



PROYECTO COFINANCIADO  
POR LA UNIÓN EUROPEA  
FONDO EUROPEO  
DE DESARROLLO REGIONAL

# *Curso de gvSIG Sextante*

## Índice de contenido

<b>SEXTANTE en gvSIG .....</b>	<b>3</b>
<b>Introducción.....</b>	<b>3</b>
<b>Elementos de SEXTANTE.....</b>	<b>3</b>
<b>Ficheros a utilizar.....</b>	<b>4</b>
<b>Ejercicio 1: Creación de un MDE .....</b>	<b>5</b>
MDE a partir de curvas de nivel.....	5
MDE a partir de puntos con cota.....	9
<b>Ejercicio 2: Cálculos ráster.....</b>	<b>14</b>
Cálculo de volúmenes .....	14
Cálculo de áreas de influencia.....	14
Cálculo de perfiles.....	17
<b>Ejercicio 3: Cálculo de mapas ráster.....</b>	<b>21</b>
Mapa de pendientes .....	21
Mapa de orientaciones.....	22
Mapa de cuencas visuales.....	23
Mapa de exposición visual.....	25
<b>Ejercicio 4: Herramientas básicas de capas ráster.....</b>	<b>28</b>
Recorte de un ráster.....	28
Vectorización de capa ráster.....	29
<b>Ejercicio 5: Modelizador gráfico de procesos.....</b>	<b>32</b>
Mapa de índice de protección desde curvas de nivel.....	32
<b>Referencias en internet .....</b>	<b>35</b>
<b>GNU GENERAL PUBLIC LICENSE .....</b>	<b>36</b>

## **SEXTANTE en gvSIG**

### **Introducción**

Para la realización de estos ejercicios se ha utilizado la versión 1.1.2 estable de gvSIG (build number 1045) más las siguientes instalaciones sucesivas:

- ✓ versión 0.2 de la extensión de gestión de sistemas de referencia,
- ✓ piloto Raster (build number 7),
- ✓ extensión Sextante, versión 0.15.

Cada ejercicio se compone de 3 apartados:

- el primero donde se explica el Propósito,
- el segundo donde se explica la cartografía y otros parámetros necesarios,
- y por último la secuencia de pasos hasta obtener el resultado deseado.

En dicha secuencia de pasos se han hecho algunas explicaciones más teóricas de conceptos de cálculo, con el único fin de dar más herramientas al lector para que pueda hacer más pruebas de las aquí descritas.

La cartografía que se utiliza en estos ejemplos tiene como créditos:

CARTOGRAFÍA DE CURVAS DE NIVEL DPTO. DE CANELONES

Propiedad: Ministerio de Transportes y obras Públicas. Uruguay.

Estos ejercicios se han preparado para el curso avanzado de gvSIG a realizar en Facultad de Ciencias de la Universidad de la República (República Oriental del Uruguay), con aplicación al proyecto C ("*Desarrollo de instrumentos para el monitoreo ambiental y territorial*") en el marco del Programa Piloto "Unidos en la Acción" que el Gobierno Uruguayo y el Sistema de Naciones Unidas.

### **Elementos de SEXTANTE**

SEXTANTE se compone de un conjunto de extensiones (más de 228), cada una de las cuales implementa un proceso de análisis basado en datos espaciales, y que amplía las capacidades de gvSIG como SIG de escritorio, especialmente en lo relativo al análisis de datos geográficos.

El Gestor de Extensiones es el elemento principal para el control de las extensiones o módulos. Este gestor conforma un conjunto de herramientas con todas las extensiones de SEXTANTE, que pueden ejecutarse desde el mismo de forma individual. Cada herramienta también puede ejecutarse por lotes sin necesidad de tener la cartografía cargada en gvSIG previamente.

El Modelizador gráfico es una herramienta que permite la creación de modelos complejos mediante una interfaz sencilla, simplificando procesos que impliquen el uso de varias extensiones de SEXTANTE de forma encadenada.

La línea de comandos de SEXTANTE permite a los usuarios más avanzados hacer un uso más ágil del programa y automatizar tareas mediante la creación de sencillos scripts. En caso de estar haciendo pruebas con algún algoritmo en particular, gracias a la línea de comandos no es necesario repetir la introducción de los parámetros relacionados con el algoritmo. En la Ayuda de SEXTANTE se encuentra la sintaxis de los comandos mediante los cuales se debe llamar a cada algoritmo de cálculo.

El historial de procesos registra los distintos procesos que se han llevado a cabo con SEXTANTE, bien desde el gestor de extensiones o bien desde la línea de comandos. En caso de haber ejecutado las acciones desde línea de comandos, además de poder ver las diferentes acciones que hemos realizado con SEXTANTE podremos ejecutarlas sistemáticamente, facilitando la repetición de procesos.

## Ficheros a utilizar

La siguiente tabla resume los ficheros de cartografía y definición de sistema de referencia, necesarios para la realización de este curso.

Ficheros	Ej1	Ej2	Ej3	Ej4	Ej5	Observaciones
alalti_trozo.shp	√				√	Campo COTA en tabla atributos.
Yacare.txt	√					Definición CRS Yacaré (ROU - USAMS)
poligono_buffer.shp		√				Polígonos a rasterizar para Buffer
perfil.shp		√				Línea de perfil longitudinal
MDT_sin_depresiones.tif		√	√	√		Georreferenciado internamente
ruta_exposicionvisual.shp			√			
pol_recorte.shp				√		Polígono para recortar el MDT
ruta_exposicionvisual_raster.tif				√		
polignos_raster.tif				√		

## Ejercicio 1: Creación de un MDE

### MDE a partir de curvas de nivel

#### Propósito:

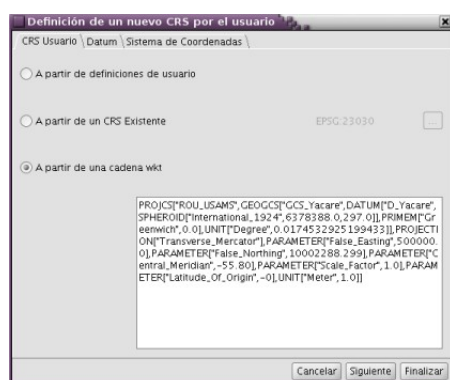
Crear un Modelo Digital del Elevaciones a partir de un fichero vectorial de curvas de nivel. Se rellenarán los valores de celdas sin datos por 2 caminos: definiendo un Umbral de tensión y por el método de vecindad. El MDE resultante se reclasificará para obtener un ráster discreto.

#### Cartografía necesaria:

Capa vectorial de curvas de nivel de la zona de estudio con campo COTA con altura de las curvas de nivel, como por ejemplo *alalti\_trozo.shp*. Además serán necesarios los parámetros de definición del CRS Yacaré (ROU - USAMS).

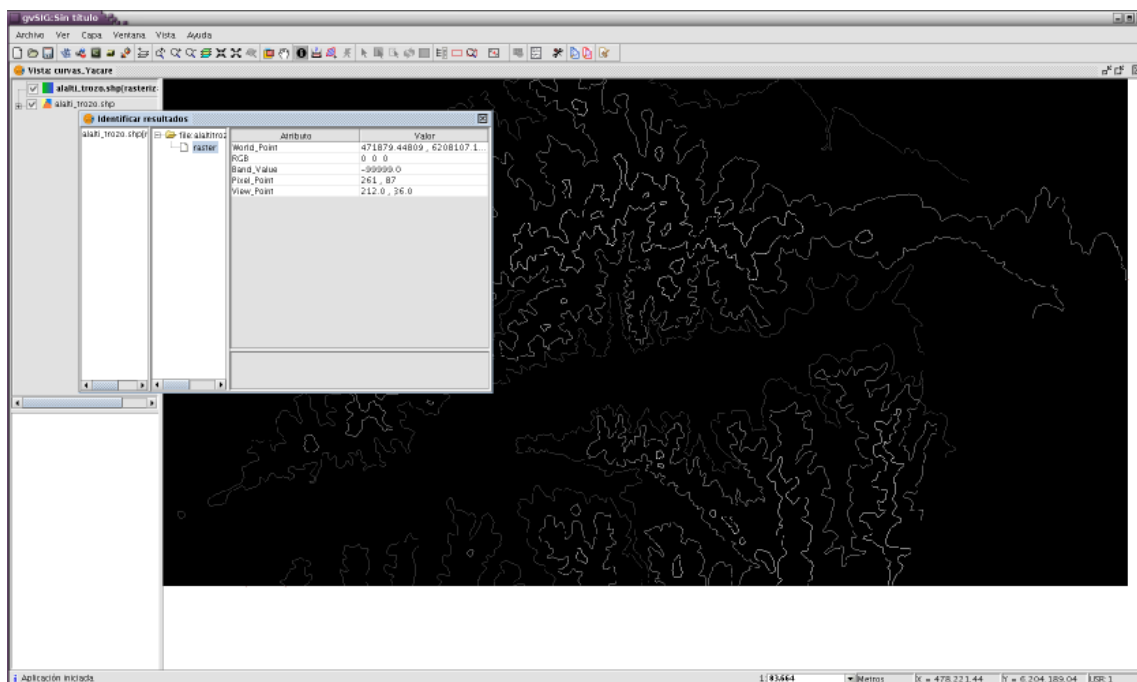
#### Secuencia:

1. **Creación del sistema de referencia:** crear una vista nueva y desde sus Propiedades crear CRS de usuario con parámetros de Yacaré. Estos parámetros están disponibles en el fichero *Yacare.txt*. Para ello seleccionar Tipo CRS de Usuario y pinchar sobre el botón Nuevo. Seleccionar la creación a partir de una Cadena wkt y pegar el el cuadro el contenido del fichero *Yacare.txt*.



En las siguientes pestañas comprobar que los parámetros del CRS creado son efectivamente los del sistema de referencia Yacaré.

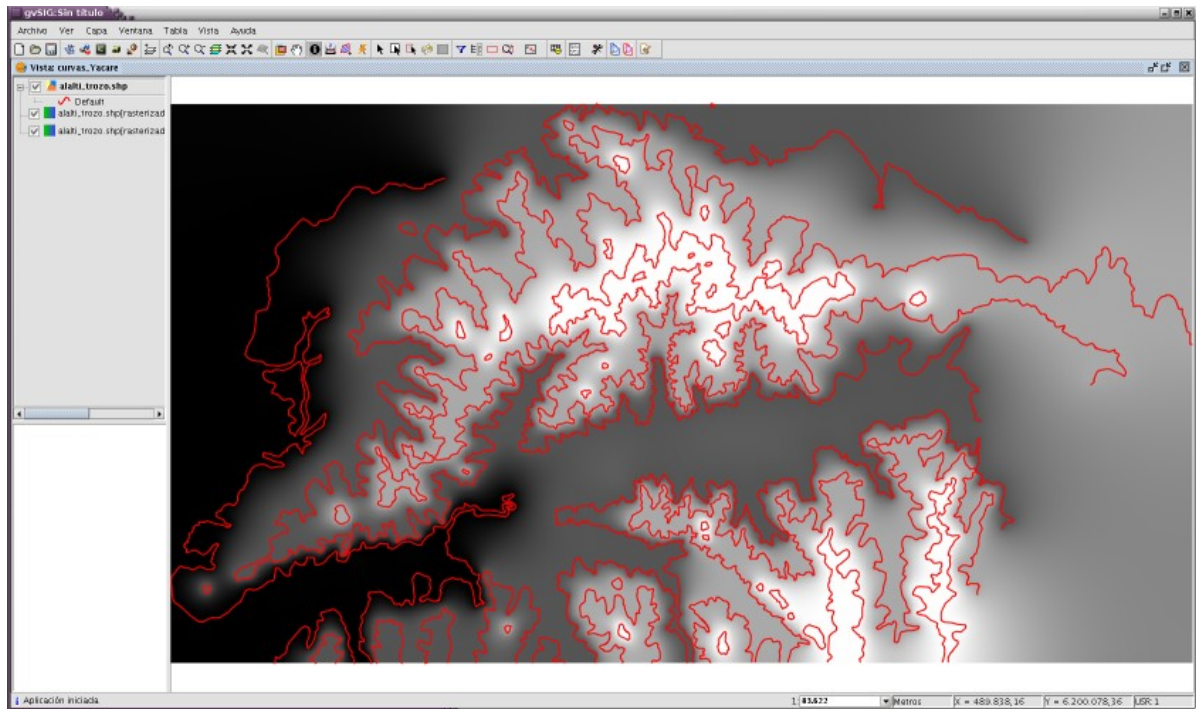
2. **Añadir capa vectorial:** abrir la capa de curvas de nivel *alalti\_trozo.shp*. Trabajaremos con líneas de cota, explorar tabla atributos de la capa exportada.
3. **Rasterización de la capa:** desde Sextante, módulo Rasterización e Interpolación, seleccionar Rasterizar capa vectorial. Campo a conservar COTA, se utilizará la extensión de la capa vectorial, con una resolución de 15 m. y generando archivo temporal.



4. **Propiedades del fichero generado:** ver en Propiedades del ráster el número de bandas , la extensión del ráster (Sextante genera .tiff georreferenciados), ubicación en disco duro, etc.
5. **Valores de celda:** observar que los valores en color negro son valores sin dato. Usar herramienta info para consulta de píxeles con dato y sin dato.
6. **Rellenar celdas sin datos:** desde Sextante, módulo Herramientas básicas para capas raster, seleccionar Rellenar celdas sin datos. Definir el Umbral de tensión como "0.5".

Umbral de tensión: este parámetro regula la forma en que la información local disponible se utiliza para rellenar los huecos, utilizando algoritmos de curvas adaptativas (splines, funciones polinómicas por tramos) con tensión. Gracias al parámetro de tensión se pueden controlar las oscilaciones artificiales que pueden aparecer en puntos cercanos con variabilidad importante del valor a interpolar. Valores altos de tensión implican un tiempo proceso menor. El resultado es un ráster continuo.

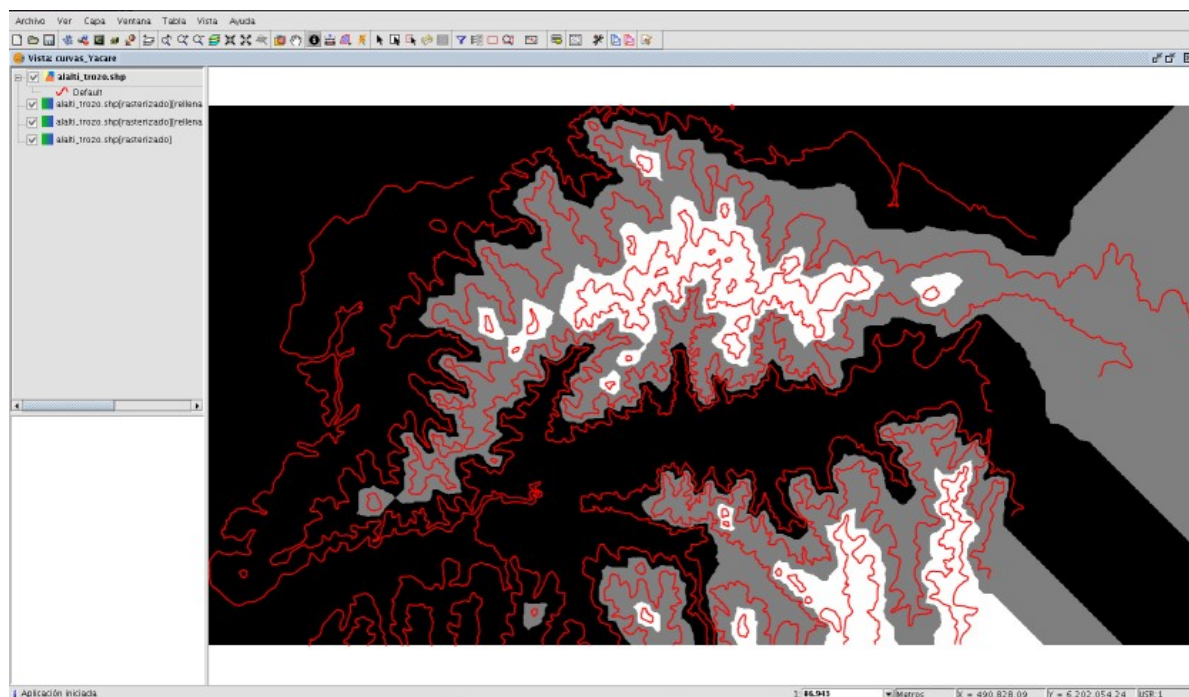
7. **Resultado del rellenado:** observar el resultado sin límites precisos, este es el resultado que se espera de la interpolación de una variable continua. Ver también las zonas de diferente cota en la capa ráster y su diferencia respecto a las curvas de nivel.



8. **Rellenar celdas sin datos:** desde Sextante, módulo Herramientas básicas para capas ráster, seleccionar Rellenar celdas sin datos (por vecindad)

Interpolación por vecindad (vecino más próximo): este algoritmo de interpolación local asocia a cada celda el valor del punto a menor distancia de dicha celda a rellenar. El resultado es un ráster discreto y por tanto este método es adecuado para el cálculo de variables categóricas.

9. **Resultado del rellenado:** observar que el resultado es más nítido en los bordes de clases de diferentes cotas, resultado que se espera de la interpolación de una variable discreta (o discontinua).



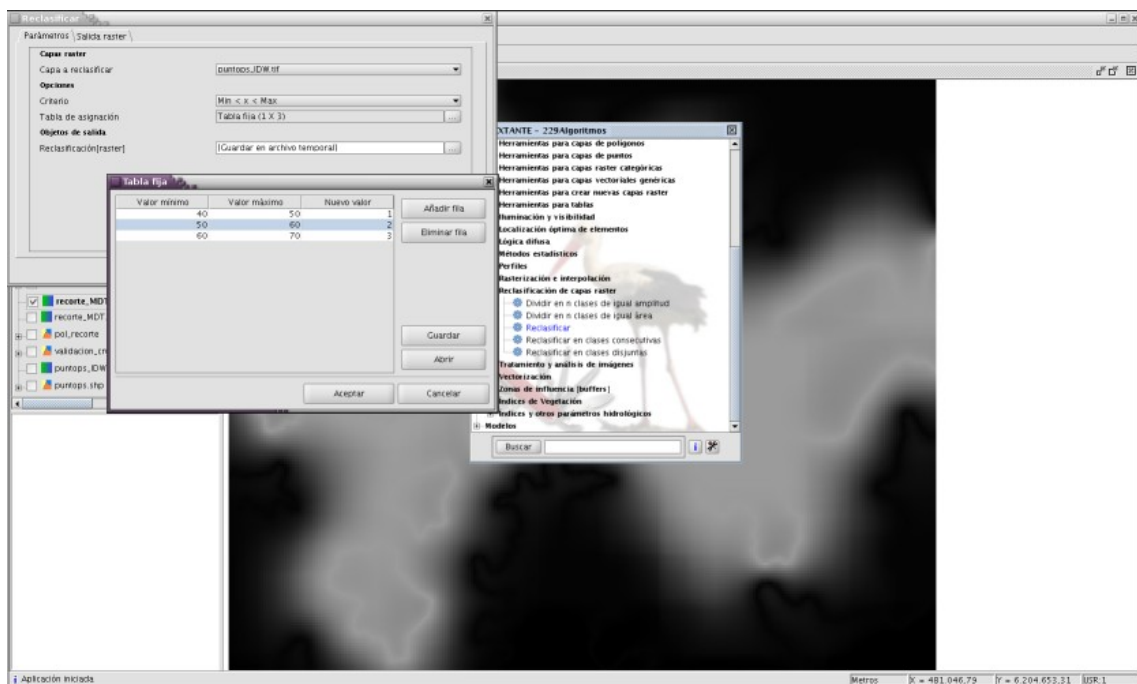
10. **Comprobación del relleno:** comparar los valores de cotas del raster respecto de la capa vectorial original. Quitando el realce lineal, desde Propiedades del ráster/Realce, se aprecian las 4 categorías de cotas.

Tipos de interpolación: la bondad de cada método no va sólo ligada al método en sí, sino también a la variable que se esté interpolando y al uso que se de del resultado de la interpolación.

11. **Histograma del MDE:** para ver los valores de cotas disponibles, desde Sextante abrir el módulo Herramientas básicas para capas ráster, seleccionar Histograma. Los resultados acumulados de la sesión están disponibles desde el icono de resultados de Sextante.
12. **Reclasificación del MDE:** desde Sextante abrir módulo Reclasificación de capas raster, seleccionar Reclasificar. Se aplicará este algoritmo sobre el MDE con valores continuos (relleno por umbral de tensión). Para decidir los intervalos de cada nueva clase, usar el Histograma del MDE. La tabla de asignación se definirá con el número de clases resultantes y seleccionar como criterio  $\text{Min} < x \leq \text{Max}$ . Ajustar la capa de salida a los datos de entrada. Volver a calcular el histograma para comprobar que los valores se hayan modificado correctamente.

Reclasificación: es el procedimiento por el cual se modifican los valores de celda del ráster. Se deben definir el criterio de asignación de valores y los valores mínimo, máximo y Nuevo de cada clase.





## MDE a partir de puntos con cota

### Propósito:

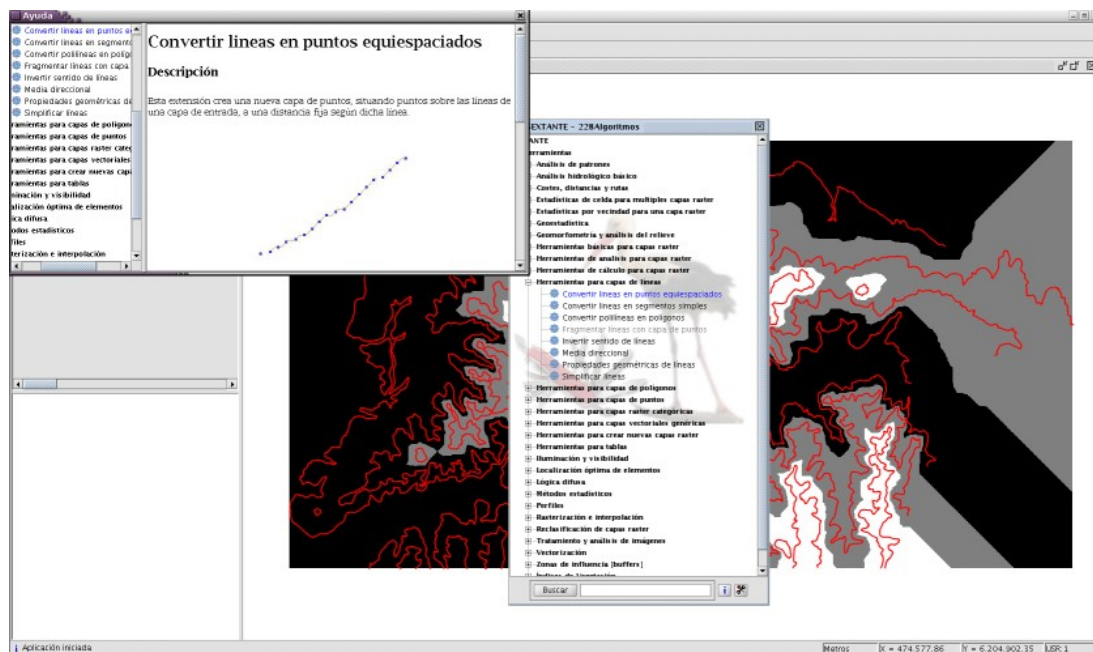
Crear un Modelo Digital del Elevaciones a partir de un fichero vectorial de puntos con cota. Se rellenarán los valores de celdas sin datos por 3 caminos: interpolando por el método de Distancia inversa (IDW), por el método de Decremento lineal y por el método de Kriging.

### Cartografía necesaria:

Capa vectorial de curvas de nivel de la zona de estudio con campo COTA con altura de las curvas de nivel, como por ejemplo *alalti\_trozo.shp*.

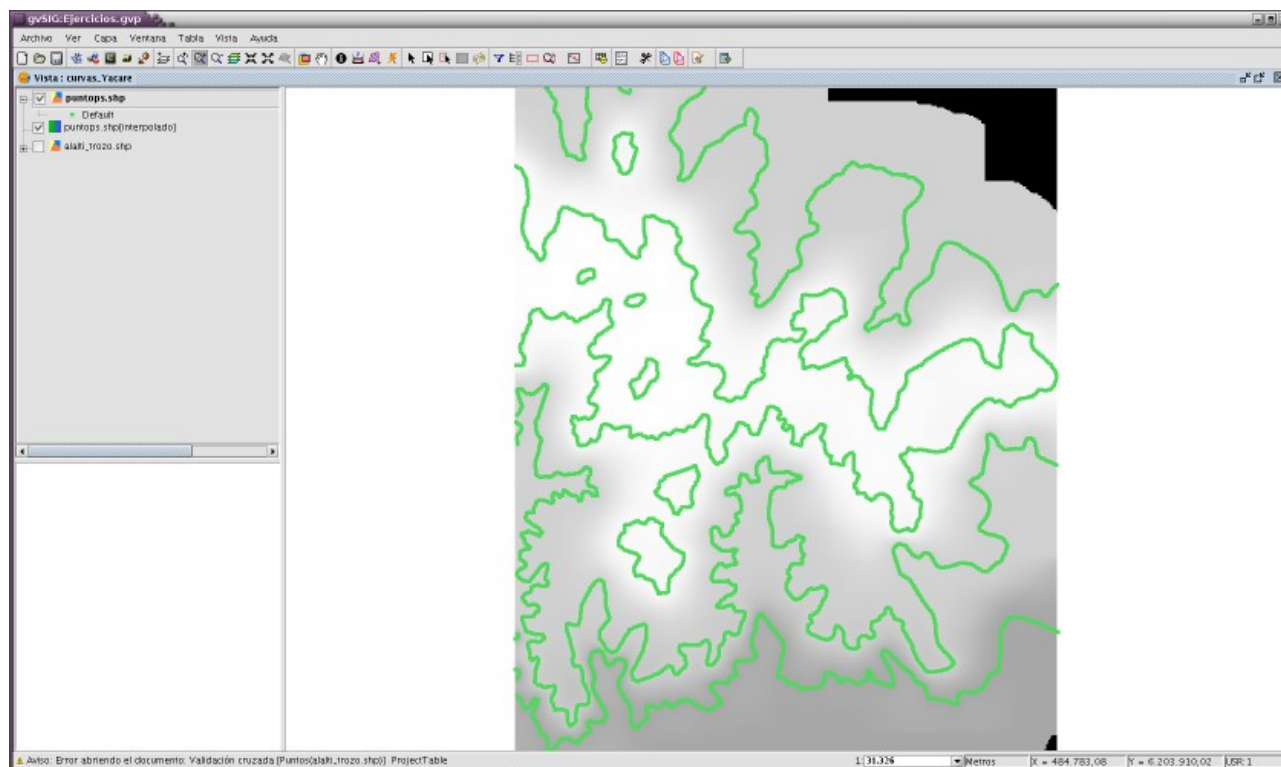
### Secuencia:

1. **Generar capa de puntos equidistantes:** Transformar la capa original de curvas de nivel en puntos equidistantes. Desde Sextante, abrir el módulo Herramientas para capa de líneas y seleccionar Convertir líneas en puntos equiespaciados. Poner equidistancia entre puntos a 20 m, por ejemplo. Ver que la tabla de atributos ha conservado el campo COTA.



2. **Interpolación por Distancia inversa:** Interpolamos valor de COTA de los puntos y generamos nuevo ráster MDE interpolado. Desde Sextante, abrimos módulo de Rasterización e interpolación, seleccionamos método de Distancia inversa, con Radio de búsqueda: 500 m, exponente 2 y tamaño de píxel 10 m. Seleccionamos Utilizar la extensión de otra capa, refiriéndonos a la capa de puntos recién creada.

Interpolación por distancia inversa: esta interpolación es de tipo local y determinística, acotada por el radio de búsqueda. Se debe utilizar un radio mínimo que asegure que alrededor de todas las celdas se encuentra un número suficiente de puntos. Si ningún punto cae dentro del radio, la interpolación dará como resultado una celda sin datos (valor -99999.0). El parámetro exponente es el que se utiliza en el cálculo de los pesos de forma inversamente proporcional a la distancia. Este tipo de interpolación tiene en cuenta para la ponderación (asignación de pesos) el alejamiento entre las celdas pero no su posición. El ráster resultante es de tipo continuo.



3. **Tabla validación cruzada:** Para consultar la tabla de validación cruzada, hacer Ver/Gestor de proyectos y en el lista de documentos Tabla seleccionar la última tabla creada.

Validación cruzada: valores que sirven para verificar la calidad de los valores interpolados. Se hace una estimación de los valores (mediante la interpolación) en una serie de puntos de control (o muestreo) de los cuales se conoce su valor real. El valor estimado es el valor que le correspondería a un punto si se interpolan todos los puntos menos el punto en cuestión.

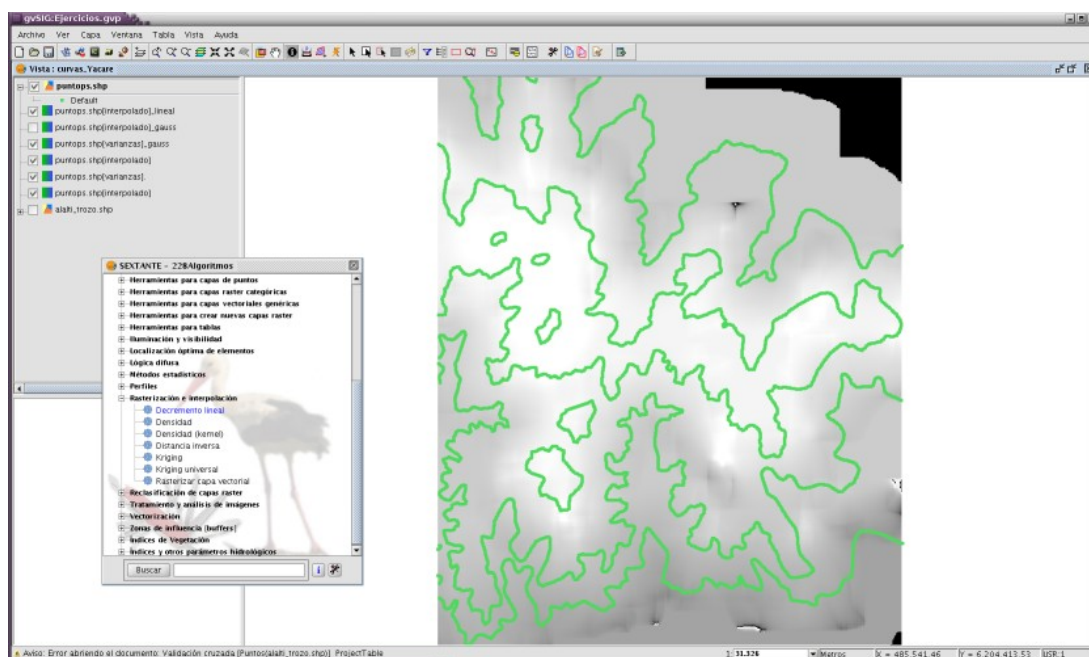
La tabla de validación cruzada muestra la diferencia entre valor real e interpolado para todos los puntos con valores utilizados para la interpolación.

Tabla: Validación cruzada (puntos.shp)					
X	Y	Valor real	Valor esti.	Diferencia	
480421.0	6206715.0	60.0	59.679297	-0.320703	
480405.058157	6206727.077154	60.0	59.999979	-2.1E-5	
480389.116314	6206739.154308	60.0	59.999922	-7.8E-5	
480396.918926	6206754.124295	60.0	59.999889	-1.11E-4	
480392.068214	6206773.727145	60.0	59.999809	-1.91E-4	
480381.642615	6206790.476514	60.0	59.999997	-1.0E-6	
480369.642615	6206806.476514	60.0	59.999997	-1.0E-6	
480358.372214	6206822.940338	60.0	59.999984	-1.0E-5	
480348.979955	6206840.597686	60.0	59.999955	-4.5E-5	
480339.587695	6206858.255133	60.0	59.999695	-3.05E-4	
480340.074984	6206876.668272	60.0	59.999947	-1.53E-4	
480346.66864	6206895.551107	60.0	59.999985	-1.5E-5	
480353.262297	6206914.432942	60.0	59.999974	-2.6E-5	
480357.168979	6206933.567838	60.0	59.999973	-2.7E-5	
480354.216608	6206953.148725	60.0	59.999865	-1.35E-4	
480351.264237	6206973.129633	60.0	59.999896	-1.04E-4	
480348.311866	6206992.9105	60.0	59.999949	-5.1E-5	
480361.295566	6207006.966009	60.0	59.999992	-8.0E-6	
480377.136081	6207019.090657	60.0	59.999951	-4.0E-5	
480395.777007	6207030.184671	60.0	59.9999	-2.0E-4	
480410.418013	6207041.278675	60.0	59.999713	-2.87E-4	
480429.259353	6207044.079232	60.0	59.999966	-3.4E-5	
480440.211391	6207042.69366	60.0	59.999804	-1.96E-4	
480469.16325	6207041.308108	60.0	59.999811	-1.89E-4	
480488.698336	6207039.127079	60.0	59.999916	-8.4E-5	
480494.929556	6207021.757583	60.0	59.999697	-3.03E-4	
480492.353785	6207001.924142	60.0	59.999971	-2.9E-5	
480489.778013	6206982.0907	60.0	59.999959	-4.1E-5	

4. **Interpolación por decremento lineal:** desde la capa de puntos volver a interpolarse pero esta vez con la opción de Decremento lineal. Como parámetros

utilizar por ejemplo: radio de búsqueda 500 m y exponente 2. Extensión de la capa de puntos y resolución 10 m.

**Decremento lineal:** esta interpolación es de tipo local y determinística, acotada por el radio de búsqueda. Se diferencia de la interpolación por distancia inversa en la función utilizada para el cálculo de los pesos. En caso de que el exponente sea 1, la función de pesos es lineal.



5. **Tabla de validación cruzada:** investigar las estadísticas de la tabla de validación asociada abriendo la tabla, seleccionando el campo de Diferencia y haciendo Tabla/Estadísticas.
6. **Interpolación por método Kriging:** desde la capa de puntos, volver a interpolar pero esta vez con la opción de Kriging.

**Kriging:** este método estocástico (no determinístico) puede aplicarse tanto de forma global como de forma local. Gracias a este método se consigue una interpolación donde se conoce el error cometido en la predicción. La ponderación de los valores a estimar se hace a través de un variograma teórico, ya que a través de éste se puede describir la correlación espacial de los datos. Los parámetros Nugget, Sill, Rango y modelo caracterizan el variograma:

*Nugget:* variabilidad en los valores estimados que no puede explicarse mediante la estructura espacial.

*Sill:* Máxima variabilidad en ausencia de dependencia espacial

*Rango:* Máxima distancia a partir de la cual desaparece la correlación espacial.

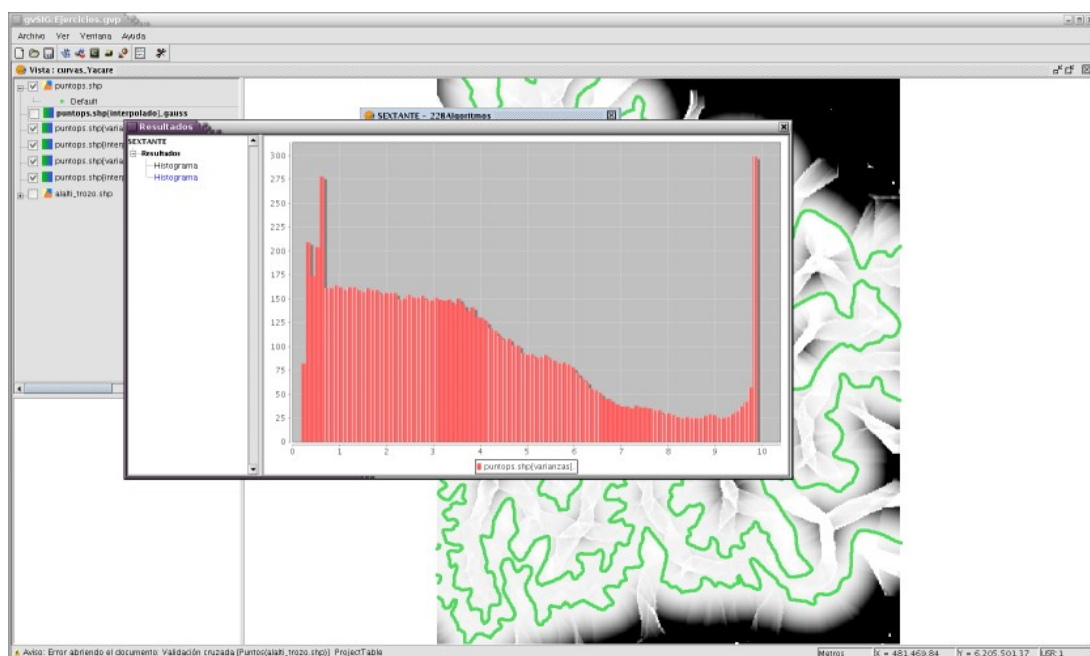
*Modelo:* función que se utiliza para definir el variograma.

El error cometido en la interpolación viene dado en la capa de varianzas. Esta capa representa una medida del error que se ha cometido al calcular la predicción de valores, y puede ser utilizada para conocer la bondad de los cálculos en las distintas zonas.

7. **Parámetros método Kriging:** utilizar Radio y Rango 500 m, mínimo número de puntos 4 y máximo 25, modelo de variograma esférico, nugget 0.0 y Sill 10.0. Resolución de 10 m y la extensión de la capa original de los puntos de cota. Repetir proceso pero con modelo gaussiano y mismos parámetros anteriores.

*Nota: se han dejado los parámetros por defecto que caracterizan el modelo de variograma, en lugar de estimarlos en base a los datos de cota originales. Para el proceso de cálculo de los parámetros del variograma empírico puede utilizarse la extensión Autocorrelación espacial presente en el módulo Herramientas para capa de puntos.*

8. **Resultados método Kriging:** a los datos de salida (ráster interpolado y tabla de validación cruzada) se añade una capa ráster con las varianzas de los valores de COTA interpolados. Utilizar la herramienta de Histograma sobre la capa de varianzas para ver el rango de valores de dichos errores.



9. **Salvar a disco duro los MDE creados:** si hemos generado las capas en temporales, tendremos que hacer Capa/Exportar a/Raster (formato tif) habiendo previamente seleccionado la capa ráster en el TOC. En la ventana de diálogo de Exportar a ráster seleccionaremos desde la Vista la zona a exportar.
10. **Salvar a disco duro las tablas creadas:** crearemos una capa de eventos a partir de la tabla y ésta puede exportarse desde Capa/Exportar a/ a los formatos vectoriales soportados por gvSIG.

## Ejercicio 2: Cálculos ráster

### Cálculo de volúmenes

**Propósito:**

Calcular el volumen de terreno existente entre un nivel de referencia y la superficie definida por el MDE.

**Cartografía necesaria:**

MDE generado a partir de las curvas de nivel, o bien MDE generado a partir de puntos con cota espaciados de forma equidistante. Puede usarse también el fichero *MDT\_sin\_depresiones.tif*.

**Secuencia:**

1. **Cálculo del volumen:** utilizaremos el algoritmo Cálculo de volúmenes (Herramientas básicas raster). Pondremos como capa de entrada el ráster interpolado y como nivel de referencia la cota 10.0 m. Queremos que calcule todo el volumen comprendido entre los 10.0 m y el MDE que le proporcionamos.
2. **Resultado:** las unidades del resultado dependerán de las unidades que las del tamaño de celda de nuestro MDE. El resultado puede ser recuperado pulsando el icono de resultados de Sextante.

### Cálculo de áreas de influencia

**Propósito:**

Calcular las áreas de influencia sobre polígonos en formato ráster. Normalmente se piensa esta funcionalidad sobre formatos vectoriales, pero también existen algoritmos para aplicarla sobre capas ráster.

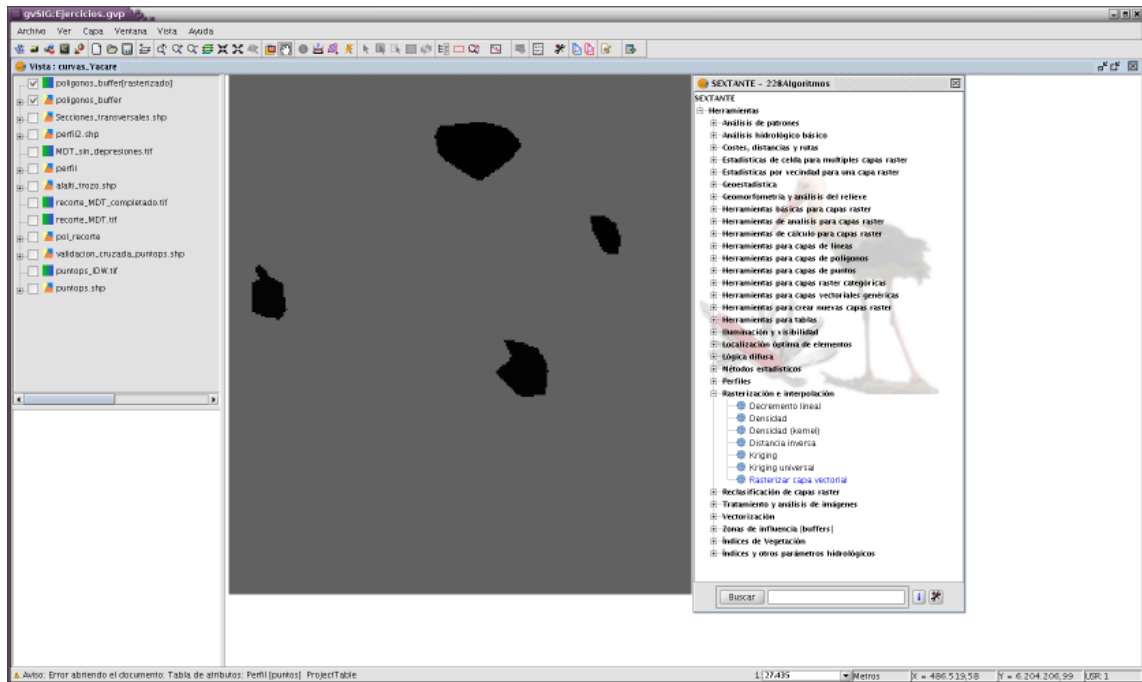
**Cartografía necesaria:**

Se utilizará una capa de polígonos que solapen parte de la extensión del MDE, con campo ID (de tipo integer), como por ejemplo *poligono\_buffer.shp*. Además se utilizará también el fichero *MDT\_sin\_depresiones.tif*

**Secuencia:**

1. **Rasterizar capa vectorial de polígonos:** hacerlo con el módulo de Rasterización e interpolación. Esta capa es la base para el buffer. Tener la precaución de poner campo ID como campo a conservar en la capa ráster, y la extensión igual a la del MDE sobre el que estamos trabajando. Poner el tamaño de celda a 10.0 m.
2. **Cambiar Propiedades de la capa ráster de polígonos:** una vez calculada la capa ráster, puede verse completamente negra. Esto ocurre porque la opción Realce de la capa está activa por defecto. Desde el menú contextual, desactivarla en la solapa Realce.



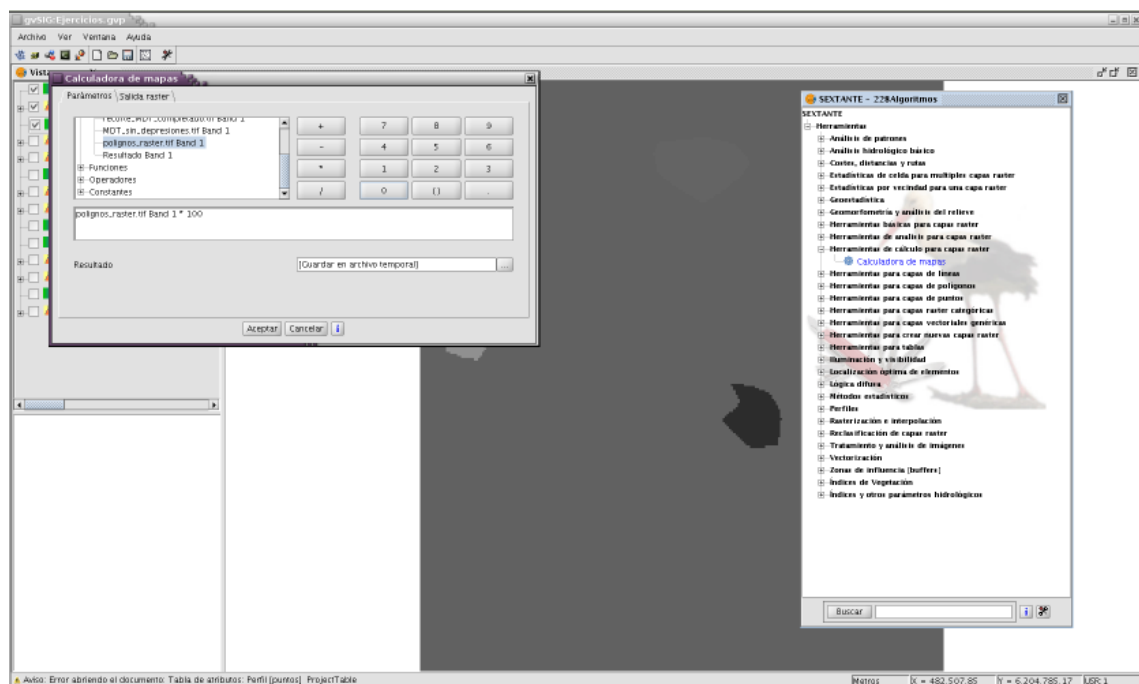


3. **Buffer sobre polígonos:** el buffer sobre estos polígonos se calcula con el módulo Zonas de influencia, seleccionando zona de influencia. Seleccionamos la capa con los polígonos, y la opción de usar valor de celda como distancia. La salida tendrá el mismo extent que el ráster original. El resultado no difiere en nada a la capa de entrada. Esto sucede porque el máximo valor de ID es 4 que no supera el tamaño del pixel, por lo que ninguno de los polígonos presenta buffer realmente.
4. **Aumentar valor de buffer:** ampliaremos el valor asociado a cada polígono de la capa ráster, con el módulo Herramientas de cálculo para capas ráster, seleccionando calculadora de mapas .

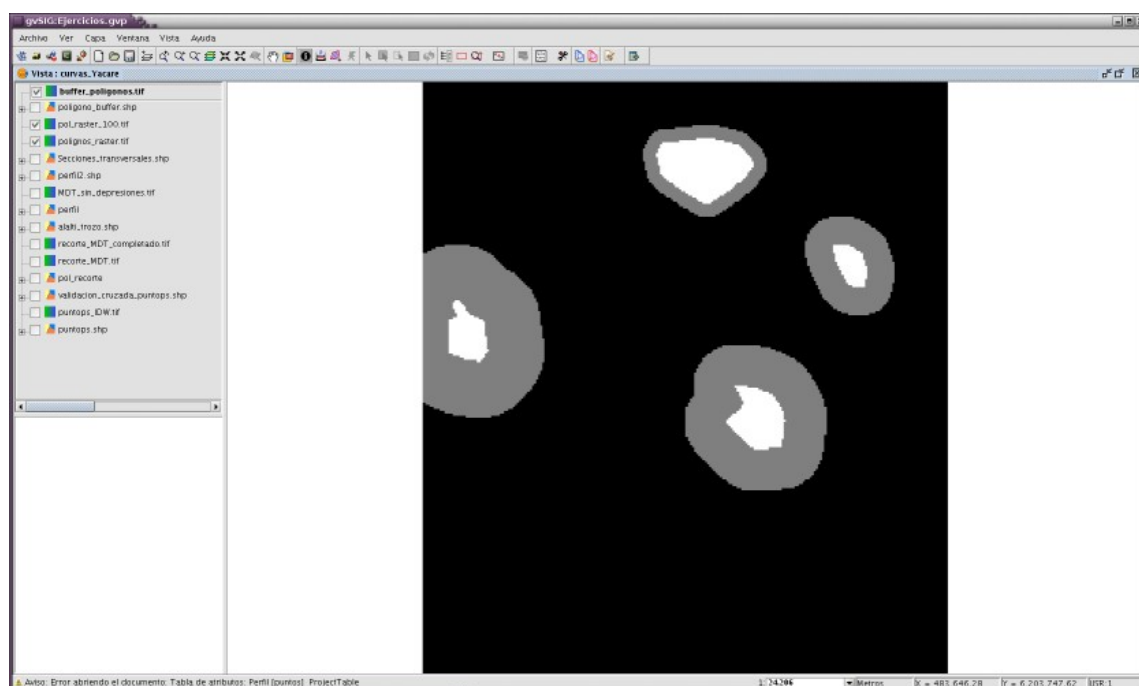
Calculadora de mapas: con esta herramienta se pueden hacer cálculos algebraicos sobre los valores asociados a una capa ráster. Para los cálculos pueden ser seleccionados distintos tipos de datos: capas, funciones, operadores y constantes.

Multiplicar por 100, por ejemplo, el valor de la capa de polígonos. Comprobar que los valores de píxel asociados a los polígonos son: 100, 200, 300 y 400 respectivamente.

Hacer nuevamente el buffer sobre los polígonos desde el módulo Zonas de influencia, seleccionando zona de influencia.



5. **Comprobar cálculo:** comprobar con la herramienta Información que el cálculo se ha hecho correctamente. Lo mismo puede hacerse calculando el histograma de nueva capa ráster (valores: 2 en interior polígonos, 1 en buffer, 0 en el resto).



6. **Zona de influencia mediante umbral:** el otro algoritmo que podemos usar para buffers ráster es el de Zona de influencia mediante umbral. Existen 2 tipos de umbral que podemos usar: absoluto y relativo.

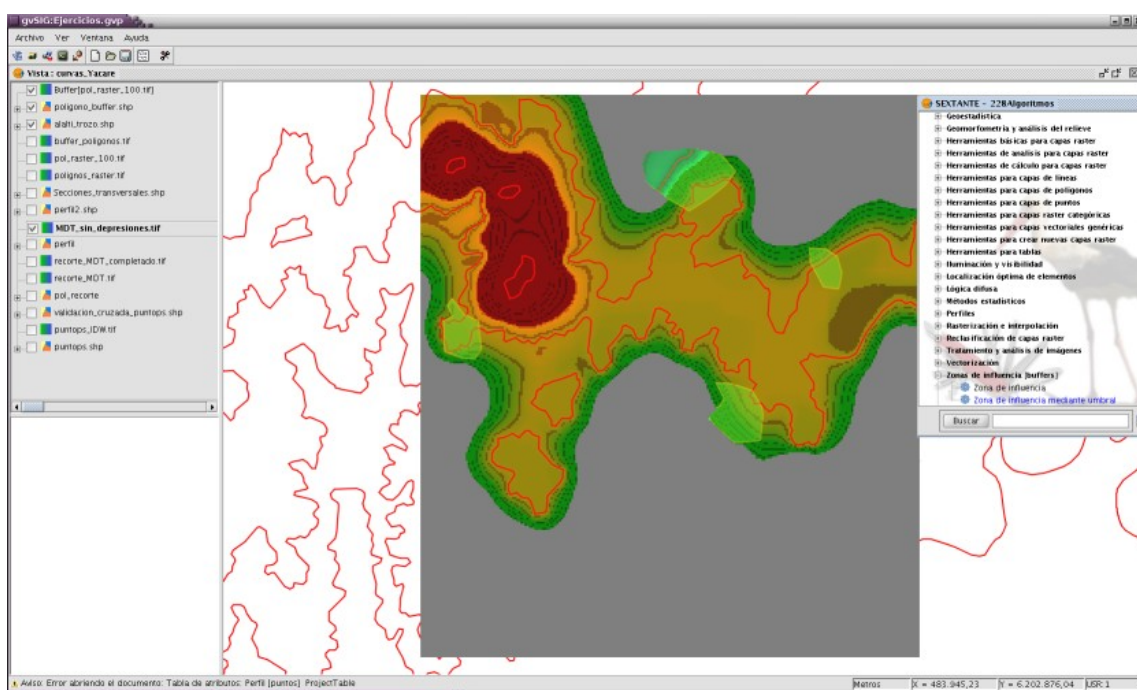
Umbral absoluto: empleando este umbral, todas las celdas circundantes hacia las



que se expande el buffer pasarán a formar parte de éste si el valor en las mismas es menor que dicho umbral.

Umbral relativo: se seleccionan para el buffer en este caso las celdas contiguas para las que la diferencia entre cada una de ellas y la celda inicial es menor que el umbral definido.

Utilizaremos la capa ráster donde tenemos definidos los polígonos (los vectores rasterizados), la capa de parámetro umbral será el MDE, y el cálculo según umbral absoluto de 55.0 m (para saber qué valor poner, investigar las cotas de los polígonos que se quiere extender con el buffer). El extent será el mismo que el del ráster de polígonos. En la imagen siguiente, la zona gris es la correspondiente a los buffer extendidos desde los polígonos que cumplen que la cota es menor de 55.0 m.



## Cálculo de perfiles

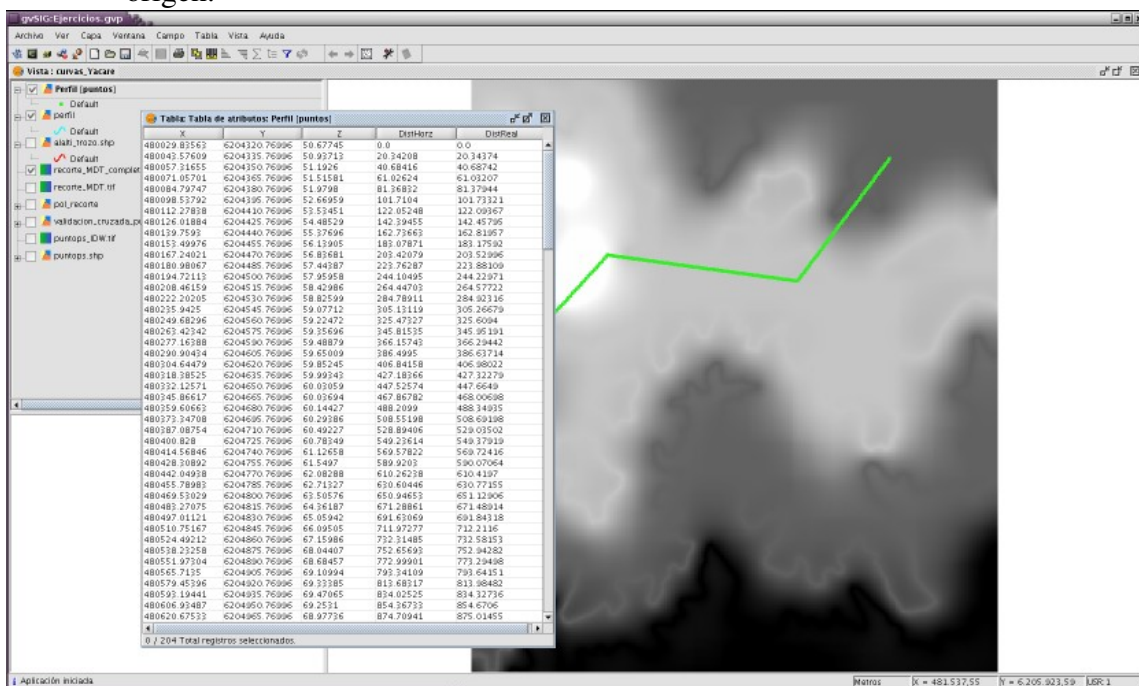
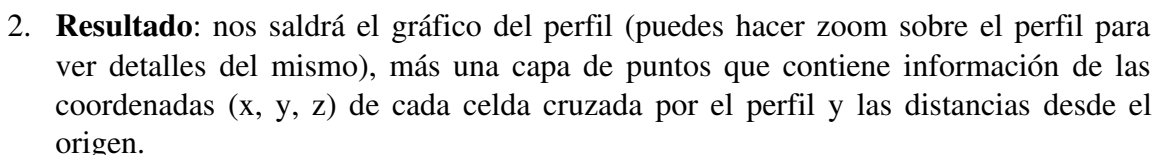
### Propósito:

Calcular el perfil longitudinal de una ruta que une dos puntos. También se calcularán los perfiles transversales a dicha ruta. Calculando el perfil longitudinal que une un punto con otro (o incluso una ruta compuesta por una serie de segmentos lineales) nos permite saber si la línea que los une pasa por terreno llano o no.

### Cartografía necesaria:

Es necesaria una capa vectorial lineal que represente la ruta desde A hasta B. Para ello utilizar la capa *perfil.shp* que contiene los campos ID y nombre. Además será necesario el MDE, como por ejemplo *MDT\_sin\_depresiones.tif*.

1. **Cálculo de perfil longitudinal:** abrir el módulo de Perfiles de Sextante, y seleccionar Perfil longitudinal. Utilizaremos el MDE y la planta del perfil que acabamos de definir.

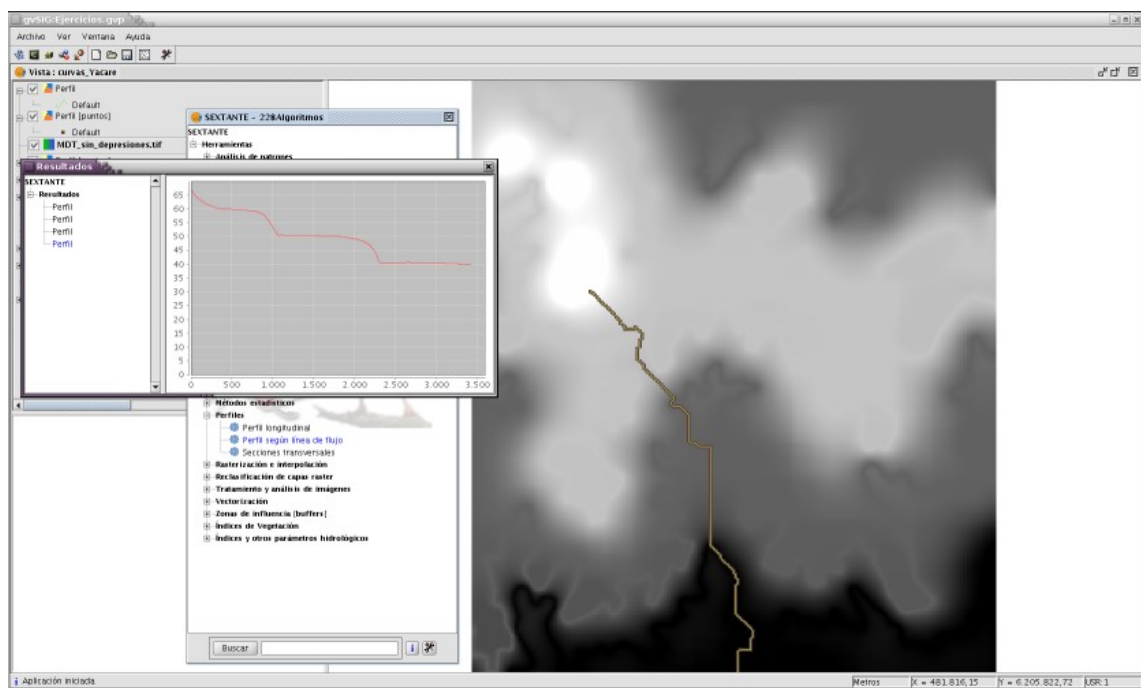


Para guardar esta capa de forma permanente es necesario exportarla a disco duro.

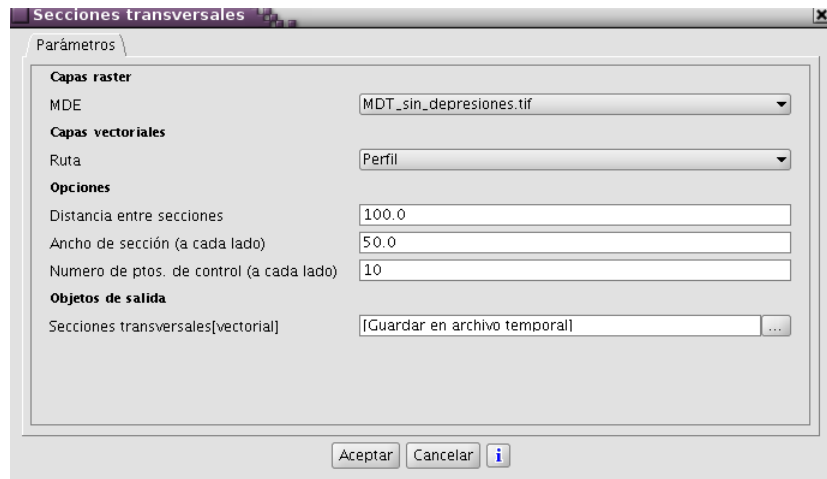
3. **Eliminar depresiones del MDE:** es necesario tratar previamente el MDE utilizando el módulo Eliminar depresiones (de Análisis hidrológico básico) ya que sino el cálculo del perfil puede quedar trunco. Dejaremos el ángulo de inclinación por defecto. Este ángulo hace referencia a la inclinación que queremos que se rellene cada celda que conforma la depresión.
4. **Perfil por línea de máxima pendiente (aguas abajo):** utilizaremos el módulo de Perfil según línea de flujo. Nos hará falta consultar las coordenadas del punto desde el cual queremos el perfil. Utilizar las coordenadas (480733; 6204952).

Perfil según línea de flujo: en caso de que el perfil resulte muy pequeño puede ser debido a que no se ha procesado previamente el MDE para eliminar depresiones. En este caso el algoritmo se trunca al encontrar valores de celdas menores que todas las de su entorno, y por lo tanto no puede proseguir con el cálculo.

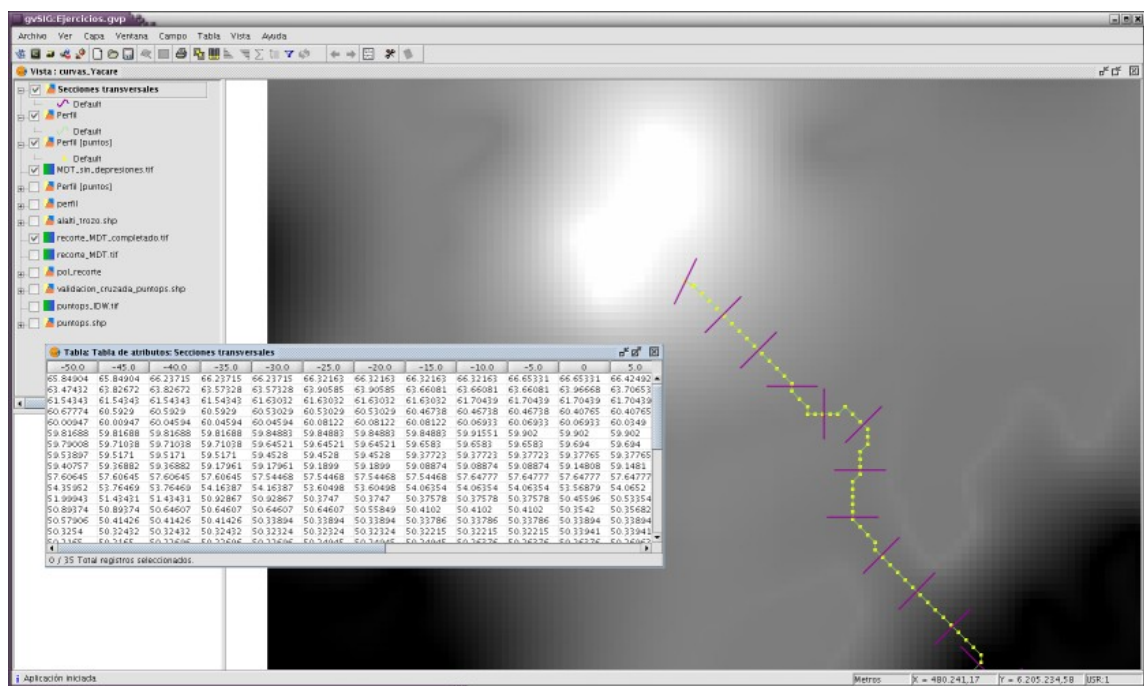
5. **Resultados:** se ha generado el gráfico del perfil, una capa de líneas con la planta del perfil y una capa de puntos con las coordenadas de los mismos y las distancias acumuladas desde el origen que seleccionamos.



6. **Cálculos secciones transversales:** también podemos calcular secciones transversales (es decir, perfiles de un ancho fijo y perpendiculares al perfil longitudinal) . Utilizar la herramienta Secciones transversales (de Perfiles). El MDE es el mismo que hemos utilizado anteriormente, como ruta seleccionaremos la ruta calculada en el apartado anterior, los demás parámetros dejaremos los que aparecen por defecto.



7. **Resultados:** se añade al proyecto una capa vectorial lineal, en donde cada entidad tiene definida las cotas de los puntos pertenecientes a la sección. Si algún punto de una sección no tiene intersección con la superficie, su cota aparecerá con valor -99999.0



## Ejercicio 3: Cálculo de mapas ráster

### Mapa de pendientes

#### Propósito:

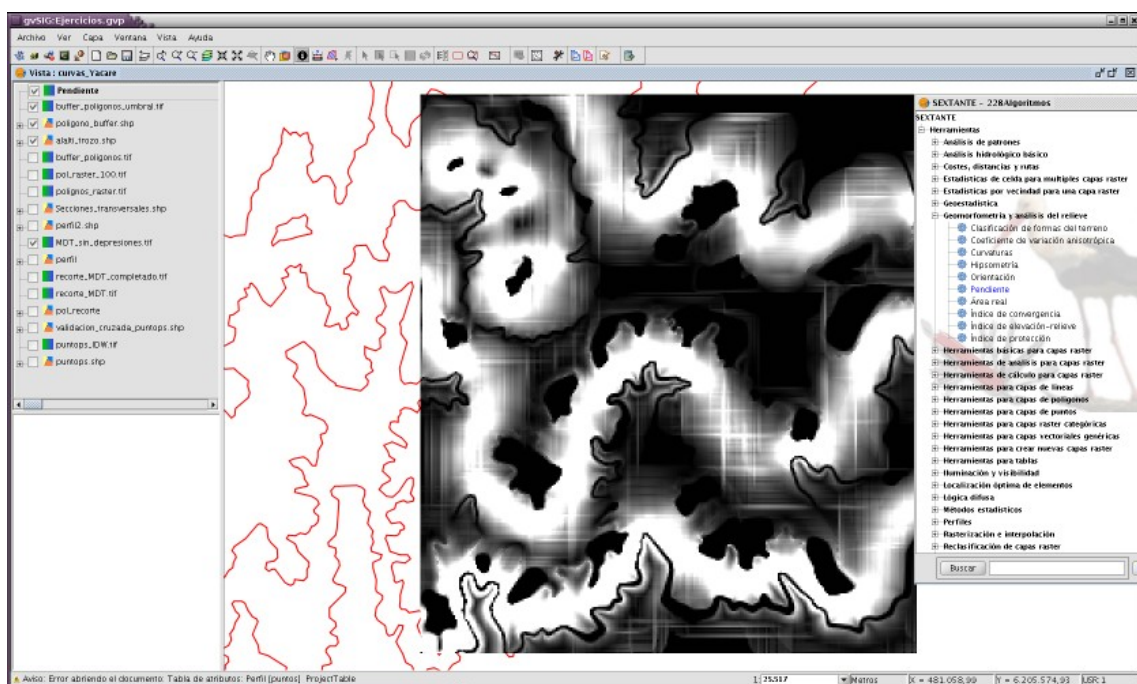
Calcular un mapa de pendientes a partir de un MDE de la zona de estudio. Desde Sextante es posible hacerlo por varios algoritmos. El valor calculado es el ángulo existente entre el vector normal a la superficie en ese punto y la vertical.

#### Cartografía necesaria:

Será necesario el MDE de la zona, como por ejemplo *MDT\_sin\_depresiones.tif*.

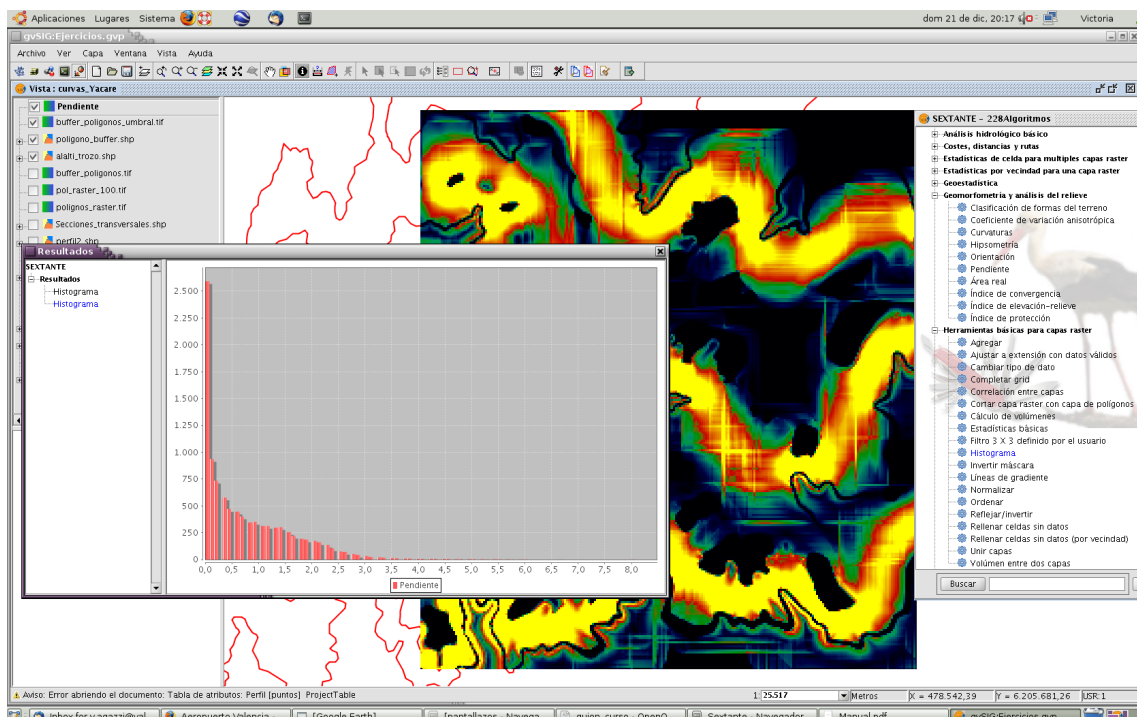
#### Secuencia:

1. **Cálculo de la pendiente:** abriremos la extensión Pendiente del módulo Geomorfometría y análisis del relieve. Para el cálculo de la pendiente seleccionaremos el MDE sin depresiones. Como método seleccionaremos primero por máxima pendiente, y como unidades Grados.



2. **Analizando valores de pendiente:** haciendo un histograma podemos averiguar los valores presentes en nuestro mapa de pendientes para saber, por ejemplo, qué simbología aplicarle.





3. **Tabla de color para mapa:** en el menú contextual de la capa que acabamos de crear, acceder a *Tablas de color* y activar la tabla. Seleccionar la tabla *blue-green-red-yellow* de las predefinidas y aplicarla, hacer Salvar como y ponerle el nombre Pendientes. Borrar las filas y definir los valores límite de los intervalos de pendientes que queremos representar (por ejemplo 40, 50, 60 y 70) y asociar un valor a cada uno de ellos. De este modo hemos definido la simbología de nuestro mapa de pendientes.
4. **Cálculo con otros algoritmos:** repetir este proceso, pero seleccionando otro algoritmo de cálculo y otras unidades.

## Mapa de orientaciones

### Propósito:

Calcular un mapa de orientaciones a partir de un MDE de la zona de estudio. Para cada píxel se calcula el ángulo (medido en sentido horario) existente entre el vector que señala el Norte y la proyección horizontal del vector normal a la superficie en ese píxel. Los algoritmos por el cual calcular este mapa son los mismos que los vistos para el mapa de pendientes. Los valores del ráster resultante serán orientaciones expresadas en radiantes.

### Cartografía necesaria:

Será necesario el MDE de la zona, como por ejemplo *MDT\_sin\_depresiones.tif*.

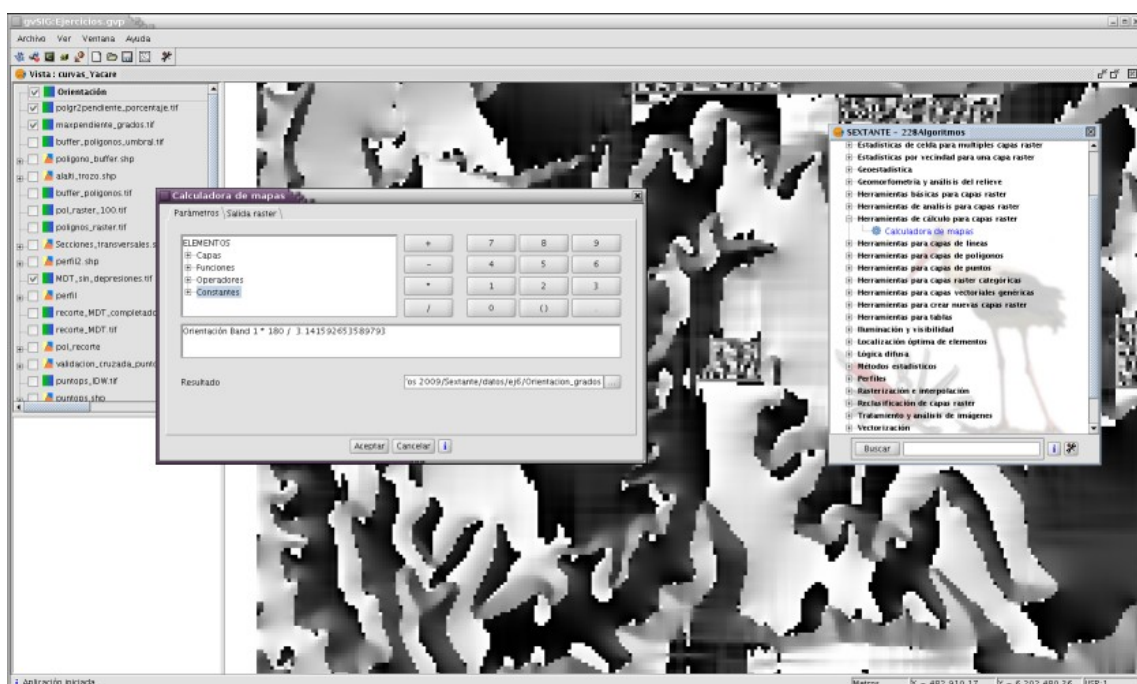
### Secuencia:

1. **Cálculo de mapa de orientaciones:** abriremos la extensión Orientación del módulo Geomorfometría y análisis del relieve. Para el cálculo de la orientación seleccionaremos el MDE sin depresiones. Como método seleccionaremos Ajuste a Polinomio de Grado 3, y la extensión será la

misma que la de los datos de entrada (el MDE). Para analizar los resultados obtenidos se puede utilizar la herramienta de Histograma de Sextante.

2. **Cambio de unidades del mapa:** si se quiere, por ejemplo, pasar los valores de orientaciones a grados (ya que se calculan en radianes), es necesario usar la calculadora de mapas desde Herramientas de cálculo para capas ráster.

Desde Calculadora de mapas, seleccionar la banda de la capa de orientaciones del elemento CAPA, multiplicarla por el valor 180 y dividir la expresión entre la constante pi del elemento CONSTANTES. La salida tendrá la misma extensión que los datos de entrada.



3. **Comparaciones:** comparar ambas capas (en radianes y en grados) en base a sus histogramas, y comprobar visualmente que son exactamente iguales.

## Mapa de cuencas visuales

### Propósito:

Definir el mapa de cuencas visuales a partir de una celda emisora. La extensión de visibilidad permite saber si 2 celdas cualesquiera tienen conexión visual.

### Cartografía necesaria:

Será necesario el MDE de la zona, como por ejemplo *MDT\_sin\_depresiones.tif*.

### Secuencia:

1. **Cálculo de cuencas visuales:** Desde la extensión de Iluminación y visibilidad se accede al módulo Visibilidad. Existen 4 algoritmos (visibilidad, distancia,

iluminación y tamaño) para el cálculo de estas cuencas de visibilidad.

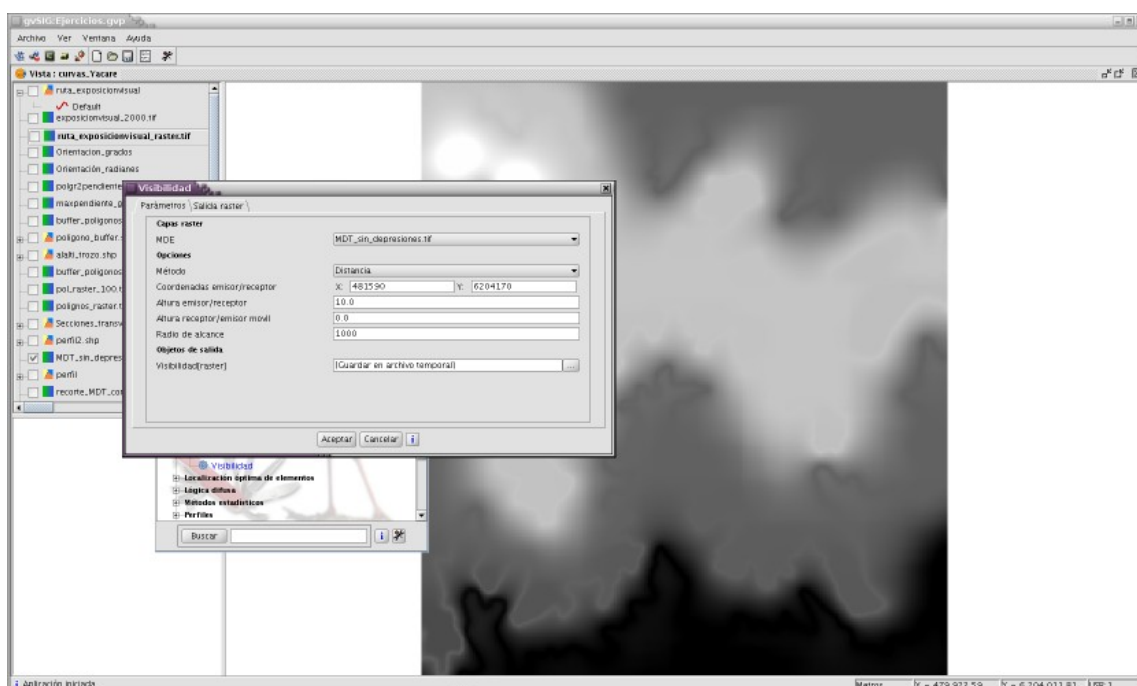
Método de visibilidad: asigna valor 1 a las celdas visibles y 0 a las no visibles.

Método de distancia: nos permite saber la distancia, en unidades de mapa, a la que se encuentra cada celda de la cuenca visual

Método de iluminación: para las celdas visibles, se guarda el valor del ángulo de inclinación

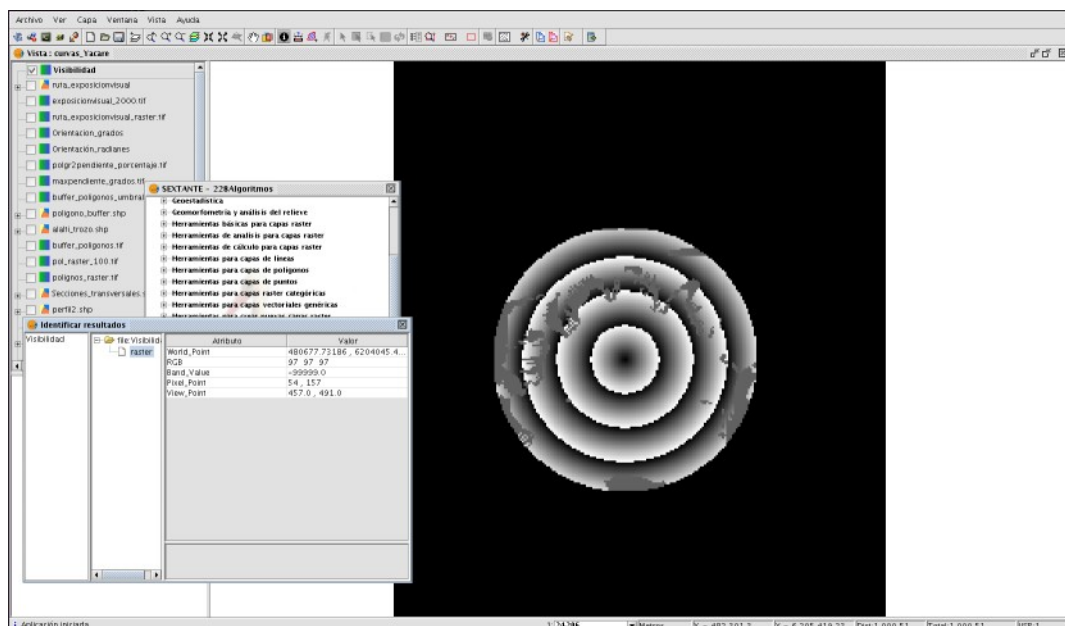
Método de tamaño: se estima el tamaño relativo con el que se verán las celdas, desde la celda emisora, usando el ángulo de inclinación y la distancia a la celda.

Se usará el método de distancias, las coordenadas desde donde se quiere calcular la cuenca (481590; 6204170) y un radio de alcance de 1000 m. Los demás parámetros dejaremos los que vienen por defecto. La extensión de la capa de salida será la misma que la de entrada.



- Resultados:** los valores de la capa resultante serán de 3 tipos: en las celdas que caen dentro del radio de alcance y que pertenecen a la cuenca visual el valor será la distancia al punto emisor, las celdas que caen dentro del radio de alcance pero no pertenecen a la cuenca visual el valor será -99999.0, para las celdas que disten del punto emisor más que el radio de alcance el valor será 0.0.





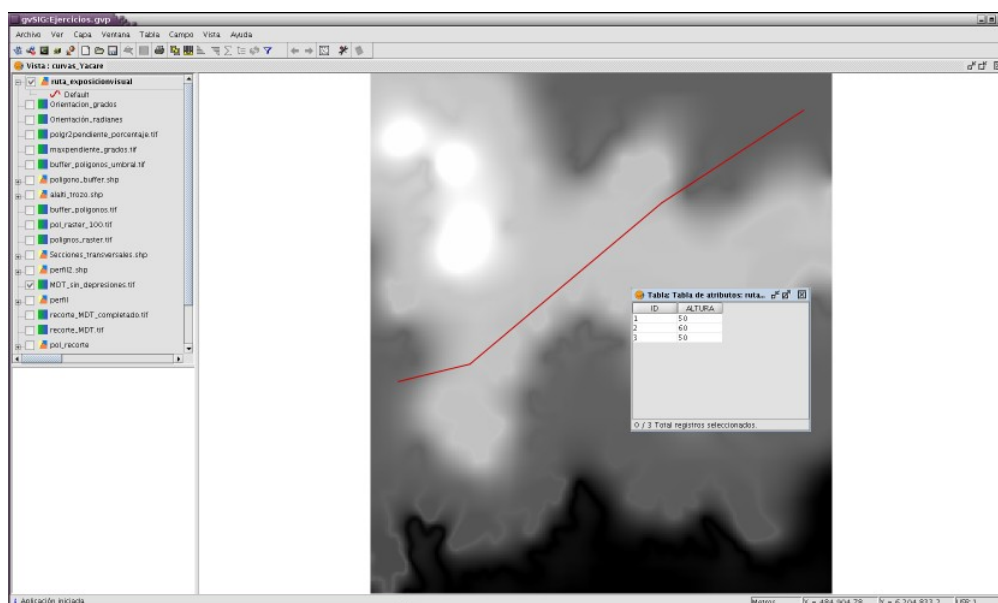
### Mapa de exposición visual

#### Propósito:

Calcular el mapa de exposición visual a partir de un MDE. Dicho de otro modo, se quiere localizar las celdas que son visibles desde una celda o un conjunto de celdas predefinido, como por ejemplo un camino o carretera. Si el elemento desde el que se calcula la exposición visual es una carretera, podremos saber, por ejemplo, la zona en la cual se debe colocar paneles informativos.

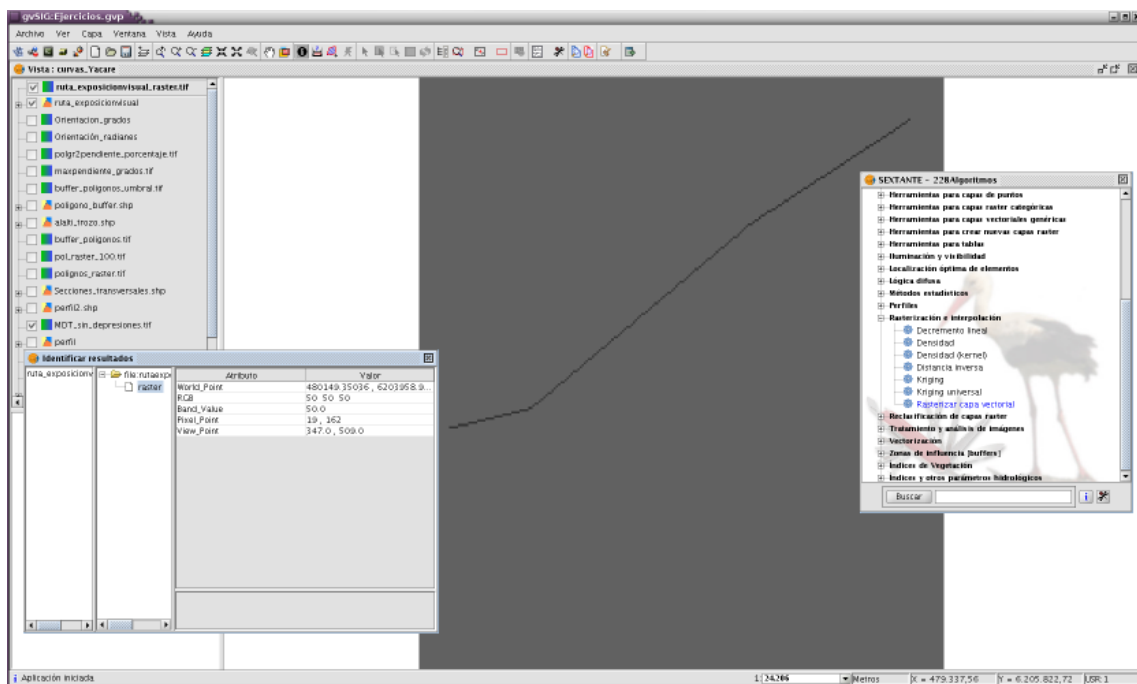
#### Cartografía necesaria:

Será necesario el MDE, como por ejemplo *MDT\_sin\_depresiones.tif*. Además se utilizará la capa *ruta\_exposicionvisual.shp* que representa el eje de la carretera desde la cual se quiere calcular la zona de exposición visual. Esta capa tiene definido un campo de ID y otro numérico cuyo valor es la altura de este tramo de carretera.



**Secuencia:**

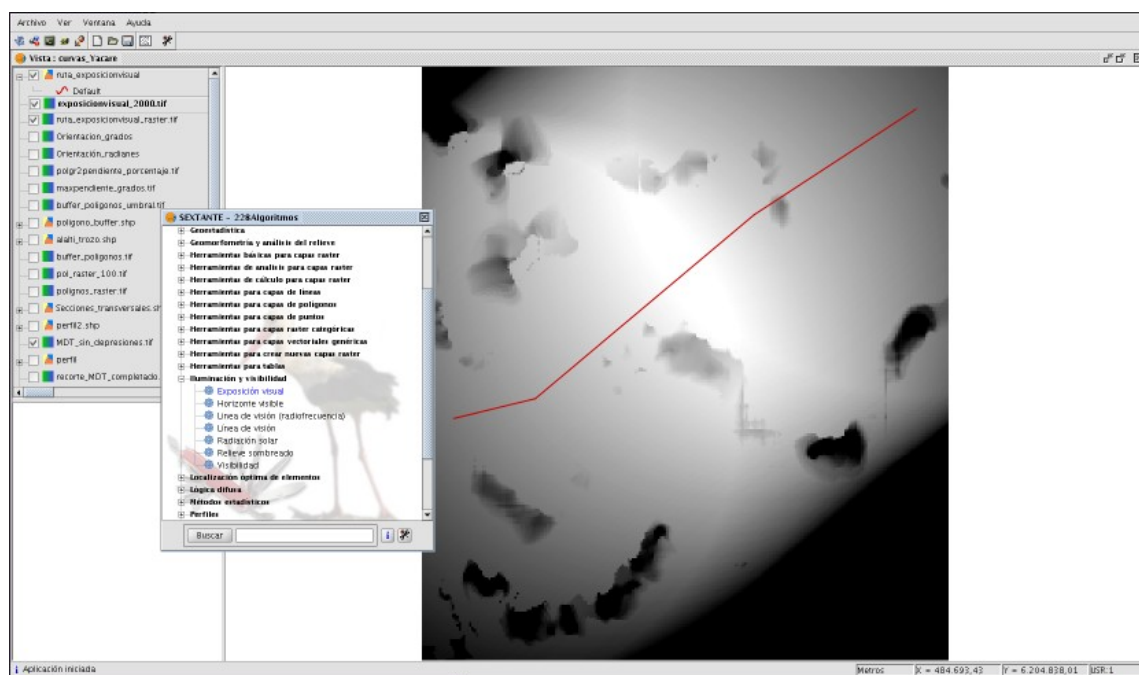
1. **Rasterizar ruta:** es necesario convertir a capa ráster la capa de ruta para poder utilizarla en el cálculo de zonas de exposición visual. Para ello utilizamos el módulo de Rasterizar capa vectorial del apartado Rasterización e interpolación. El campo que queremos conservar es la altura de cada celda. Como extensión de la capa de salida se seleccionará el MDE. Comprobar que el resultado tiene valor de cota en las celdas de la ruta y valor -99999.0 en las demás celdas.



2. **Cálculo de exposición visual:** desde la extensión Iluminación y visibilidad accedemos al módulo Exposición visual. Para el cálculo de las zonas de exposición visual se usará el MDE, la ruta rasterizada como capa de elementos, la poderación la haremos con el propio MDE, por el método de Irradiar valores y la distancia dejaremos la que viene por defecto. Si aumentamos dicha distancia veremos que aparecerán más zonas no visibles (valores nulos en la capa resultado).

Método de irradiar valores: este método analiza desde cada entidad de la capa todas las celdas que se verían alrededor del valor del radio de alcance. A cada celda del mapa le suma la cantidad de veces que es vista de la capa de elementos.

3. **Resultado:** el ráster resultante contiene, para cada celda, el valor de la cantidad de celdas de la capa elemento que le ven. Las celdas con mayor valor serán las que son más visibles desde la ruta de cálculo.



## Ejercicio 4: Herramientas básicas de capas ráster

### Recorte de un ráster

#### Propósito:

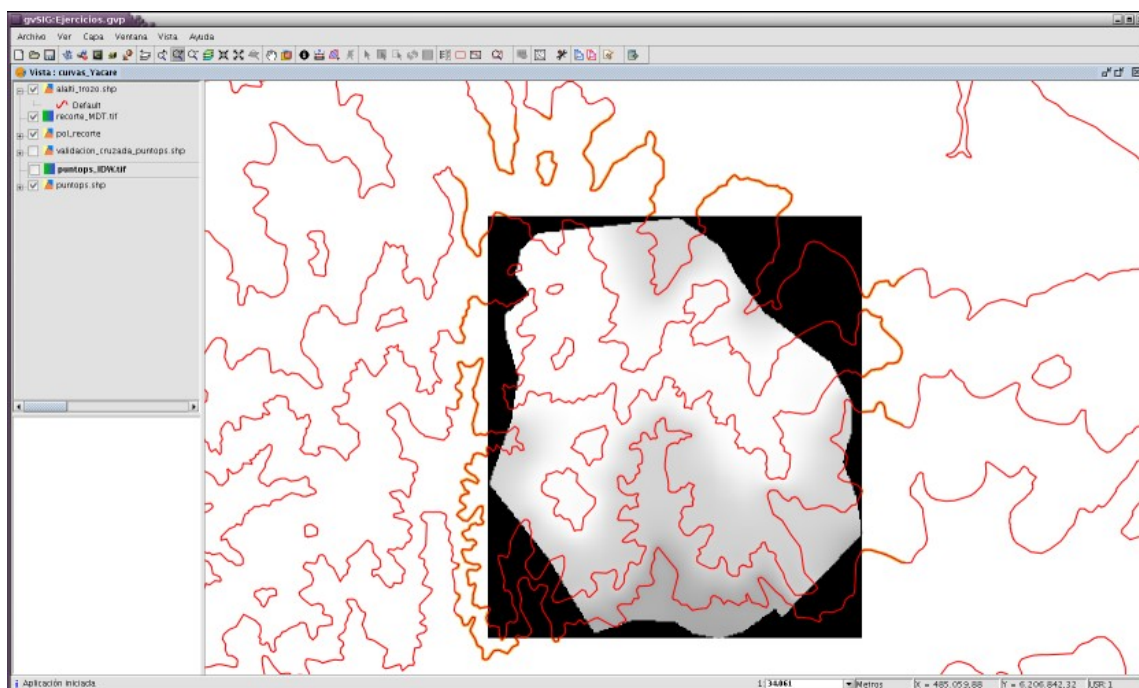
Generar una capa ráster a partir de la superficie de una capa vectorial poligonal. Si tenemos un MDE de todo un país, con esta herramienta podremos obtener la parte del MDE correspondiente a un departamento.

#### Cartografía necesaria:

Será necesario el MDE, como por ejemplo *MDT\_sin\_depresiones.tif*. Además utilizaremos la capa *pol\_recorte.shp* que contiene un polígono irregular. Esta capa debe tener solape con el MDE.

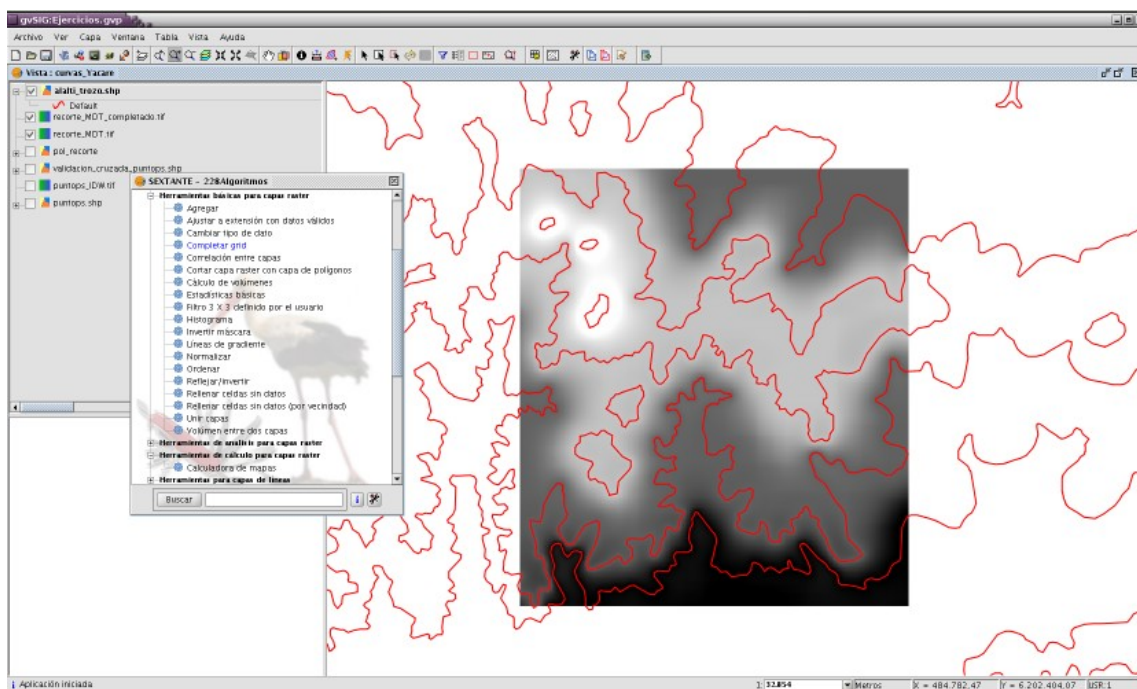
#### Secuencia:

1. **Recorte del MDE:** desde Sextante, módulo Herramientas básicas para capas ráster seleccionar Cortar capa ráster con capa de polígonos.
2. **Resultado:** obtenemos un ráster rectangular, las celdas fuera de cualquier polígono pero que quedan dentro de los límites rectangulares de la capa tienen valor de sin datos (-99999.0).



3. **Rellenado de celdas sin datos:** puede utilizarse uno de los algoritmos visto anteriormente (Rellenar celdas sin datos, rellenar celdas sin datos por vecindad, etc). Además existe la posibilidad de rellenar esos valores con datos provenientes de otra capa ráster, usando el algoritmo Completar grid desde Herramientas básicas para capas ráster.

Seleccionar Completar grid, como capa base el recorte anterior, capa adicional el MDE completo y el método de interpolación Distancia inversa. La extensión de la capa de salida ráster será la misma que la de la capa base de entrada.



Evidentemente, si el polígono vectorial del que partimos es un rectángulo, todas las celdas tendrán valor de cota y no hará falta completar el grid resultante.

## Vectorización de capa ráster

### Propósito:

A partir del MDE que hemos utilizado toda la práctica, generaremos una capa de curvas de nivel. Además vectorizaremos otras geometrías (puntos, polígonos y líneas).

### Cartografía necesaria:

Será necesario el MDE, como por ejemplo *MDT\_sin\_depresiones.tif*. Se debe investigar cuáles son los valores máximo y mínimo presentes en el MDE ya que son parámetro que necesitaremos indicar.

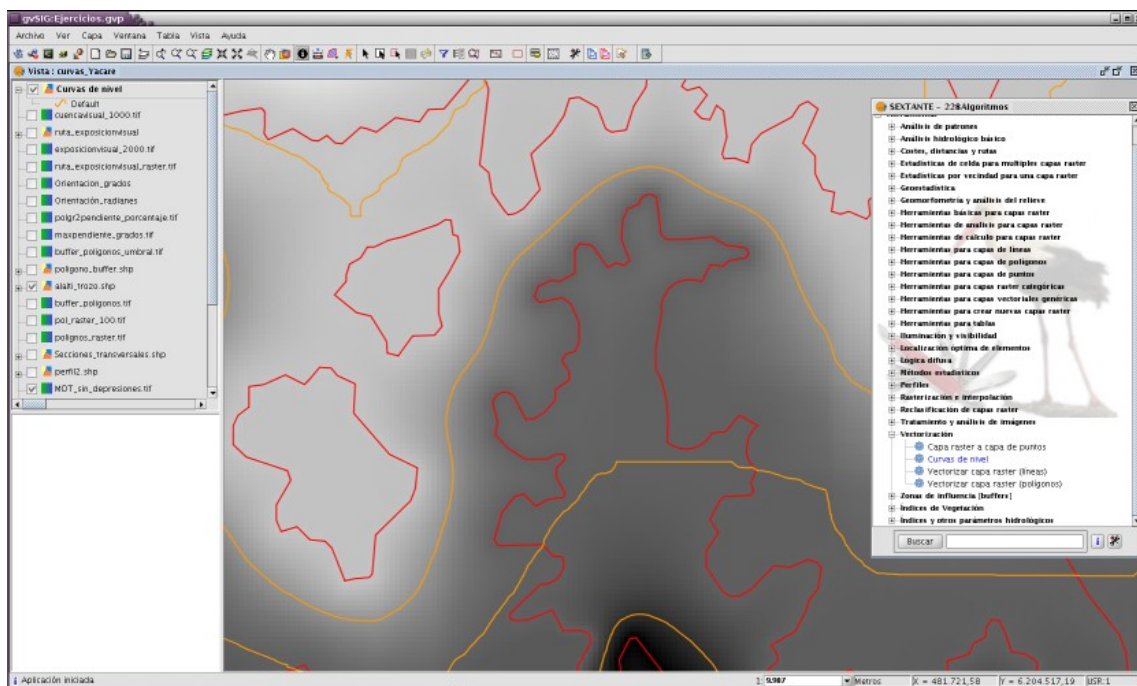
Además se utilizará una ruta rasterizada, como por ejemplo *ruta\_exposicionvisual\_raster.tif*. Por último también es necesario el ráster de polígonos, como por ejemplo *polignos\_raster.tif*.

### Secuencia:

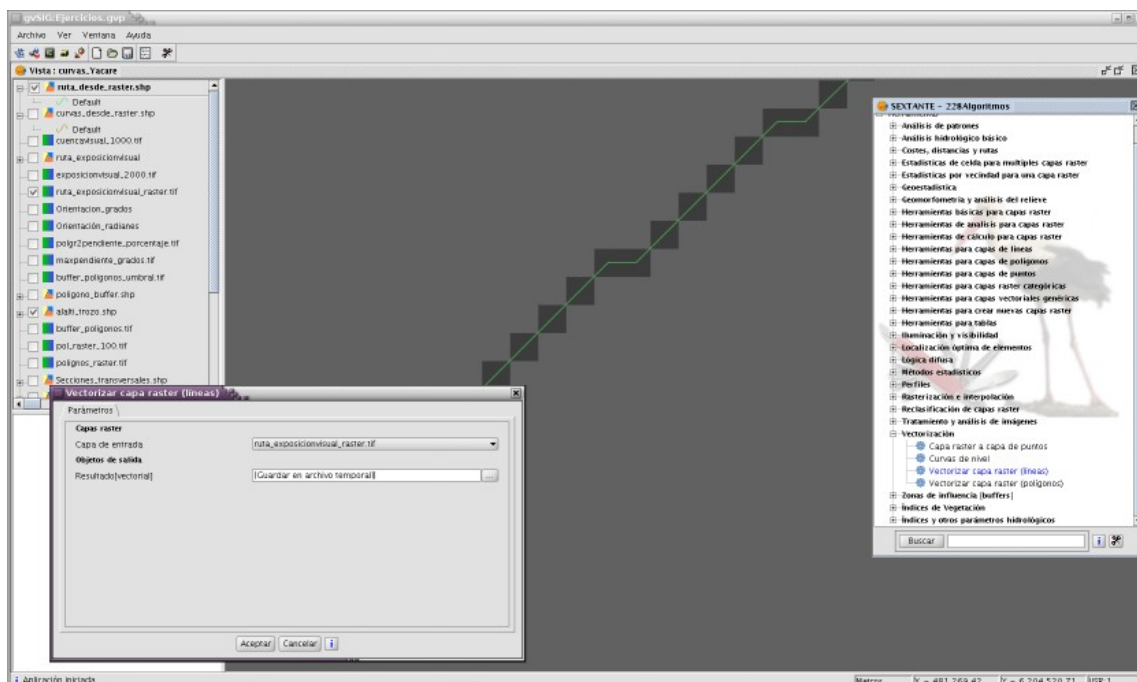
1. **Generar curvas de nivel desde ráster continuo:** en la extensión Vectorización seleccionar Curvas de nivel. Como parámetros seleccionar el MDE, las cotas mínimas y máximas que queremos calcular (en este caso serán 40.0 m y 70.0 m) y la equidistancia de 10.0 m.
2. **Resultado:** el resultado se verá mucho más simplificado que las curvas originales, ya



que los valores de cota del MDE han sido manipulados (rasterización, interpolación, recorte y completado, etc.).

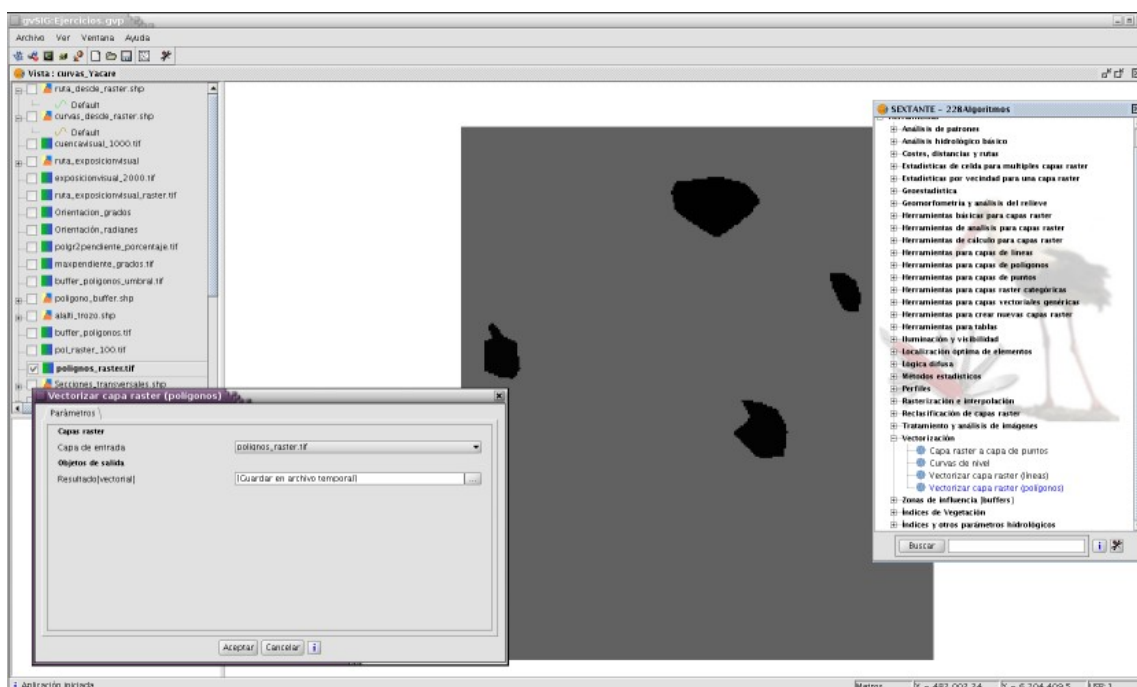


3. **Comprobación:** para cada entidad lineal, en la tabla asociada tiene la cota correspondiente.
4. **Generar malla de puntos de cota:** abrir el módulo Capa ráster a capa de puntos para generar una malla de puntos con cota a partir de un ráster. Seleccionar el MDE ráster y ver que el resultado tiene en su tabla asociada la cota de cada punto de la malla.
5. **Generar ruta desde ráster discreto:** añadir el ráster *ruta\_exposicionvisual.tif* y desde el módulo Vectorizar capa de líneas, seleccionar el ráster de ruta de exposición visual y aplicar el módulo. Las líneas resultantes unen centros de celdas del ráster.



6. **Vectorizar polígonos:** abrir el módulo Vectorizar capa ráster (polígonos). Usaremos los polígonos ráster del ejercicio Cálculo de zonas de influencia en cartografía ráster (*polignos\_raster.tif*).

Cada celda del ráster tiene asociado un valor, y al crear la capa vectorial poligonal dicho valor se conserva en la tabla de atributos (si es diferente de -999999.0). Las celdas cuyo valor sea -999999.0 serán tratadas como Nodata.



## Ejercicio 5: Modelizador gráfico de procesos

### Mapa de índice de protección desde curvas de nivel

#### Propósito:

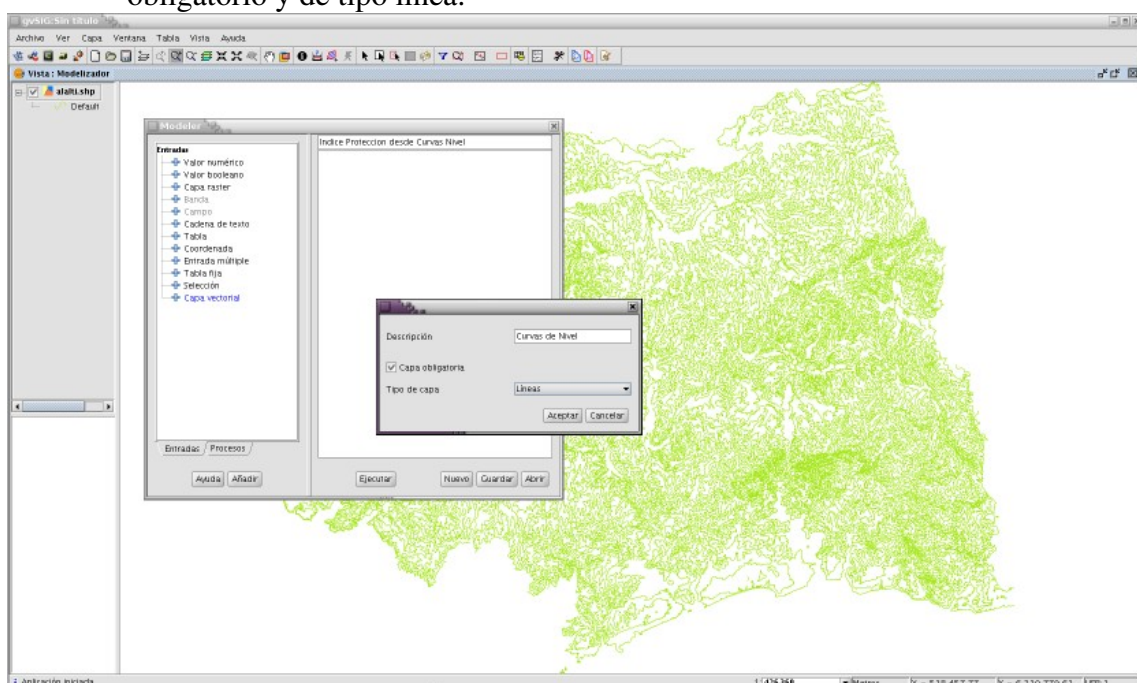
Diseñar de forma gráfica un modelo conceptual que contenga todos los pasos para calcular de un sólo click un mapa de índice de protección.

#### Cartografía necesaria:

Una capa vectorial de curvas de nivel de la zona de estudio con campo COTA con altura de las curvas de nivel, como por ejemplo *alalti\_trozo.shp*.

#### Secuencia:

1. **Añadir capa origen del modelo:** añadir la capa vectorial de curvas de nivel en una vista nueva. El sistema de referencia definido en la vista será el Yacaré (ROU - USAMS).
2. **Modelizador gráfico:** abrir el Modelizador gráfico de Sextante. En el lienzo debemos definir las entradas necesarias y el flujo de datos entre entradas y resultados intermedios, hasta llegar a un resultado final. Poner nombre del módulo nuevo: Índice Protección desde Curvas de Nivel.
3. **Capa entrada modelo:** añadir capa vectorial como entrada única del modelo. Llamar a esta entrada curvas de nivel. Se debe establecer que este parámetro es obligatorio y de tipo línea.



4. **Rasterizar capa vectorial:** añadir el proceso Rasterizar capa vectorial desde Rasterización e Interpolación. Seleccionar capa de Curvas de nivel como capa

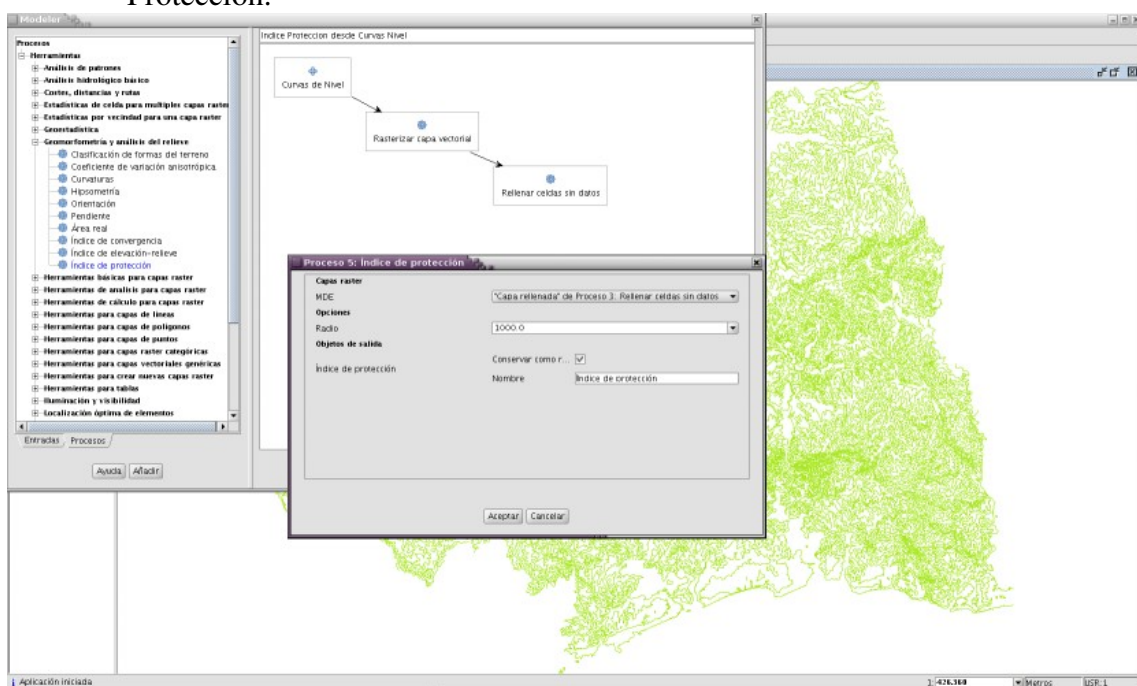


vectorial, el campo será llamado COTA (en mayúsculas, y así deberá llamarse en cada capa vectorial donde se quiera aplicar el proceso que estamos definiendo). No se seleccionará la opción de Conservar como capa intermedia y dejamos el nombre que nos propone Sextante por defecto. Los parámetros de Salida del ráster se definirán al ejecutar este nuevo módulo, como con cualquier otro módulo de Sextante.

5. **Rellenar celdas sin datos:** iremos a la extensión Herramientas básicas de capas ráster y utilizaremos Rellenar celdas sin datos. Como capa origen se seleccionará el resultado del proceso anterior, y como Umbral de tensión se pondrá el valor “0.5”.
6. **Índice de protección:** desde la extensión Geomorfometría y análisis del relieve añadir el algoritmo Índice de protección, aplicándolo a la última capa generada, con un radio de 1000.0 m (unidades de la vista).

El Índice de protección se calcula analizando el entorno inmediato de cada celda hasta una distancia establecida y evalúa cómo el relieve "protege" a la misma. Este índice de protección puede ser útil para un estudio ecológico de la zona. A mayor radio de entorno, mayor tiempo de ejecución.

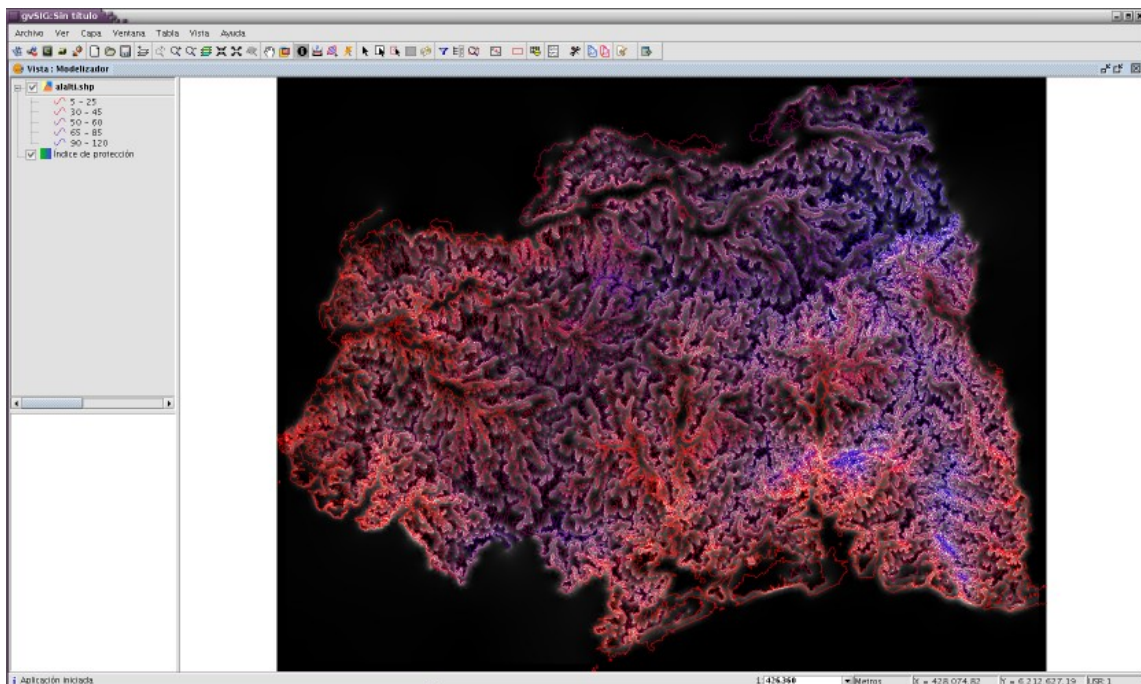
Seleccionar Conservar la capa como resultado y poner como nombre Índice de Protección.



7. **Guardar modelo nuevo:** Debemos guardar el modelo para poder recuperarlo posteriormente.
8. **Ejecutar el proceso:** para ejecutar el nuevo proceso será necesario tener añadida a

una vista una capa vectorial de curvas de nivel. Ejecutar dicho proceso desde la ventana del modelizador directamente.

9. **Salida ráster:** utilizar la misma extensión que la capa de curvas de nivel y poner como tamaño de celda 15.0 m. En caso que Sextante nos avise que el tamaño de la capa a generar es demasiado grande, aumentar dicho tamaño de celda.
10. **Resultado:** luego de que cada proceso se ejecute de forma encadenada, obtenemos el mapa de Índice de protección de nuestra zona.  
Se puede apreciar que las zonas más elevadas son las zonas menos protegidas (valores menores en el mapa resultante).



11. **Configurar Gestor de extensiones:** para tener disponible nuestro nuevo modelo desde el Gestor de extensiones ir a las Propiedades de configuración del mismo, definir una carpeta para los modelos.  
Reiniciar el Gestor de extensiones y ver que aparece un nuevo nodo llamado Modelos en donde tenemos disponible nuestro modelo creado.



## **Referencias en internet**

(Últimos accesos Febrero de 2009)

Descarga de binarios gvSIG:

[http://www.gvsig.gva.es/index.php?id=gvsig\\_desktop&L=0&K=1](http://www.gvsig.gva.es/index.php?id=gvsig_desktop&L=0&K=1)

Manual de gvSIG versión 1.1:

<https://gvsig.org/web/docusr/userguide-gvsig-1-1>

Descarga de binario de SEXTANTE:

<http://forge.osor.eu/plugins/wiki/index.php?Downloads&id=13&type=g>

Conceptos generales de SEXTANTE:

<http://www.sextantegis.com/docs.htm>

Tutoriales de video de SEXTANTE:

<http://www.sextantegis.com/video.htm>

## **GNU GENERAL PUBLIC LICENSE**

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,  
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA  
Everyone is permitted to copy and distribute verbatim copies  
of this license document, but changing it is not allowed.

### Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and

modification follow.

GNU GENERAL PUBLIC LICENSE  
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those

sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to

address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

#### NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

#### END OF TERMS AND CONDITIONS

#### How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
```

```
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.
```



This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.
```

```
<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.