

GRASS GIS 7 workshop

GRASS GIS 7 intro

Markus Neteler

OSGeo Ireland – 1st National Symposium 2017

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





Session Objectives

- Database structure of GRASS GIS
- About the course data set
- First steps in using GRASS GIS 7
 - the graphical user interface (GUI)
 - displaying raster and vector maps
 - querying maps
 - adding map elements
 - map swiping with two different Landsat coverages
 - bivariate scatterplots
 - GRASS GIS command structure
 - command line or GUI?
 - Creating a perspective view



GRASS GIS Database concept

GRASS Database

folder with **Locations** ("projects")

Location

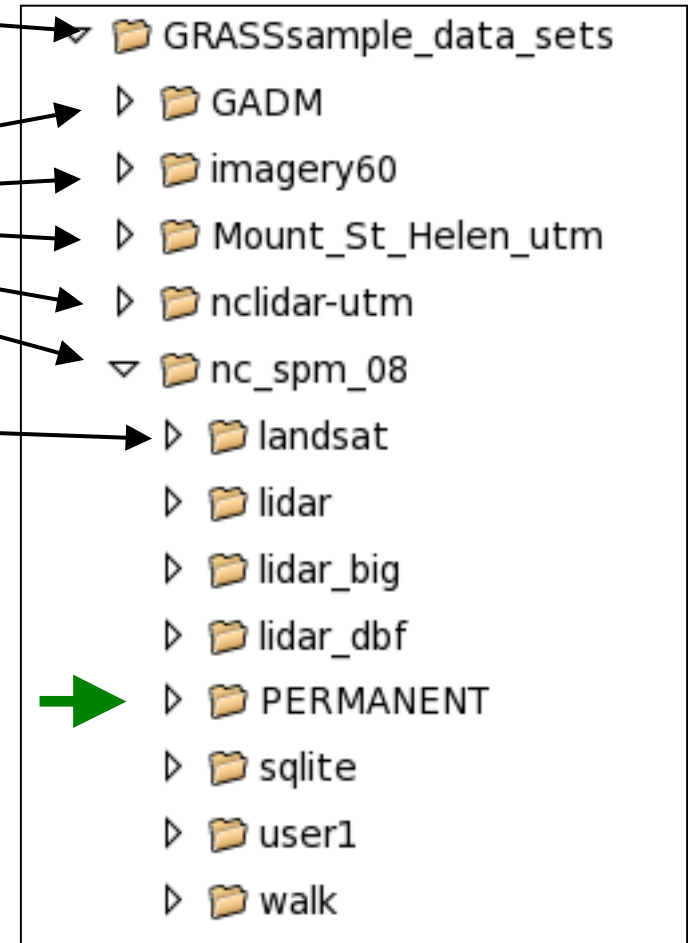
folder with **Mapsets**

Mapset

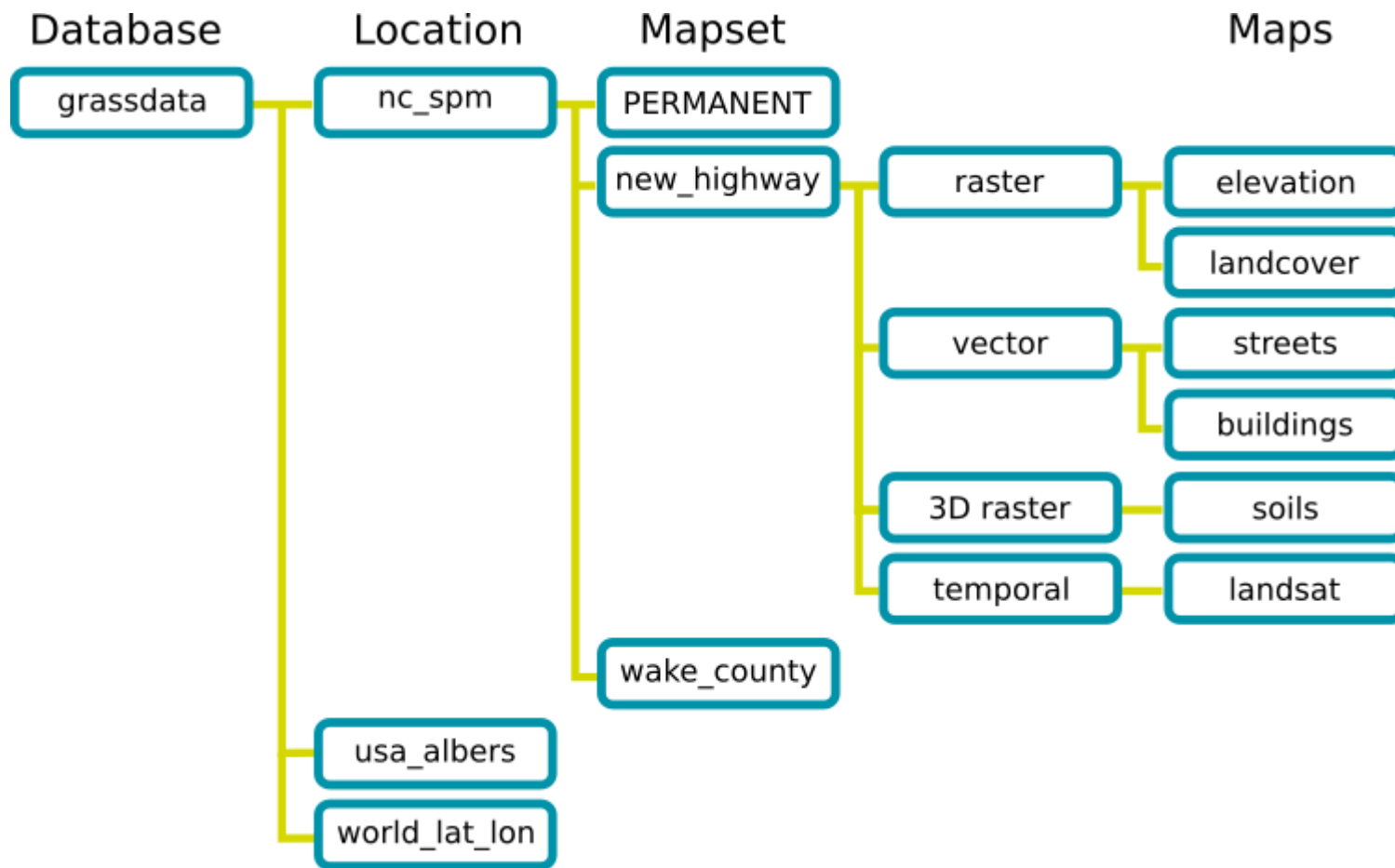
collection of **maps** and **support data**

Each Location...

- is defined by a **projection**
- can hold several mapsets
- has a **PERMANENT mapset** for base cartography



GRASS 7 Database concept: graphical view

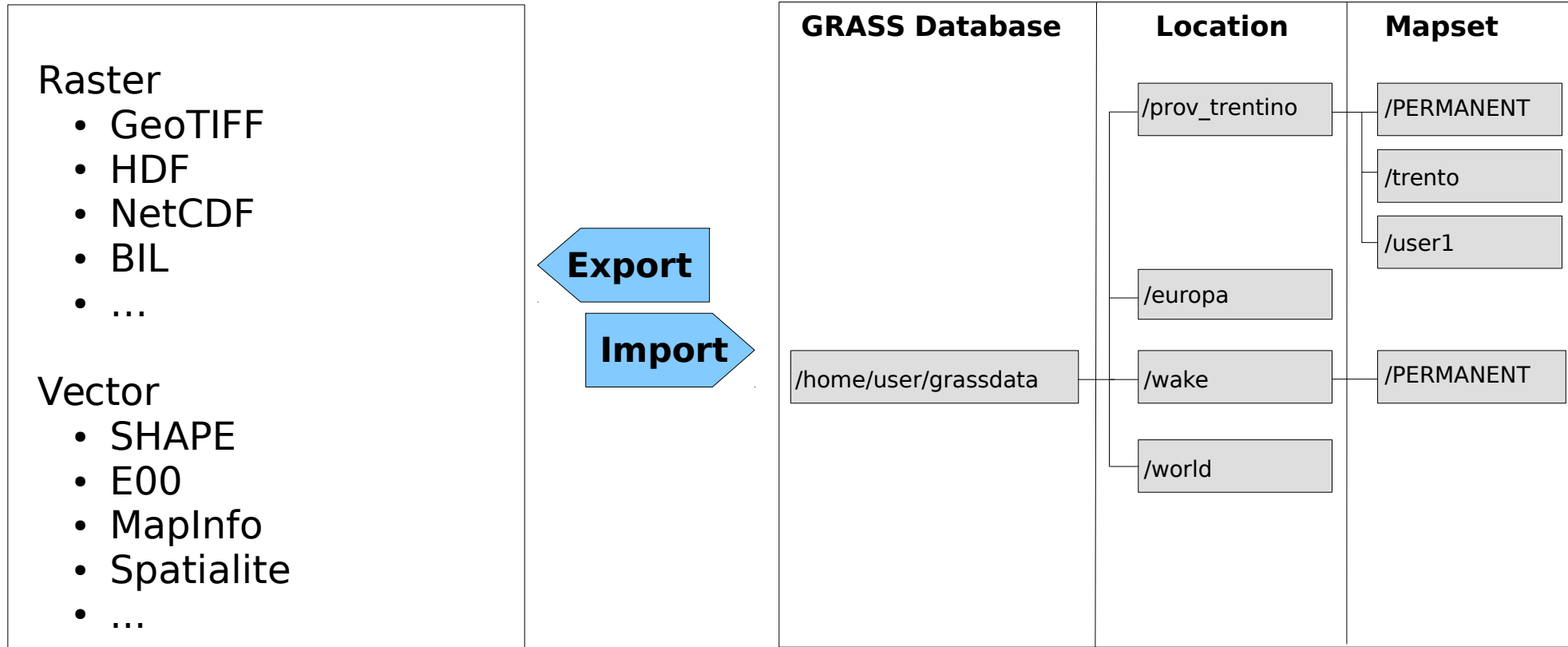


... effectively a tree of subdirectories.

GRASS GIS is a multi-user system when storing “**grassdata**” on a network drive. Users can share “**locations**” and manage therein own “**mapsets**” for the data.

GRASS GIS will do the file management for you!

Standard GIS formats versus GRASS 7 Database



Data in standard GIS formats.

Store in directory:

/home/user/gisdata/

or a shared network directory

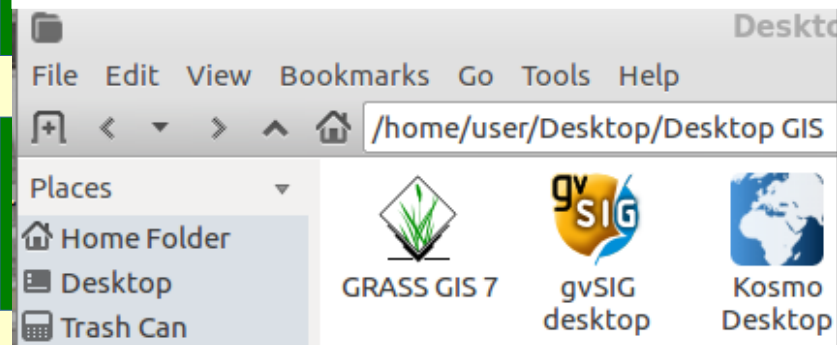
Data in GRASS GIS formats.

Store in directory:

/home/user/grassdata/

or a shared network directory

Exercise – GRASS startup and first steps



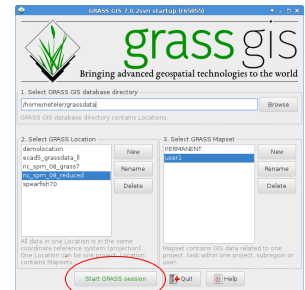
Alternative to GUI: cmd line

```
> grass70 -gui
```



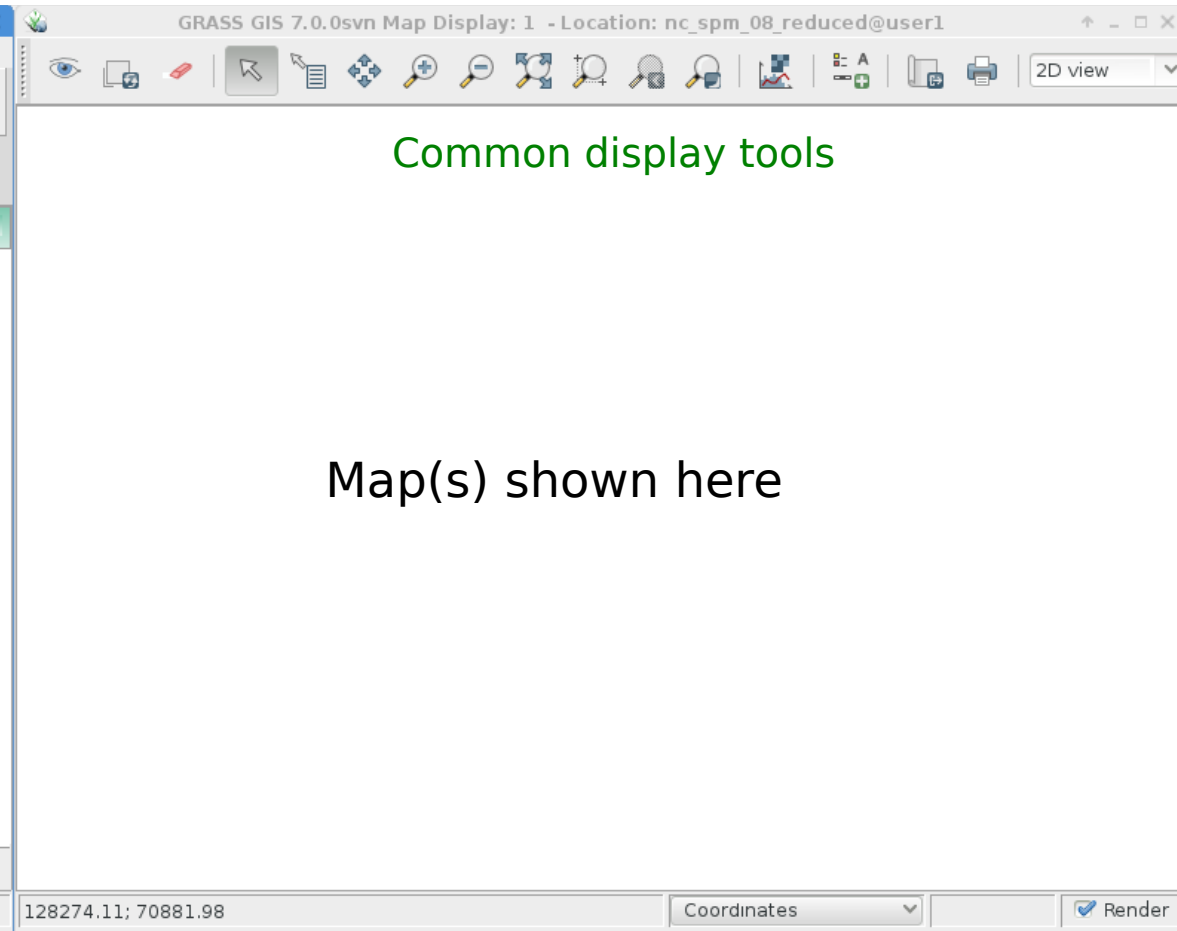
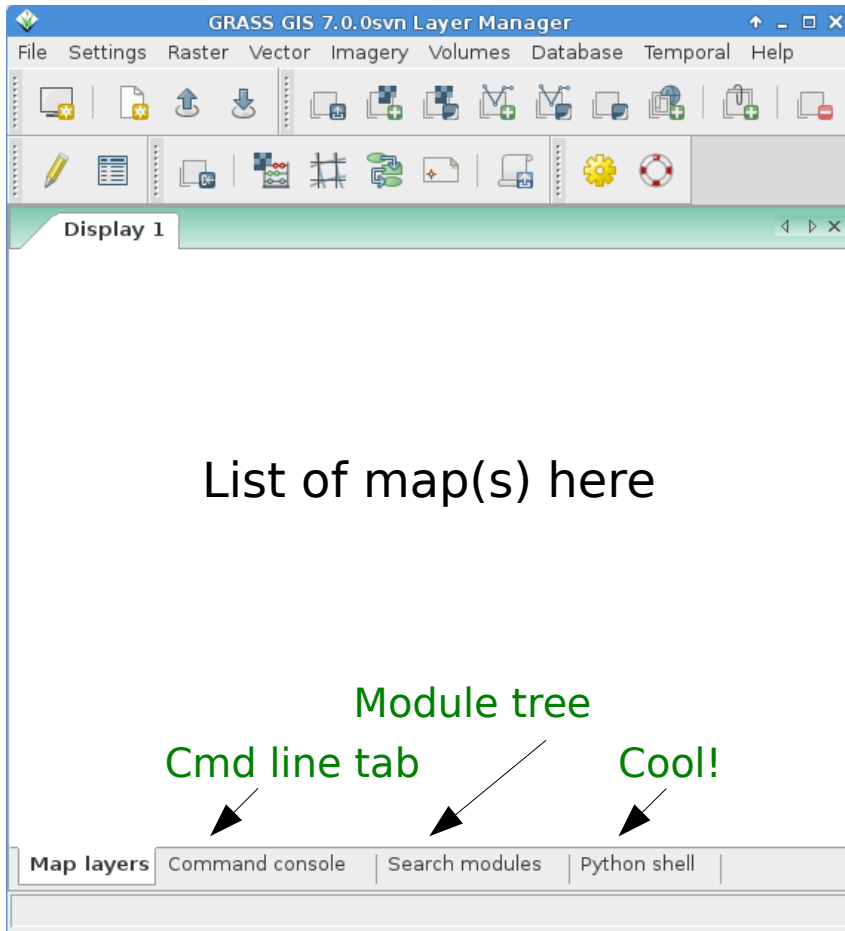
Exercise – GRASS 7 startup and first steps

The graphical user interface at startup:



Main menu

Map display



List of map(s) here

Map(s) shown here

Module tree
Cmd line tab
Cool!

Common display tools



GRASS GIS GUI at full usage

(we'll reach that shortly)

The screenshot displays the GRASS GIS GUI with several windows open. The 'GRASS GIS Layer Manager' window shows a list of layers, including 'roadsmajor@PERMANENT' and 'soils_wake@PERMANENT'. The 'GRASS GIS Map Display: 1 - Location: nc_spm_08' window shows a colorful map with a 5 km scale bar. The 'GRASS GIS Attribute Table Manager - <soils_wake@PERMANENT>' window is the central focus, displaying a table of attribute data for the 'soils_wake' table.

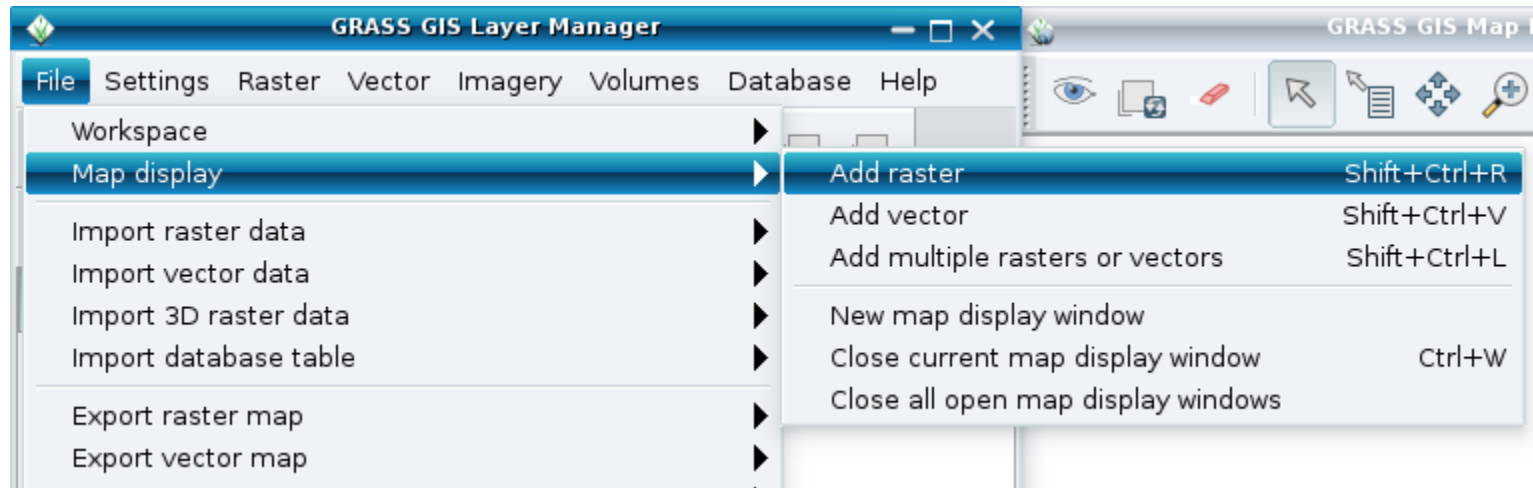
cat	AREA	PERIMETER	SOILS_	SOILS_ID	DSL_NAI	HYDRIC
1	34308200	715717	411	45556	w	
2	24420300	141501	19488	18683	CeB2	
3	7864140	77984.4	20036	46346	CeC2	
4	136123000	494143	20179	46348	CeB2	
5	4255490	36089.4	20209	19401	CeD	
6	4528200	29120.8	20246	46349	CeD	
7	17014600	152098	20256	19447	CeC2	
8	3248240	30345	20357	19548	CeD	
9	4342250	28882.4	20833	46355	ApC2	
10	1885750	22272.4	20966	20149	Cm	B
11	13345800	115666	20970	20153	CeC2	
12	1270500	17500.7	21020	20212	Cm	D

The Attribute Table Manager window also includes a SQL Query section with a 'Simple' query: `SELECT * FROM soils_wake WHERE AREA`. The 'Map layers' window at the bottom shows the command `d.vect -c map=soils_wake@PERMANENT` and the coordinates `634177.17; 225212.69`.



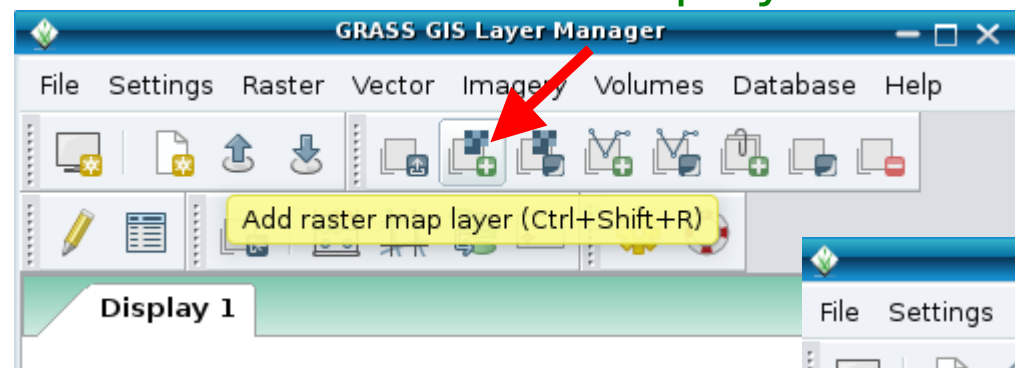
Displaying raster and vector maps

A) Using the menu

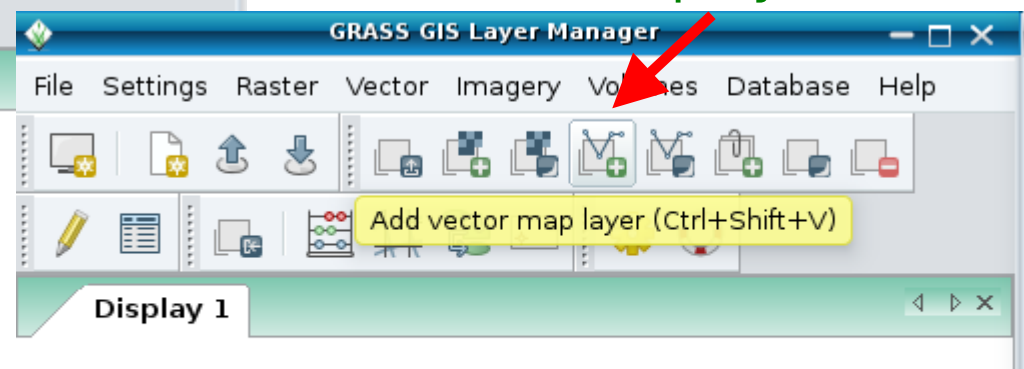


B) Using the icons

Display raster maps



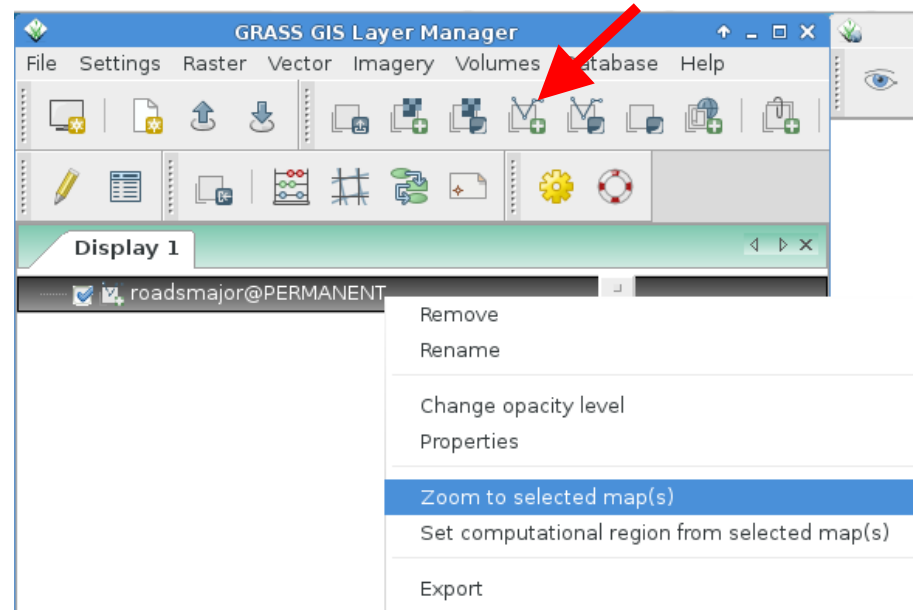
Display vector maps



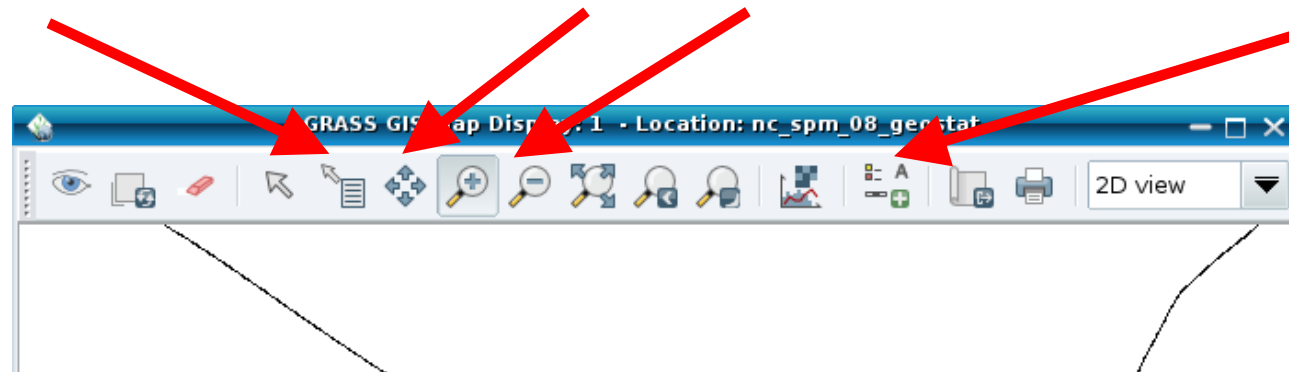


Exercise – Display the “roadsmajor” vector map

- Load the “roadsmajor” vector map into the canvas, zoom to map if needed: **Display vector maps**



- **Query** map elements, **pan**, **zoom** in and out, add a **scale bar**





Exercise – Show vector map attributes

Open the attribute table of the “roadsmajor” vector map by

- ... either *right-mouse* clicking in layer tree on map name
- ... or using the related “Show Attribute table” icon

The screenshot shows two windows from the GRASS GIS interface. The left window is the 'GRASS GIS Layer Manager' with a red arrow pointing to the 'Show attribute data for selected vector map' icon. The right window is the 'GRASS GIS Attribute Table Manager - READONLY - <roadsmajor@PERMANENT>' showing the attribute table for the 'roadsmajor' vector map.

cat	MAJORRDS_	ROAD_NAME	MULTILA	PROPYEAR	OBJECTID	SHAPE_LEN
1	1	NC-50	no	0	1	4825.369405
2	2	NC-50	no	0	2	14392.589058
3	3	NC-98	no	0	3	3212.981242
4	4	NC-50	no	0	4	13391.907552
5	5	NC-98	no	0	5	7196.001495
6	6		no	0	6	10185.513951
7	7	US-1	yes	0	7	13655.438596
8	8		no	0	8	797.901095
9	9	NC-98	no	0	9	14772.176241
10	10	NC-98	no	0	10	8446.822876
11	11	NC-98	no	0	11	14876.323626
12	12	NC-98	no	0	12	11610.268716

SQL Query:

Buttons:

Number of loaded records: 355



Exercise – SQL queries of attributes

Select the **single lane roads** in the “roadsmajor” vector map

- Use “Simple” SQL query
- The selected vectors will be highlighted in the map display

GRASS GIS Attribute Table Manager - <roadsmajor@PERMANENT>

1 / Table roadsmajor

Attribute data - right-click to edit/manage records

cat	MAJORRDS_	ROAD_NAME	MULTILANE	PROPYEAI	OBJECTID	SHAPE_LEN
1	1	NC-50	no	0	1	4825.369405
2	2	NC-50	no	0	2	14392.589058
3	3	NC-98	no	0	3	3212.981242
4	4	NC-50	no	0	4	13391.907552
5	5	NC-98	no	0	5	7196.001495
6	6		no	0	6	10185.513951
8	8					
9	9	NC-98				
10	10	NC-98				
11	11	NC-98				
12	12	NC-98				
13	13					

SQL Query

Simple SELECT * FROM roadsmajor WHERE MULTILANE = 'no'

Advanced SELECT * FROM roadsmajor

Number of loaded records: 115



Exercise – Adding map elements

Using the Wake county “elevation” and “roadsmajor” maps:

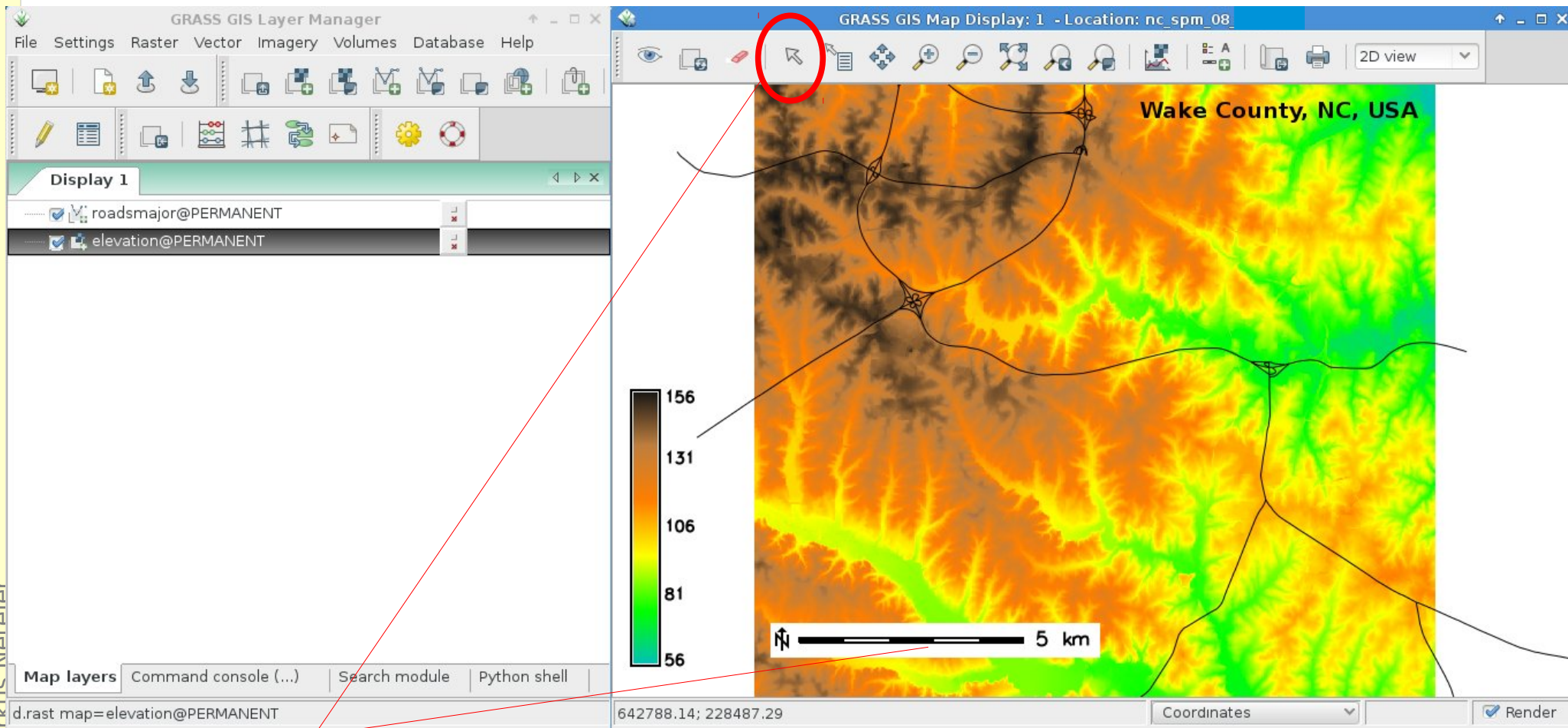
- Add scalebar and north arrow
- Add legend
- Add text layer

The screenshot shows the GRASS GIS interface. On the left is the 'GRASS GIS Layer Manager' window with a red arrow pointing to the 'Layers' tab. It lists two layers: 'roadsmajor@PERMANENT' and 'elevation@PERMANENT'. The main window is 'GRASS GIS Map Display: 1 - Location: nc_spm'. The map shows 'Wake County, NC, USA' with a color-coded elevation map and a network of roads. A vertical color scale on the left indicates elevation values from 56 to 156. A scale bar at the bottom indicates 5 km. The bottom status bar shows coordinates '642788.14; 228487.29' and a 'Render' button.

Exercise – Modify element settings and position



Using the Wake county “elevation” and “roadsmajor” maps:



- Use pointer to
- move map elements
 - edit element settings with a click



Exercise – Map histogram tool

Using the Wake county “elevation” map:

The screenshot shows the GRASS GIS Map Display window with an elevation map. A red arrow points to the histogram tool icon in the toolbar. A context menu is open, listing several options, with "Create histogram of raster map" selected. Below this, the "Select raster map or imagery group to histogram" dialog box is shown. It has two radio buttons: "Histogram single raster" (selected) and "Histogram imagery group". The "Select raster map:" field contains "elevation@PERMANENT". The "Number of bins (for FP maps)" is set to 255, and the "Histogram type" is set to "count".

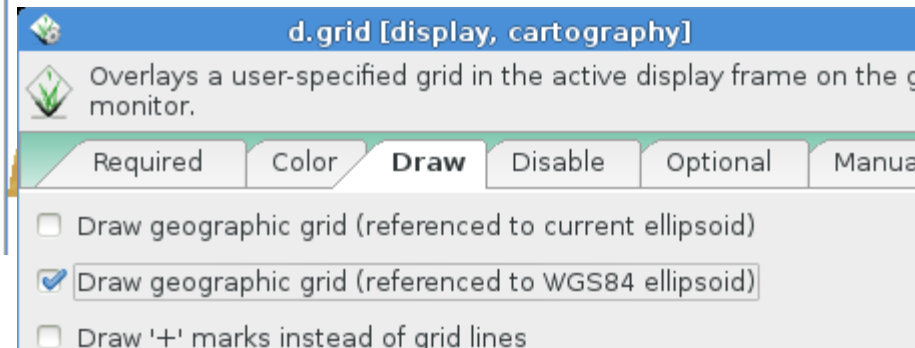
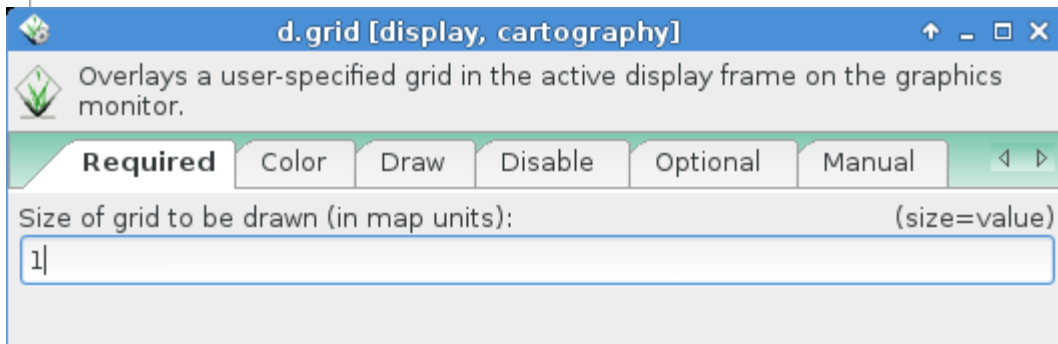
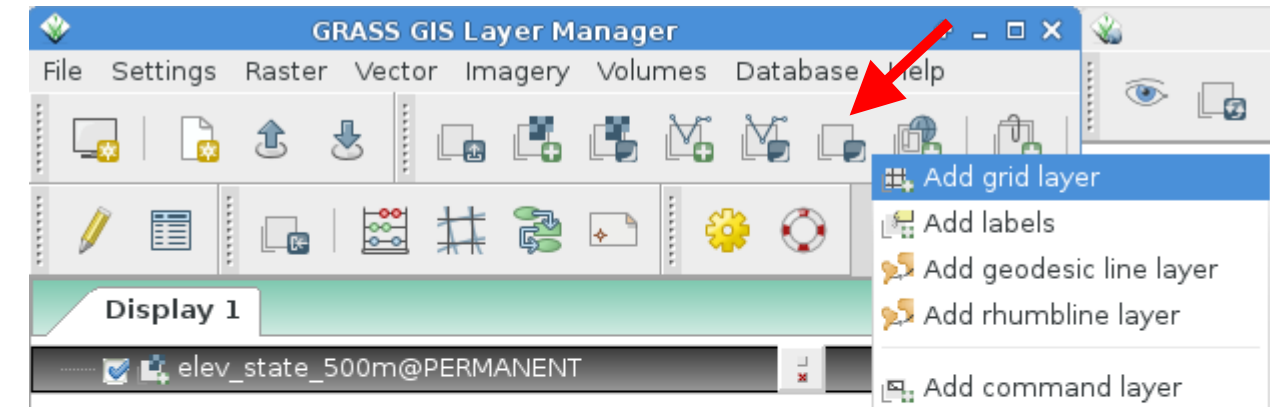
The GRASS Histogramming Tool window displays a histogram titled "Histogram of elevation". The y-axis is labeled "Cell counts" and ranges from 0 to 900. The x-axis is labeled "Raster cell values" and ranges from 60 to 160. The histogram shows a distribution of cell counts across the elevation values, with a peak around 100.

Map will be preselected if selected in Layer Manager



Exercise – Adding a Lat-Long grid to the map

Using the NC state “elev_state_500m” map:

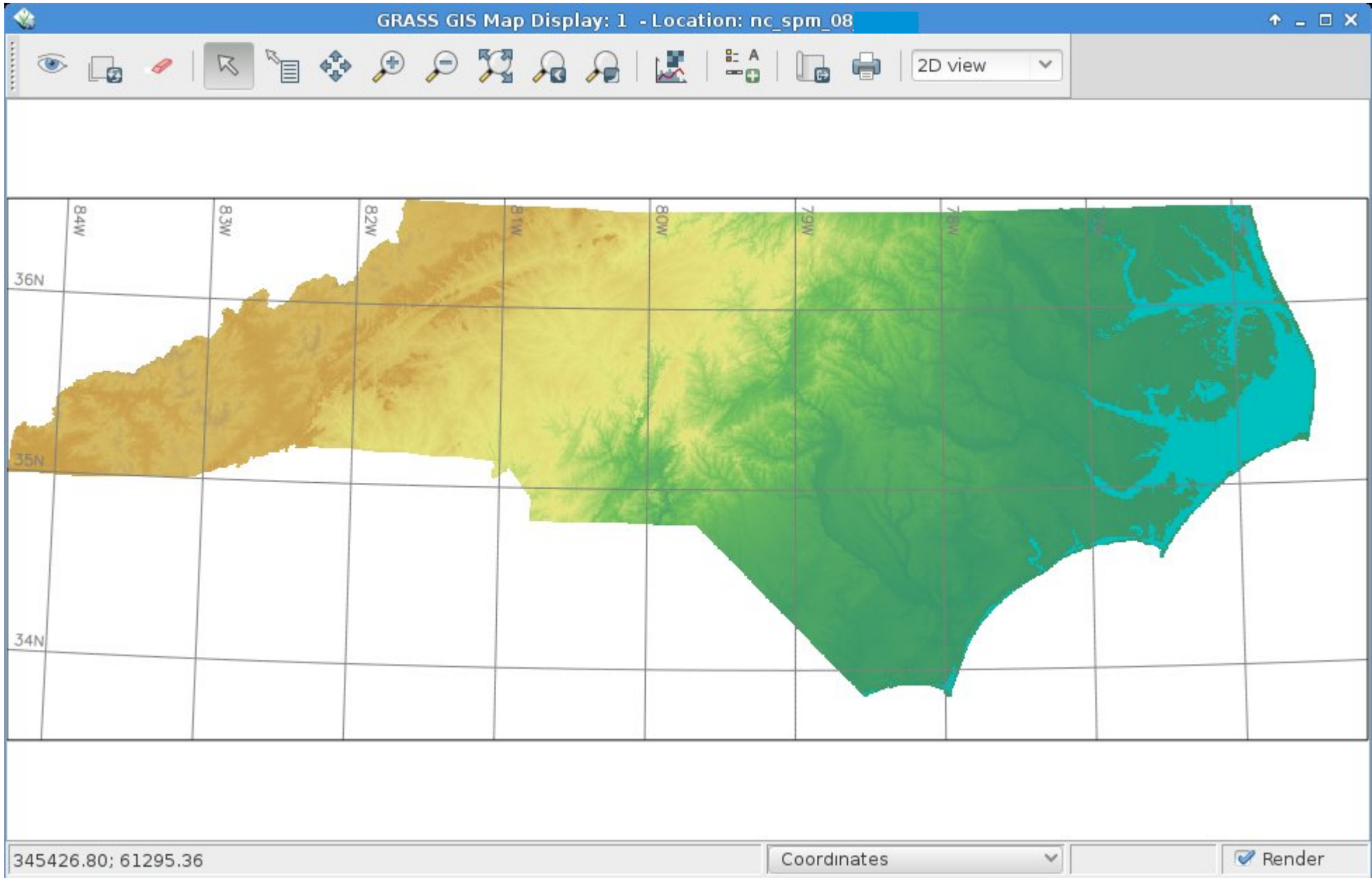


Select a grid size in grid units (here: LL – 1 deg) → Draw grid as LatLong WGS84 grid



Exercise – Adding a Lat-Long grid to the map

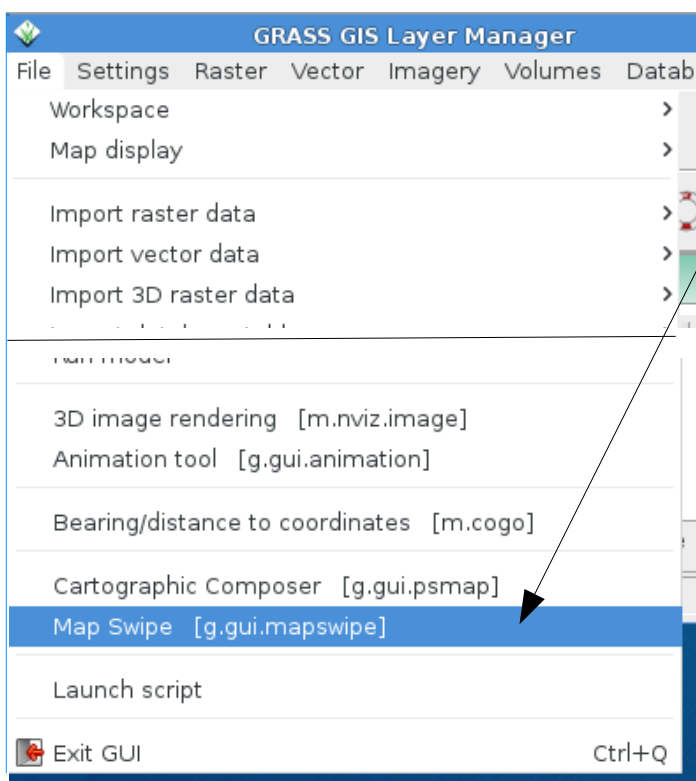
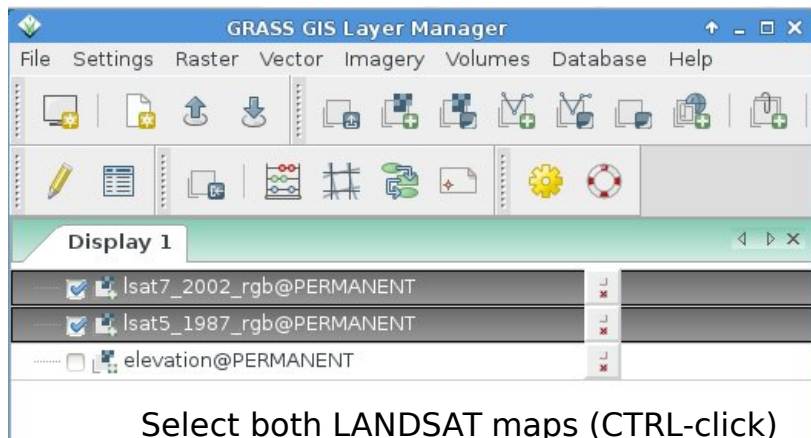
Using the NC state “elev_state_500m” map:



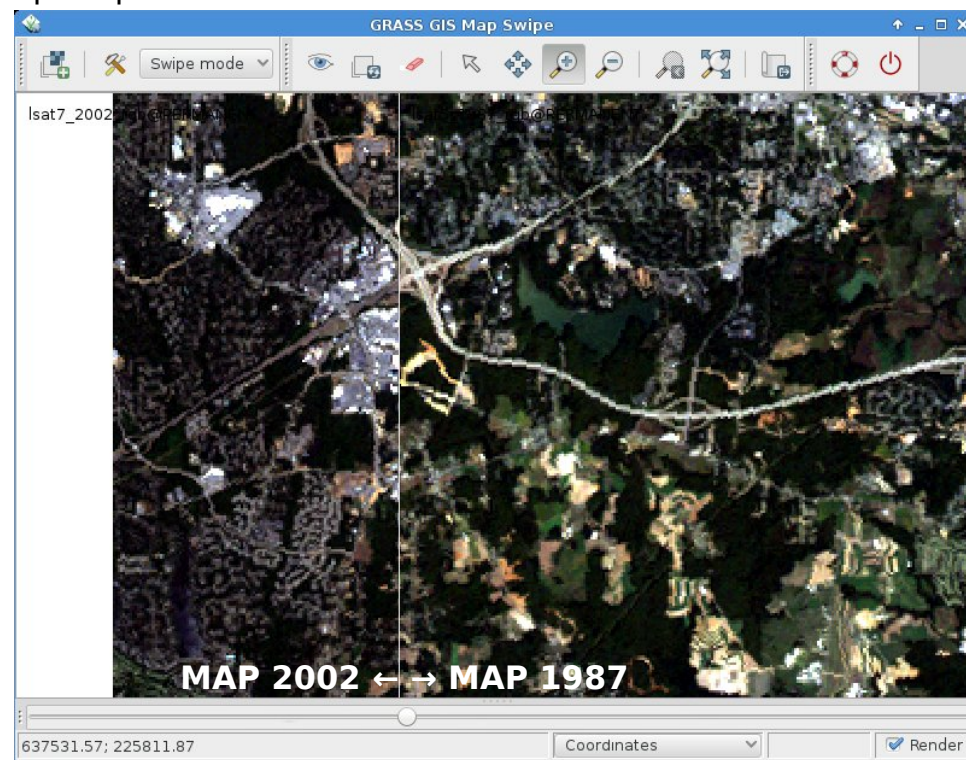


Exercise – Map swiping for multitemporal maps

Load the LANDSAT 5 (1987) and 7 (2002) **RGB composites** of Wake county:



Switch to “map swipe” view





Exercise – Convenient map selector

Load the LANDSAT channels red and green using the convenient map selector

Add selected map layers into layer tree

Map type: raster Select toggle **Toggle all/none**

Mapset: landsat **Select mapset**

Pattern: | **Convenient pattern matching**

List of maps:

- lsat5_1987_10
- lsat5_1987_20
- lsat5_1987_30
- lsat5_1987_40 **Select individual maps**
- lsat5_1987_50
- lsat5_1987_60
- lsat5_1987_70
- lsat5_1987_80
- lsat7_2000_10
- lsat7_2000_20
- lsat7_2000_30
- lsat7_2000_40
- lsat7_2000_50
- lsat7_2000_60
- lsat7_2000_70
- lsat7_2000_80

Use fully-qualified map names

Apply Cancel OK

Add selected map layers into layer tree

Map type: raster Select toggle

Mapset: landsat

Pattern: 2000

List of maps:

- lsat7_2000_10
- lsat7_2000_20
- lsat7_2000_30
- lsat7_2000_40
- lsat7_2000_50
- lsat7_2000_61
- lsat7_2000_70
- lsat7_2000_80

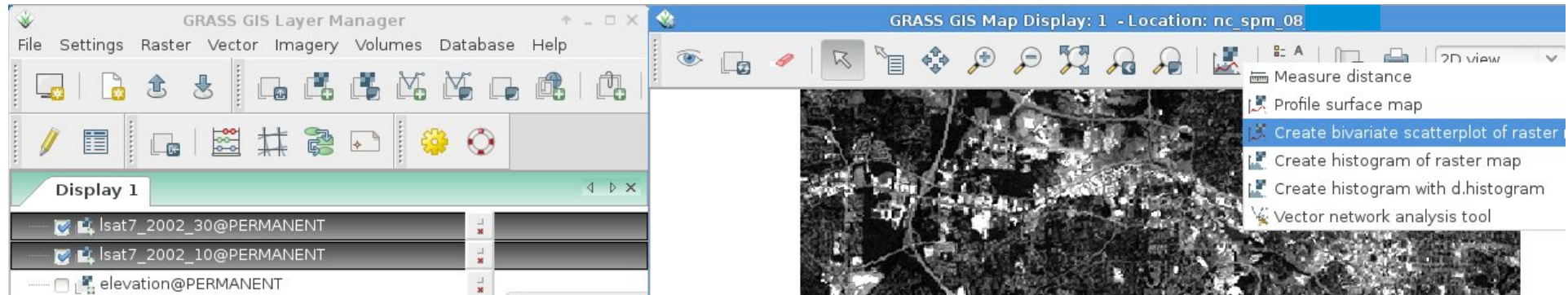
Use fully-qualified map names

Apply Cancel OK

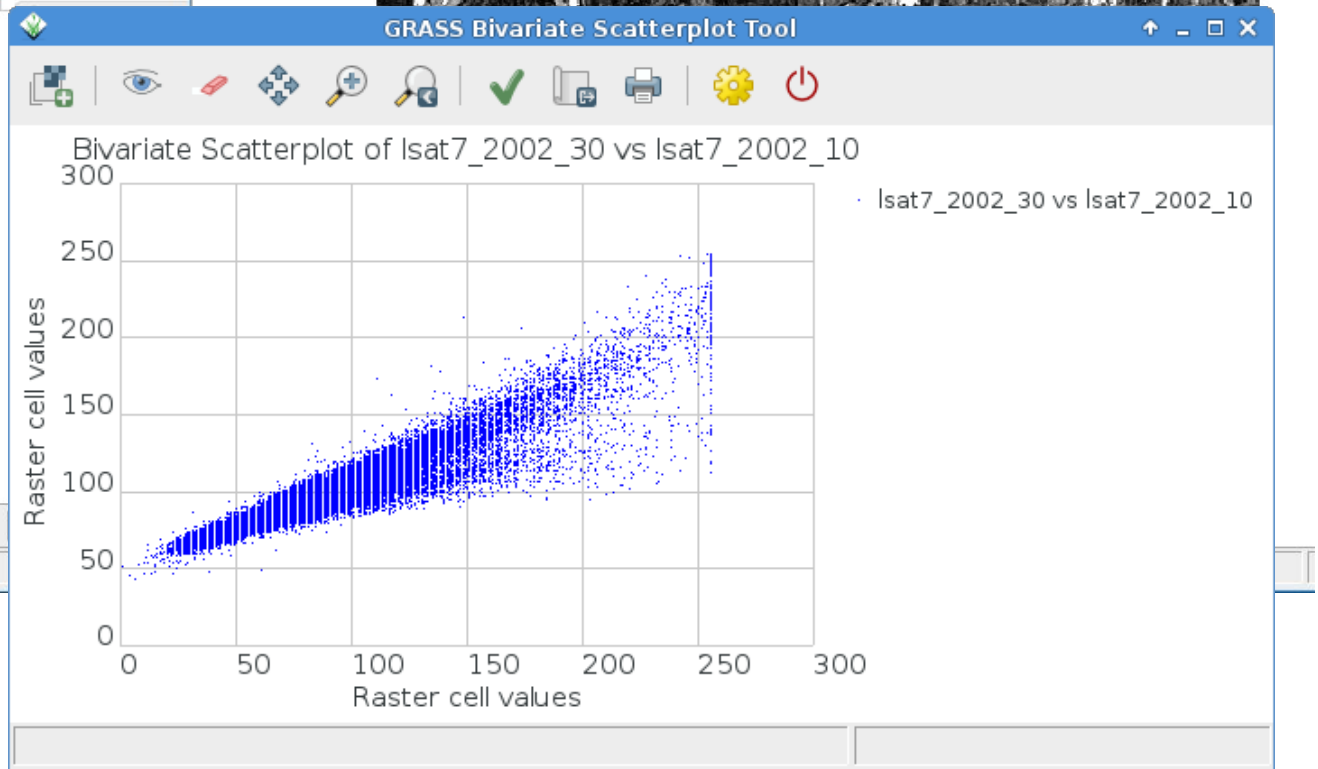


Exercise – Bivariate Scatterplots

Load the LANDSAT 7 2002 **channels 1 and 3** of Wake county.



Select both LANDSAT maps (CTRL-click)





Overview: GRASS GIS command structure

prefix	function class	type of command	example
g.*	general	general data management	<i>g.rename: renames map</i>
d.*	display	graphical output	<i>d.rast: display raster map</i> <i>d.vect: display vector map</i>
r.*	raster	raster processing	<i>r.mapcalc: map algebra</i> <i>r.univar: univariate statistics</i>
v.*	vector	vector processing	<i>v.clean: topological cleaning</i>
i.*	imagery	imagery processing	<i>i.pca: Principal Components Analysis on imagery group</i>
r3.*	voxel	3D raster processing	<i>r3.stats: Voxel statistics</i>
db.*	database	database management	<i>db.select: select value(s) from table</i>
ps.*	postscript	map creation in PostScript format	<i>ps.map: PostScript map creation</i>
t.*	temporal	Space-time cubes	<i>t.rast.aggregate: Raster time series aggregation</i>

Ninja trick: on command line, type the desired prefix (e.g. v.) and then <tab><tab> to complete the command name



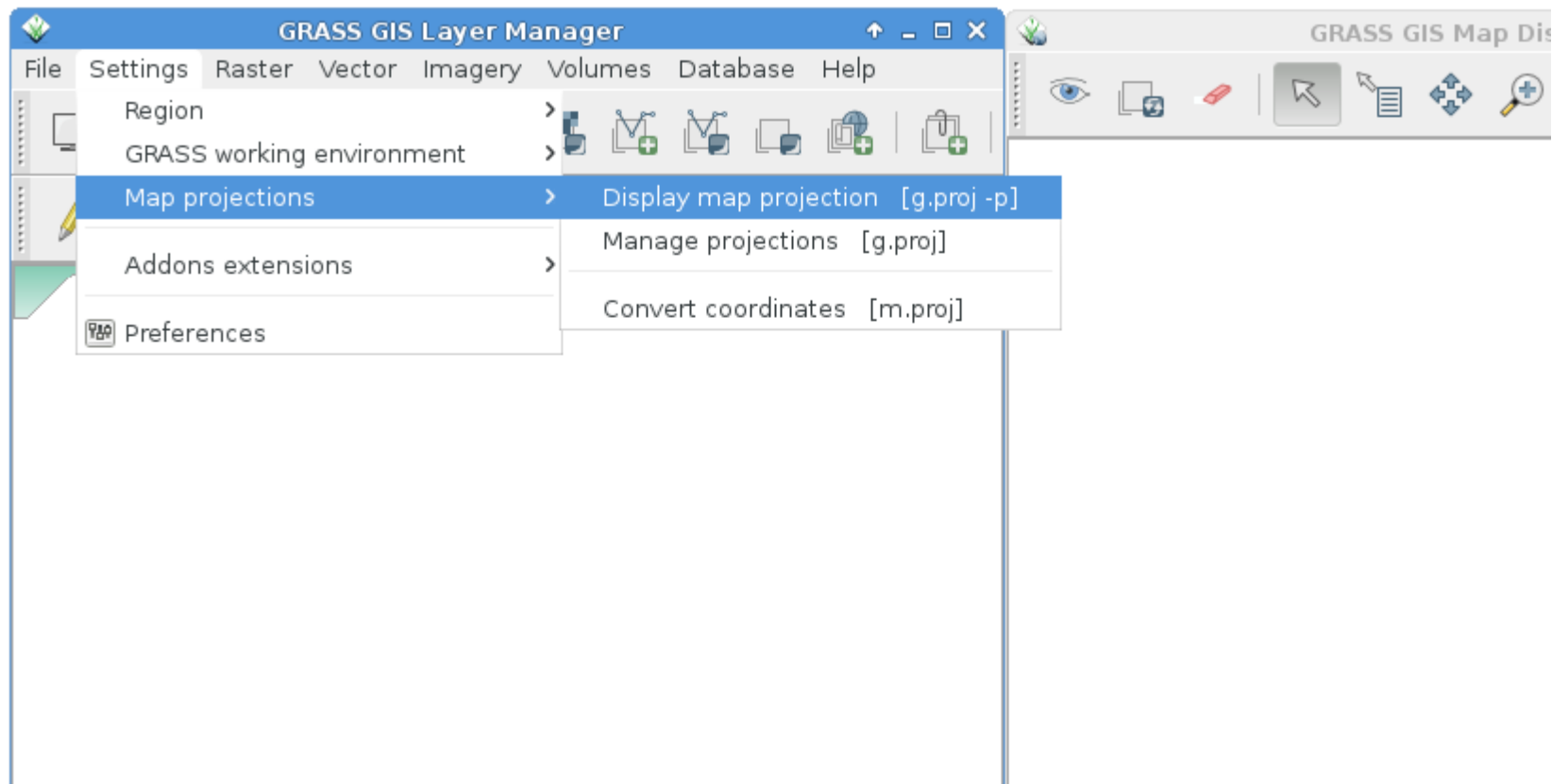
Exercise – GRASS startup and first steps

Know where you are...

get projection information for the North Carolina Location

command: `g.proj`

wxGUI: Settings → Map projections → Manage projections





Exercise – GRASS startup and first steps

Know where you are...

get projection information for the
North Carolina sample data set:

```
name          : Lambert Conformal Conic
proj          : lcc
datum        : nad83
a            : 6378137.0
es          : 0.006694380022900787
lat_1       : 36.166666666666666666
lat_2       : 34.333333333333333334
lat_0       : 33.75
lon_0       : -79
x_0         : 609601.22
y_0         : 0
units       : meters
```

Different formats

```
> g.proj -p
> g.proj -w
> g.proj -t
> g.proj -e
```

GRASS Database concept – Projection



Know where you are...

get projection information for the
North Carolina sample data set:

```
name      : Lambert Conformal Conic      Projection name
proj      : lcc                          projection
datum     : nad83                         geodetic datum
a         : 6378137.0
es        : 0.006694380022900787 } ellipsoid GRS80
lat_1     : 36.166666666666666
lat_2     : 34.333333333333334 } standard parallels
lat_0     : 33.75                         reference latitude
lon_0     : -79                           reference longitude
x_0       : 609601.22                      } lon and lat shifts
y_0       : 0                             (false easting/northing)
units     : meters
```

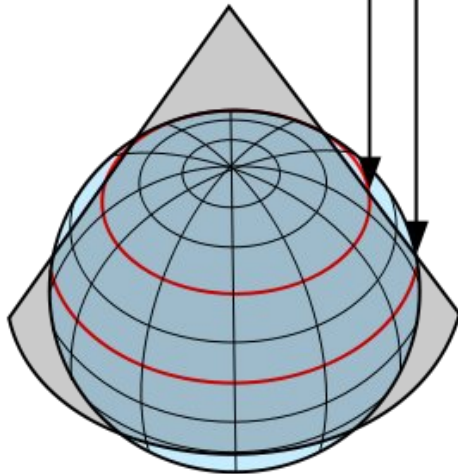



GRASS Database concept – Projection

Projection of the North Carolina Location:

NAD83(HARN) / North Carolina – EPSG code: 3358

Two standard parallels
(selected by mapmaker)



Cone



```
+proj=lcc  
+lat_1=36.166666666666666 +lat_2=34.333333333333334  
+lat_0=33.75 +lon_0=-79  
+x_0=609601.22 +y_0=0  
+ellps=GRS80 +units=m +no_defs
```

Graphical user interface versus Command line



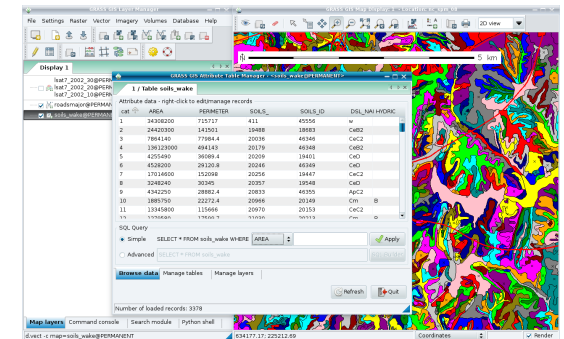
- GRASS GIS offers a graphical user interface
- On command line, there is (text) help:

```
> r.univar --help
```

There are flags (e.g. `-g`) and parameters (e.g. `map=`)

- From command line, you can open the module's GUI (just call the command without parameters)

```
> r.univar
```



```
GRASS 7.0.2svn (nc_spm_08_grass7):~> r.univar --help
Description:
Calculates univariate statistics from the non-null cells of a raster map.
Statistics include number of cells counted, minimum and maximum cell values, range,
arithmetic mean, population variance, standard deviation, coefficient of variation,
and sum.

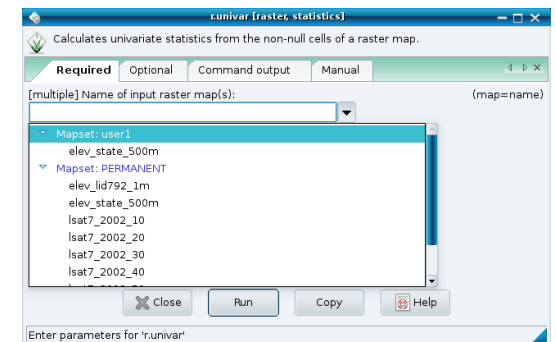
Keywords:
raster, statistics, univariate statistics, zonal statistics

Usage:
r.univar [-get] map=name[,name,...] [zones=name] [output=name]
[percentile=value[,value,...]] [separator=character] [--overwrite]
[--help] [--verbose] [--quiet] [--ui]

Flags:
-g Print the stats in shell script style
-e Calculate extended statistics
-t Table output format instead of standard output format
-o Allow output files to overwrite existing files
-h Print usage summary
-v Verbose module output
-q Quiet module output
--ui Force launching GUI dialog

Parameters:
map Name of raster map(s)
zones Raster map used for zoning, must be of type CELL
output Name for output file (if omitted or "-" output to stdout)
percentile Percentile to calculate (requires extended statistics flag)
options: 0-100
default: 90
separator Field separator
Special characters: pipe, comma, space, tab, newline
default: pipe

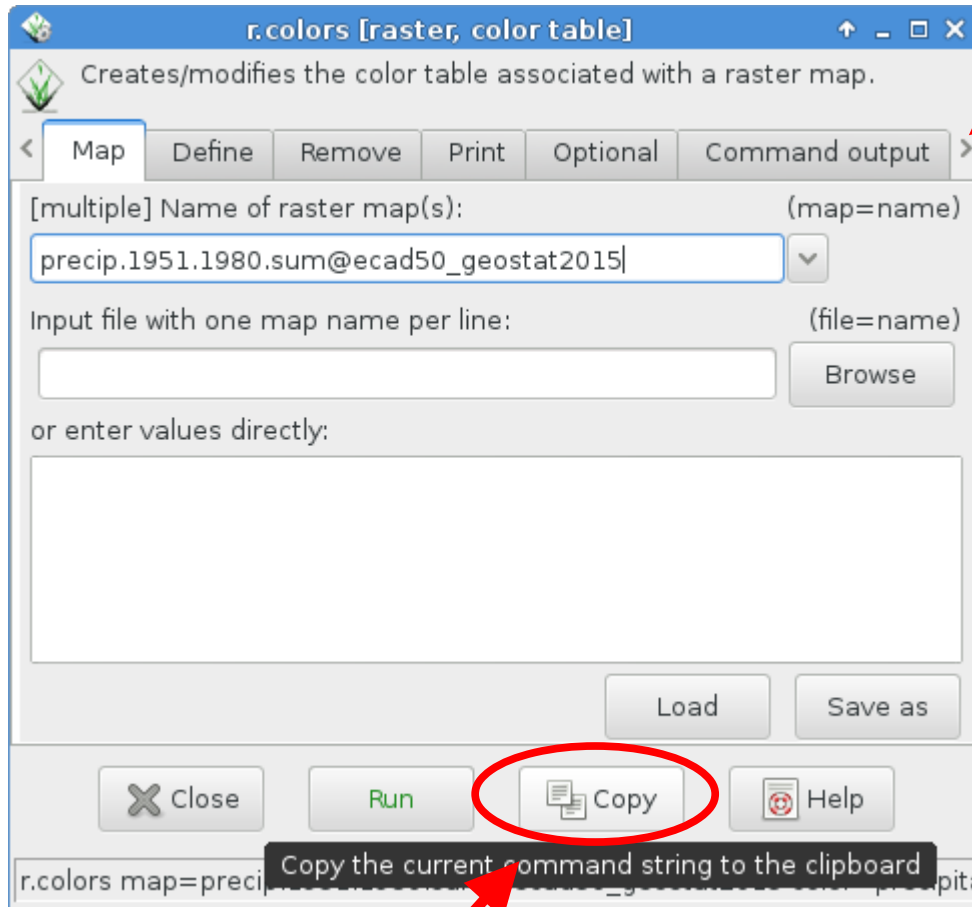
GRASS 7.0.2svn (nc_spm_08_grass7):~> >
```



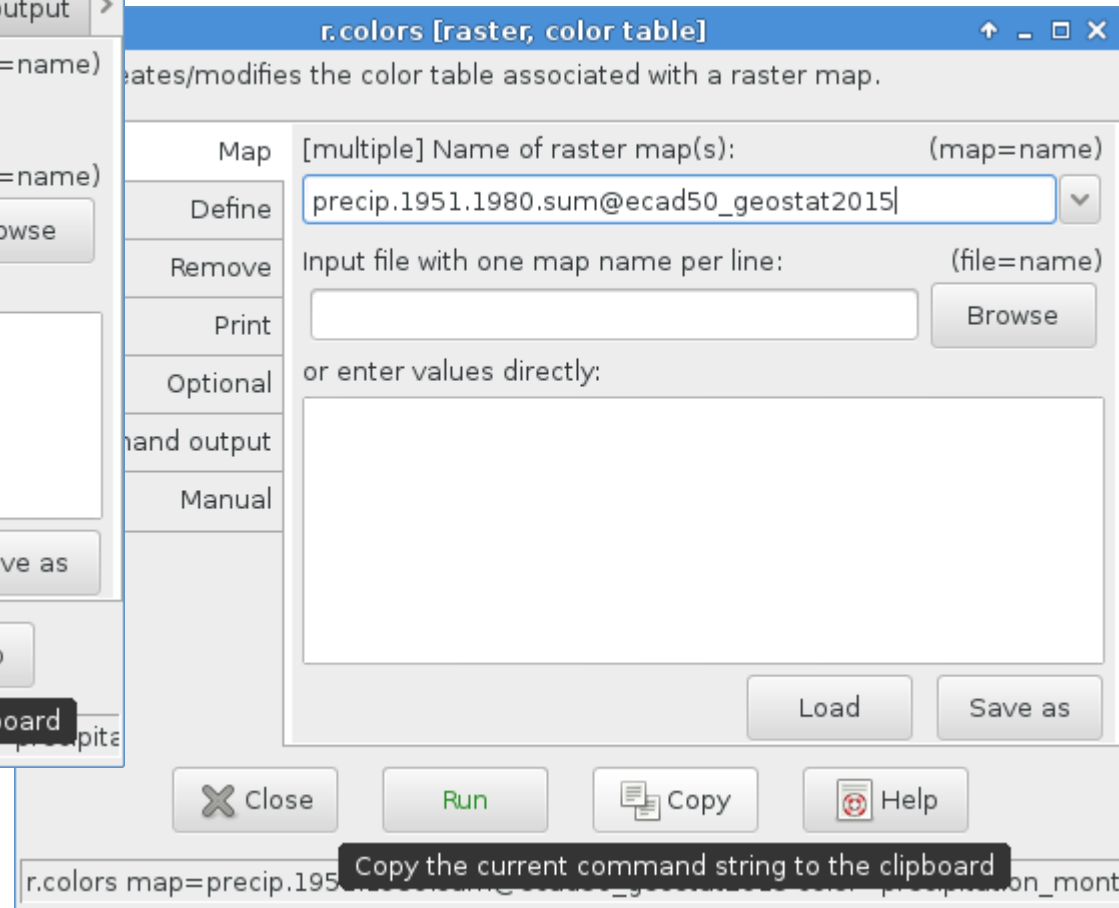


Graphical user interface versus Command line

STYLE: Menu: Settings → GUI Settings → Appearance → Module dialog style: Basic top/left



Cycle through the various tabs



You may copy the command to your documentation

The graphical user interface effectively generates the respective command for the command line (and also writes to the shell “history”)

Command line at its best: 1/2



Advantages of the command line

- Run “history” to see all your previous commands, “forever”
- **History** is stored individually per MAPSET (note that the history of each map is stored within the map's metadata, for this use `r|r3|v.info`)

- **Search** in history with CTRL-R

- **Save it to a file:**

```
history > my_protocol.sh
```

- Note for Windows users:
no history command but
“Command console” tab → “Log file”

```
GRASS 7.0.4 (piemont):~ > v.univar --help

Description:
Calculates univariate statistics of vector map features.
Variance and standard deviation is calculated only for points if specified.

Keywords:
vector, statistics, univariate statistics, attribute table, geometry

Usage:
v.univar [-gewd] map=name [layer=string] [type=string[,string,...]]
[column=name] [where=sql_query] [percentile=value] [--help]
[--verbose] [--quiet] [--ui]

Flags:
-g Print the stats in shell script style
-e Calculate extended statistics
-w Weigh by line length or area size
-d Calculate geometric distances instead of attribute statistics
--h Print usage summary
--v Verbose module output
--q Quiet module output
--ui Force launching GUI dialog

Parameters:
  map      Name of vector map
           Or data source for direct OGR access
  layer    Layer number or name
```



Command line at its best: 2/2

Advantages of the command line

- Polish protocol file, use “# comment” to annotate it
- **Rerun** such a protocol file in a later GRASS GIS session with
`sh my_protocol.sh`

Note: a simplified command line is included in the graphical user interface, tab “Command console”.

It offers a “Command prompt protocol” button.

- These script will work for decades...

```
#!/bin/sh
# MN, 2009, 2013
# convert ECAD gridded data to final Celsius data
# run in
# grass70 /grassdata/latlong/ecad90_climate/

VER=9.0
STARTDATE=19500101
ENDNUM=23191
# FYI:
# date -d '1950-01-01 23191 days' +%Y-%m-%d
# 2013-06-30

# tx_0.25deg_reg_v9.0.nc
IN=tx
VAR=txmax

#####
if [ -z "$GISBASE" ] ; then
    echo "You must be in GRASS GIS to run this program." >&2
    exit 1
fi
export GRASS_OVERWRITE=1
export GRASS_MESSAGE_FORMAT=plain

# import. Band 1 = first day
r.in.gdal -o input=${IN}_0.25deg_reg_v${VER}.nc output=${VAR} memory=6000

# initialization, use YYYY.DOY format
MYDATE=`date -d "$STARTDATE 0 days" +%Y.%j`
# generates: 1950.001

# i is NC layer number, GDAL bands start with 1
for i in `seq 1 $ENDNUM` ; do

    g.region rast=${VAR}.$i
    # save with YYYY.DOY
    r.mapcalc "${VAR}.$MYDATE = if(${VAR}.$i == -9999, null(), ${VAR}.$i / 100.)"
    g.remove --q rast=${VAR}.$i
    r.colors ${VAR}.$MYDATE color=celsius

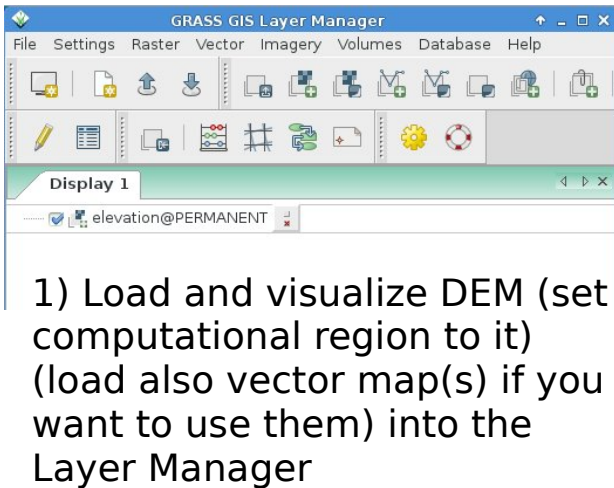
    # careful: $i starts with 0!!
    DAYOFFSET=`expr $i - 1`
    MYDATE=`date -d "$STARTDATE $DAYOFFSET days" +%Y.%j`
done
exit 0
```



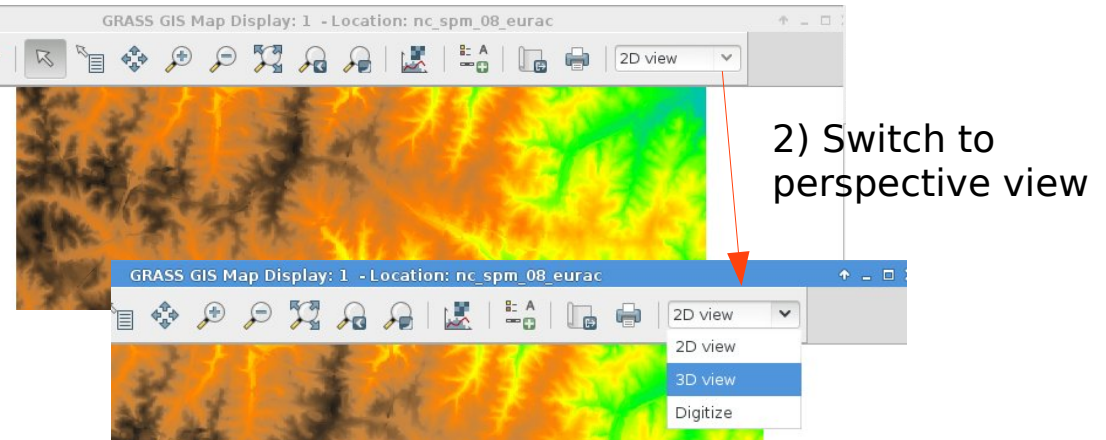
Exercise – Perspective view

Load the Wake county “elevation” map (this tool requires OpenGL support):

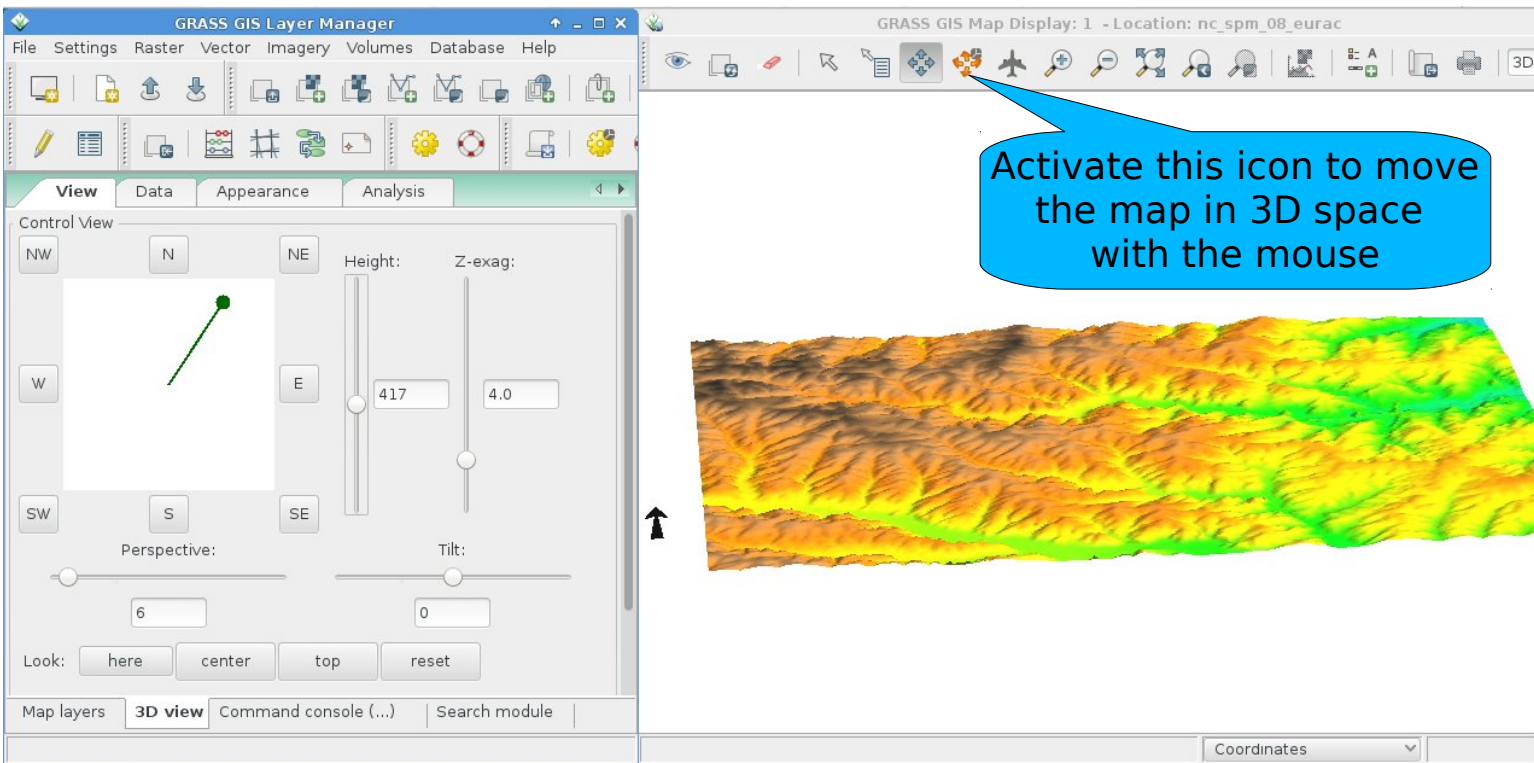
1) Load and visualize DEM (set computational region to it) (load also vector map(s) if you want to use them) into the Layer Manager



2) Switch to perspective view



Activate this icon to move the map in 3D space with the mouse



Eventually switch back to 2D mode...