

# GRASS 6: Una guía de inicio

-

## GRASS 6 in a nutshell



Por Markus Neteler  
(neteler at itc dot it)

*Material originalmente preparado para:* Open Source Geospatial '05 Conference,  
Junio 16-18 - 2005

University of Minnesota, Minneapolis, MN USA

<http://mapserver.gis.umn.edu/mum/mtg2005.html>

Taller 16 Junio 2005, 8:00 - 12:00

Título : Open Source Geoinformatics with GRASS GIS

Ponentes: Markus Neteler, Kristen Perry

Traducido al español por Guillermo Martínez

<http://www.colnodo.apc.org> (guillermo at colnodo)

Revisada la traducción y actualizado a la versión 6.3 de GRASS por Carlos Dávila  
(carluti at users dot sourceforge dot net)

## Prefacio

En estos momentos, el software libre se ha convertido en un sinónimo de innovación y progreso. Uso libre, modificación y distribución de programas y su código fuente garantizan el libre intercambio de ideas entre usuarios y desarrolladores al igual que un sistema de licencias apropiado. GRASS (por su siglas en inglés, the Geographic Resources Analysis and Support System, <http://grass.itc.it>) es un Sistema de Información Geográfica (SIG) gratuito y de código abierto integrado con subsistemas de procesamiento de imágenes y visualización de datos. Proporciona muchos módulos para el procesamiento de información ráster y vectorial, producción de imágenes en pantalla o en papel, geocodificación y procesamiento de imágenes multiespectral, administración de datos puntuales y de datos en forma general. GRASS cuenta con interfaces para PostgreSQL, MySQL, DBF y bases de datos conectadas con ODBC. Además puede ser conectado a UMN/MapServer, R-stats, gstat, Matlab, Octave, Povray y otros paquetes de software.

Este manual da una introducción compacta a GRASS 6. Nosotros queremos ilustrar las funcionalidades básicas del programa. Para una mayor perspectiva de las capacidades del software, consulte las referencias en la bibliografía. Los conjuntos de datos usados para este taller y este documento (en inglés) están disponibles en <http://mpa.itc.it/markus/mum3/>.

## Índice

<b>1. Introducción</b>	<b>4</b>
1.1. Nuevas características en GRASS 6	4
1.2. Bajando el software	4
1.3. Documentación disponible y datos de ejemplo	4
1.4. Estructura de las bases de datos de GRASS: el “GRASS Project”	4
1.4.1. Crear una base de datos de GRASS	5
1.4.2. Instalar la base de datos de ejemplo Spearfish	5
1.5. Características multi-usuario para equipos de trabajo	6
1.6. Fundamentos de GRASS: estructura de comandos e interfaz de usuario	7
<b>2. Bienvenido a GRASS</b>	<b>10</b>
2.1. Iniciar Linux, login	10
2.2. Introducción a Linux en unos pocos minutos	10
2.3. Sesión de prueba: primeros pasos	11
2.3.1. Iniciar GRASS	11
2.3.2. Iniciar el Administrador GIS gis.m: cargar mapas tipo ráster y vectorial, visualizar mapas	12
2.3.3. Guardar los cambios del Administrador GIS	13
2.3.4. Guardar los cambios en la región actual	13
2.3.5. La herramienta de visualización NVIZ	14
2.3.6. Ayuda en línea: botón de ayuda, -help y g.manual	14

---

2.3.7.	Usar la consola de comandos . . . . .	15
2.3.8.	QGIS: visualizar mapas de GRASS, agregar leyendas, nombres y más . . . . .	16
2.3.9.	QGIS: visualizar información externa de mapas vector y ráster (Shape, GeoTiff, etc) . . . . .	17
2.3.10.	QGIS: visualizar mapas PostGIS . . . . .	17
2.3.11.	Crear mapas en papel con QGIS . . . . .	17
2.3.12.	QGIS: Exportar a un archivo map de MapServer . . . . .	17
2.3.13.	Cerrar la Sesión de GRASS . . . . .	18
<b>3.</b>	<b>Trabajar con su propia información: importar y exportar mapas y crear localizaciones</b>	<b>18</b>
3.1.	Importar información SIG . . . . .	18
3.1.1.	Iniciar GRASS con Spearfish . . . . .	18
3.1.2.	Importar información de archivos Shape de ESRI (TIGER 2000) . . . . .	19
3.1.3.	Importar archivos ráster ERDAS/IMG (LANDSAT-7) . . . . .	20
3.1.4.	Importar archivos ráster GeoTIFF . . . . .	21
3.1.5.	Crear nuevas localizaciones para GRASS a partir de datos existentes . . . . .	21
3.1.6.	Un gran número de formatos de SIG... . . . .	21
3.2.	Crear una nueva localización . . . . .	22
3.2.1.	Definir una nueva localización interactivamente . . . . .	22
3.2.2.	Crear su propia localización a partir de un código EPSG . . . . .	24
<b>4.</b>	<b>Análisis de mapas ráster</b>	<b>24</b>
4.1.	Análisis de modelos de elevación digital . . . . .	25
4.2.	Álgebra de mapas ráster . . . . .	25
4.3.	Georreferenciar un mapa escaneado (4 puntos en las esquinas): mapa topográfico 1:24.000	25
4.4.	Procesamiento y visualización de información de volúmenes . . . . .	26
<b>5.</b>	<b>Procesamiento de imágenes</b>	<b>26</b>
5.1.	Clasificación de imágenes . . . . .	26
5.2.	Fusión de imágenes: Transformada Brovey . . . . .	28
<b>6.</b>	<b>Trabajar con información vectorial</b>	<b>29</b>
6.1.	Importar mapas vectoriales . . . . .	29
6.2.	Administración de atributos . . . . .	30
6.3.	Operaciones de Buffer . . . . .	30
6.4.	Extracciones . . . . .	30
6.5.	Seleccionar, cortar, uniones, intersecciones . . . . .	31
6.6.	Conversiones ráster-vector y viceversa . . . . .	31
6.7.	Digitalización con GRASS . . . . .	32
6.8.	Digitalizar en QGIS . . . . .	32
6.9.	Trabajar con geometrías vectoriales . . . . .	33

---

<b>7. Análisis de redes</b>	<b>33</b>
7.1. Análisis de la ruta más corta . . . . .	34
7.2. Más herramientas de análisis de redes . . . . .	34
<b>8. Interfaz entre GRASS y R-stats</b>	<b>35</b>
8.1. Instalación de los paquetes relacionados con R . . . . .	35
8.2. Sesión de ejemplo de R-stats/GRASS . . . . .	35

## 1. Introducción

### 1.1. Nuevas características en GRASS 6

En los últimos años se ha hecho un número significativo de mejoras a GRASS. Se han añadido un nuevo motor de topologías para información vectorial en 2D y 3D y soporte para análisis de redes vectoriales. Los atributos son ahora administrados en un sistema de base de datos relacional. La herramienta de visualización NVIZ ha sido mejorada para desplegar información vectorial en 3D y volúmenes en vóxeles. Los mensajes se encuentran parcialmente internacionalizados (i18N) con soporte para fuentes Freetype, incluyendo caracteres asiáticos tipo multibyte. GRASS está integrado con las librerías GDAL/OGR para soportar un extenso rango de formatos ráster y vectorial, incluyendo características simples conformes con el OGC (Open Gis Consortium). Un nuevo visualizador y la integración con QGIS (<http://www.qgis.org>) proporcionan una mejora en la falicidad de uso.

### 1.2. Bajando el software

GRASS está disponible desde ITC-irst en Italia (<http://grass.itc.it>) y desde numerosos sitios espejo (Consultar <http://grass.itc.it/mirrors.php>). También se puede obtener en CD-ROM o en distribuciones derivadas de KNOPPIX como "Live Linux GIS".

### 1.3. Documentación disponible y datos de ejemplo

Libros, manuales, cursos en línea y más documentación se encuentra listada en el "Proyecto para la Documentación de GRASS"(<http://grass.itc.it/gdp/>). Información de ejemplo para explorar las funcionalidades del software se encuentra disponible en la página de descargas del proyecto (<http://grass.itc.it/download/>) así como el suplemento en línea (<http://mpa.itc.it/grasstutor/>) del libro publicado por Neteler y Mitasova en 2004.

### 1.4. Estructura de las bases de datos de GRASS: el "GRASS Project"

La información de GRASS se encuentra almacenada en un directorio conocido como *database* (también llamado "GISDBASE" o base de datos) . Este directorio tiene que ser creado con `mkdir` o un administrador de archivos antes de comenzar a trabajar con GRASS. Dentro de esta *base de datos*, los

proyectos se organizan por áreas de proyectos en subdirectorios llamados *localizaciones* (locations en inglés). Una *localización* esta definida por su sistema de coordenadas, proyección cartográfica y límites geográficos. Los subdirectorios y archivos que definen una *localización* se crean automáticamente cuando se inicia GRASS la primera vez con una nueva *localización*.

Cada *localización* puede tener múltiples *directorios de mapas* (mapsets en inglés). Un motivo para mantener múltiples *directorios de mapas* es almacenar mapas relacionados con algún aspecto específico del proyecto o subregiones. Otro motivo es permitir el acceso simultáneo de varios usuarios a capas almacenadas dentro de la misma *localización*, por ejemplo, equipos trabajando en el mismo proyecto. Para trabajos en equipo, se crearía una base de datos de GRASS centralizada en un sistema de archivos de red (por ejemplo, NFS). Además de acceder el propio *directorio de mapas*, cada usuario puede leer capas en los *directorios de mapas* de otros usuarios, aunque cada uno sólo puede modificar o borrar las capas en su propio *directorio de mapas*.

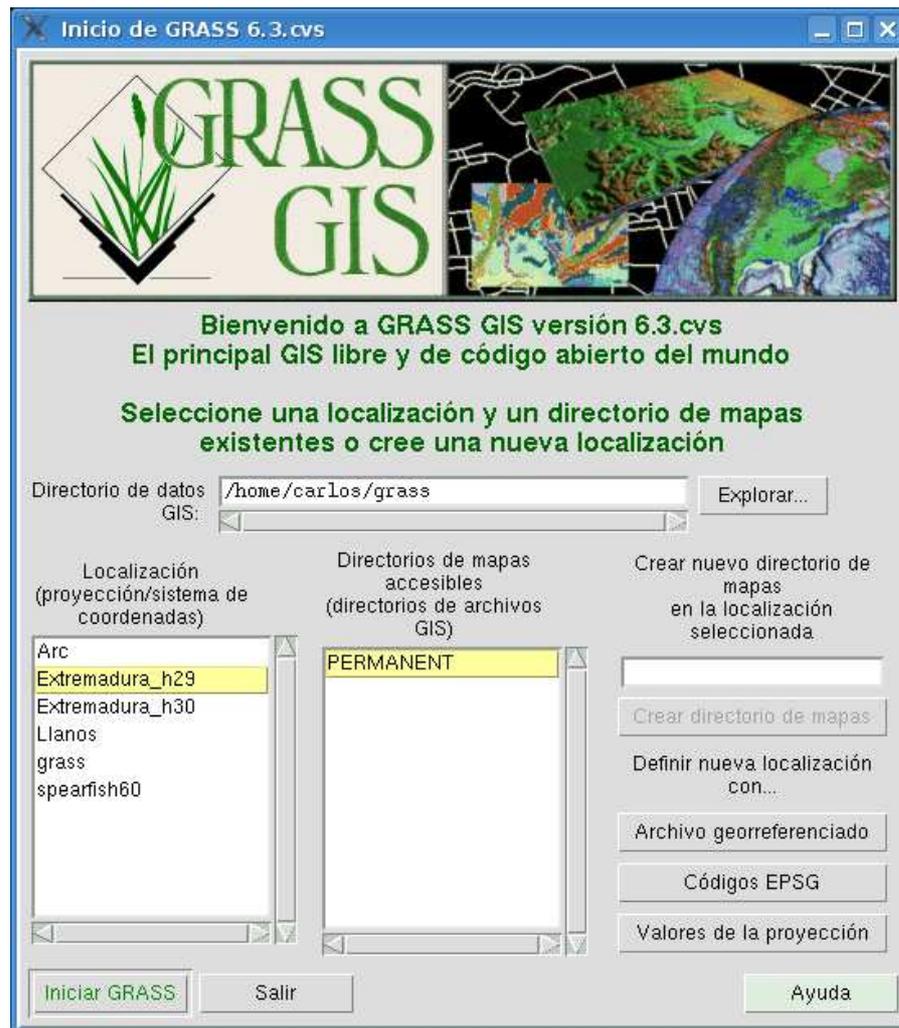
Cuando se crea una nueva *localización*, GRASS automáticamente crea un *directorio de mapas* especial llamado PERMANENT donde se puede almacenar la información principal del proyecto. Los datos en el *directorio de mapas* PERMANENT sólo pueden ser añadidos, modificados o borrados por el propietario del *directorio de mapas* PERMANENT, sin embargo, los datos pueden ser consultados, analizados y copiados a sus respectivos *directorios de mapas* por otros usuarios. El *directorio de mapas* PERMANENT es útil para proporcionar información general (ejemplo, un modelo de elevación digital), accesible pero protegida contra escritura de todos los usuarios que se encuentran trabajando en la misma *localización* del propietario de la base de datos. Para manipular o agregar datos al PERMANENT, el propietario debe iniciar GRASS y escoger la *localización* y el directorio de mapas PERMANENT. Este *directorio de mapas* también contiene el archivo DEFAULT\_WIND, el cual almacena las coordenadas de los límites de la región por defecto para la *localización* (los cuales todos los usuarios heredarán cuando inicien la base de datos). Además, un archivo WIND es mantenido en todos los *directorios de mapas* para almacenar los límites actuales y la resolución ráster seleccionada actualmente. Los usuarios tienen la opción de regresar a la región por defecto en cualquier momento.

### 1.4.1. Crear una base de datos de GRASS

Para crear una nueva base de datos de GRASS, busque un directorio donde tenga permiso de escritura. La partición del disco debe tener suficiente espacio libre para almacenar su información espacial. Cree un directorio que represente la base de datos de GRASS, (ejemplo, `mkdir /data/grassdata/` o `mkdir /home/sucuenta/grassdata/`). Esta es la ruta que debe ser colocada en la línea "Directorio de datos GIS"(Database) de la pantalla de inicio (vea la figura 1).

### 1.4.2. Instalar la base de datos de ejemplo Spearfish

Existen un par de *localizaciones* de ejemplo para GRASS disponibles para bajar de la red. En nuestro taller usaremos la *localización* *Spearfish* (<http://grass.itc.it/download/data6.php>), la cual se



**Figura 1:** Pantalla de Inicio de GRASS

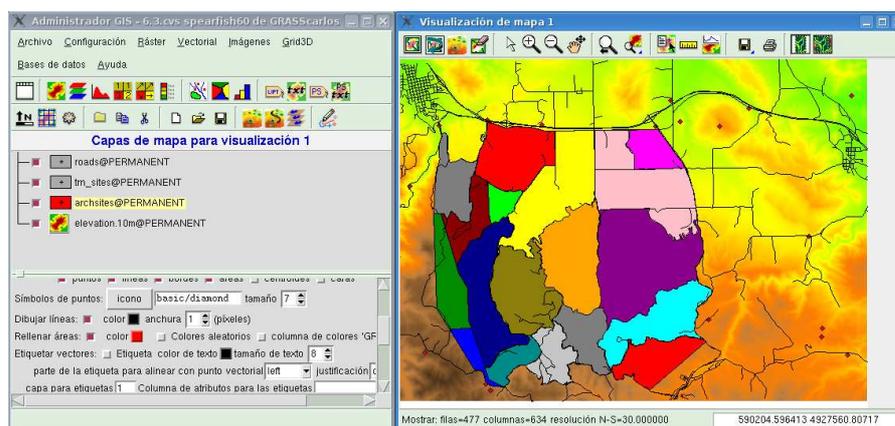
debe descomprimir en el nuevo directorio para la base de datos. Ejemplo:

```
cd /data/grassdata/ ó cd /home/sucuenta/grassdata/
tar -xvzf spearfish_grass60data.tar.gz
```

Una vez hecho esto, los datos se encuentran listos para ser utilizados. Antes de empezar, daremos una vista rápida de la estructura general del programa.

### 1.5. Características multi-usuario para equipos de trabajo

GRASS permite a cualquier número de usuarios trabajar en una misma *localización* (pero con diferentes directorios de mapas) simultáneamente. Los usuarios únicamente pueden leer los mapas de otros *directorios de mapas* si reciben los permisos, pero nunca pueden escribir en otras *localizaciones*. Los mapas



**Figura 2:** El Administrador GIS de GRASS y Visualización de mapa

para todos los miembros del grupo deben ser almacenados en el directorio de mapas *PERMANENT*. Este sencillo esquema permite una fácil administración de grandes proyectos de SIG.

### 1.6. Fundamentos de GRASS: estructura de comandos e interfaz de usuario

GRASS es un software para SIG completo, híbrido y estructurado modularmente con funciones ráster y vectoriales. Cada función SIG es administrada por su propio módulo. De esta forma el sistema es claramente estructurado y actúa en forma transparente. Otra ventaja de su modularidad es que únicamente son ejecutados los módulos necesarios, lo cual preserva los recursos del sistema.

Actualmente hay disponibles tres interfaces gráficas de usuario (con siglas en inglés, GUI) de manera adicional a la tradicional consola de comandos. La GUI por defecto es el Administrador GIS (Gis Manager). El visor de mapas NVIZ incluye soporte para información ráster y vectorial, visualización de volúmenes, animaciones, perfiles y más (vea las figuras 2 y 3).

Un proyecto externo, el visor de datos geográficos amigable con el usuario QGIS, proporciona soporte directo para GRASS. Desde la versión 0.7 de QGIS en adelante esta incluida una extensiva interfaz de GRASS con un digitalizador en pantalla y funcionalidades de SIG. También está incluida una nueva herramienta para la producción de mapas en papel (vea la figura 4).

Finalmente, existe JAVAGRASS (JGRASS), que es una herramienta multiplataforma y multisesión (vea la figura 5). JGRASS prepara a GRASS para ser usado en ambientes de producción, de forma opuesta a los ambientes de investigación. La arquitectura de JGRASS sigue internamente un modelo cliente-servidor, separando la Interfaz Gráfica de Usuario del motor de procesamiento espacial. Esta separación permite el fácil desarrollo de mecanismos de acceso remoto.

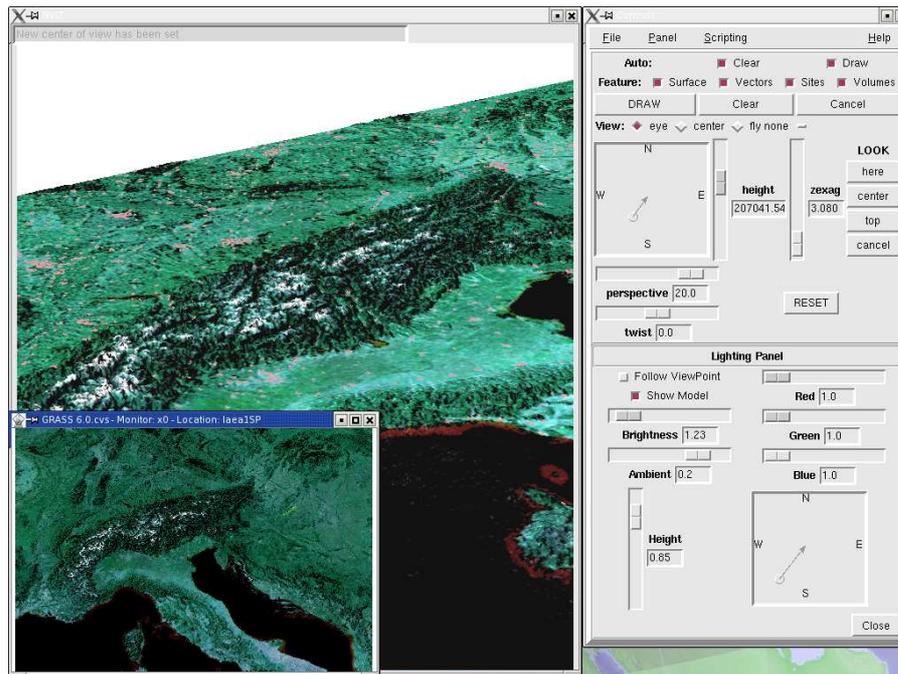


Figura 3: El visualizador NVIZ con imágenes de satélite

<i>Sumario de comandos en GRASS</i>			
<i>prefijo</i>	<i>clase de función</i>	<i>tipo de comando</i>	<i>ejemplo</i>
d.*	visualización (display)	despliegue gráfico	d.rast: muestra un mapa ráster d.vect: muestra un mapa vectorial
db.*	bases de datos  (data base)	administración de la base de datos	db.select: selecciona valores de una tabla
g.*	general	operaciones generales con los archivos	g.rename: renombra un mapa
i.*	imágenes	procesamiento de imágenes	i.smap: clasificador de imágenes
ps.*	postscript	creación de mapas en formato Postscript	ps.map: creación de un mapa
r.*	ráster	procesamiento de datos ráster	r.buffer: crea una región alrededor de elementos ráster
r3.*	vóxel	procesamiento de datos en vóxeles	r3.mapcalc: álgebra de mapas de volumen
v.*	vectorial	procesamiento de información vectorial	v.overlay: intersección de mapas vectoriales

Ayuda en línea con un navegador HTML: `g.manual <comando>&`

Ayuda en línea en formato MAN: `g.manual -m <comando>`

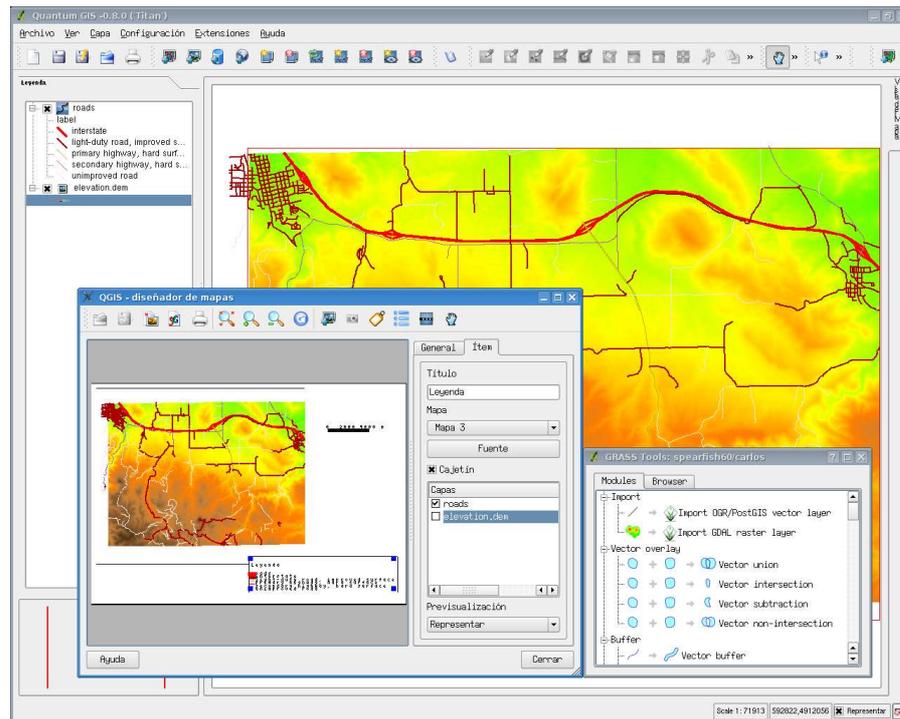


Figura 4: El visor de información geográfica QGIS con la interfaz de GRASS

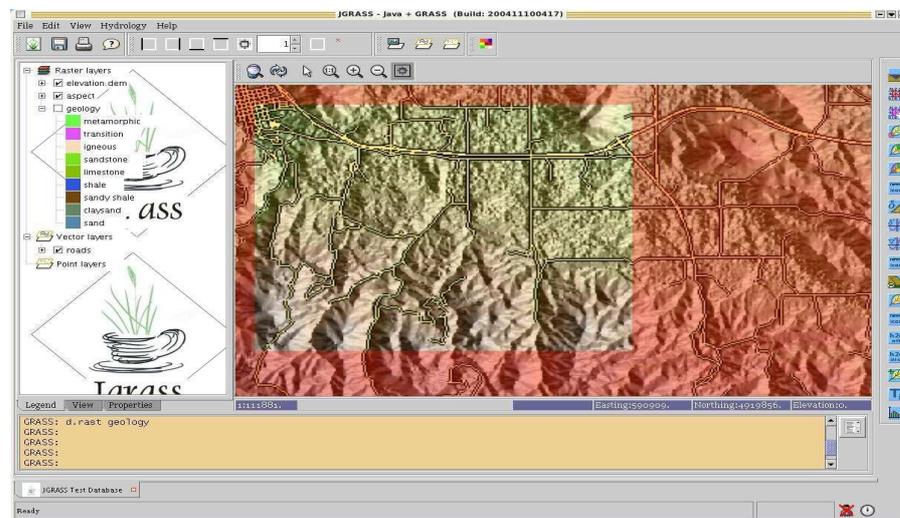


Figura 5: La interfaz JAVAGRASS

En la siguiente sección vamos a mostrar una sesión de ejemplo.



**Figura 6:** Spearfish, South Dakota (SD, USA)

## 2. Bienvenido a GRASS

En esta sección vamos a trabajar con la base de datos *Spearfish60*, la cual ha sido descomprimida en la base de datos de GRASS (ver sección anterior). Se encuentra situada en Dakota del Sur (SD), en los Estados Unidos (ver figura 6). La mayoría de los mapas en *Spearfish60* fueron generados en los años 80, con algunas actualizaciones y adiciones en los últimos años. La base de datos está compuesta por mapas ráster y vectoriales de dos cuadrángulos escala 1:24.000 del USGS (cuadrángulos “Spearfish” y “Deadwood North”) y cubren la mayor parte de la reserva nacional forestal Black Hills (Monte Rushmore).

### 2.1. Iniciar Linux, login

El inicio del PC depende de la instalación local. Explicaremos esto durante el taller.

### 2.2. Introducción a Linux en unos pocos minutos

GRASS es un paquete de software diseñado para correr en varios ambientes UNIX, tales como GNU/Linux, SUN-Solaris, Irix, MacOS X, así como bajo MS-Windows/NT/2000/XP (actualmente con Cygwin). Las arquitecturas de 32 y 64 bits se encuentran soportadas. Por lo tanto, el uso efectivo de GRASS requiere cierta familiaridad con UNIX y disponer del hardware adecuado. Actualmente el uso de GNU/Linux es más simple debido al desarrollo de interfaces gráficas comunes (como el escritorio KDE), donde se pueden iniciar programas desde entornos orientados a menús. Sin embargo es útil aprender como iniciar programas desde la consola de comandos, debido a la forma como incrementa su productividad. En particular, los usuarios pueden combinar comandos de GRASS con el *shell* y otros comandos del sistema para crear scripts más útiles, sin tener que aprender lenguajes de programación adicionales.

Después de abrir una consola de comandos se verá un prompt como el siguiente:

```
[sunombre@sumaquina]
```

o algo similar (la apariencia del cursor `prompt` se puede personalizar). Es aquí donde uno puede introducir los comandos de UNIX e iniciar las aplicaciones. Dentro de la consola de comandos, el “shell” interpreta sus comandos. Éste recibe los comandos desde el teclado y los transfiere al sistema operativo. El “shell” se carga automáticamente cuando se abre una consola de comandos. Hay diferentes shells disponibles: C-shell (csh), bash, y el tsh. Todos aceptarán cualquier comando, aunque difieren en el comportamiento, por ejemplo, como manejan las teclas y como se completan los nombres de los archivos.

El relleno automático de nombres de archivo ahorra bastante tiempo cuando se escribe en el teclado porque sólo se necesita introducir los primeros caracteres del archivo o del nombre del comando para que el “shell” lo complete después de haber utilizado la tecla para completar nombres específica de cada consola. El nombre de la tecla para completar nombres en tsh es <ESC>(presionando dos veces), en bash es <TAB>. Los comandos usados anteriormente se pueden seleccionar y editar con las flechas <ARRIBA>y <ABAJO>. También se puede transferir texto de una consola a otra usando las funciones de “copiar y pegar” con el ratón. Use el botón izquierdo del ratón para marcar y copiar el texto, luego péguelo donde lo necesite usando el botón central o derecho del ratón.

## 2.3. Sesión de prueba: primeros pasos

### 2.3.1. Iniciar GRASS

Dependiendo de la instalación local y la versión, se puede iniciar GRASS 6 desde un menú o desde la ventana de comandos tecleando:

```
grassXX
```

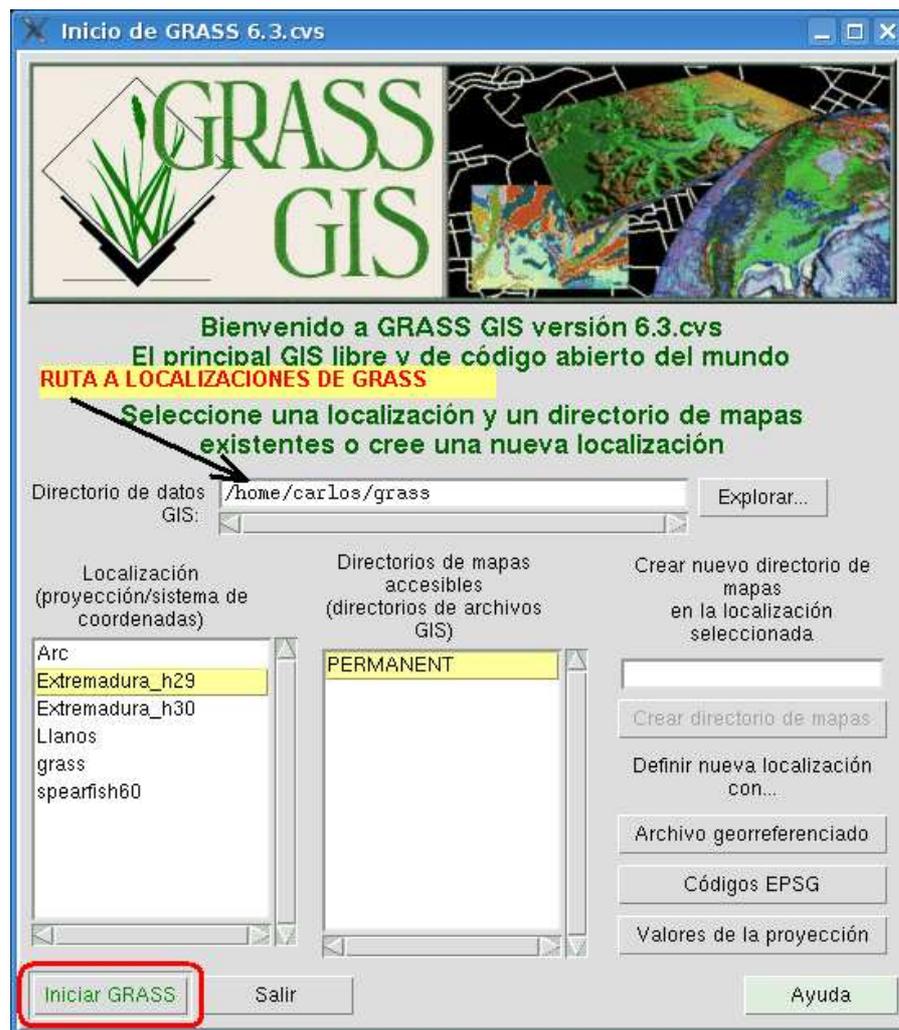
(donde XX corresponde a los dos primeros números de la versión de GRASS que estemos usando. Ej.: GRASS-6.2 ->grass62)

Una interfaz de usuario debe aparecer, tal como se muestra en la figura 7. Si no apareciera, se puede usar:

```
grass63 -gui
```

La ruta a la base de datos **Database** se debe ingresar en el primer campo (Directorio de datos GIS). Si no tiene ninguna base de datos, cree un directorio nuevo (ej. `grassdata/`) en su directorio home. Para el taller la base de datos será preparada e indicada. Después de ingresar la base de datos puede elegir entre una **localización** existente (aquí vamos a usar “spearfish60”) o definir una nueva localización mediante un archivo georreferenciado, códigos EPSG o indicando los valores de la proyección. Seleccionamos “spearfish60” y creamos un nuevo **directorio de mapas** dentro de Spearfish introduciendo un nuevo nombre (ej. el nombre de su cuenta) en la derecha de la pantalla de inicio y luego seleccionando el botón “Crear directorio de mapas”. El nombre del nuevo directorio de mapas aparecerá en la columna central; selecciónelo y entre en GRASS utilizando el botón “Iniciar GRASS” en la esquina inferior izquierda.

Algunas explicaciones:



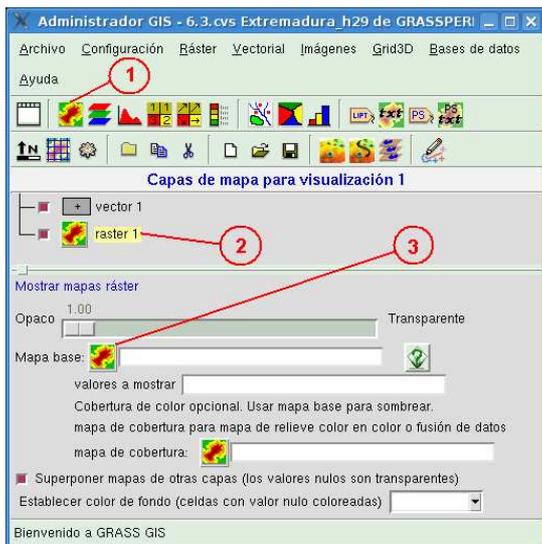
**Figura 7:** Pantalla de inicio de GRASS 6 con la selección de la base de datos, la localización y el directorio de mapas

- Una base de datos (database) es la ruta completa a la base de datos de GRASS, la cual puede contener una o más localizaciones, cada una con sus propios directorios de mapas.
- Una *localización* es el nombre de la región para un proyecto.
- Un *directorio de mapas* está contenido dentro de una localización y se usa para organizar los mapas en carpetas y los archivos por proyecto, subregión o cualquier nombre que resulte adecuado.

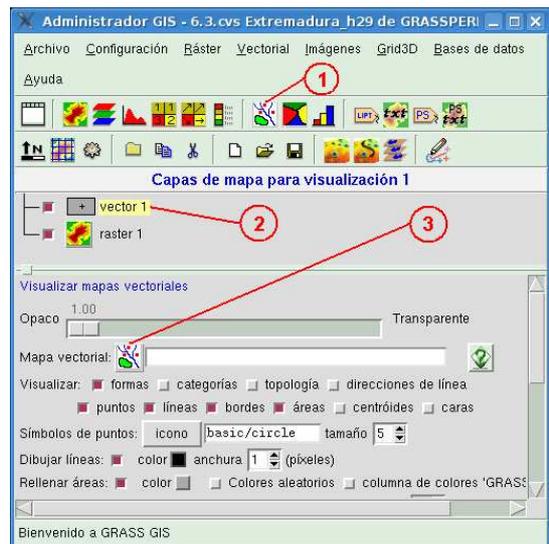
### 2.3.2. Iniciar el Administrador GIS gis.m: cargar mapas tipo ráster y vectorial, visualizar mapas

El Administrador GIS gis.m debe abrirse automáticamente. Si no es así, inícielo desde la consola con:

```
gis.m
```



**Figura 8:** Administrador GIS: cargando un mapa ráster



**Figura 9:** Administrador GIS: cargando un mapa vectorial

Ahora cargue el mapa ráster `elevation.dem` y el mapa vectorial `roads` como se indica en las figuras 8 y 9. En general, se añadirá el tipo de mapa deseado (ráster o vectorial) pulsando el botón correspondiente, luego se seleccionará el mapa añadido en la parte central del Administrador GIS, con lo que las opciones de visualización aplicables al mismo aparecerán en la parte inferior; finalmente se seleccionará un mapa de la lista que aparece al pulsar el icono correspondiente al tipo de mapa y luego se visualizará el mapa en el monitor. Existen opciones para controlar los detalles del mapa.

Los mapas se visualizan en unas ventanas gráficas llamadas Visualización de mapas (Map Display). Se pueden abrir varias Visualizaciones de mapas, numeradas 1, 2, ... Desde las ventanas de visualización se puede exportar el contenido de la ventana a distintos tipos de archivos de imagen, tales como `bmp`, `jpg`, `ppm/pnm`, `png` o `tif`.

### 2.3.3. Guardar los cambios del Administrador GIS

Antes de interrumpir el trabajo, se deben guardar todos los cambios hechos en el Administrador GIS, de forma que se pueda volver donde se dejó el trabajo más tarde. Esto se hace fácilmente seleccionando `Archivo ->Entorno de trabajo ->Guardar como...` Los cambios se guardan en un archivo de texto con extensión `.grc`. Este archivo se puede cargar de nuevo en el Administrador GIS en cualquier momento.

### 2.3.4. Guardar los cambios en la región actual

Al trabajar en un SIG, puede que no sólo queramos guardar los cambios en los mapas, sino también en los límites geográficos. Para hacer una región accesible fácilmente (haciendo `zum`), podemos guardar los límites geográficos y la resolución ráster actuales. Por ejemplo, podemos hacer `zum` en los mapas

roads y elevation.dem visualizados previamente usando la herramienta `.^cercar` de la ventana de visualización (ver siguiente párrafo) o escribiendo `d.zoom` en la consola de comandos (`d.z<tab>` debe completarlo en la consola bash).

Para ampliar una zona del mapa, seleccione el icono de la lupa con un signo `+` en la ventana Visualización de mapas y haga clic con el botón izquierdo del ratón en una de las esquinas que definirá la zona a ampliar. Sin soltar, arrastre el ratón en diagonal hacia la esquina opuesta de la zona que desee ampliar. Este proceso se puede repetir tantas veces como haga falta, hasta ampliar a la zona del mapa que se desee.

Para navegar por el mapa, use la herramienta `.EnCuadrar`, cuyo icono es una mano con una cruz de cuatro puntas.

Para guardar los cambios en los límites de la región como predefinidos, introduzca:

```
g.region save=roadmap (roadmap es el nombre usado en este ejemplo. Elija el que desee en su caso).
```

Ahora queremos devolver la localización Spearfish a sus valores iniciales (1) y luego hacer zum en la subregión guardada anteriormente (2).

(1) Pulsamos el botón “Zum a...” y seleccionamos la opción “Zum a la región por defecto”

(2) Pulsamos el botón “Zum a...” y seleccionamos la opción “Zum a la región actual (establecida con `g.region`)”

El monitor deberá desplegar la región guardada de nuevo. Si tenemos varias regiones guardadas, podemos elegir la opción “Zum a una región guardada” y seleccionar cualquiera de las regiones que tengamos guardadas en la ventana que aparece.

Note que el navegador QGIS viene con una herramienta de zum intuitiva. (por lo que no la explicaremos aquí).

### 2.3.5. La herramienta de visualización NVIZ

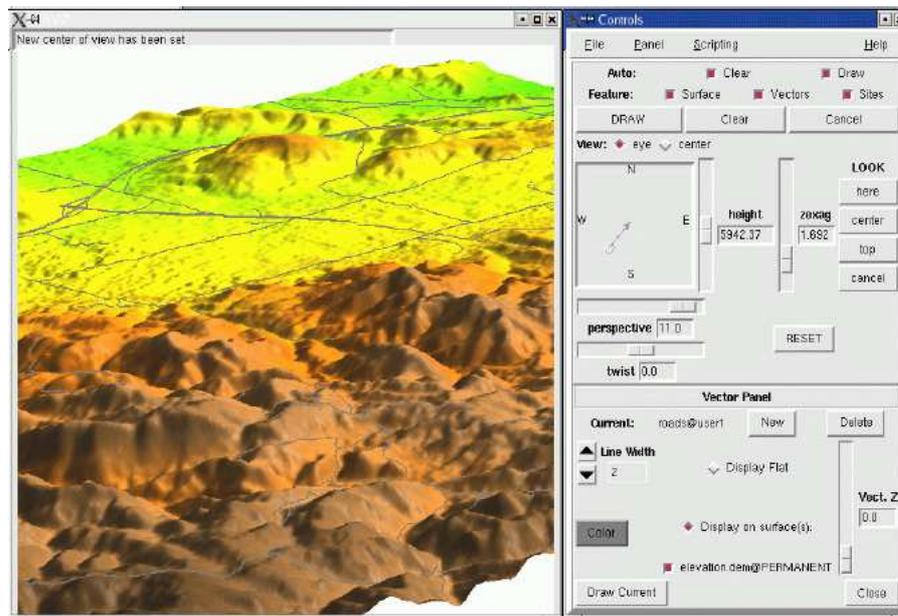
La herramienta de visualización NVIZ es una herramienta poderosa para explorar información ráster y vectorial, así como los volúmenes ráster (ver la figura 10). Permite colocar mapas sobre modelos de elevación digital, acomodar varios mapas, generar perfiles y crear animaciones de sobrevuelos en la región. Para probarla, podemos iniciarla desde la consola de comandos (también se puede iniciar desde el Administrador GIS o la Visualización de mapa):

```
nviz elev=elevation.dem vect=roads
```

La utilización del menú debe ser intuitiva. La última versión de NVIZ soporta diferentes modos de vista, algunos de ellos son similares a la perspectiva de un simulador de vuelo.

### 2.3.6. Ayuda en línea: botón de ayuda, `-help` y `g.manual`

¿Cómo hacer esto y aquello?, se preguntará. Con seguridad, a menudo sólo queremos mirar la sintaxis de un parámetro o indicaciones especiales para un comando. La ayuda se puede encontrar en diferentes niveles:



**Figura 10:** Herramienta de visualización NVIZ con mapas ráster y vectoriales de Spearfish

- Iniciar el comando de GRASS sin ningún parámetro (en la mayoría de los casos) abre una ventana gráfica:

<comando> ej.: d.rast

Un botón de ayuda está disponible en la parte de abajo.

- Para ver la disponibilidad de opciones y parámetros de un comando de GRASS:

<comando>-help ej.: d.rast -help

- Para ver la página del manual para un comando en el navegador web:

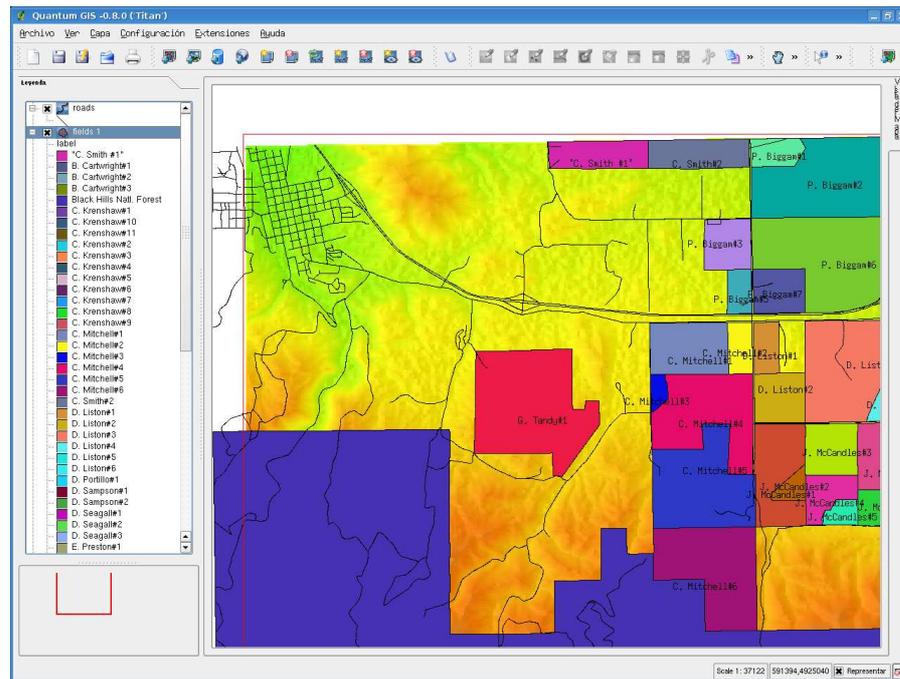
g.manual <comando> ej.:g.manual d.rast

- Para ver la página del manual de un comando en estilo MAN:

g.manual -m <comando> ej.:g.manual -m d.rast

### 2.3.7. Usar la consola de comandos

...aunque ya se ha usado: usar en GRASS la consola de comandos se entiende como introducir los comandos con opciones y parámetros. Usando comandos de consola, es posible crear scripts poderosos. Puede recordar ahora la función para completar comandos mencionada anteriormente. Esto acelera la velocidad con que se escriben los comandos. También se pueden reutilizar los comandos utilizados anteriormente usando las flechas de dirección. Algunos comandos importantes son:



**Figura 11:** Visor de información geográfica QGIS con la interfaz de GRASS: los datos de Spearfish

- para abrir un monitor: `d.mon x0`
- para cerrar un monitor: `d.mon stop=x0` (se puede simplemente cerrar la ventana)
- para hacer una lista de los mapas vectoriales: `g.list type=vect`
- para hacer una lista de los mapas ráster: `g.list type=rast`

### Interfaz de usuario alternativa: QGIS

Hasta el momento solo hemos visto “puro” GRASS. Pero existe algo más para explorar: Quantum GIS (QGIS). Este es un navegador de información geográfica con funcionalidades de GIS que funciona de forma independiente, el cual se encuentra integrado ahora con GRASS.

Para iniciarlo, simplemente ingrese desde dentro (o no) de una sesión de GRASS:

```
qgis &
```

#### 2.3.8. QGIS: visualizar mapas de GRASS, agregar leyendas, nombres y más

Ahora vamos a cargar algunos mapas de GRASS en QGIS. Para ello, debemos activar la extensión GRASS en el Administrador de extensiones de Quantum GIS (menú Extensiones) y luego usar las opciones “Añadir capa vectorial (o ráster) de GRASS”. Cargamos los mapas vectoriales “roads” y “fields” y el mapa ráster “elevation.dem”. Trate de replicar la vista como se muestra en la figura 11. Si además

se carga el mapa ráster “aspect”, se puede usar el cursor de transparencia para integrar visualmente el modelo de elevación digital con las sombras del mapa de aspecto para generar un modelo de elevación digital sombreado. El cursor se encuentra al abrir las propiedades de la capa haciendo doble clic sobre su nombre en la leyenda. Es ahí donde también se pueden seleccionar las leyendas de la información vectorial, textos y más. El orden de los mapas en la leyenda define como son visualizados.

### **2.3.9. QGIS: visualizar información externa de mapas vector y ráster (Shape, GeoTiff, etc)**

Puesto que QGIS es un visualizador de SIG que funciona en forma independiente, también podemos cargar información de fuentes externas tales como archivos Shape, GeoTiff o ERDAS/img. De esta forma la información de GRASS puede ser integrada fácilmente con otras fuentes si las proyecciones son las mismas. Desde QGIS 0.7 en adelante está implementada la reproyección de mapas vectoriales “en caliente”, simplificando la integración de fuentes de datos heterogéneas.

Añada algunos mapas SHAPE TIGER 2000 e imágenes GeoTIFF LANDSAT-7 a su vista de QGIS. Estos archivos se encuentran disponibles para el taller y se encuentran reproyectados de la proyección original a UTM13/NAD27.

### **2.3.10. QGIS: visualizar mapas PostGIS**

Si QGIS fue instalado con soporte para PostGIS, podemos cargar mapas directamente desde una base de datos PostGIS usando el botón *Añadir capa de PostGIS*. PostGIS es una extensión espacial para PostgreSQL para almacenar objetos (vectoriales) espaciales.

Si PostGIS se encuentra disponible, se puede crear una conexión con `db.connect` (consulte el manual) y luego un mapa existente en GRASS se puede copiar a PostGIS con `g.copy`.

### **2.3.11. Crear mapas en papel con QGIS**

Seleccionando el botón con la impresora se accede a la herramienta de composición de mapas, que le permite la creación de mapas para impresión. La figura 12 muestra el diseñador de mapas. También añade un título, una leyenda vectorial y una escala. Note que el botón para refrescar la vista actualiza el contenido del diseñador de mapas con el contenido de la vista principal. Esto es conveniente si se decide modificar el mapa de nuevo antes de finalizarlo. Los mapas se pueden imprimir, exportar como EPS, SVG o archivos de alta resolución PNG.

### **2.3.12. QGIS: Exportar a un archivo map de MapServer**

Una característica especial de QGIS es la capacidad de exportar la vista actual a un archivo de mapa de UMN/MapServer. Primero se debe construir la vista con todas las leyendas vectoriales, transparencias, etc, luego desde el menú principal seleccionar: Archivo ->Exportar a mapa de MapServer. Este procedimiento incluye las rutas a los archivos en GRASS.

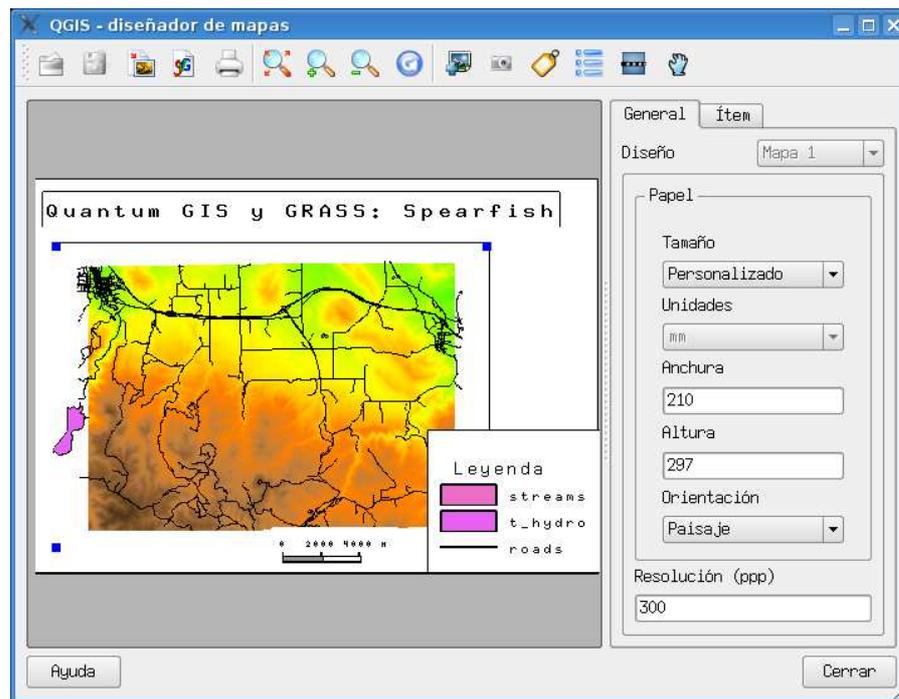


Figura 12: QGIS: Herramienta para la composición de mapas

### 2.3.13. Cerrar la Sesión de GRASS

Ahora cerramos nuestro rápido tour a través de GRASS y QGIS. Primero cierre la ventana de QGIS, luego el Administrador GIS y finalmente, teclee:

```
exit
```

en la consola de comandos para salir de GRASS. Los monitores se cerrarán automáticamente.

## 3. Trabajar con su propia información: importar y exportar mapas y crear localizaciones

### 3.1. Importar información SIG

Para ilustrar mejor la vida diaria de un SIG, vamos a mostrar como importar un par de archivos de SIG. Se ha preparado un conjunto de mapas para el taller usando información de la región de Spearfish.

#### 3.1.1. Iniciar GRASS con Spearfish

Para iniciar GRASS con Spearfish, teclear:

```
grass63
```

En la pantalla de selección, seleccione “spearfish60” en la columna izquierda y su directorio de mapas en la columna central (vea la sección 2.3.1 si no se ha creado un directorio de mapas antes). Luego pulsar el botón “Iniciar GRASS”.

### 3.1.2. Importar información de archivos Shape de ESRI (TIGER 2000)

Seleccione los archivos shape TIGER 2000 que han sido preparados para el taller (archivo: *tiger2000\_latlong\_nad83.tar.gz*).

Estos archivos shape se encuentran originalmente en longitud/latitud NAD83 (grados; código EPSG 4269). Antes de importarlos dentro de la localización Spearfish, tenemos que reproyectar los mapas a UTM de forma que concuerden con las características de la base de datos de ejemplo Spearfish (Zona UTM 13N, NAD27/Clarke66; código EPSG 26713).

Esto se puede hacer de forma eficiente con la herramienta `ogr2ogr`. Como los archivos originales no tienen el archivo `.prj` que contiene la información de la proyección, le asignamos la información “en caliente” usando el parámetro `'-s_srs'` (en inglés, source spatial reference system). El objetivo SRS se define con `'-t_srs'`. Para simplificar la definición de las proyecciones, usamos los códigos EPSG que definen completamente la proyección. El orden de la especificación de archivos en la línea de comandos es (quizás sorprendentemente) `<salida><entrada>`

```
ogr2ogr -s_srs "+init=epsg:4269" -t_srs "+init=epsg:26713" \
    tgr460811kA_UTM13_nad27.shp tgr460811kA.shp
```

Este paso se debe hacer para todos los archivos SHAPE Lat/Log TIGER 2000. Para este taller, se puede saltar este paso, dado que tenemos preparado el paquete *tiger2000\_UTM13\_nad27.tar.gz* que contiene mapas que pueden utilizarse directamente. El archivo *readme.html* explica los nombres y acrónimos de las capas.

La información vectorial se importa utilizando `v.in.ogr`, el cual permite importar muchos tipos de formatos. Los archivos SHAPE no se almacenan en un formato topológico. El módulo `v.in.ogr` contiene un “motor topológico” que arregla una gran cantidad de problemas comunes que contienen los archivos SHAPE durante la importación y generación de información topológica.

Ahora podemos importar los archivos SHAPE reproyectados. Aquí se muestra como, usando la línea de comandos; también se puede usar el ratón después de iniciar el comando sin parámetros o bien desde el Administrador GIS en el menú Archivo->Importar->Mapa vectorial->Varios formatos que usan OGR. Importamos el archivo de caminos (lkA) y la hidrografía (lkH):

```
v.in.ogr tgr460811kA_UTM13_nad27.shp out=tgr2000_roads
v.in.ogr tgr460811kH_UTM13_nad27.shp out=tgr2000_hydro
d.vect tgr2000_roads col=grey
d.vect tgr2000_hydro col=aqua
```

Para visualizarlos, se pueden seleccionar los mapas importados en el Administrador GIS, usar la consola de comandos o iniciar `qgis` y seleccionar los mapas allí (usando el botón *Añadir capa vectorial de GRASS*).

Nota: Si la información de la proyección no se encuentra en el archivo (y se está seguro de que corresponde a la proyección de la localización de GRASS), puede usarse la opción '-o' para no realizar la comprobación de la proyección.

### 3.1.3. Importar archivos ráster ERDAS/IMG (LANDSAT-7)

Se ha preparado una imagen LANSDAT para el área de Spearfish, la cual ha sido reproyectada y cortada para minimizar el tamaño de la información. La imagen está dividida en tres archivos (NIR: infrarojo cercano, MIR: infrarojo medio, TIR: termal):

- spearfish\_landsat7\_NAD27\_vis\_ir.img: TM10, TM20, TM30 (azul, verde, rojo), TM40 (NIR), TM50, TM70 (MIR)
- spearfish\_landsat7\_NAD27\_tir.img: TM62 (TIR ganancia baja), TM62 (TIR ganancia)
- spearfish\_landsat7\_NAD27\_pan.img: TM80 (pancromática)

Para importar datos ráster a GRASS, se utiliza `r.in.gdal` (el parametro de salida para datos con múltiples canales se usa como prefijo):

```
r.in.gdal -e in=<image.img>out=<image>
```

Esta función se puede iniciar también desde el Administrador GIS, mediante el menú Archivo->Importar->Mapa ráster->Varios formatos que usan GDAL

El módulo verifica que la definición de la proyección de la imagen concuerde con la de la localización en la que se está trabajando. Si la proyección no concuerda, se genera un error. A veces tal tipo de definiciones no se encuentran en la información a importar; si se está seguro de que la información concuerda con las definiciones de la localización, se puede utilizar el parámetro '-o' para ignorar la prueba. El parámetro '-e' se puede usar para expandir los límites de la localización y de esa forma concordar con los de la imagen. Sin embargo, el mapa es importado completamente en cualquier caso. Para nuestro ejemplo en Spearfish escribimos:

```
r.in.gdal -e in=spearfish_landsat7_NAD27_vis_ir.img out=tm
g.rename rast=tm.6,tm.7
r.in.gdal -e in=spearfish_landsat7_NAD27_tir.img out=tm6
r.in.gdal -e in=spearfish_landsat7_NAD27_pan.img out=pan
```

Para mantener la numeración correcta, renombramos *tm.6* al número correcto *tm.7*.

Para ver la información de los mapas multicanal, podemos generar una composición RGB en caliente:

```
g.region rast=tm.1 -p
d.rgb b=tm.1 g=tm.2 r=tm.3
```

También podemos usar el botón “Añadir capa RGB o HIS” de la barra de iconos del Administrador GIS.

Una vez cargado el mapa se debe poder ver el área de Spearfish en colores cercanos a los reales.

### 3.1.4. Importar archivos ráster GeoTIFF

Los datos en archivos de imagen TIFF se encuentran en formato GeoTIFF (un solo archivo que contiene los metadatos como etiquetas TIFF) o en dos archivos, un archivo plano TIFF `map.tif` y un archivo ASCII `map.tfw`. Un TFW, o archivo de proyección, es un archivo de texto tipo ASCII que contiene la información para transformar la imagen usada por el software para visualizarlo. Los archivos de proyección se pueden crear con cualquier editor de texto y también por GDAL. Asegúrese de contar con ambos archivos cuando las imágenes no se encuentran en formato GeoTIFF. El formato TIFF viene en distintas variedades, las cuales son aceptadas por GRASS. La misma imagen LANDSAT-7 post-procesada anteriormente en formato ERDAS/Img se encuentra disponible como un archivo GeoTIFF (extensión `.tif`). Utilice `r.in.gdal` tal como lo hizo con los archivos LANDSAT-7 ERDAS/Img:

```
r.in.gdal -e in=<map.tif>out=<map>
```

### 3.1.5. Crear nuevas localizaciones para GRASS a partir de datos existentes

Tanto `v.in.ogr` como `r.in.gdal` cuentan con un parámetro 'localización' que se puede usar para generar una nueva localización de GRASS (incluyendo la importación de los datos) desde una localización existente. Esto simplifica mucho el procedimiento. Note que los datos existentes deben incluir los parámetros de la proyección. Si faltan, se puede usar `ogr2ogr` o `gdal_translate` para asignarlos ('-a\_srs') antes de usar los datos.

### 3.1.6. Un gran número de formatos de SIG...

Para darle una idea, existen numerosos formatos soportados por el SIG GRASS.

*Formatos ráster:*

- `r.in.gdal`: ArcInfo, CEOS, DOQ, DTED, ENVI, Envisat, Erdas Img/LAN, FAST, (Geo)TIFF, HDF4, SAR, SDTS, ECW, ...
- `r.in.bin`: binarios, BIL, archivos GMT
- `r.in.mat`: archivos MatLab
- `r.in.srtm`: bloques ("tiles") SRTM de 1 grado

*Formatos vectoriales:*

- `v.in.ogr`: SHAPE, GML, UK.NTF, SDTS, TIGER, archivos MapInfo, DGN, VRT, ODBC, PostGIS, ArcCover
- `v.in.ascii`: GRASS ASCII
- `v.in.e00`: Formato E00 de ArcInfo
- `v.in.db`: Crear vectoriales a partir de bases de datos con coordenadas x,y, [z]

De la misma forma, existen módulos para escribir en varios formatos (`r.out.gdal` y `v.out.ogr`).

## 3.2. Crear una nueva localización

### 3.2.1. Definir una nueva localización interactivamente

Aunque algunas veces este proceso es considerado un poco truculento o peligroso, ahora vamos a aprender como crear nuestra propia localización desde cero. Recuerde que se pueden generar nuevas localizaciones desde datos existentes (ver arriba). Pero puede ser de utilidad saber como hacerlo interactivamente. Una gran diferencia entre GRASS y otros SIG es que GRASS requiere los parámetros de la proyección antes de que el usuario pueda trabajar en una localización. La ventaja de esto es que las cosas quedan bien definidas y se evita un desorden con la combinación de proyecciones.

Primero inicie GRASS:

```
grass63
```

La ventana de inicio nos permite definir una nueva localización mediante un archivo georreferenciado, códigos EPSG o introduciendo los valores de la proyección. Seleccione el botón *Valores de la proyección* para crear la nueva localización, el cual lo llevará a una nueva ventana de texto (algún día existirá una interfaz gráfica para esto). En la pantalla se introduce un nombre para la nueva localización (no puede contener espacios en blanco) y luego se continúa tras presionar la tecla “ESC”seguida de la tecla“RETURN”.

Vamos ahora a describir el procedimiento general:

Se necesitará asignar los parámetros de la localización tales como el sistema de coordenadas, el datum que se quiere utilizar, los límites de la región y la resolución por defecto para los datos ráster. El sistema los irá preguntando de forma sucesiva:

- **En primer lugar** nos avisará de que la localización introducida no existe y preguntará si queremos crearla (por defecto toma la respuesta Sí).
- **A continuación** nos muestra la información que debemos tener para poder crear la localización y nos pregunta si disponemos de ella (por defecto, Sí).
- **Sistema de coordenadas:** a continuación se debe escoger entre X-Y, Latitud-Longitud, UTM u otro sistema de coordenadas. Esta opción depende de sus datos y del uso que se les vaya a dar. Nos pedirá confirmación del sistema elegido.
- **Descripción:** se le preguntará por una línea de texto que describa el área del proyecto, por ejemplo “Mapa topográfico de los Alpes”.
- **Datum:** luego se le preguntará si quiere especificar un datum para la localización (por defecto, Sí) y en caso afirmativo nos pedirá el nombre del datum. Si se escribe "list"mostrará una lista de los datum disponibles y si se escribe custom"permitirá introducir parámetros personalizados.

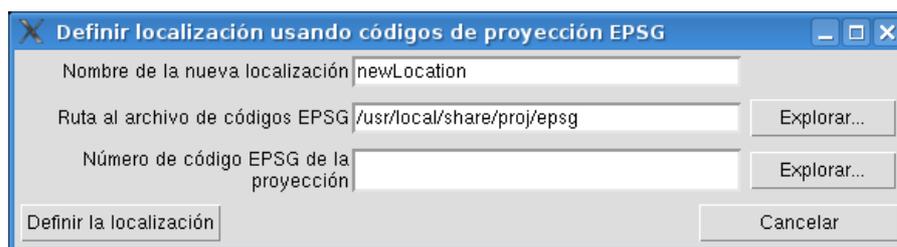
Tenga en cuenta que las preguntas variarán dependiendo de la proyección; un ejemplo es el siguiente:

- (Si se escoge “D - Otra Proyección) **Por favor, indique el nombre de la proyección:** Si teclea 'list' se muestra una lista de las proyecciones disponibles, por ejemplo “tmerc” para Transverse Mercator, “lcc” para Lambert Conforme Cónica, “moll” para Mollweide, etc.
  - **Por favor especifique el nombre del datum:** de nuevo utilice "list" para obtener una lista de los datums disponibles; ejemplos pueden ser: “wgs84”, “nad27”, “eur79”, etc.
  - **Enter Central Parallel (“introduzca el paralelo central”):** o si se quiere el Ecuador como el paralelo central.
  - **Enter Central Meridian (“introduzca el meridiano central”):** o si se quiere Greenwich como el meridiano central.
  - **Enter Scale Factor (“introduzca el factor de escala”)** en el meridiano central: 1 ó 0.9996 ó ...
  - **Enter plural form of map units (“introduzca el nombre en plural de las unidades del mapa”):** por ejemplo, "meters"(metros, en inglés).
- **Area boundary coordinates (“Coordenadas de los límites del área”):** el paso siguiente es introducir las coordenadas de los límites del área de estudio y la resolución por defecto para datos ráster.
    - **La resolución por defecto para los datos ráster** (Resolución de cuadrícula) tiene que ser seleccionada de acuerdo a sus necesidades. Generalmente, es recomendable trabajar en múltiplos de .25 (0.25, 0.5, 1.75, 2.00, 12.25 etc.). Esta resolución no se relaciona con los datos vectoriales o de puntos debido a que éstos se almacenan con sus coordenadas exactas. Tenga en cuenta que cada mapa ráster puede tener su propia resolución. Se puede salir de esta pantalla con “ESC”-“RETURN” y luego si todo es correcto aceptar la lista de parámetros que aparece en la pantalla.
  - Se regresará entonces a la pantalla de inicio: **enter the mapset’s name (“introduzca el nombre del directorio de mapas”)** (si no ha sido introducido previamente). Otro “ESC”-“RETURN” le permitirá finalmente salir de esta pantalla. El directorio de mapas se crea dentro de la nueva localización tras responder “si” a la siguiente pregunta. El directorio de mapas usará los parámetros de la localización (la región y la resolución) como parámetros por defecto.

Ahora el área del proyecto, esto es, la localización incluyendo el directorio de mapas, se ha creado. Pulse el boton *Iniciar GRASS* para entrar en GRASS y empezar a trabajar dentro de la nueva localización. Ahora se pueden verificar los parámetros de la proyección con:

```
g.proj -w
```

Si leyó esta sección sin definir realmente una nueva localización, puede probar este comando en la localización Spearfish.



**Figura 13:** Pantalla de inicio de GRASS: Creando una localización con códigos EPSG

### 3.2.2. Crear su propia localización a partir de un código EPSG

Como alternativa, para crear su propia localización de forma rápida y fácil, se pueden usar los códigos de proyección EPSG. Proyecciones y cuadrículas nacionales han sido estandarizadas por el European Petroleum Survey Group (EPSG, <http://www.epsg.org>), dando un identificador único a cada sistema de referencia. En GRASS, estos códigos se encuentran en la librería PROJ4, la cual provee una tabla de códigos EPSG. Seleccionando el botón para crear una localización con códigos EPSG, se abrirá una nueva ventana (ver figura 13). Se introduce el nombre de la nueva localización y el código EPSG. Si no se conoce el código EPSG, puede utilizarse el botón que muestra el contenido del archivo PROJ4-EPSG con los nombres de los sistemas de referencia y los códigos correspondiente entre corchetes. Se advierte que algunas veces no se encuentran los parámetros de los datum geodésicos. Estos parámetros pueden ser agregados dentro de GRASS o el sistema le puede preguntar por el datum que quiere utilizar. Después de introducir la información necesaria, seleccione el botón “Definir la localización”. Esto generará la nueva localización de GRASS, que ya estará disponibles desde la pantalla de inicio de GRASS. Seleccione la localización y el directorio de mapas y luego seleccione el botón “Iniciar GRASS” para entrar al sistema.

## 4. Análisis de mapas ráster

GRASS es conocido tradicionalmente por tener excelentes herramientas para el procesamiento de información ráster. Todas las funcionalidades clásicas más las herramientas para el análisis de datos de series de tiempo y modelos se encuentran disponibles. Aunque los nombres de los comandos para el procesamiento de imágenes difieren en el primer carácter, (i.\* en vez de r.\*), se encuentran totalmente integrados. Cualquier imagen (de fotografías aéreas o satélites) puede ser usada como un mapa ráster. Se encuentra disponible soporte adicional para manejar imágenes multiespectrales. También se han implementado métodos más sofisticados tales como generación de ortofotos y la clasificación de imágenes.

GRASS puede realizar operaciones a nivel de píxeles sobre mapas ráster, así como cálculos locales (vecindades) y globales (en todo el mapa). También se pueden crear buffers, mapas de cuencas hidrológicas, líneas de flujo, pendientes, orientación y curvatura del terreno y realizar álgebra ráster.

Para comenzar, queremos mirar los metadatos de un mapa ráster; teclee:

```
r.info <map>      ej.: r.info elevation.10m
```

### 4.1. Análisis de modelos de elevación digital

Podemos calcular los mapas de pendiente y orientación a partir de un Modelo Digital del Terreno (MDT, en inglés DEM, Digital Elevation Model) con `r.slope.aspect`. Primero restablecemos los límites de la región de forma que correspondan a los del mapa de entrada:

```
g.region rast=elevation.dem -p
r.slope.aspect el=elevation.dem as=aspect_30m sl=slope_30m
d.rast aspect_30m
d.rast.leg slope_30m
```

Los dos mapas son procesados a la vez. Tenga en cuenta que los ángulos horizontales se miden en el sentido de las agujas del reloj desde del Este. Las pendientes son calculadas por defecto en grados. El comando `d.rast.leg` añade una leyenda sencilla al monitor.

Existen módulos adicionales que trabajan con los MDT: las áreas con depresiones se pueden rellenar con `r.fill.dir` y las líneas de flujo se pueden calcular con `r.flow`. El análisis de cuencas se puede realizar con `r.watershed` y con `r.terraflow` en cuadrículas masivas.

### 4.2. Álgebra de mapas ráster

GRASS cuenta con una muy buena herramienta para álgebra de mapas `r.mapcalc`. Este módulo se usa de una forma más eficiente en la consola de comandos ya, que se cuenta con un soporte de cursor flexible que proporciona el shell. Opera celda por celda, usando una tecnología de ventanas móviles. Para empezar con algunas operaciones sencillas, vamos a filtrar todos los píxeles con una elevación mayor a 1.500 m en el MDT de Spearfish:

```
r.mapcalc .elev_1500 = if(elevation.dem >1500.0, \
    elevation.dem, null())"
d.rast elev_1500
```

El comando, dentro de las comillas dobles, contiene una sentenica “if” (si es mayor que 1.500 m) con una opción “then” (entonces copie los valores de los píxeles) y una opción “else” (si la condición no se cumple, escriba *Sin datos*). La función `null()` es una palabra reservada que inserta el valor *Sin datos* en las celdas que están siendo procesadas. Existen otras funciones disponibles tales como `mean()` (media), `min()` (mínimo), `max()` (máximo), `sin()` (seno), `cos()` (coseno) etc. La herramienta de álgebra de mapas puede aceptar más de un mapa de entrada a la vez. Los nuevos mapas se pueden calcular a partir de un conjunto de mapas de entrada. Además, los valores adyacentes pueden ser considerados, ej. para generar líneas de flujo a través del terreno. Por favor consulte el manual o los libros indicados en la bibliografía para más funciones y ejemplos.

### 4.3. Georreferenciar un mapa escaneado (4 puntos en las esquinas): mapa topográfico 1:24.000

GRASS se puede usar para georreferenciar mapas (por ejemplo escaneados) definiendo puntos de control en el terreno. El mapa escaneado debe importarse dentro de una localización XY sin ningun-

na información de proyección. Esta localización se puede crear automáticamente usando el comando `r.in.gdal` dentro de otra localización. Al usar el parámetro “localización” el comando no solamente importará el mapa sino que además creará una nueva localización (ver sección 3.1.5). Luego se debe reiniciar GRASS con la nueva localización XY que contiene el mapa escaneado. Ahora vamos a hacer una breve descripción del procedimiento sin adentrarnos en muchos detalles.

- El mapa escaneado tiene que ser integrado dentro de un grupo de imágenes (`i.group`; incluso si es un solo mapa).
- El grupo es apuntado a una localización de referencia (`i.target`).
- El usuario abre un monitor de GRASS y coloca gráficamente los puntos de control en el terreno (`i.points` o `i.vpoints`). El mapa sin referenciar se carga dentro del lado izquierdo del monitor y el mapa de referencia se carga dentro del lado derecho. Para un mapa que no tenga distorsiones, cuatro puntos en las esquinas deben ser suficientes.
- El mapa sin referenciar es rectificado en la localización de referencia (`i.rectify`). El orden polinomial para una rectificación de 4 puntos es 1.

Una vez está hecha la rectificación, GRASS tiene que ser reiniciado en la localización de referencia. Ahora se puede validar el mapa resultante.

#### 4.4. Procesamiento y visualización de información de volúmenes

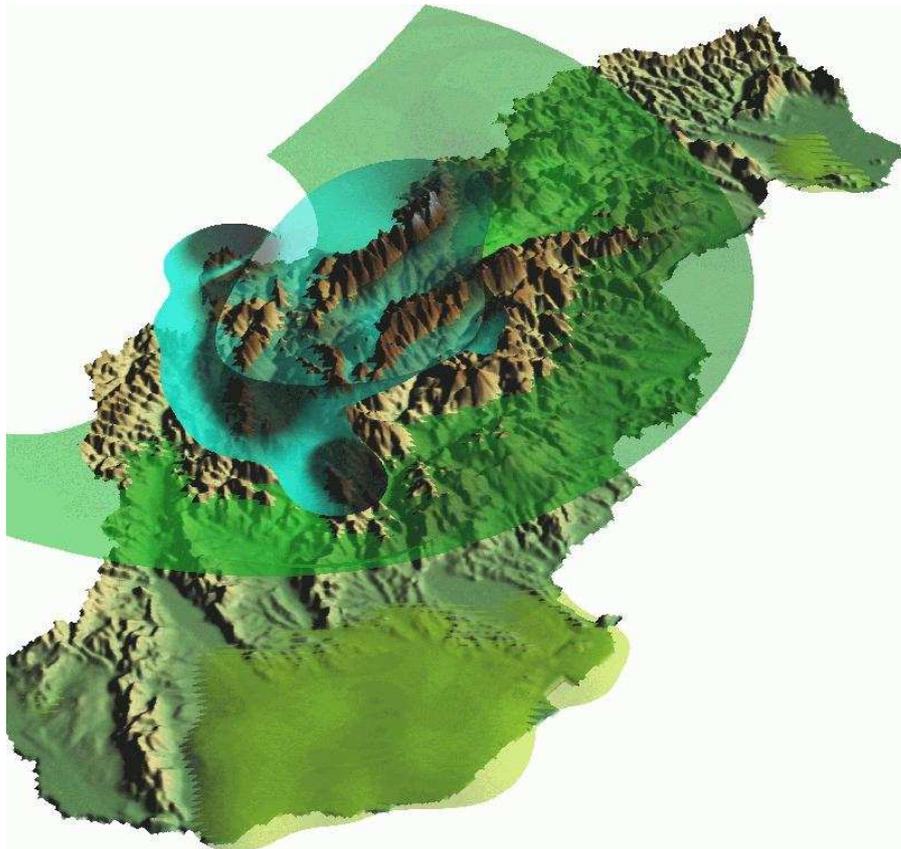
Una reciente mejora a GRASS es la capacidad para procesar volúmenes ráster (vóxeles). Esto se puede usar para visualizar datos del suelo o la atmósfera sin las limitaciones de los mapas en 2D. GRASS cuenta con herramientas de interpolación (3D spline) y álgebra de mapas como herramientas de análisis. NVIZ fue mejorado recientemente para visualizar volúmenes (ej. iso-valores, ver figura 14). Vamos a mostrar ejemplos durante el taller. El ejemplo de Eslovaquia está disponible de la página Web del libro de Neteler & Mitasova, 2004 (<http://mpa.itc.it/grasstutor/>, ver los archivos de la 2ª edición).

## 5. Procesamiento de imágenes

### 5.1. Clasificación de imágenes

En clasificación de imágenes generamos un mapa temático a partir de un (conjunto de) canal (es) de entrada. Esos canales de entrada son normalmente datos aéreos o de satélite. Los datos multiespectrales se pueden considerar como un lote de mapas ráster con la misma referencia espacial. Durante el proceso de clasificación de imágenes se analiza la respuesta espectral de los objetos y se separada en clases. Los mapas resultantes contienen un conjunto de clases que pueden representar usos del suelo o el tipo de cobertura.

GRASS puede procesar múltiples canales que se pueden agrupar con `i.group`. Luego existen dos posibilidades, un análisis estadístico automático sobre los canales de entrada (clasificación no supervisada)

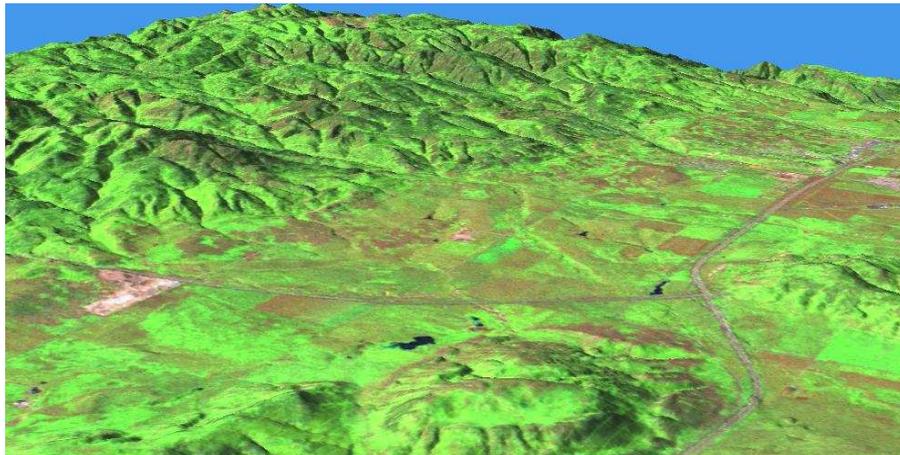


**Figura 14:** Visualización de los datos de precipitación de Eslovaquia mediante volúmenes en 3D

o bien se tienen que digitalizar por el usuario áreas de entrenamiento para definir áreas de uso del suelo o cobertura conocidos (clasificación supervisada). GRASS luego genera firmas espectrales para las clases deseadas y realiza un análisis final sobre todos los píxeles de todos los canales de entrada, asignando cada píxel a una clase. En el caso de clasificación no supervisada las clases son simplemente enumeradas, mientras que en el caso de clasificación supervisada las clases corresponden a los nombre de las áreas de entrenamiento.

Dado que las herramientas más sofisticadas para la clasificación supervisada son explicadas en la literatura, aquí vamos a mostrar una clasificación no supervisada simple (algoritmo de Máxima Verosimilitud):

```
i.group group=lsat subgroup=lsat in=tm.1,tm.2,tm.3,tm.4,tm.5,tm.7
i.cluster group=lsat subgroup=lsat sig=sig.cluster \
  classes=15 sep=1.5
i.maxlik group=lsat subgroup=lsat sig=sig.cluster \
  class=tm.class rej=tm.class.rej
d.rast.leg tm.class
```



**Figura 15:** NVIZ mostrando una composición Brovey de una imagen LANDSAT 7

```
d.rast.leg tm.class.rej
```

El mapa *tm.class* almacena los resultados y el mapa *tm.class.rej* el nivel de confianza para cada pixel.

## 5.2. Fusión de imágenes: Transformada Brovey

Se puede hacer un ejemplo ilustrativo de como mejorar visualmente una escena LANDSAT-7 con la transformada Brovey. Aquí se fusionan tres canales multiespectrales (resolución 28.5m) y el canal pancromático (resolución 14.25m) en tres nuevos canales de rojo, azul y verde. Después de importar la subescena para Spearfish ejecutamos:

```
i.fusion.brovey -l ms1=tm.2 ms2=tm.4 ms3=tm.5 pan=pan out=brovey
g.region -p rast=brovey.red
```

```
r.composite r=brovey.red g=brovey.green b=brovey.blue \
out=tm.brovey
```

```
d.rast tm.brovey
```

Los canales de entrada tienen que ser 2, 4, 5 y el canal pancromático con la opción '-l' para el sensor LANDSAT-7. Luego ajustamos la región de GRASS a la alta resolución resultante de la imagen y creamos un nuevo mapa a partir de los tres nuevos canales Brovey. Podemos visualizar este mapa usando el Administrador GIS o QGIS. Comparando con *tm.4* o con otros canales se puede observar como mejora la resolución espacial.

Podemos colocar la imagen resultante sobre el MDT de alta resolución con NVIZ (ver figura 15):

```
nviz elevation.10m col=tm.brovey
```

---

## 6. Trabajar con información vectorial

GRASS 6 viene con un motor vectorial completamente renovado que permite administrar información topológica y vectorial en 2D y 3D. El nuevo formato interno para la información vectorial es portátil entre plataformas de 32-bit y 64-bit. Además, un nuevo sistema de indexación espacial acelera el acceso a la información vectorial y un nuevo sistema de indexación de categorías acelera las consultas a los atributos. Se pueden importar datos vectoriales de otros programas de GIS (permitiendo la limpieza de la información topológica) y también se pueden enlazar con la base de datos de GRASS como mapas virtuales. Una nueva biblioteca *Directed Graph Library* permite el análisis de redes vectoriales. Se han implementado herramientas para la superposición, intersección y extracción de elementos de mapas vectoriales. El nuevo motor vectorial incluye una integración completa y flexible con sistemas de administración de bases de datos (DBMS) para la administración de los atributos (actualmente se encuentran soportados DBF, PostgreSQL, MySQL y ODBC). Las sentencias SQL se utilizan para administrar los atributos. Se ha implementado la actualización gráfica de atributos al igual que una herramienta de consulta vectorial interactiva.

En esta sección vamos a explorar las funcionalidades básicas con datos vectoriales.

Los tipos de geometrías soportadas son puntos, centroides, líneas, bordes, áreas (borde + centroide), caras (áreas en 3D), kernel (centroides en 3D) y volúmenes (caras + kernel). El almacenamiento de la geometría es 3D verdadero:  $x,y,z$  con  $z=0$  en el caso 2D.

### 6.1. Importar mapas vectoriales

Los mapas vectoriales se pueden importar desde varias fuentes tales como coberturas de ArcInfo, archivos CSV, DGN, SHAPE, GML, MapInfo, MySQL, ODBC, OGDI, PostgreSQL/SQL, S57, SDTS, TIGER, UK, NTF y VRT. El módulo para importar los mapas vectoriales es `v.in.ogr`, que puede iniciarse desde la consola o mediante el menú Archivo->Importar->Mapa vectorial->Varios formatos que usan OGR, en el Administrador GIS. El parámetro “dsn” (data source name) de entrada puede ser un archivo, un directorio o una conexión a una base de datos, dependiendo del formato de los datos. Como GRASS es un SIG topológico, estructuras simples no topológicas tales como los archivos SHAPE son transformados en representaciones topológicas durante la importación. La calidad de los datos es verificada durante la importación y estructuras vectoriales que violan las condiciones topológicas son almacenadas en capas separadas para su inspección posterior. Para más detalles por favor consulte la página del manual de `v.in.ogr`.

Como ejemplo, importamos un archivo SHAPE generado a partir de los datos TIGER 2000 en la localización Spearfish y lo miramos.

```
v.in.ogr dsn=tiger_lines.shp out=tiger_lines
d.vect tiger_lines
```

Los mapas también se pueden registrar simplemente usando `v.external`. En este caso se genera únicamente una pseudo-topología y el mapa será de lectura solamente (por lo que no se podrán hacer modificaciones).

## 6.2. Administración de atributos

Por defecto GRASS 6 administra atributos vectoriales en archivos dBase (xBase). Para añadir o eliminar vínculos entre un mapa vectorial y su tabla de atributos se utiliza el comando `v.db.connect`. Este comando también imprime las conexiones actuales. Veamos dos mapas (en el mapset PERMANENT):

```
v.db.connect -p roads
v.db.connect -p streams
```

Mientras que el mapa roads está vinculado a una tabla de atributos, el mapa streams no la tiene.

Cuando se utiliza una base de datos externa, el módulo `db.connect` se usa para definir los parámetros de la conexión, luego `db.login` para introducir el nombre de usuario y la contraseña. Esto es necesario para conexiones a PostgreSQL y PostGIS y para conexiones a otras bases de datos.

Hay disponible un conjunto de módulos para mostrar los nombres de las tablas, columnas y los tipos, para hacer consultas SQL y para crear o alterar las definiciones de las tablas. Podemos consultar los atributos del mapa roads con:

```
echo "SELECT * FROM roads" | db.select
```

Esto funciona para cualquier tabla dependiendo de como esté conectado GRASS a una base de datos externa (ver `db.connect`). Una forma más conveniente de consultar las tablas asociadas es usando `v.db.select`. Por ejemplo, podemos mostrar los atributos del mapa roads con:

```
v.db.select roads
```

Con `v.report` se genera un informe que contiene los tamaños de las áreas o las longitudes de las líneas.

## 6.3. Operaciones de Buffer

Se pueden hacer operaciones de tipo buffer en mapas vectoriales usando el comando `v.buffer`. Aquí mostramos como generar buffers de 300 metros alrededor de sitios arqueológicos en la localización Spearfish:

```
d.vect archsites
v.buffer archsites out=archsites_buf300 buffer=300
d.vect archsites_buf300 col=red
```

Para generar solo medio buffer a un lado de líneas, se puede usar `v.parallel`. Esto añade una sola línea paralela al lado derecho o izquierdo.

## 6.4. Extracciones

Se pueden extraer elementos vectoriales de los mapas de diferentes formas: se pueden seleccionar por el identificador ID (llamado “cat” o “número de categoría” en el lenguaje de GRASS), por el valor de los atributos vía sentencias SQL “where” o por tipos geométricos (punto, línea, etc.). Como ejemplo podemos extraer las autopistas (“interstates”) del mapa roads por atributo. Primero visualizamos la tabla de atributos para ver cómo está escrita y cómo se llama la columna a consultar y luego extraemos las líneas de tipo vectorial a un nuevo mapa:

```
d.erase  
d.vect roads  
v.db.select roads  
v.info -c roads  
v.extract roads out=interstates where="label='interstate'"  
d.erase  
d.vect interstates
```

## 6.5. Seleccionar, cortar, uniones, intersecciones

Para nuestro ejemplo de cortar vamos a importar las áreas urbanas de los archivos TIGER 2000 (de nuevo dentro del archivo *tiger2000\_UTM13\_nad27.tar.gz* para el área Spearfish):

```
v.in.ogr dsn=UA_46081_UTM13_nad27.shp out=urban_areas  
d.vect urban_areas
```

Queremos extraer todas las carreteras que estén dentro de áreas urbanas. Para ello usamos `v.select` y especificamos el mapa de polígonos de áreas urbanas y el mapa de líneas roads como parámetros (`ainput` y `binput`):

```
v.select ain=roads bin=urban_areas out=urban_roads  
d.vect urban_roads col=red
```

Como segundo ejemplo queremos cortar los distritos unificados de escuelas de los archivos TIGER 2000 con las áreas urbanas que importamos primero. En este caso usamos un comando diferente que nos permite usar mapas de polígono como entrada:

```
v.in.ogr dsn=tgr46081uni_UTM13_nad27.shp out=school_dist_unified  
v.overlay ain=urban_areas bin=school_dist_unified \  
    out=urban_school_dist op=and  
d.vect urban_school_dist fcol=yellow
```

Para verificar, podemos consultar los nuevos mapas creados:

```
d.what.vect urban_school_dist
```

Existen opciones más avanzadas para estos dos comandos, vea el manual para más detalles.

Para extraer datos de un solo archivo según el identificador o basándose en una sentencia SQL, utilice `v.extract`.

## 6.6. Conversiones ráster-vector y viceversa

GRASS es capaz de convertir entre modelos (representaciones de mapas) ráster y vectoriales, incluyendo transferencia de atributos. Para convertir mapas vectoriales a ráster, se usa `v.to.rast`. Esto permite asignar valores fijos al mapa resultante (útil cuando se genera una máscara MASK) o transferir los atributos de una columna específica.

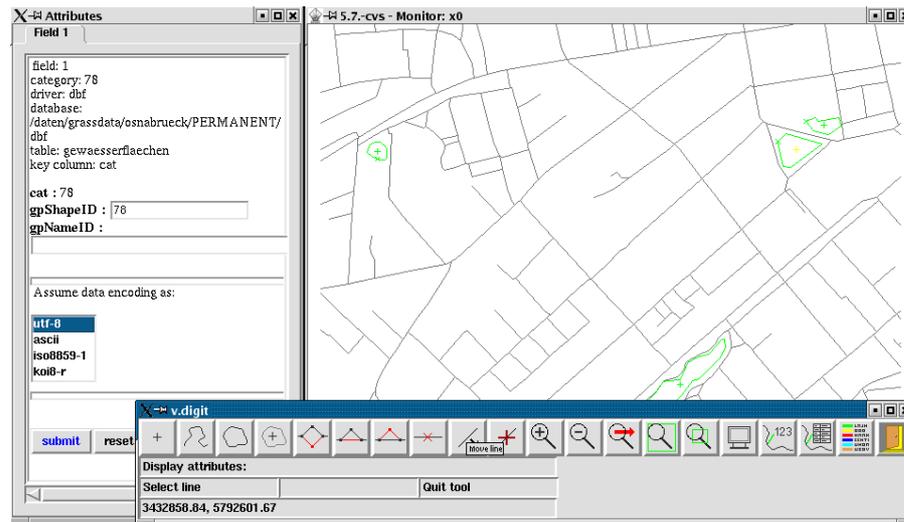


Figura 16: Digitalizador de GRASS: v.digit

El módulo `r.to.vect` hace lo contrario, vectoriza puntos, líneas y áreas ráster. Mientras que los puntos se pueden vectorizar directamente, las líneas deben ser adelgazadas (esqueletizadas) con `r.thin` antes de ser usadas.

## 6.7. Digitalización con GRASS

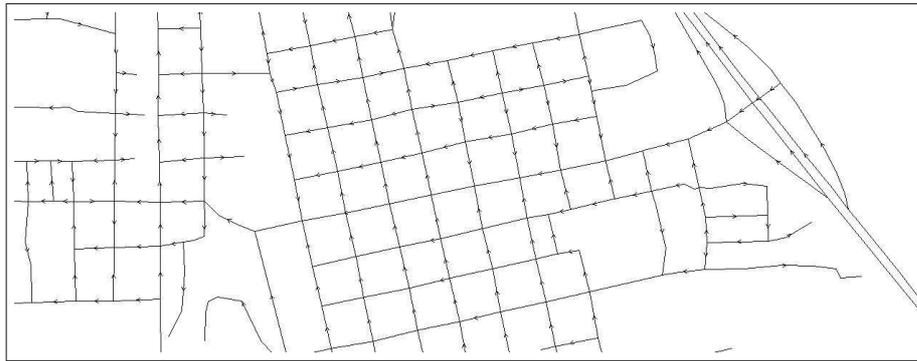
La herramienta de digitalización es `v.digit`, que ha sido completamente rescrita recientemente y ahora está totalmente basada en una interfaz gráfica. Los botones deben ser auto explicatorios. Para comenzar con un nuevo mapa, hay que añadir la opción `'-n'` (o activar el botón en el menú). Luego, en el menú de configuración, se puede definir una nueva tabla de atributos, junto con la distancia de ajuste. Se puede cargar un mapa de fondo de forma opcional antes de comenzar la digitalización. Las áreas (actualmente) tienen que ser digitalizadas en dos partes. Las áreas cerradas se vuelven verdes, mientras que las estructuras topológicas inválidas permanecen rojas. En este caso es recomendable hacer `z` para identificar el error. Una vez que el elemento ha sido digitalizado, una ventana se abre de modo que puedan ser introducidos los atributos del mismo. La figura 16 ilustra el ejemplo.

## 6.8. Digitalizar en QGIS

Una alternativa interesante es usar el digitalizador dentro de QGIS. Para hacerlo, primero hay que crear un mapa vacío:

```
v.in.ascii -e out=newmap
```

Luego inicie QGIS dentro de la terminal de GRASS y cargue este nuevo mapa o un mapa existente dentro de QGIS usando el icono de Vectorial de GRASS.



**Figura 17:** Direcciones nativas en datos vectoriales de GRASS usados para análisis de redes

## 6.9. Trabajar con geometrías vectoriales

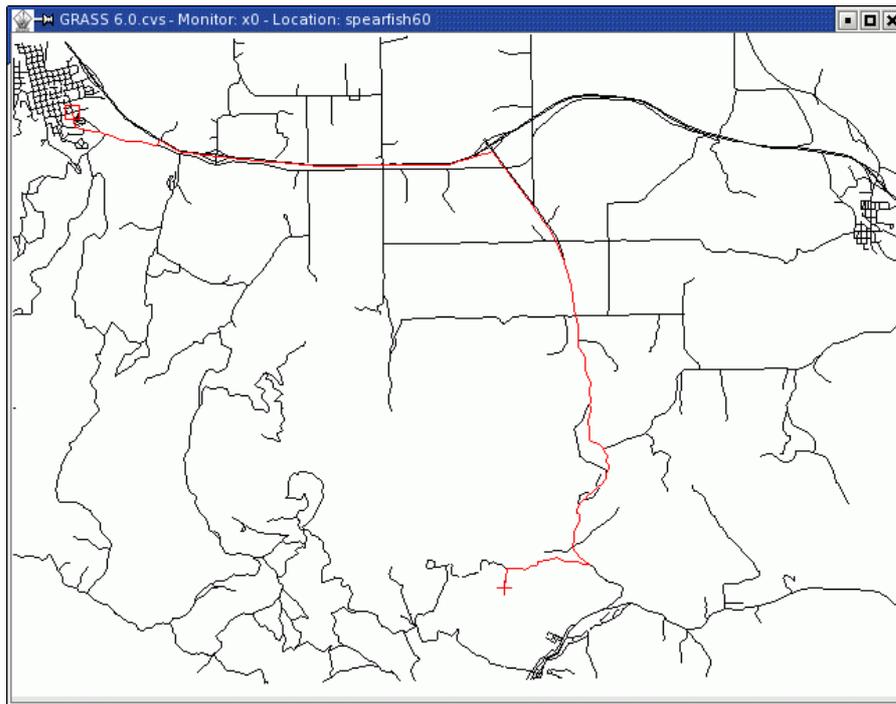
En GRASS un área poligonal se define por la suma de un contorno y un centroide. Las líneas pueden ser (poli) líneas o contornos.

Se encuentran varias herramientas disponibles para consultar y manipular geometrías vectoriales:

- `v.build`: generar topología (se hace automáticamente), escribe vectores con problemas en un nuevo mapa de error para ser inspeccionados luego.
- `v.build.polylines`: hace polilíneas de líneas vectoriales conectadas.
- `v.category`: muestra información de los identificadores vectoriales (llamados “categorías” o “cats” en GRASS), asigna automáticamente nuevas categorías a vectoriales y añade los centroides que falten.
- `v.clean`: limpia los problemas topológicos, nodos de ajuste, elimina irregularidades, áreas pequeñas, fisuras, recortes, etc.
- `v.to.db`: transfiere los valores del vectorial a la base de datos.
- `v.type`: convierte tipos de geometría vectorial (punto vs. centroide; punto 3D vs. kernel; líneas vs. contorno; área 3D vs. caras).
- `d.vect`: visualiza la dirección de las líneas vectoriales (indicadas por flechas pequeñas, vea la figura 17)

## 7. Análisis de redes

Un nuevo conjunto de módulos de GRASS soporta operaciones sobre redes vectoriales. Los cálculos por defectos están basados en las longitudes de los vectores. Pero es posible asignar atributos de costos a los nodos y para dos direcciones de cada línea vectorial (ej. para simular flujos de tráfico).



**Figura 18:** Cálculos para la ruta más corta

### 7.1. Análisis de la ruta más corta

La ruta de conexión entre dos puntos en una red vectorial se puede analizar gráficamente con `d.path`. El módulo necesita un monitor de GRASS abierto. Si no se especifica ningún atributo, únicamente se toman en consideración las longitudes de los vectores. Podemos experimentar con el mapa de carreteras:

```
d.path roads
```

Hay que utilizar los botones del ratón, su funcionamiento se explica en la consola de comandos. La ruta más corta es resaltada inmediatamente en el monitor de GRASS (ver Figura 18). Para guardar la ruta más corta hay que utilizar el módulo `v.net.path`.

### 7.2. Más herramientas de análisis de redes

Los siguientes métodos de análisis de redes vectoriales se encuentran implementados actualmente en GRASS:

- `v.net.path`: ruta más corta (conexión entre dos puntos).
- `v.net.salesman`: problema del vendedor viajero (viaje completo).
- `v.net.alloc`: colocación de recursos (crear subredes, ej. brigadas de bomberos).

- 
- `v.net.steiner`: árboles mínimos de Steiner (conexiones de tipo estrella, ej. cables de fibra óptica).
  - `v.net.iso`: iso-distancias (desde el centro).
  - `v.net`: mantenimiento de redes.

## 8. Interfaz entre GRASS y R-stats

La interfaz entre GRASS y el lenguaje estadístico R (R) está sufriendo actualmente cambios significativos (Bivand, 2005). El nuevo diseño está incluido dentro de los nuevos esfuerzos para desarrollar clases espaciales coherentes para R. La página web básica para la nueva interfaz con GRASS está alojada en SourceForge (<http://r-spatial.sourceforge.net>). El nuevo paquete se llama “spgrass6”. Las clases de objetos espaciales básicos son mantenidos en el paquete “sp”. La clase adicional “spGDAL” es un empaquetador para las funciones en el paquete “rgdal” el cual se conecta con la librería GDAL. Otro paquete es “spmaptools” que se conecta con la librería SHAPE.

### 8.1. Instalación de los paquetes relacionados con R

La instalación se hace como sigue (se necesita una versión de R-stats superior a 2.1.0):

```
R
> install.packages(c("sp", "rgdal", "maptools"), dependencies=TRUE)
> rS <- "http://r-spatial.sourceforge.net/R"
> install.packages(c("spgrass6", "spGDAL", "spmaptools"),
                  repos=rS, dependencies=TRUE)
> q()
```

### 8.2. Sesión de ejemplo de R-stats/GRASS

Para tener una idea de como trabaja el lenguaje de R-stats, podemos iniciar GRASS/Spearfish para nuestra sesión de pruebas y volver a colocar los valores por defecto de las regiones, luego iniciamos R dentro de la consola de GRASS. Se muestran un par de comandos:

```
grass60
g.region -dp
```

```
R
> library(spgrass6)
> G <- gmeta6()
```

Los comandos anteriores cargan la interfaz y luego el ambiente de GRASS en la sesión de R. El siguiente comando muestra las variables de ambiente.

```
> str(G)
```

Ahora cargamos un mapa ráster de GRASS dentro de R:

```
> geology <- readCELL6sp("geology")
> summary(geology)
> str(geology)
```

Ya que estamos usando datos de SIG, queremos ver el mapa:

```
> image(geology, "geology", col = terrain.colors(10))
```

Para agregar una leyenda, primero tenemos que controlar el número de clases en el mapa geológico, luego visualizamos la leyenda:

```
> system("r.info -r geology")
> legend(c(590000, 605000), c(4912570, 4913850),
         legend = 1:9, fill = terrain.colors(10),
         cex = 0.8, bty = "n", horiz = TRUE)
> q()
```

La nueva interfaz R/GRASS6 puede cambiar en el futuro. Para mantener este manual corto, sugerimos que consulten más información acerca de herramientas geo-estadísticas con R-stats. Se pueden encontrar vínculos en la sección de Aplicaciones/Geoestadística del sitio web de GRASS.

## Conclusión

Este corto manual trató de mostrarle el poder de la nueva versión de GRASS 6. Esperamos que haya sacado ideas y motivos para usar GRASS en su propio trabajo futuro. Por favor, consulte las páginas Web regularmente ya que el desarrollo avanza rápidamente. ¡Y no dude en participar, enviar sus comentarios, sugerencias e incluso código fuente!

## Reconocimientos

El autor agradece a Ken Boss y Kristen Perry por los comentarios acerca de las versiones anteriores de este texto.

## Bibliografía

- Bivand, R., 2005, Interfacing GRASS 6 and R: Status and development direction. GRASS Newsletter June 2005, Vol.3, <http://grass.itc.it/newsletter/>

- GDF Hannover, 2005, An introduction to the practical use of the Free Geographical Information System GRASS 6.0. Published under GNU FDL, <http://www.gdf-hannover.de/literature>
- Neteler, M., H. Mitasova, 2004, Open Source GIS: A GRASS GIS Approach. 2nd Edition. 424 pages, Kluwer/Springer, ISBN: 1-4020-7088-8. Online supplement: <http://mpa.itc.it/grasstutor/>
- Erle, S., R. Gibson, J. Walsh, 2005, Mapping Hacks. 564 pages. 1st Edition June 2005 (est.) <http://mappinghacks.com/>, ISBN: 0-596-00703-5
- Mitchell, T., 2005, Web Mapping Illustrated: Using Open Source GIS Toolkits. 1st Edition June 2005 (est.), ISBN: 0-596-00865-1