

# FILTRO POR ATRIBUTOS PARA EL VISOR DE SHAPEFILES DEL COMPONENTE MAPWINGIS

Germán Alonso Carrillo Romero  
[carrillo.german@gmail.com](mailto:carrillo.german@gmail.com)  
<http://geotux.tuxfamily.org>  
Mayo de 2008

Cuando necesitamos visualizar subconjuntos de datos espaciales contenidos en una capa vectorial empleamos filtros espaciales. A continuación se muestra una implementación para filtrar datos provenientes de archivos tipo *Shapefile* en [el visor construido en un blog anterior](#).

## Introducción

Algunos programas para SIG permiten construir filtros sobre los datos espaciales con el fin de prescindir de la información que no se requiera visualizar. Para datos de *Postgres/PostGIS* se pueden emplear vistas, muy útiles para representar información con filtros. Para datos en formato *Shapefile* podemos emplear programas como *Kosmo SAIG* (*gvSIG* y *Quantum GIS* no han implementado esta funcionalidad a la fecha) o incluso optar por componentes SIG programables como *MapWinGIS* del proyecto *MapWindow*.

## ¿Qué se busca con el filtro?

Al implementar esta funcionalidad, se espera poder definir reglas para visualizar información geográfica con base en sus atributos alfanuméricos.

Para lograr una interacción rápida y sencilla de la herramienta con el usuario, se va a limitar el filtro así:

- Solo se podrá seleccionar un atributo de la tabla.
- Solo se podrá seleccionar un valores del atributo.

Esta simplificación permite hacer más claro para el usuario la definición del filtro, dejando de lado operadores de comparación (*Mayor que*, *menor que*, *mayor o igual que*, *menor o igual que*, *diferente*, entre otros) y conectores (*AND*, *OR*, *XOR*, entre otros) que pueden complejizar relativamente el desarrollo.

El filtro solo aplicará para datos provenientes de archivos *Shapefile* y aceptará las geometrías punto, línea y polígono.

## Interfaz para filtrar

La interfaz de filtrado a implementar luce así:



Imagen 1: Interfaz de filtrado.

El usuario selecciona la capa sobre la cual se realizará el filtro y posteriormente define la regla de visualización a partir de un campo y su valor.

## ¿Cómo se implementa?

Para implementar la funcionalidad, se debe descargar la clase *Filtro* desde [aquí](#) (archivo Zip, 4,7 Kb), descomprimirla y cargala al proyecto de *Visual Basic .Net* que contiene el código del visor.

Llamamos la forma para filtrar desde el evento *click* de un botón así:

```
Sub BtnFiltroClick(ByVal sender As Object, ByVal e As EventArgs)
    If axMap1.NumLayers = 0 Then
        MsgBox ("Debes cargar al menos una capa al mapa.")
    Else
        Dim pFiltro As New MapWindowVisor_1.Filtro(axmap1)
        pFiltro.Show()
    End If
End Sub
```

En el anterior bloque de código, *BtnFiltro* es el nombre del botón que carga la ventana para hacer los filtros, *axMap1* es el nombre del control *AxMap* y *MapWindow\_1* es el nombre del proyecto de *Visual Basic .Net*.

## Filtrando datos

Para filtrar datos en el visor de *Shapefiles* debemos cargar por lo menos una capa y dar *click* en el botón de filtrado, con lo cual se despliega la ventana para definir el filtro:

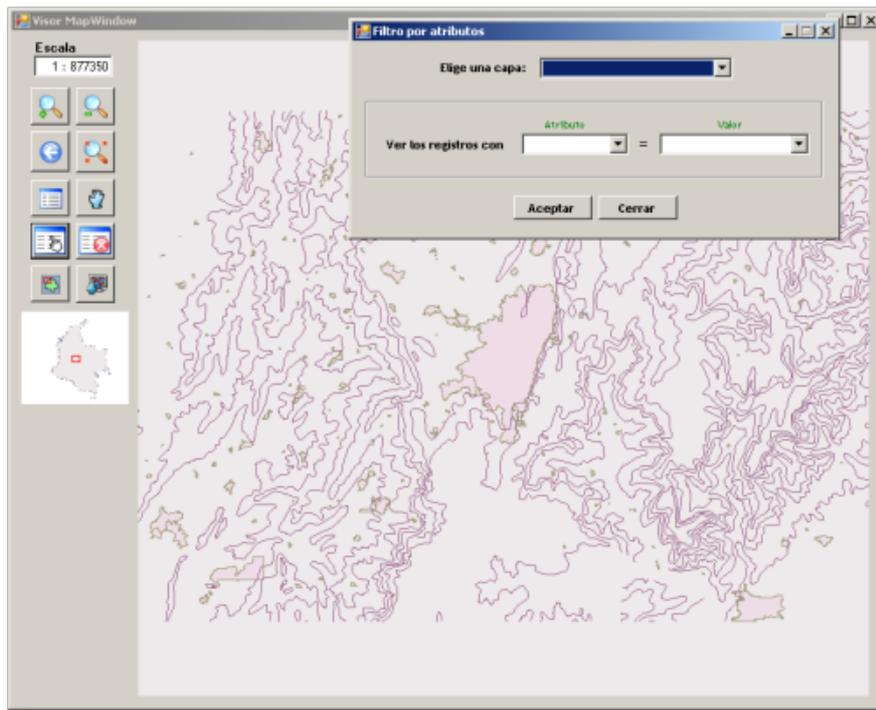


Imagen 2: Filtrando datos.

Para el ejemplo se construye el filtro para visualizar únicamente las curvas de nivel de tipo índice:

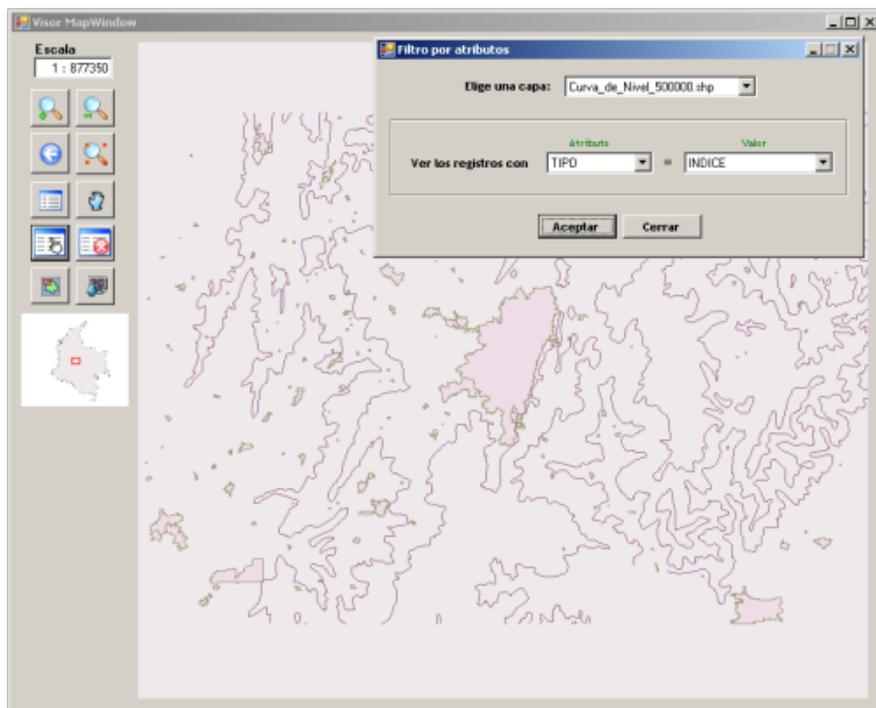


Imagen 3: Mapa filtrado.

De este modo, la visualización es más clara y solo presenta la información que se necesita.

## Detalles técnicos

En cuanto al desarrollo se refiere, el componente *MapWinGIS* provee el método *set\_ShapeVisible* para el objeto *AxMap*, que permite definir la visibilidad de cada elemento espacial dentro de la capa de interés.

Para realizar el filtrado se siguen estos pasos a nivel de código:

1. Leer la capa de interés.
2. Leer el campo que se usará como base para definir el filtro.
3. Leer el valor asociado al campo.
4. Para cada elemento espacial (*Shape*) leer el valor asociado al campo que fue seleccionado por el usuario.
5. Comparar el valor leído en el punto 4 con el valor seleccionado por el usuario.
6. Asignar el valor verdadero a la visibilidad de los elementos espaciales (*Shapes*) que cumplan con la comparación y el valor falso a las restantes.

Se podrían construir consultas más robustas en la interfaz agregando los conectores y operadores de *SQL* para lograr mayor control sobre el filtro, sin embargo, para efectos de este ejercicio, se ha optado por una consulta sencilla que aún así, permite obtener una funcionalidad útil al momento de discriminar datos de una misma capa espacial.

## Borrando el filtro

Se debe dar al usuario la posibilidad de borrar los filtros construidos previamente, para ello, agregamos un nuevo botón y escribimos el siguiente código en su evento *click*:

```
Sub btnBorrarFiltroClick(ByVal sender As Object, ByVal e As EventArgs)

    Dim lHandleCapa As Integer
    Dim pCapa As MapWinGIS.Shapefile

    If axMap1.NumLayers = 0 Then Return

    For i As Integer = 0 To axMap1.NumLayers - 1

        lHandleCapa = axMap1.get_LayerHandle(i)
        pCapa = axMap1.get_GetObject(lHandleCapa)

        For j As Integer = 0 To pCapa.NumShapes - 1
            axMap1.set_ShapeVisible(lHandleCapa, j, true)
        Next

    Next

    MsgBox("Se han quitado los filtros sobre el mapa.")

End Sub
```

En este código se recorren los elementos espaciales (*Shapes*) de cada una de las capas del mapa y se asigna el valor verdadero a la visibilidad de cada uno.

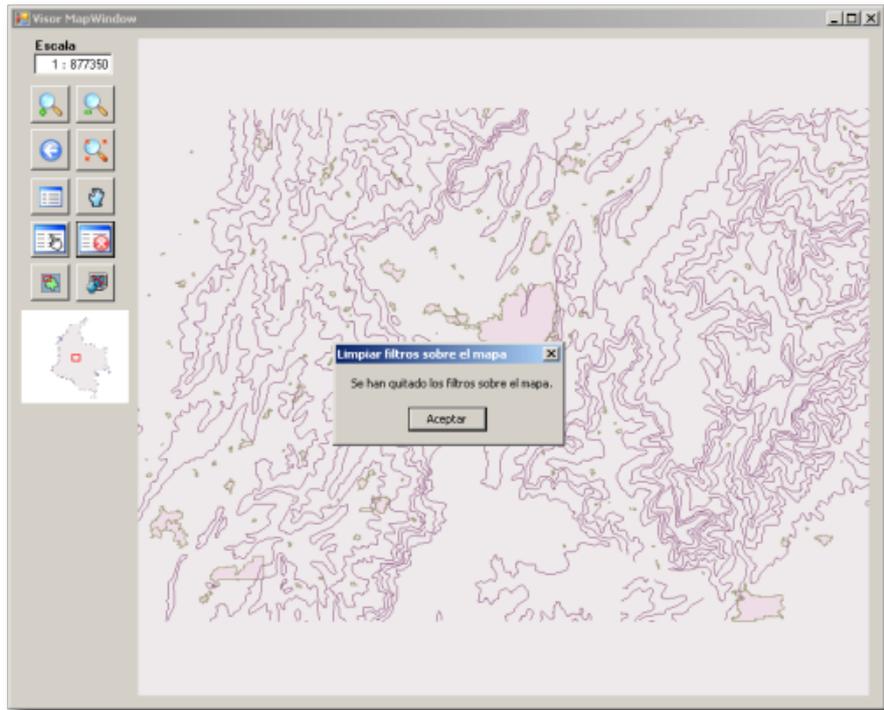


Imagen 4: Borrando el filtro.

## Conclusiones

El filtrado de datos espaciales es útil para discriminar información de interés y facilitar la visualización y el análisis.

A través del componente *MapWinGIS* es posible desarrollar la funcionalidad de filtrado para datos en formato *Shapefile*.