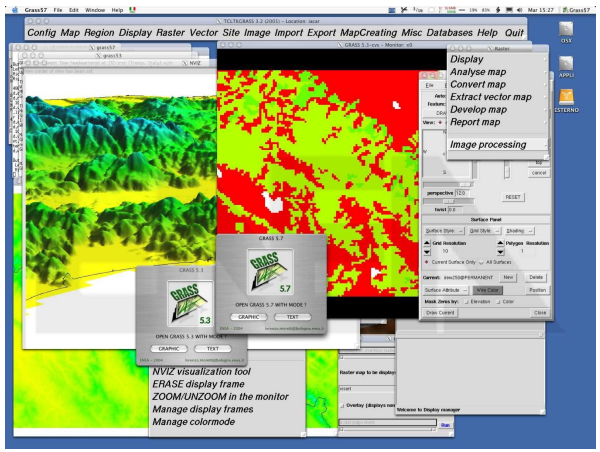
- **G**eographic **R**esources **A**nalysis **S**upport **S**ystem
- **Open Source GIS**, developed since 1984, since 1999 GNU GPL
- **Portable code** (many operating systems, 32/64bit)
- Your **GIS backbone** – linkable to:

GDAL, Q, R, SEXTANTE, ParaView, ZOO (http://zoo-project.org), PyWPS

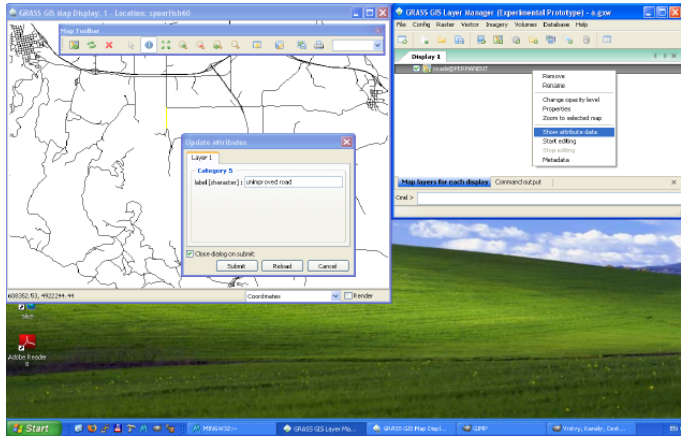
GNU/Linux



MacOSX

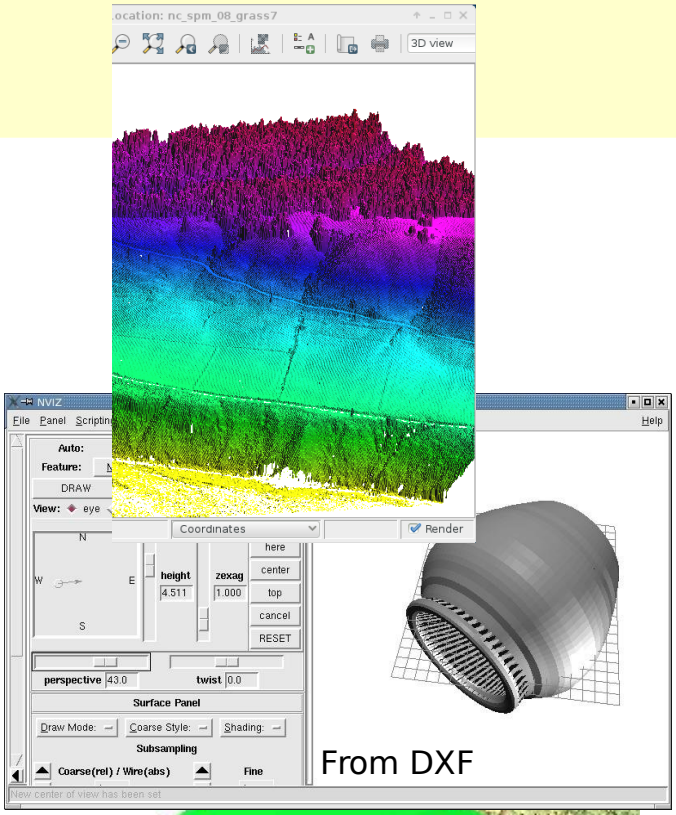


MS-Windows (IBM AIX, *BSD, ...)

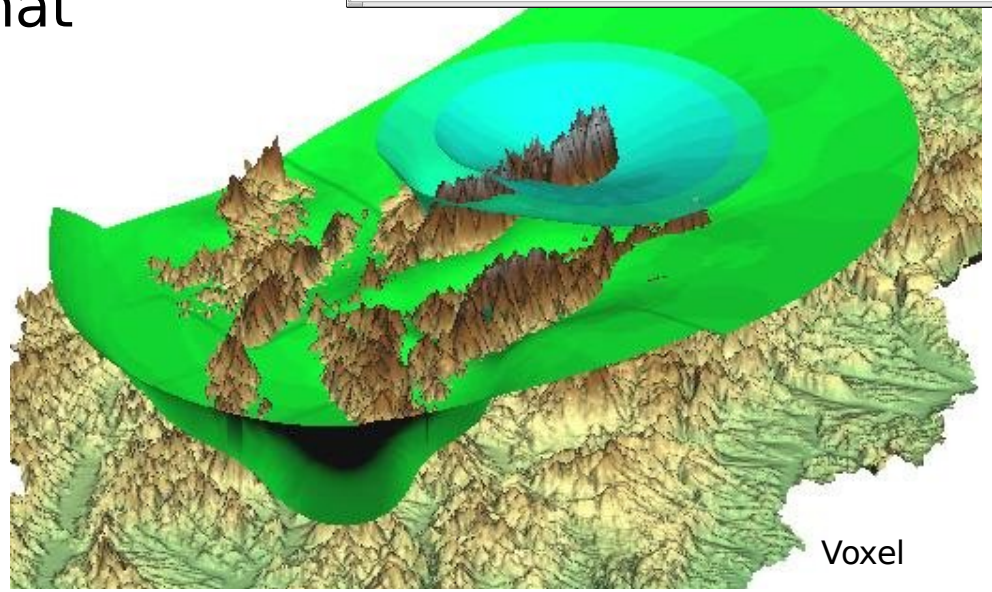


What's GRASS GIS?

- Raster 2D/3D (voxel) processing
- Vector 2D/3D topological processing
- Vector network analysis support
- Image processing system
- Space-time cubes, temporal GIS
- Native raster and vector format
- 3D Visualization system
- DBMS integrated (SQL) with SQLite, DBF, PostgreSQL, MySQL, and ODBC drivers



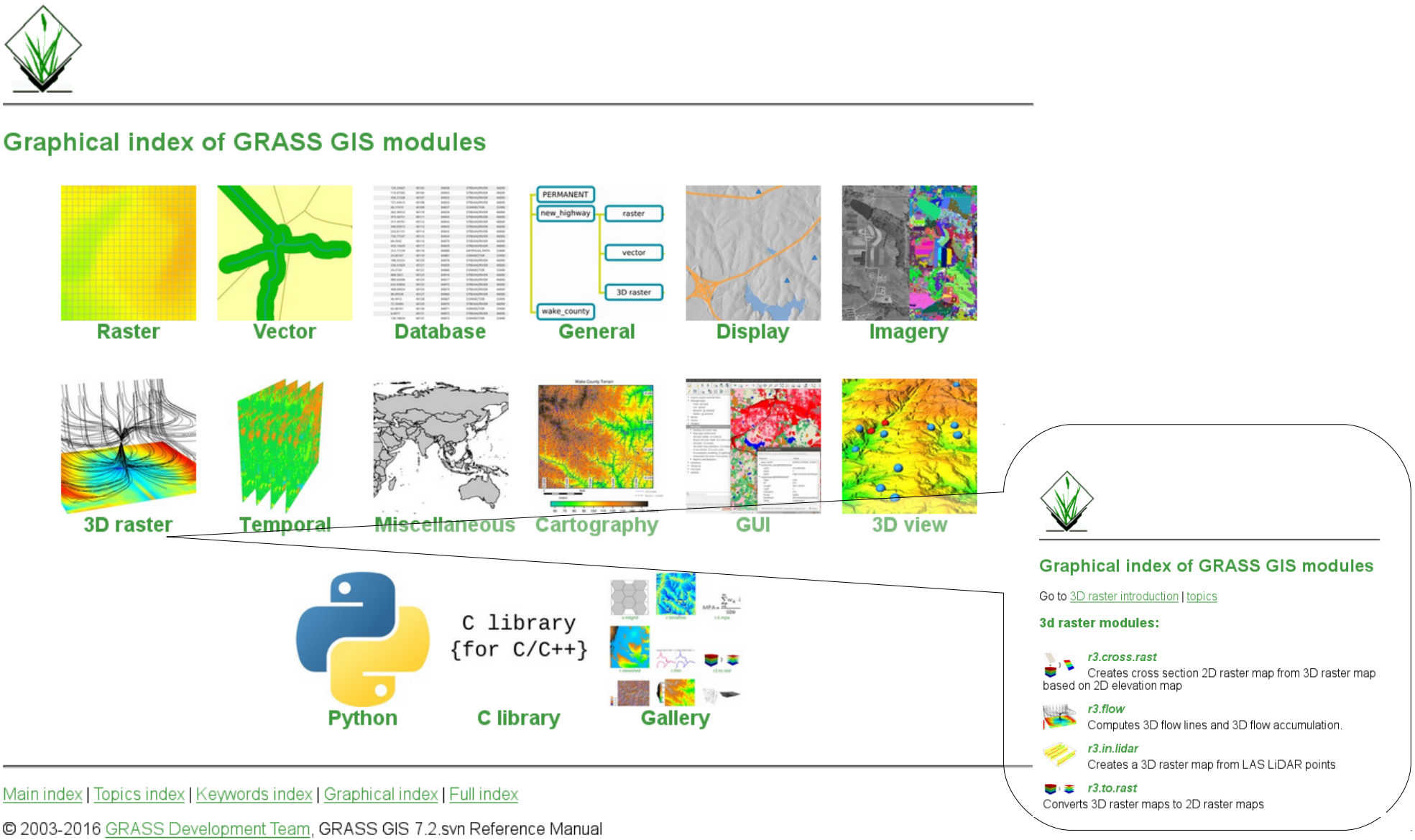
From DXF


Voxel

GRASS GIS 7 capabilities: a graphical overview:
http://www.slideshare.net/markusN/grass-gis-7-capabilities-a-graphical-overview

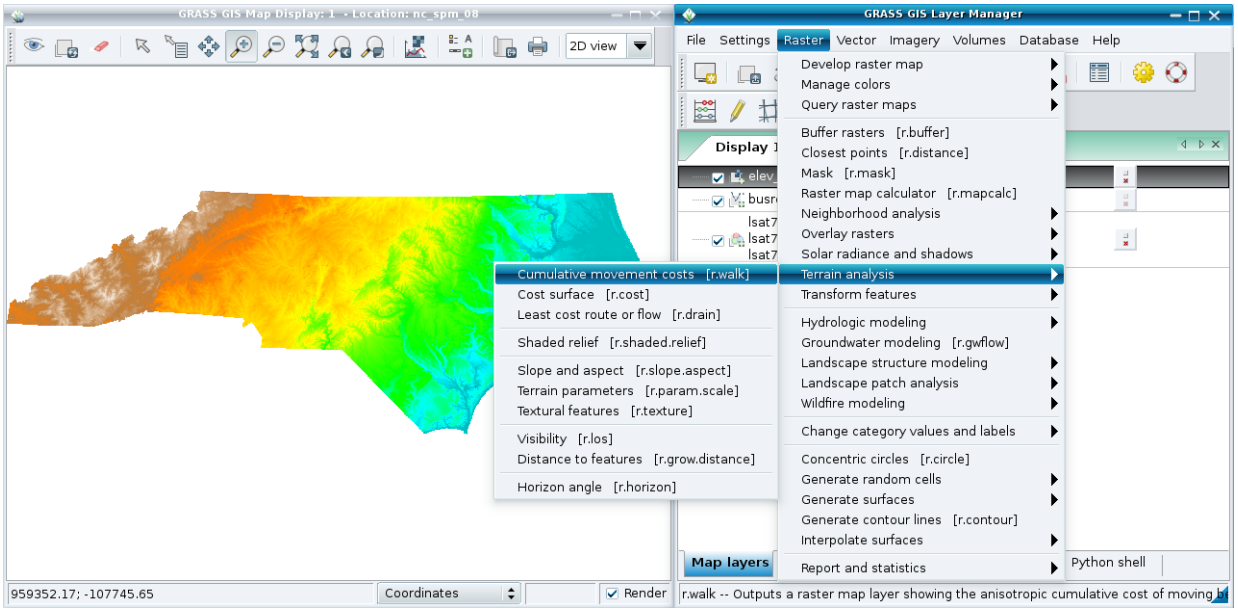
What's GRASS GIS?

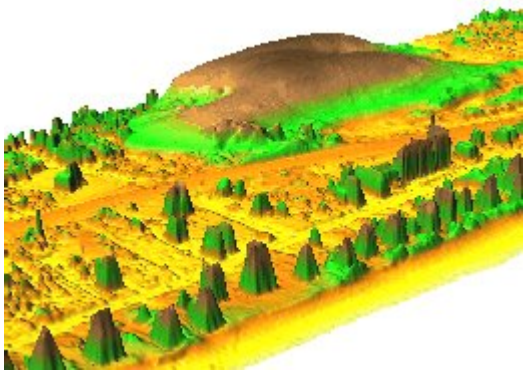
Graphical index of GRASS GIS modules



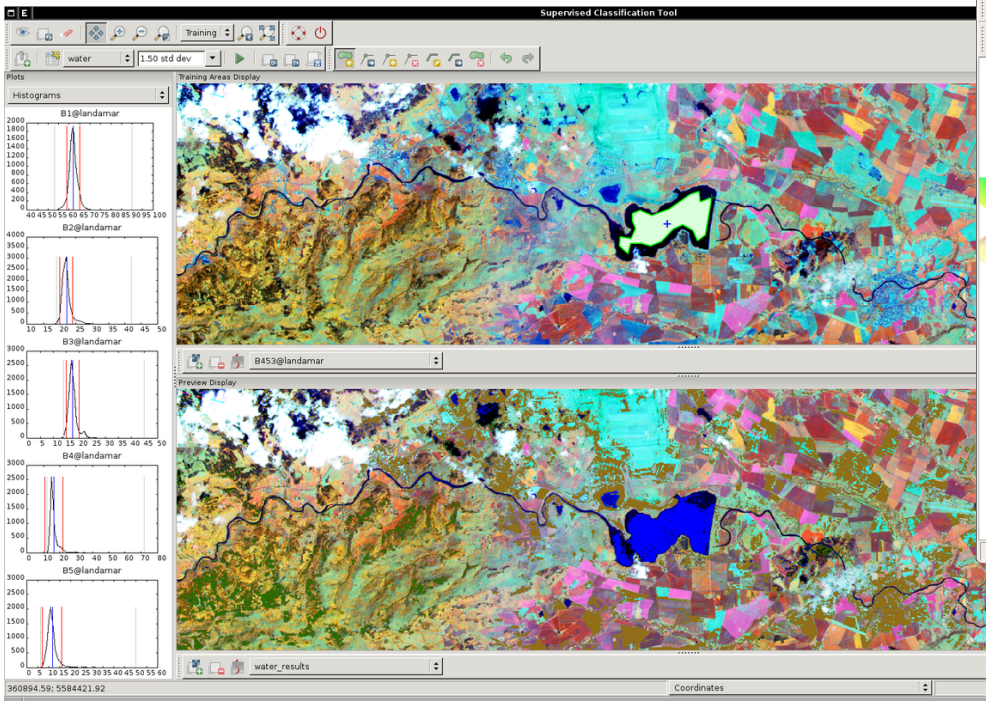
https://grass.osgeo.org/grass72/manuals/graphical_index.html

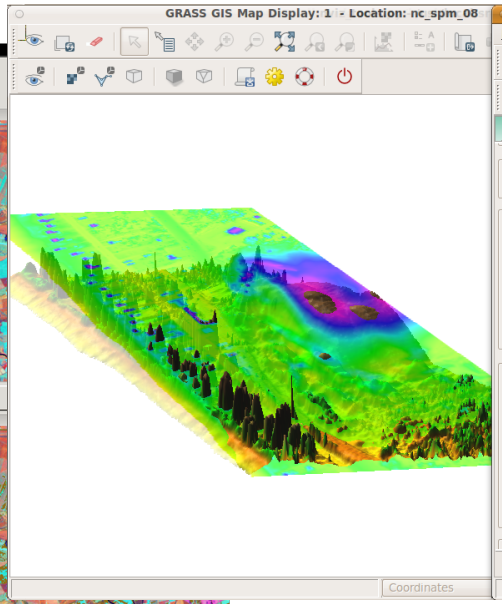
GRASS GIS 7 User interface

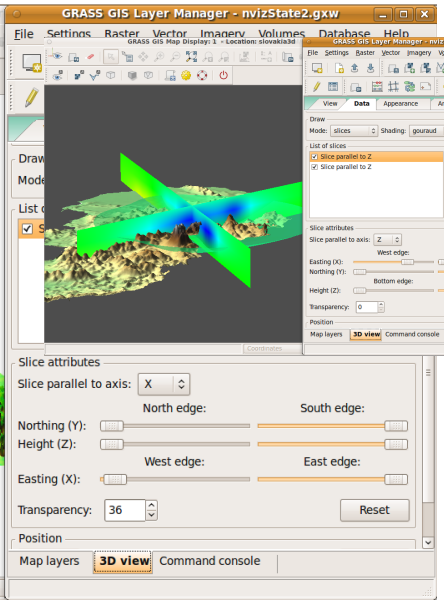




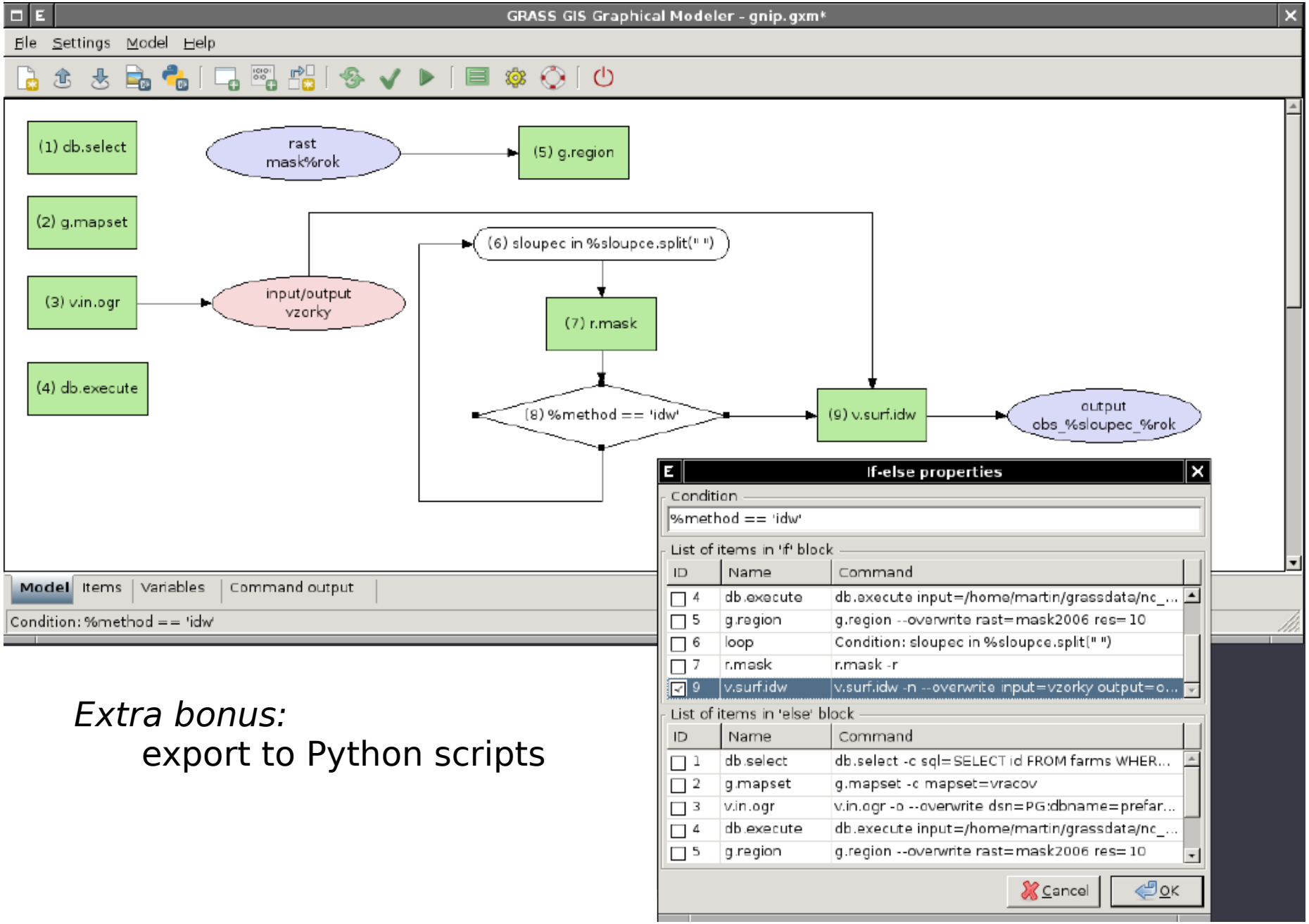
Nagshead LiDAR time series: dune moving over 9 years (NC, USA)







GRASS GIS 7: Geospatial Modeller

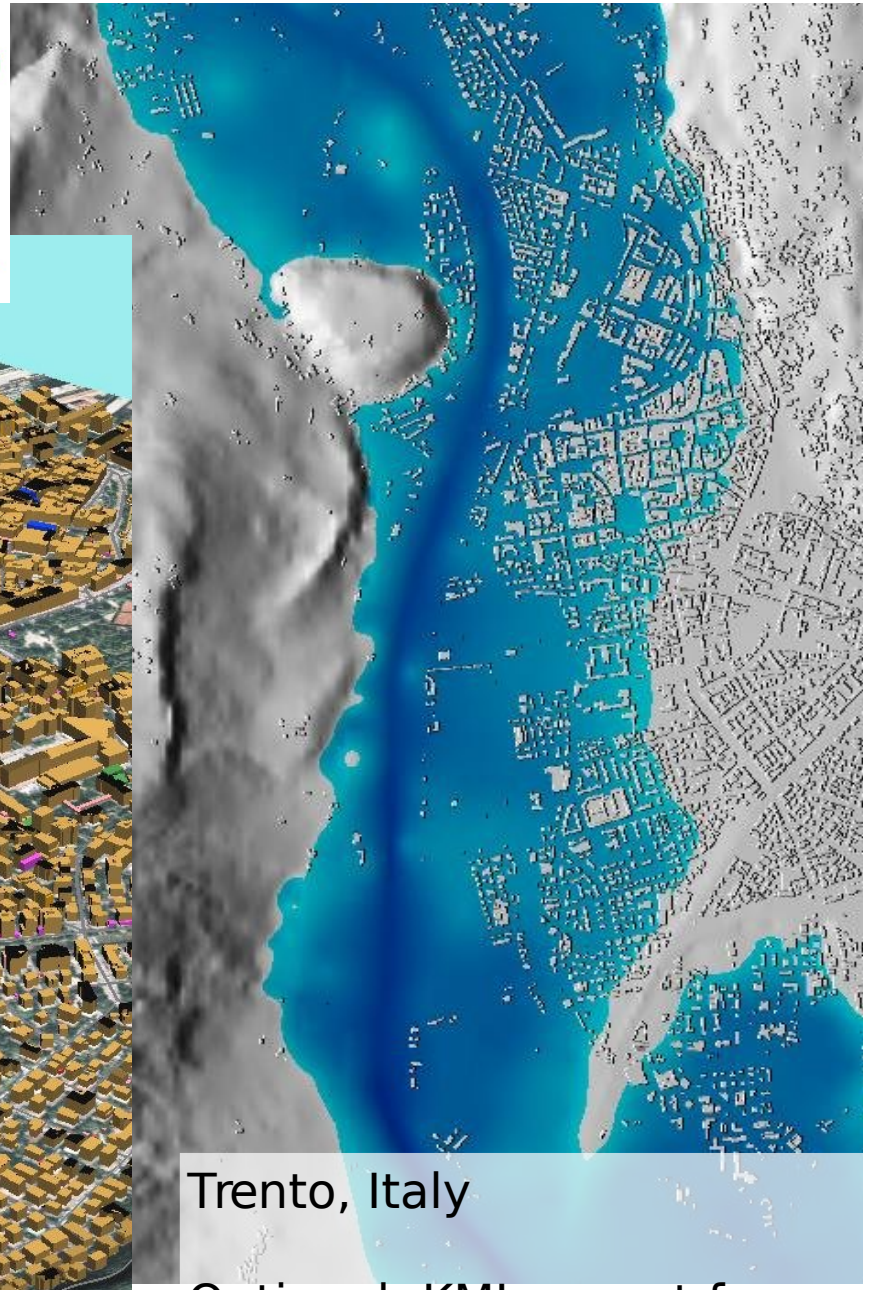


Extra bonus:

export to Python scripts

Raster and 3D vector

Elevation model combined with extruded 3D buildings;
also true 3D vector supported



Trento, Italy

Optional: KML export for virtual globes

GRASS Topological 2D/3D Vector model

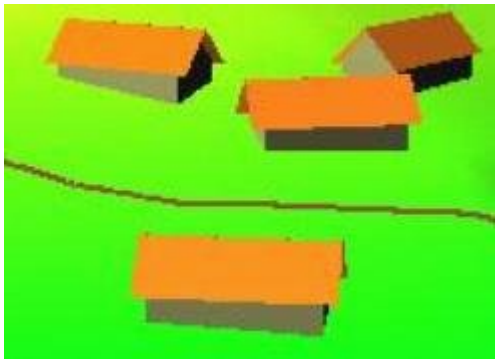
Vector geometry types

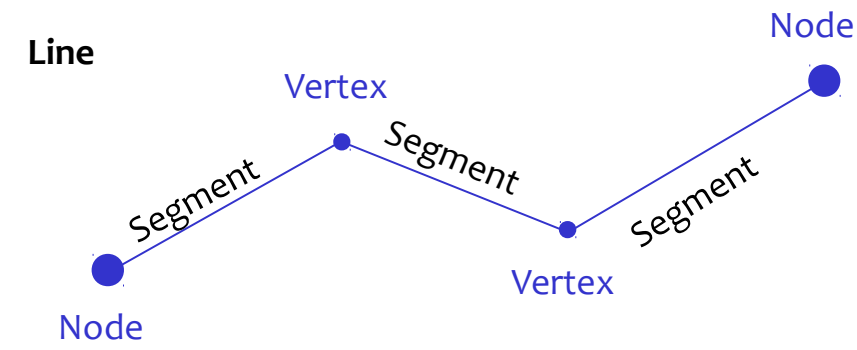
- Point
- Centroid
- Line
- Boundary
- Area (boundary + centroid)
- face (3D area)
- [kernel (3D centroid)]
- [volumes (faces + kernel)]

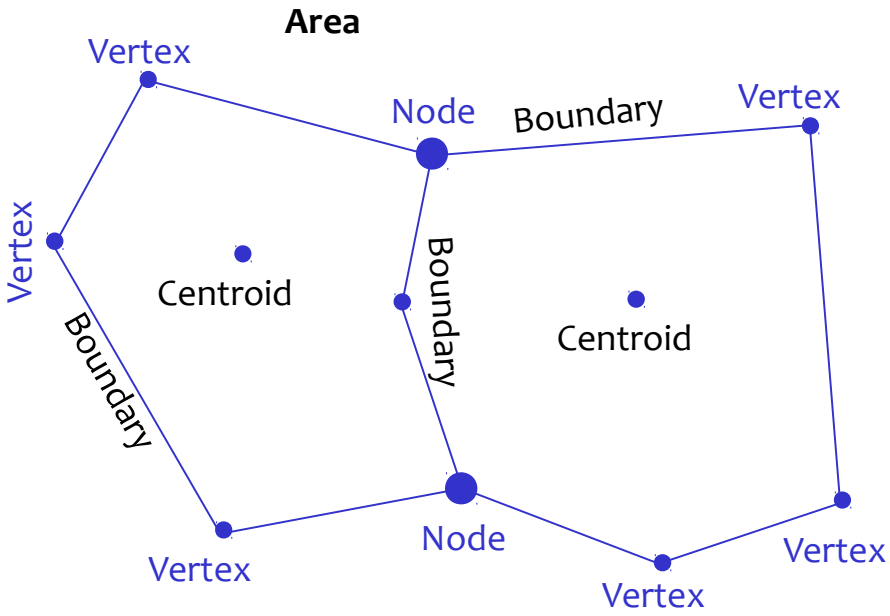
Geometry is **true** 3D when: x, y, z

not in all GIS!

Faces

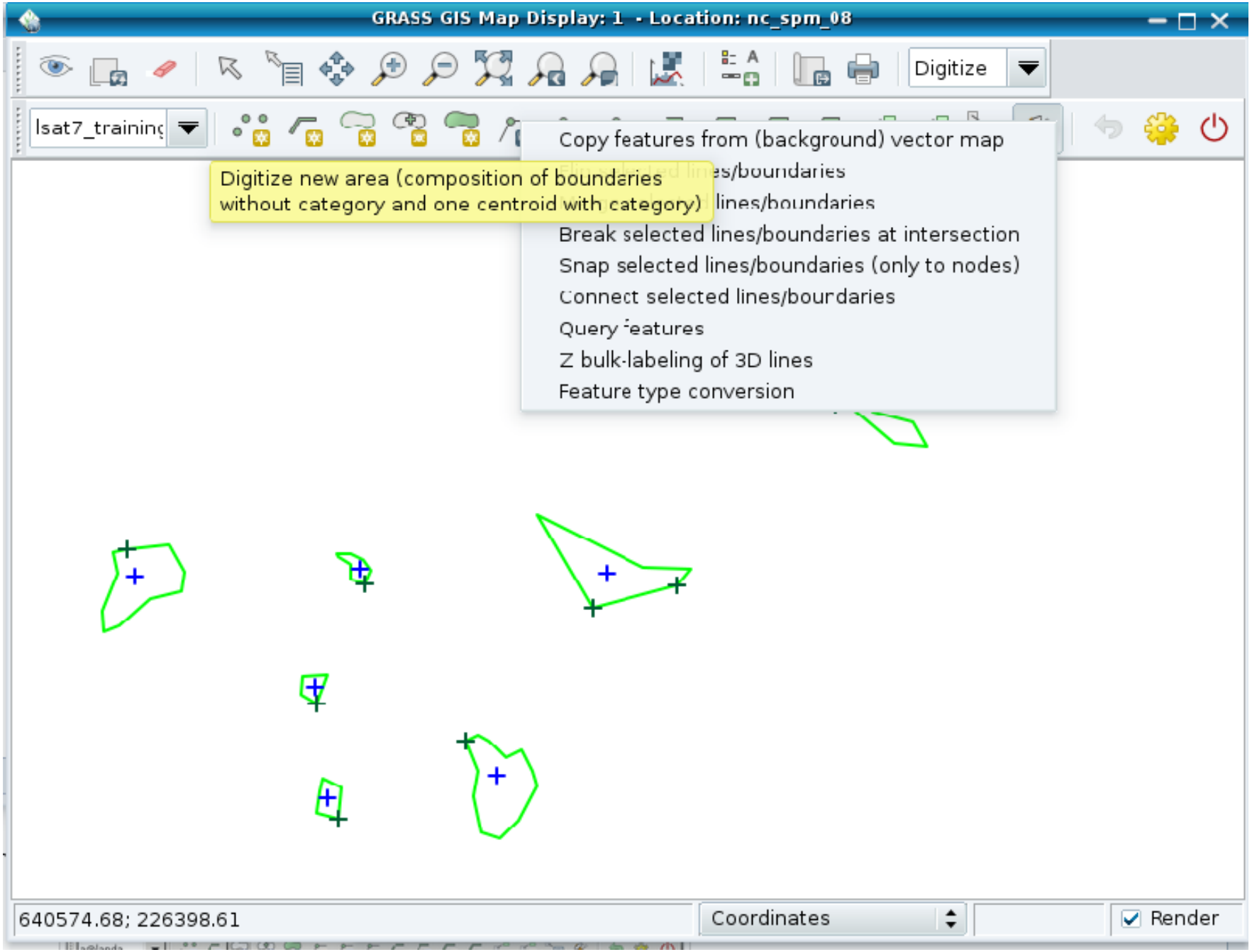






Use of Spatial Index

GRASS Topological Vector Digitizer



New Space-Time functionality in GRASS 7

Temporal data processing in GRASS GIS

The temporal GIS framework in GRASS introduces three new datatypes that are designed to handle time series data:

- *Space time raster datasets* (strds) are designed to manage raster map time series. Modules that process strds have the naming prefix *t.rast*.
- *Space time 3D raster datasets* (str3ds) are designed to manage 3D raster map time series. Modules that process str3ds have the naming prefix *t.rast3d*.
- *Space time vector datasets* (stvds) are designed to manage vector map time series. Modules that process stvds have the naming prefix *t.vect*.

Temporal data management in general

List of general management modules:

- t.connect
- t.create
- t.remove
- t.register
- t.unregister
- t.info
- t.list
- t.rast3d.list
- t.vect.list
- t.vect.db.select
- t.sample
- t.support
- t.topology

Export/import conversion

- t.rast.export
- t.rast.import
- t.rast.out.vtk
- t.rast.to.rast3
- r3.out.netcdf
- t.vect.export

Statistics and gap filling

- t.rast.gapfill
- t.rast.univar

Querying and map calculation

- t.rast.list
- t.rast.extract
- t.rast.gapfill
- t.rast.mapcalc
- t.rast3d.extract
- t.rast3d.mapcalc
- t.rast3d.univar
- t.vect.extract
- t.vect.import
- t.vect.observe.strds
- t.vect.univar
- t.vect.what.strds

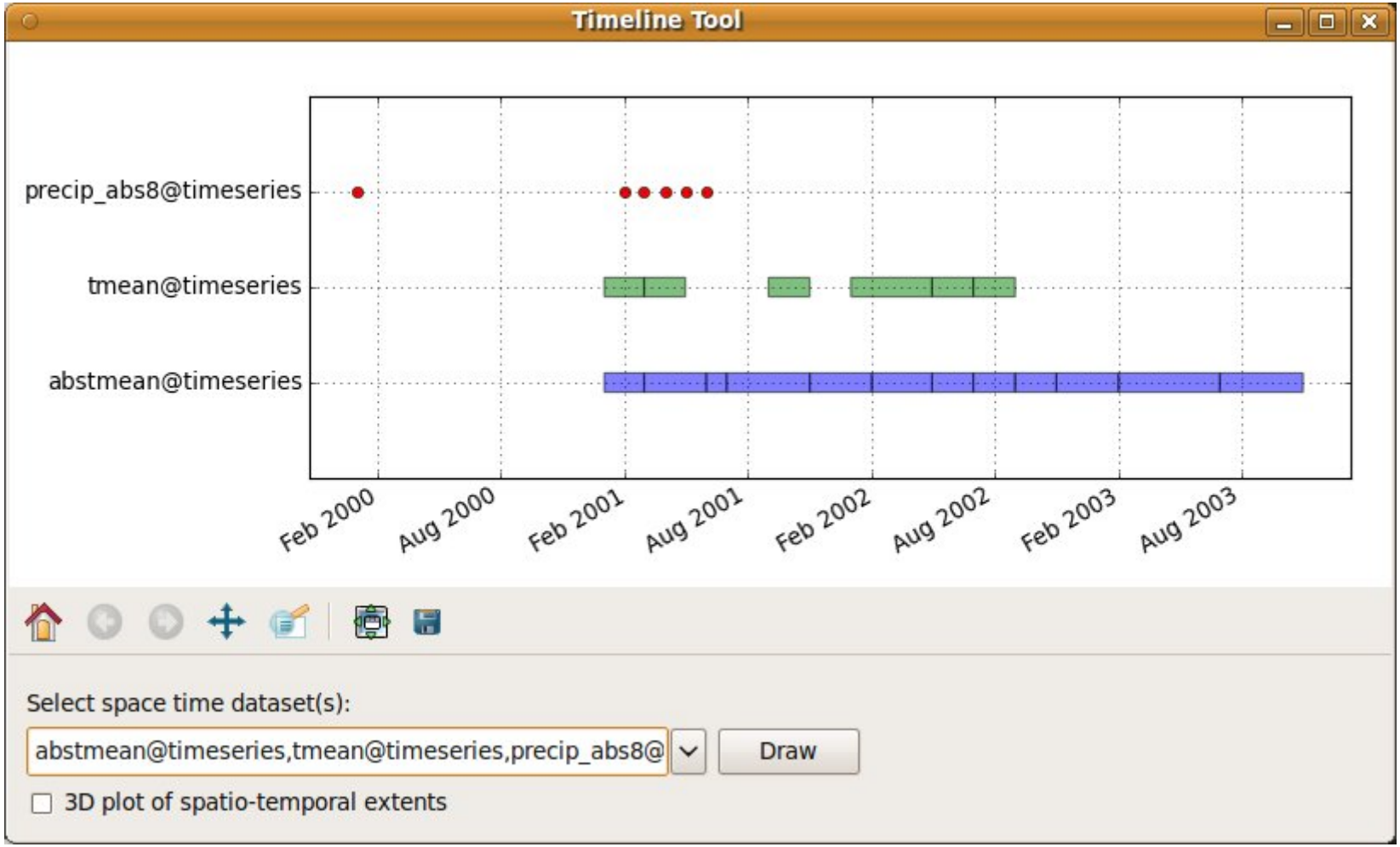
Aggregation

- t.rast.aggregate.ds
- t.rast.aggregate
- t.rast.series

Space time datasets are stored in a temporal database. SQLite3 or PostgreSQL are supported as SQL database back end. Connection settings are performed with t.connect. As default a sqlite3 database will be created in the PERMANENT mapset that stores all space time datasets and registered time series maps from all mapsets in the location.

https://grass.osgeo.org/grass72/manuals/temporalintro.html

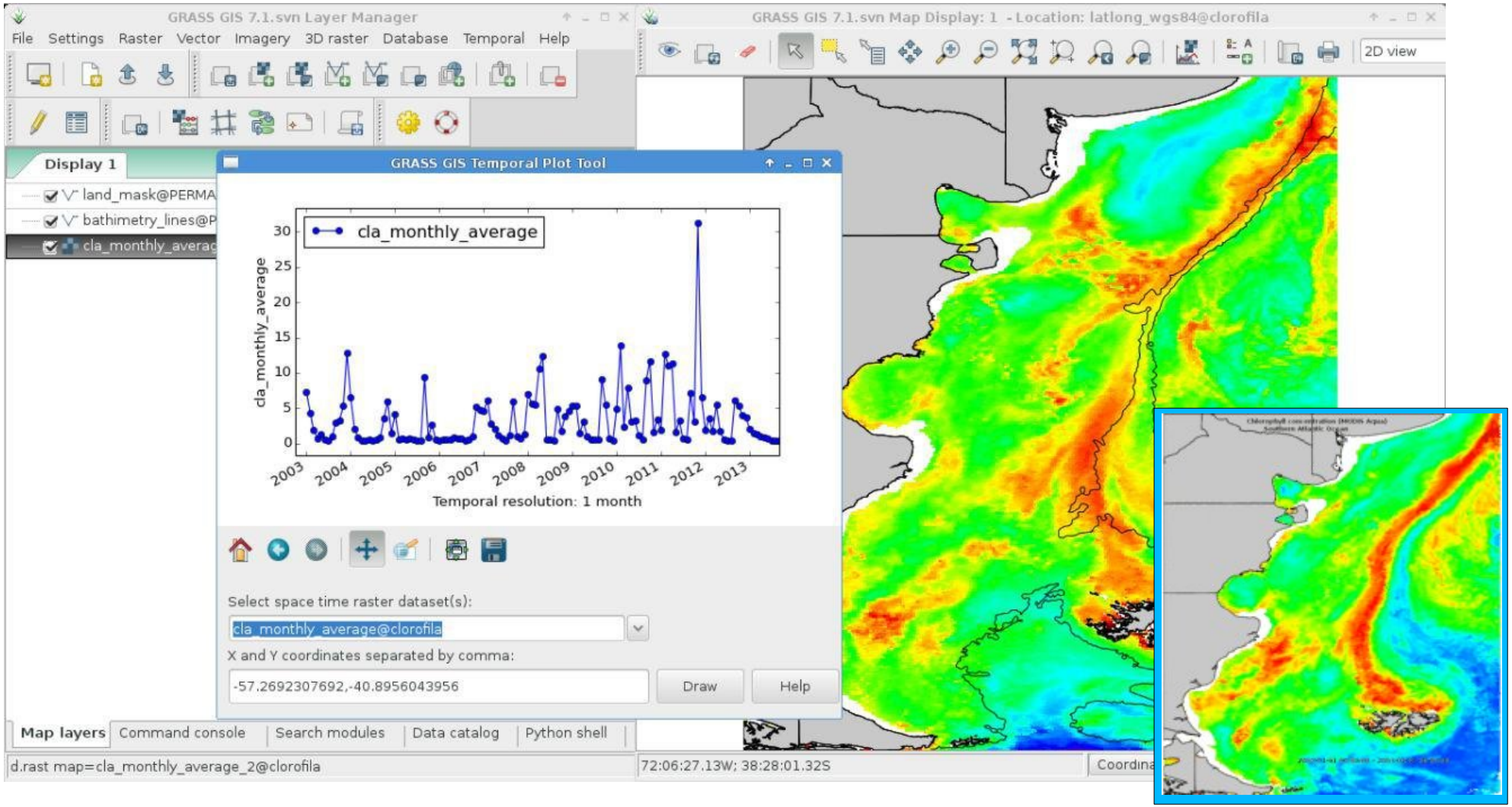
GRASS GIS 7: Space-time functionality



Screenshot: S Gebbert/A. Petrasova

t.register: Registers raster, vector and raster3d maps in a space time dataset

GRASS GIS 7: Space-time functionality



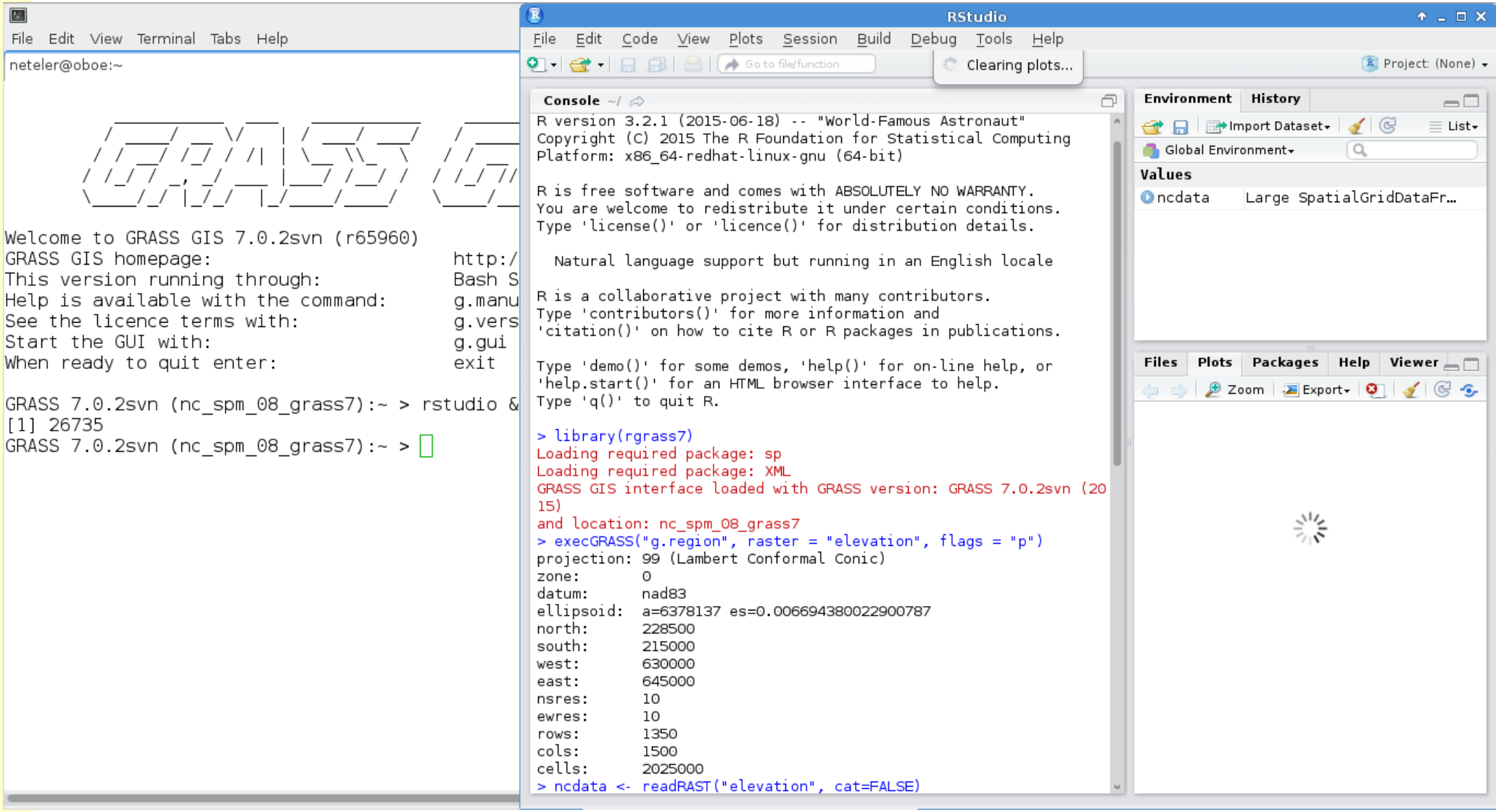
g.gui.tplot: plots the values of one or more temporal raster datasets for a queried point defined by a coordinate pair

(in PDF, click for animation)

GRASS GIS 7 and R integration

There is a dedicated R packages “rgrass7” for GRASS GIS data exchange

https://grasswiki.osgeo.org/wiki/R_statistics/rgrass7



Python API integration

http://grass.osgeo.org/wiki/GRASS_and_Python

Page | Discussion | Read | Edit | View history | Go | Search

GRASS and Python

Navigation

- Main Page
- Community
- Development
- Documents
- GRASS Help
- Recent changes
- Help

Toolbox

- What links here
- Related changes
- Upload file
- Special pages
- Printable version
- Permanent link

Introduction to Vector classes — Python library documentation documentation - Konqueror

File Edit View Go Bookmarks Tools Settings Window Help

http://grass.osgeo.org/grass71/manual/ Google

Python library documentation documentation » PyGRASS documentation » previous | next | modules | index

Introduction to Vector classes

Details about the architecture can be found in the GRASS GIS 7 Programmer's Manual: GRASS Vector Library

Instantiation and basic interaction.

```
>>> from pygrass.vector import VectTopo
>>> municip = VectTopo('boundary_municp_sqlite')
>>> municip.is_open()
False
>>> municip.mapset
''
>>> municip.exist()  # check if exist, and if True set mapset
True
>>> municip.mapset
'user1'
```

Open the map with topology:

```
>>> municip.open()

get the number of primitive:
```

Previous topic
Introduction to Raster classes

Next topic
Interface to GRASS GIS modules

Quick search

Go

Enter search terms or a module, class or function name.

Using Python and GRASS GIS 7 with ipython

An interactive (Web based!) shortcourse on writing GRASS scripts in Python

https://github.com/wenzeslaus/python-grass-addon

Introduction to the GRASS GIS 7 Python Scripting Library

The GRASS GIS 7 Python Scripting Library provides functions to call GRASS modules within scripts as subprocesses. The most often used functions include:

- **run_command**: most often used with modules which output raster/vector data where text output is not expected
- **read_command**: used when we are interested in text output
- **parse_command**: used with modules producing text output as key=value pair
- **write_command**: for modules expecting text input from either standard input or file

Besides, this library provides several wrapper functions for often called modules.

Calling GRASS GIS modules

We start by importing GRASS GIS Python Scripting Library:

In [ ]:
```
import grass.script as gscript
```

Before running any GRASS raster modules, you need to set the computational region using g.region. In this example, we set the computational extent and resolution to the raster layer elevation.

In [ ]:
```
gscript.run_command('g.region', raster='elevation')
```

The run_command() function is the most commonly used one. Here, we apply the focal operation *average* (r.neighbors) to smooth the elevation raster layer. Note that the syntax is similar to bash syntax, just the flags are specified in a parameter.

GRASS Addons: User contributed extensions

The Addons repository is SVN based:

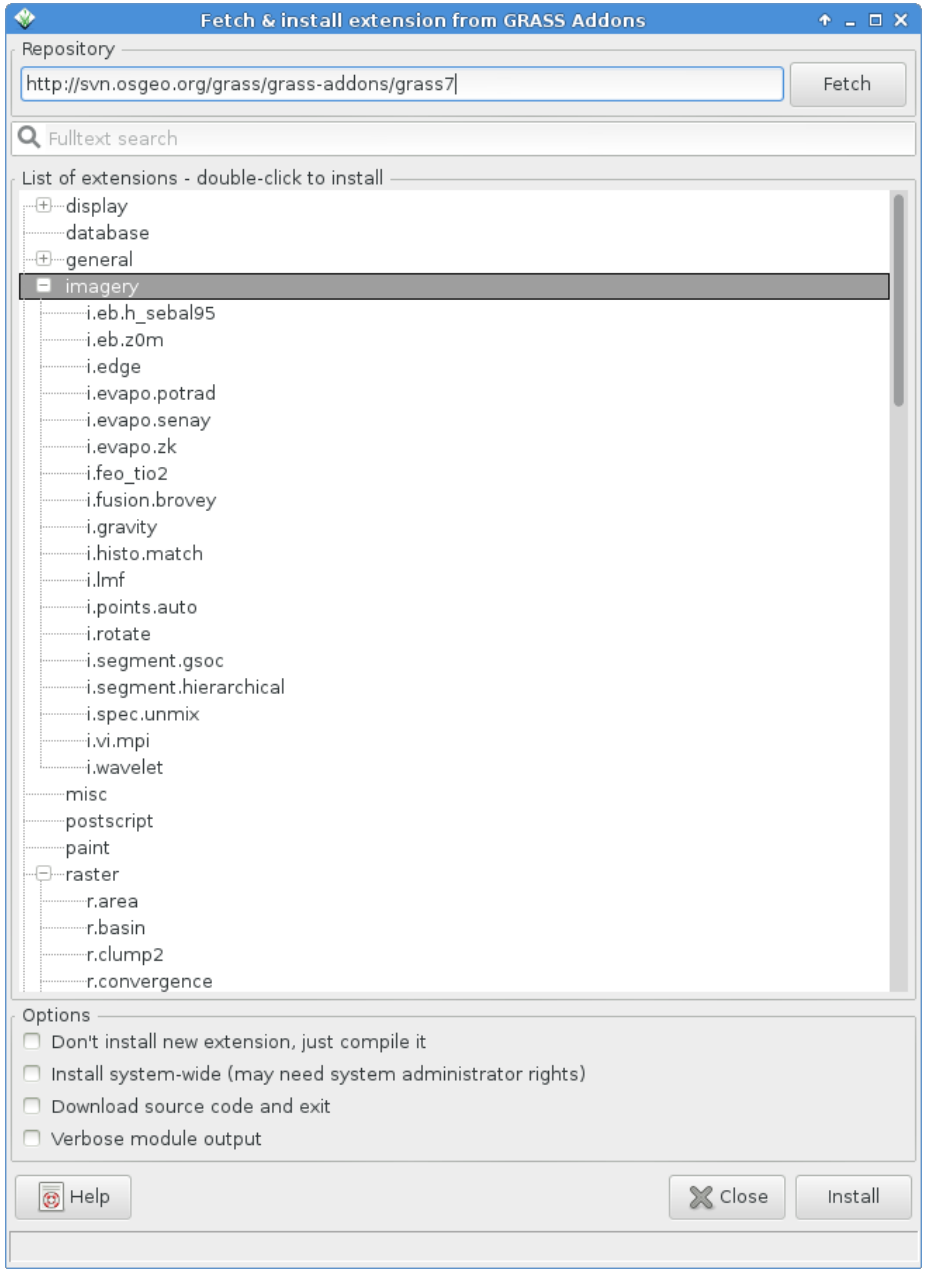
One-click installation with extension manager

Increasing inflow of Python scripts

Users can easily obtain **write** access to develop new functionality

Peer review through SVN commit email list

Also github, gitlab etc. now supported



https://grass.osgeo.org/grass70/manuals/addons/

Where's the stuff?

GRASS GIS 7 Software:

Free download for MS Windows, MacOSX, Linux and source code:

https://grass.osgeo.org/download/

Addons (user contributed extensions):

https://grass.osgeo.org/grass70/manuals/addons/

Free sample data:

Rich data set of North Carolina (NC)

... available as GRASS GIS location and in common GIS formats

https://grass.osgeo.org/download/sample-data/

User Help:

Mailing lists (also in different languages):

https://grass.osgeo.org/support/

Wiki:

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

https://grasswiki.osgeo.org/wiki/R_statistics/rgrass7

Manuals:

https://grass.osgeo.org/documentation/manuals/