

Einführung in GRASS

Erste Schritte

- Computer als erstes ganz herunterfahren und auch hinten am Kippschalter ausmachen: damit wird auch das Netzteil ausgeschaltet
- Unter Betriebssystem Linux Computer starten
- Mit Ihrer Kennung und Paßwort einloggen (wie unter Windows)
- Terminal starten: in unterer Menüleiste einmal auf kleinen Bildschirm klicken -> es erscheint ein sogenanntes Terminal, indem alle Kommandos eingegeben werden können.
- Zum Starten von GRASS: im Terminal **grass5** eingeben
- Alle GRASS-Kommandos einer Sitzung werden in diesem Terminal abgesetzt. Sie starten als eigene Prozesse. In diesem Terminal können auch weiterhin UNIX-Befehle ausgeführt werden.
- Zum Beenden von GRASS: **exit**
- LOCATION **spearfish** bezeichnet den GRASS-Beispieldatensatz, mit dem wir trainieren. Ein MAPSET sollte unter Ihrer Kennung angelegt werden. DATABASE ist für alle /usr/local/grassdata.
- Mit ESC und Enter bestätigen...

- GRASS-Monitor starten: `d.mon start=x0`
- GRASS Map darstellen: `d.rast geology`
- Legende im 2. Monitor: `d.mon start=x1`
`d.legend geology - return - 10`
- Arbeiten mit dem Manual: `g.manual`, `list`

Rastern, Klassifizieren, Pufferzonen und Kombinieren

Übersicht, Liste der Datensätze

- Welche Raster- und Vektordaten sind Bestandteil des Beispieldatensatzes spearfish?
 - **g.list type=vect**
 - **g.list type=rast**
- GRASS-Monitor starten **d.mon start=x0**
- Darstellen von Vektordaten am Beispiel der Strassen **d.vect roads**

Umwandlung von Vektordaten in Raster

(Analyse von Datensätzen in GRASS meist in Rastern)

- Region definieren: Größe der Rasterzellen auf 25m einstellen: **g.region**
`1 modify current region directly`
- Umwandlung: **v.to.rast**
 - Name existing vector file: roads
 - Name raster file: roads.ras
 - Number of rows to hold in memory etc.: mit return die voreingestellten Werte übernehmen
- Anzeige der Rastermap im 2. Monitor: **d.mon start=x1, d.rast roads.ras**

Interaktion mit der Map

Anzeige der Informationen, die in der ausgewählten Rasterzelle gespeichert sind. Hier Strassenkategorien **d.what.rast roads.ras**

Berechnen von Pufferzonen

Beispiel: Puffer im Raster *landuse*

- Befehl: **r.buffer**
 - Name input map: landuse
 - Name output map: landuse.buffer
 - Zone 1: 250m Abstand
 - Zone 2: 500m Abstand
- Löschen des Monitorinhaltes: **d.erase**
- Darstellung des Puffers **d.rast -o landuse.buffer**

Studenten:

Beispiel Puffer um Flüsse (Raster *streams*)

- Befehl: **r.buffer**
 - Name input map: streams
 - Name output map: streams.buffer
 - Zone 1: 250m Abstand
- Löschen des Monitorinhaltes: **d.erase**
- Darstellung des Puffers **d.rast -o streams.buffer**

Beispiel: Anwendung von **r.buffer** zur Berechnung von Lärmzonen bei verschiedenen Strassenkategorien (Verwendung von **r.reclass** und **r.buffer**)

- Welche Strassenkategorien sind im Raster *roads.ras* unterschieden? **r.cats roads.ras**
- 1. Isolieren der **Interstates** mit **r.reclass**
 - Name map: roads.ras
 - Name reclassified map: roads.interstate
 - Init: 2 (Initiale Annahme aller Klassen = NULL)
 - Interstate: 1, alle anderen NULL
 - Title: Interstate
- Löschen des Monitorinhaltes: **d.erase**
- Darstellung des Puffers **d.rast -o roads.interstate**
- 2. Pufferzone **500m** um die Interstates mit **r.buffer**
 - Name input map: roads.interstate
 - Name output map: roads500.ras
 - Zone 1: 500m Abstand
- Löschen des Monitorinhaltes: **d.erase**
- Darstellung des Puffers **d.rast -o roads500.ras**

3. Isolieren der **Highways**

r.reclass

- Name map: roads.ras
- Name reclassified map: roads.highway
- Init: 2 (Initiale Annahme aller Klassen = NULL)
- highway (primary & secondary): 1, alle anderen NULL
- Title: Highway

4. Pufferzone **250m** um die Highways

r.buffer

- Name input map: roads.highway
- Name output map: roads250.ras
- Zone 1: 250m Abstand

5. Kombinieren von Kategorien verschiedenen Maps mit

r.combine

am Beispiel der berechneten Lärmzonen: r.combine wird mit einer Kommandosprache gesteuert.

Boolsche Kombinationen (AND, OR, NOT) können auf ausgewählte Kategoriegruppen (GROUP) verschiedener Raster angewendet werden. Es ist vorteilhaft, die Befehle zunächst in einer Datei zu arrangieren.

- Mit dem Texteditor xedit kann eine Datei erzeugt werden:

xedit noise

Die Anweisung wird in einzelne Zeilen zergliedert. Das Einrücken (mit Tabulator) hilft dabei, die Übersicht zu bewahren:

```
(NAME roads.noise
  (OVER roads.ras red
    (OR
      (GROUP 1-2 (roads250.ras))
      (GROUP 1-2 (roads500.ras))
    )
  )
)
```

Bedeutung der Befehle in der Datei noise:

Ein neues Raster **roads.noise** wird aus der transparenten Überlagerung (OVER) des *roads.ras* mit den Zellen der Kategorie 1 (Strasse) und 2 (Pufferzone) (GROUP 1-2) der beiden (OR) Raster *roads250.ras* und *roads500.ras* kombiniert.

- Start mit

r.combine < noise

r.combine liest den Inhalt der Datei noise (Symbol '<')

- Löschen des Monitorinhaltes:

d.erase

- Darstellung des Puffers

d.rast -o roads.noise

- GRASS beenden