

DESARROLLO DE UN S.I.G. PARA EL DISEÑO DE PROGRAMAS DE GESTIÓN DE RESIDUOS SÓLIDOS DE CANTABRIA.

César Otero; Almudena de la Pedraja; Reinaldo Togores

Departamento de Ingeniería Geográfica y Técnicas de Expresión Gráfica.

Escuela de Caminos. Universidad de Cantabria. España.

Avda. de Los Castros s/n. 39005. Santander.

Tfno: 00 34 42 201794

Fax: 00 34 42 201703

E-mail: otero@ccaix3.unican.es

1. Resumen

La intención primordial de la comunicación es describir los trabajos contemplados para la generación de un Sistema de Información Geográfica específicamente orientado a presentar gráficamente los puntos incontrolados de vertido existentes en la Comunidad de Cantabria y sus características geográficas y medioambientales más relevantes. Por razones que se argumentan, en la segunda parte de la comunicación se describen aquellos procedimientos de selección y designación de objetos gráficos, siempre necesarios al realizar comandos personalizados, que resultan ser cuatro y tratables de forma sistemática, sea cual sea el entorno base de desarrollo desde el que se trabaja.

2. Abstract.

The aim of the paper is to describe the work implemented for making up a GIS specifically focused on the representation of the illegal points of solid waste residues existing in the region of Cantabria (Spain) as well as their geographical and environmental features. Once it is previously justified, the second part of the paper presents the main procedures for constructing selection sets and fences, always needed when a customised graphic user command is implemented. It results that it is very possible to propose four standard ways to do that, whatever the CAD system is used as the programming environment.

3. Objetivos de la asesoría sig.

La intención primordial de los trabajos contemplados dentro de esta asesoría ha sido comenzar la generación de un Sistema de Información Geográfica específicamente orientado a presentar gráficamente los puntos incontrolados de vertido existentes en la Comunidad de Cantabria y sus características geográficas y medioambientales más relevantes. Es, en consecuencia, un esfuerzo técnico que otorga un valor añadido completamente coherente con la inversión principal que da origen a este Proyecto: una toma de datos rigurosa y exhaustiva sobre la ubicación de residuos incontrolados en nuestra región. Dado que en un principio se desconocía la cantidad de puntos de vertido existentes y, en consecuencia, era poco previsible el tiempo preciso para su digitalización en el sistema, se determinaron los siguientes objetivos básicos:

?? Representación de los vertederos con un fondo cartográfico apropiado de la región de Cantabria.

?? Desarrollo de utilidades básicas de consulta individual de vertederos.

?? Desarrollo de utilidades básicas de gestión y producción de planos.

Satisfechos estos objetivos, la Asesoría SIG se orientaría más abiertamente hacia la investigación, enfocada hacia la búsqueda de nuevas propuestas de representación de datos, modos de depuración de cartografía o análisis espacial de los mismos. Metodológicamente, se concedía importancia a la búsqueda de otras iniciativas SIG semejantes de las que extraer referencias.

4. Descripción del trabajo realizado en la asesoría.

4.1. Valoración y elección de la plataforma SIG.

La plataforma elegida, Geographics de Bentley Systems y Access de Microsoft, lo fue frente a otras también consideradas (Map Info, AutoCAD Map, Arc View, Arc-Info Pc, ArcCAD, MGE de Intergraph en cuanto a soporte SIG y Oracle en cuanto a Base de Datos relacional e incluso entornos Delphi o similares como gestores de gráficos y B.D.) por las siguientes razones:

?? 1. Se requería del sistema capacidad de desarrollo de macros y programas propios. A su vez, por razones de tiempo de duración del proyecto, se requería que el entorno, el lenguaje y la extensión de funciones de desarrollo ya fueran conocidas y no exigieran al equipo inversión de tiempo en su aprendizaje, lo que podría suponer una merma de rendimiento en los resultados propuestos.

?? 2. Se requería del sistema la mayor compatibilidad con los formatos de la cartografía digital elegida como fondo cartográfico.

?? 3. Se requería un sistema que fuese completo (es decir, que tuviese capacidades de carga y enlace de datos, de visualización, de construcción de queries SQL y de análisis espacial) y compacto (

estas prestaciones, ofertadas en un solo módulo) de manera que todas las operaciones precisas para el desarrollo del proyecto no exigieran la compra de varios módulos.

- ?? 4. Se requería un sistema que implicase las menores inversiones de software al usuario final y, muy en especial, en lo que se refiere a los costes de mantenimiento.
- ?? 5. Se requería un sistema que fuera directamente compatible con las dos Bases de Datos alfanuméricas indicadas y, además, compatible con el entorno operativo de Microsoft (Windows 95 o NT) pues todo el soporte hardware quedaría en una plataforma tipo Pc.
- ?? 6. Se requería un sistema CAD utilizable al 100% junto con las prestaciones SIG. Ello obligaba a éste a ser construido como aplicación vertical dentro del sistema CAD.

Sin entrar en detalles más precisos, una plataforma como la indicada al inicio de este apartado se ajusta perfectamente a las especificaciones propuestas.

4.2. Estructura de los datos.

4.2.1. Alfanuméricos.

Los atributos alfanuméricos relativos a cada vertedero se han establecido sobre una única tabla, cuyos campos se adjuntan en la página siguiente. Prácticamente todos ellos tienen una interpretación inmediata. Dentro de la

Asesoría SIG se estudió la posibilidad de diseñar una estructura relacional con diversas tablas y sus relaciones correspondientes; dado que prácticamente todas las condiciones de atributo son individuales, se llegó a la conclusión de que no se deriva ninguna ventaja operativa por la separación de tablas. Aunque resulte una única de un volumen aparatoso, resulta ser la mejor solución.

mmlink	Numérico
mapid	Numérico
Patrulla	Texto
Numerovert	Numérico
CódigoMunicipio	Texto
Municipio	Texto
Pueblo	Texto
Barrio	Texto
Acceso	Texto
Mapa	Texto
Escala	Texto
Latitud	Texto
Longitud	Texto
XUTM	Texto
YUTM	Texto
Huso	Texto
EPE	Texto
Altitud	Numérico
Fecha	Texto
Película	Texto
Negativo	Texto
Fotografía	Texto
tipo	Texto
largo	Numérico
ancho	Numérico
alto	Numérico
Descripción1	Texto
Domésticos	Numérico
Construcción	Numérico
Industriales	Numérico
Ganaderos	Numérico
LíneaBM	Numérico
RTPs	Numérico
Neumáticos	Numérico
Voluminosos	Numérico
Cadáveres	Numérico

Superficie	Numérico
Volumen	Numérico
Descripción2	Texto
Proximario	Numérico
Afectario	Numérico
Proximomar	Numérico
Afectamar	Numérico
Presencialxiv	Numérico
Presenciahumos	Numérico
Afectaotrosecosist	Texto
Distanciazonashabit	Numérico
Proximoturístic	Numérico
Proxiplayas	Numérico
Proxibosques	Numérico
Proxilabranza	Numérico
Proxireligiosos	Numérico
Visibledesdecarretera	Texto
Visibletren	Numérico
Descripción3	Texto
Códigoanterio	Texto
Estadoactual	Texto

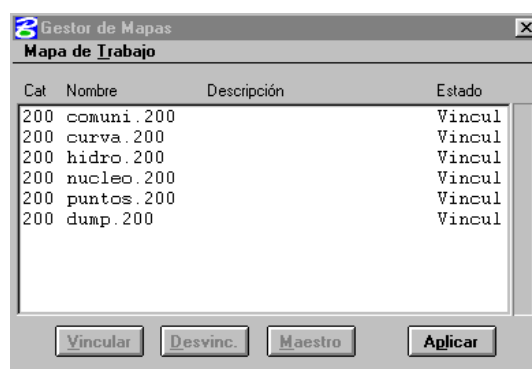
4.2.2. Datos Gráficos.

Los datos gráficos se han estructurado según las siguientes definiciones:

CATEGORÍA. Una única categoría denominada 200, sobre la que se generan 6 mapas y una única feature.

FEATURES. Un único tipo de feature, de nombre VERTEDEROS. El fichero DUMP.200, contiene la digitalización de las ocurrencias de la FEATURE de nombre VERTEDEROS, cada una de las cuales está enlazada con un registro de la tabla de atributos TVERTEDERO.

?? MAPAS de BCN200. 6 mapas pertenecientes a la categoría 200, según la nomenclatura y sentido lógico que se desprende de la figura adjunta.



Cat	Nombre	Descripción	Estado
200	comuni.200		Vincul
200	curva.200		Vincul
200	hidro.200		Vincul
200	nucleo.200		Vincul
200	puntos.200		Vincul
200	dump.200		Vincul

?? FONDO CARTOGRÁFICO. 65 hojas de la BCN25 sin adscribir a categoría alguna, memorizados con extensión .DGN e incorporables al dibujo de trabajo como referencia externa independiente. Se han desarrollado utilidades específicas de manejo del fondo cartográfico, tal como se desprende del apartado 2.9.

4.2.3. Digitalización de vertederos.

Básicamente, la digitalización ha sido una operación realizada desde el entorno CAD, pero utilizando directamente una ocurrencia de feature, que se ha ubicado sobre el fondo cartográfico (en su inicio BCN200 y posteriormente BCN25). La determinación adoptada de no dar naturaleza de Categoría a la BCN25 y, en consecuencia, utilizarla sólo como fondo cartográfico simplifica enormemente la referenciación de los vertederos sobre una u otra escala, como puede verse en las figuras adjuntas.

En todo caso, se considera recomendable dedicar un tiempo a la reposición de los primeros vertederos, ubicados



VERTEDEROS SOBRE BCN200



VERTEDEROS SOBRE BCN25

sobre la BCN200.

4.2.4. Diseño de formularios de consulta individual.

La consulta individual de vertederos es ofrecida de manera directa por un comando de Geographics; no obstante, una de las primeras operaciones requeridas de un Sistema de Información Geográfica es la capacidad de generar una ficha individualizada de información para cada vertedero. Esta operación se ha acometido en los inicios del proyecto, generando un formulario de Base de Datos que pudiese ser desplegado desde el interfaz gráfico. Tres operaciones de depuración han sido llevadas a cabo a este respecto:

- ?? Se ha dotado a la ficha de la capacidad de ser impresa directamente.
- ?? Se han efectuado diversos ensayos al objeto de que el tiempo de regeneración de la imagen ráster (la fotografía del vertedero) fuese lo más rápida que fuese posible.
- ?? Se ha depurado la macro de gestión de esta ficha, que resultaba inicialmente inestable (su despliegue en pantalla provocaba la pérdida del enlace del SIG con la Base de Datos).

FICHA TÉCNICA DE VERTEDERO Número de Vertedero 37

Código del inventario anterior Estado Actual Fecha: 22/05/98

Situación

Municipio: Castro Urdiales Código del Municipio: 20

Pueblo: Mapa: 36 III Laredo

Lugar / Barrio:

Latitud: 43° 22' 02" S UTM (m): 482.128 Escala: 1:25.000 Huso: 30

Longitud: 3° 13' 10" Y UTM (m): 4.801.747 Altitud: 20 EPE: 22

Acceso a la zona

Acceso: Desde la rotonda de Escorial se toma dirección Semario y al pasar por debajo de la autopista se gira a mano derecha y se cruza un puente. La zona de vertido está a unos 150 m a la orilla de la carretera.

Foto: C:\WIN32\

Página 1 Página 2

Registros: 14 1 de 1 (Filtrado)

FICHA TÉCNICA DE VERTEDERO

4.2.5. Constructor de consultas.

La construcción de consultas es ofertado por el Sistema como un comando integrado de amplia capacidad: desde él es posible vincular elementos, realizar sentencias Select, localizar los resultados en pantalla, revisarlos alfanuméricamente e incluso actualizar su información alfanumérica asociada o sus vínculos. Estas capacidades son interesantes para el administrador del Sistema pero pueden ser poco recomendables a disposición de un usuario sin una especialización suficiente. Por esa razón, se ha desarrollado un conjunto de funciones específicas para realizar consultas directas, con las que el usuario nunca corre el riesgo de crear alguna alteración en los datos.

4.3. Generación de planos.

La resimbolización temática es otro de los resultados que, con prontitud, se requiere de un Sistema de Información Geográfica; en este proyecto se ha abordado este aspecto bajo criterios similares a los expuestos en apartados anteriores, es decir, se mantiene el acceso al comando integrado tal como está construido por el Sistema pero también se ofrecen al usuario menos experto una serie de criterios prediseñados para obtener temáticos e incluso se han construido algunos que son directamente accesibles como dibujo. Siempre es posible que tales usuarios menos expertos puedan provocar un cambio en los criterios de creación de los temáticos, de manera que se recomienda que accedan exclusivamente a los dibujos ya generados.

Por otra parte, la validez del temático depende directamente de la estrategia de mantenimiento del SIG, toda vez que si se incorporan nuevos vertederos detectados posteriormente a la realización de este estudio, todos los temáticos han de ser reconstruidos. En tal caso, los ficheros de extensión .THM seguirán siendo válidos y el administrador del Sistema será el encargado de generar los nuevos temáticos. Como resultado demostrativo en este proyecto, se han elaborado los siguientes:

- ?? Mapa temático de vertederos por Municipio.
- ?? Mapa temático de vertederos por Número de vertedero.
- ?? Mapa temático de vertederos por Altitud.
- ?? Mapa temático de vertederos por Altura de vertedero.
- ?? Mapa temático de vertederos por superficie.
- ?? Mapa temático de vertederos por volumen de vertedero.
- ?? Mapa temático de vertederos por distancia a zonas habitadas.
- ?? Mapa temático de vertederos visibles desde carretera o ferrocarril.
- ?? Mapa temático de vertederos próximos al mar o a un río.

4.4. Tratamiento de las hojas de la BCN25.

Las circunstancias que condicionan esta operación de tratamiento de la cartografía BCN25 han sido referidas en el apartado 2.3. Se ha debido proceder en consecuencia, afrontando los siguientes trabajos:

- ?? Transformación de los planos restituidos, de naturaleza 3D, en 2D.
- ?? Eliminación de leyendas, rotulación de hojas y elementos similares de los planos tratados.
- ?? Unificación de la simbología de los planos de las tres campañas (restituidos, tratados en versión actualizada y tratados en versión no actualizada). Se han definido macroinstrucciones específicas para que todos los elementos gráficos se distribuyan sobre una secuencia de niveles uniforme. Dado que las tres campañas poseían codificación completamente diferente, se ha seguido como base la de las hojas restituidas (la que dispone de un menor número de niveles diferentes) y se han adecuado las otras.
- ?? Desarrollo de programas específicos para la inserción interactiva de las hojas 1/25000 como referencias externas sobre el dibujo maestro, normalmente el WORK.DGN.
- ?? Desarrollo de programas específicos para el control de niveles en las hojas 1/25000. Al ser incorporadas éstas al dibujo de trabajo en forma de referencias externas, se hacía necesaria una función global capaz de encender o apagar los niveles propuestos por el usuario mediante una sola orden, independientemente de cuántas y cuáles fueran las hojas insertadas en un momento dado.

5. Descripción técnica del desarrollo efectuado.

5.1. Sobre la plataforma SIG y su interés.

Con independencia de las especificaciones de plataforma y la elección indicados en el apartado 4, diversos especialistas indican que Geographics no cumple todos los requisitos de un SIG; en efecto, creemos que eso es cierto pero no lo es menos que diversos proyectos de nivel medio o alto en España ya la contemplan como una herramienta más de entre las utilizadas a lo largo de los mismos. Ello es, probablemente, porque surgiendo como desarrollo vertical desde una aplicación CAD (Microstation), sus capacidades de manipulación de grandes volúmenes de datos gráficos son muy notables. Además, el desarrollo personalizado de macros y comandos se realiza realmente en entorno CAD, lo que resulta particularmente adecuado para quien accede al SIG con buena formación en aquella especialidad. En esta segunda parte de la comunicación se describen aquellos procedimientos de selección y designación de objetos gráficos, siempre necesarios al realizar comandos personalizados, que resultan ser realmente tratables de forma sistemática, sea cual sea el entorno base de desarrollo desde el que se trabaja. Precisamente esa sistematización pretende ser la aportación más destacable de este trabajo.

5.2. Procesos de selección de objetos gráficos: su categorización a nivel de sistemas cad.

Desde el punto de vista aplicado, a nivel CAD las dos operaciones que permiten manipular a nivel programación elementos que ya pertenecen al dibujo son la **manipulación de conjuntos designados o elementos individuales a nivel de variable** y la **manipulación de cualquier dato existente en el registro de una entidad, también a nivel de variable**. En general, ambas técnicas mantienen una estructura conceptual semejante en cualquier plataforma CAD integrada. Para manipular conjuntos designados, Microstation crea un tipo de variable específico que se denomina **MbeElementSet**; cada uno de estos contiene identificadores de las entidades que contiene mediante un nuevo tipo de variable que se denomina **MbeSetMember**. A partir de uno de estos puede accederse a un **MbeElement**, que es un **registro de entidad**. Diversas **funciones del objeto Element** permiten acceder a sus datos concretos. Se pueden determinar cuatro procesos elementales:

?? Generación y recorrido de un conjunto designado interactivamente.

?? Designación interactiva individual de un elemento.

?? Acceso al último elemento de la B.D.G.

?? Filtrado de los elementos de la B.D.G. (o de un conjunto designado) mediante algún criterio relativo a su tipo, sus atributos o sus propiedades, generándose un conjunto designado por un procedimiento no interactivo sino de tipo “query”, convencional en las Bases de Datos.

5.2.1. Generación y manipulación de conjuntos designados.

En entorno MBE, la selección de objetos se efectúa mediante el método **conjunto.fromFence**, del objeto **MBEElementSet**, que valora la ocurrencia **conjunto** con los elementos que están dentro del recinto de selección. En el caso de nuestro código, el usuario ha tenido que definir previamente a la ejecución de la macro un cercado rectangular de esquinas P1 y P2. Una vez comprobado que el cercado estaba abierto, se lanza un ciclo While en el que se va recorriendo el contenido de la ocurrencia *conjunto*, a razón de un elemento por cada vuelta del ciclo. El primer elemento se captura con el método **conjunto.getFirst**, del objeto **MBEElementSet**, que valora la ocurrencia **miembro** como una del tipo **SEtMember**. Como este método **getFirst** devuelve la

```
Sub main
  Dim status As Integer
  Dim bloqueo As Integer
  Dim conjunto As New MbeElementSet
  Dim miembro As MbeSetMember
  Dim pto As MbePoint
  Dim elem As New MbeElement
  Dim posicion As Long
  Dim mensajes As Integer
  status = conjunto.fromFence (bloqueo)
  if status = MBE_NoFence then
    MbeWriteError "es preciso definir un
cercado"
  Exit sub
end if
status = conjunto.getFirst (miembro)
Do While status = MBE_Success
  posicion = elem.fromFile (miembro.filepos,
miembro.fileNum)
  print "*****"
  print elem.type
  print elem.filepos
  print elem.filesize
  print elem.color
  status = conjunto.getNext (miembro)
Loop
End Sub
```


constante **MBE_Success** en caso de que exista un primer elemento en el conjunto, la condición del While utiliza ese valor para controlar la salida en primera vuelta.

Cuando ya se dispone de la ocurrencia *miembro*, se puede acceder a sus propiedades **miembro.filepos** y **miembro.fileNum**, de modo que la invocación al método **elem.fromFile (miembro.filepos, miembro.fileNum)** valora la ocurrencia **elem** del objeto **MBEElement**, tras lo cual se pueden extraer de él las propiedades **elem.type**, **elem.filepos**, **elem.filesize** y **elem.color**, que son presentadas en pantalla. Para finalizar, el método **conjunto.getNext (miembro)** recoge el siguiente miembro del conjunto, al que se aplican las mismas operaciones y consideraciones.

5.2.2. Designación individual de elementos.

La designación individual de objetos se establece en MBE a partir del comando especial de entrada **MbeStartLocate**, que hace que las entradas de datos interactivas sean tomadas como designaciones de elementos; se continúa utilizando los métodos **mbestate.getinputdatapoint** y **mbesenddatapoint**, que provocan bajo el dominio del comando **MbeStartLocate** la designación pretendida. A continuación se recurre al método **elem1.fromlocate** del objeto **MbeElement** que valora la ocurrencia **elem1** como el elemento alcanzado en la designación y define su posición en la Base de Datos Geométrica. La función *posicion* devuelve este valor, que es utilizado en la rutina *Main* para valorar de nuevo el elemento **elem1** por medio del método **elem1.fromfile**, tras lo cual cualquier propiedad del elemento puede ser tratada desde el programa. Se presentan por pantalla la clave o posición y el tipo de elemento designado.

```
function posicion () as long
dim elem1 as new mbeelement
dim pdesig as mbepoint
  mbestartlocate 1,1,1,false
  mbewriteprompt "DESIGNAR ELEMENTO"
  mbegetinput 2,3
  stat = mbestate.getinputdatapoint (pdesig)
  mbesenddatapoint pdesig
  mbesendcommand "NULL"
  posicion = elem1.fromlocate ()
end function
'*****
sub main
dim clave as long
dim elem1 as new mbeelement
  clave = posicion
  clave = elem1.fromfile (clave)
  print clave
  print elem1.type
  print
end sub
```

5.2.3. Determinación del último elemento de la base de datos geométrica.

En MBE no existe una forma directa de acceder a la última entidad de la Base de Datos Geométrica pero sí un mecanismo indirecto muy sencillo: consiste en determinar la posición de su marca de fin de fichero mediante el método **mbedgninfo.endoffile**. El procedimiento de determinación consiste entonces en averiguar cuál es la posición del fin de la B.D.G. antes de dibujar la entidad, ya que el inicio de ésta ocupará precisamente esa posición.

En la macro adjunta se incorpora una subrutina **punto** que solicita uno interactivamente y luego se escribe la rutina principal, que comienza precisamente por determinar en la variable **posicion** dónde está la marca de Fin de Fichero; la línea se dibuja y se vuelve a determinar la **posicion2** de la marca fin de fichero; al efectuar a continuación una operación de *copia paralela* y designar el elemento a copiar mediante el método **mbelocateelement posicion** se está referenciando la recta recién dibujada. El resultado de la copia ocupa la posición memorizada en **posicion2**, por lo que es posible aplicar una última operación de cambio de color sobre ella siguiendo el mismo procedimiento.

```
Private pto as mbeipoint
sub punto ()
  mbegetinput 2,3
  mbewriteprompt "introducir punto"
  stat = mbestate.getinputdatapoint (pto)
end sub
sub main
dim posicion as long, posicion2 as long
dim linea as new mbeelement
posicion = mbedgninfo.endoffile
mbesendcommand "place line"
call punto
mbesenddatapoint pto
call punto
mbesenddatapoint pto
mbesendreset
posicion2 = mbedgninfo.endoffile
mbesendcommand "copy parallel keyin"
mbelocateelement posicion
mbesenddatapoint 0,0,0
mbesendreset
color = mbesettings.color
mbesettings.color = 2
besendcommand "change color"
mbelocateelement posicion2
mbesenddatapoint 0,0,0
mbesendreset
mbesettings.color = color
end sub
```

5.2.4. Filtros y selecciones por propiedad.

Una variante del mecanismo de selección propuesto en el apartado 14.4.1 se encuentra cuando se desea mantener en el conjunto designado creado mediante un procedimiento interactivo (ventana, etc.) sólo aquellas entidades que cumplan una determinada propiedad, normalmente relativa a su tipo o a sus atributos. El cuerpo principal de la macro que resuelve el problema en MBE es prácticamente una copia del proceso de selección de conjuntos estudiado en el apartado 13.4.1, pero con la incorporación de dos llamadas a una subrutina de nombre **filtro_tipo_elem**:

filtro_tipo_elem arraytipos,mbe_lin y

filtro_tipo_elem arraytipos,mbe_text

y la invocación al método **setlocatetypemask** del objeto **MbeState** con el argumento **arraytipos**.

La subrutina **filtro_tipo_elem**, proporcionada por el constructor de Microstation, recibe como argumento de entrada un tipo de elemento (**Mbe_line**, **Mbe_text**, etc) y devuelve el array de bits **arraytipos**, preciso para que el método **setlocatetypemask** del objeto **Mbestate** efectúe el filtrado. Básicamente, **arraytipos** es un array de 8 enteros, con un bit por cada uno de los tipos de elemento gráfico. El orden más bajo del primer elemento del array corresponde al elemento "tipo 1", el siguiente para el "tipo 2", etc.; el bit más bajo del segundo elemento del array es para el elemento 17, y así sucesivamente. A los efectos de esta obra, un análisis más pormenorizado de su funcionamiento queda fuera de nuestro campo de estudio, aunque su uso es siempre tal como se presenta aquí. A continuación se ofrece la macro que resuelve el problema.

```
Private arraytipos(1 to 8) as integer
*****
sub filtro_tipo_elem ( arraytipos%(), tipo_elem% )
dim numero_bit as integer
dim indice as integer
  numero_bit = (tipo_elem - 1) mod 16
  indice = lbound(arraytipos) + ( tipo_elem - 1 ) \ 16
  if numero_bit = 15 then
    arraytipos(indice) = arraytipos(indice) or -32768
  else
    arraytipos(indice) = arraytipos(indice) or 2^numero_bit
  end if
end sub
*****
Sub main
  Dim status As Integer
  Dim bloqueo As Integer
  Dim conjunto As New MbeElementSet
  Dim miembro As MbeSetMember
  Dim pto As MbePoint
  Dim elem As New MbeElement
  Dim posicion As Long
  Dim mensajes As Integer
  filtro_tipo_elem arraytipos,mbe_line
  filtro_tipo_elem arraytipos,mbe_text
  stat = mbestate.setlocatetypemask (arraytipos)
  status = conjunto.fromFence (bloqueo)
  if status = MBE_NoFence then
    MbeWriteError "es preciso definir un cercado"
  Exit sub
  end if
  status = conjunto.getFirst (miembro)
  Do While status = MBE_Success
    posicion = elem.fromFile (miembro.filepos, miembro.fileNum)
    print "*****"
    print elem.type
    print elem.filepos
    print elem.filesize
    print elem.color
    status = conjunto.getNext (miembro)
  Loop
End Sub
```

5.3. Procesos de manejo de bases de datos.

La programación contra una Base de datos está ofertada en MBE a partir de dos objetos básicos: **db**, para manipular la base de Datos y **Tb** para manejar una tabla; definidas las ocurrencias de estos dos objetos

```
db.name="resi"
db.modeForms=TRUE          'Forms=Text Screen
tb.name = "Tvertederos"

filterString="SELECT * from tvertederos where municipio like
'" + Municipio + "'"
tb.fencefilter=filterstring
saveMessages = MbeState.messages
MbeState.messages = 0
status = conjunto.fromFence (al lowLocked)
proceso del conjunto designado
```

sobre las que se quiere trabajar, las operaciones de selección de elementos pueden efectuarse mediante una cláusula SELECT convencional de SQL o mediante un cercado, en cuyo caso es posible proponer también un filtro de tipo query. En el fragmento adjunto, se establece el nombre de la base de datos mediante el método **db.name**, la tabla mediante **tb.name** y se define un filtro de cercado mediante el procedimiento **tb.fencefilter**. A partir de estas operaciones, el proceso del conjunto designado se efectúa por medio del método **conjunto.fromfence**, exactamente igual a como se ha mostrado en 5.2.1. Los elementos gráficos que pertenecen a ese conjunto designado serán los que, estando dentro del cercado (si el modo de captura es dentro) tienen atributos asociados que satisfacen la cláusula select propuesta.

6. Conclusión.

La participación de los autores en el proyecto referido ha permitido iniciar una línea de trabajos SIG, de entre los cuales se ha querido destacar la utilidad y el aprovechamiento obtenido del entorno de programación MBE. Asimismo, se ha puesto de manifiesto que, desde un punto de vista formativo, las operaciones más especializadas de selección de objetos o, dicho de otro modo, de programación contra la Base de Datos Geométrica del Sistema Gráfico, pueden ser sistematizadas en 4 categorías, en las cuales se fundamenta la práctica totalidad de las operaciones de desarrollo requeridas.

7. Referencias.

- [1] Maguire D.J. ; Goodchild M.F. ; Rhind D. *Geographical Information Systems*. Longman Scientific and Technical. 1991.
- [2] Laurini R. ; Thompson D. *Fundamentals of Spatial Information Systems*. Academic Press. 1992.
- [3] Microstation95² . *Microstation 95 BASIC Guide* Bentley Systems. 1997.
- [4] Otero C. *CAD en Ingeniería: una introducción*. (CICYT; DOC96-1828). En trámite de edición. 1999.