

Capítulo IV

Formato Propietario de ArcView

En este capítulo se describe la estructura del formato propietario de ArcView, denominado “Shp” o “Shapefile”.

4.1 Formato de un Shapefile

Los “Shapefiles”[ESRI, 98] son archivos que almacenan por un lado datos geométricos y por otro atributos de la información espacial en un conjunto de datos. La geometría de un objeto es almacenada como un “shape” el cual comprende un conjunto de coordenadas vectoriales.

Los “shapefiles” soportan los tipos básicos de elementos, es decir el punto, la línea y el polígono. Los atributos descriptivos son almacenados en un archivo de dBase. Cada atributo tiene asociado una relación de uno a uno con un registro de un “shape” asociado.

4.2 Tipos de Shape

Los “shapes” son las geometrías que puede tener un “shapefile”. Cada “shapefile” estará asociado a un objeto geométrico, el cual puede ser cualquiera de los valores que se muestran en la tabla 4.1. El campo “Valor”, es el valor con que es representada la geometría cuando se almacena en un archivo “shape”.

Valor	Shape Tipo
0	Shape Nulo
1	Punto
3	PoliLinea
5	Polígono
8	MultiPunto
11	PuntoZ
13	PoliLineaZ
15	PolígonoZ
18	MultiPuntoZ

Valor	Shape Tipo
21	PuntoM
23	PoliLineaM
25	PolígonoM
28	MultiPuntoM
31	MultiPatch

Tabla 4.1 Tipos de “shape” que existen en ArcView

Los tipos de “Shape” que no se encuentran en la especificación (2,4,6, etc., hasta 33) están reservados para un uso futuro. Actualmente los “shapefiles” están restringidos a almacenar un solo tipo de “shape”, a futuro se planea que guarden diferentes tipos de “shape”.

La figura 4.1 muestra la interfaz de ArcView con capas compuestas de los tres tipos básicos, puntos, líneas y polígonos. La interfaz esta dividida en dos partes, los menús y la vista. Dentro de los menús se cuenta con botones que realizan las operaciones básicas como los son los acercamientos (lupa con símbolo +), alejamientos (lupa con símbolo -) y paneos (manita). La vista se divide en dos el menú de capas (seleccionar y deseleccionar) y la vista.misma cuenta con un desplegado de coordenadas en la esquina superior derecha sobre la vista

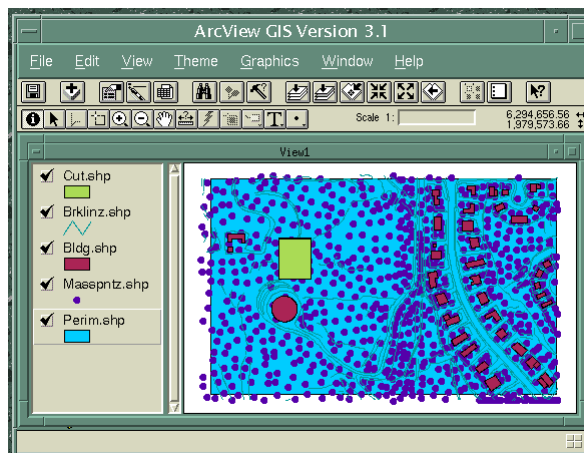


Figura 4.1 Interfaz de ArcView con diferentes “shapes”

4.3 Creación de un ShapeFile

Los “shapefiles” pueden ser creados a través de cuatro formas:

1. Exportar datos de cualquier fuente a un “shapefile” usando ARC/INFO[®], PC ARC/INFO[®], Spatial Database Engine[™] (SDE[™]), ArcView[®] GIS o BusinessMap[™] software.
2. Digitalizar a través del empleo de la herramienta de digitalización de ArcView GIS.
3. Programar usando Avenue[™] (ArcView GIS), MapObjects[™] ARC Macro Language (AML[™]) o Simple Macro Language (SML[™]).
4. Escribir directamente las especificaciones de un “shapefile” escribiendo un programa. También SDE, ARC/INFO , PC ARC/INFO, Data Automation Kit (DAK) y ArcCAD proveen software para hacer traducciones de datos. ARC/INFO de igual manera provee un traductor de coberturas a “shape”.

4.4 Elementos de un ShapeFile

Un ShapeFile de ESRI consta de un archivo principal, un archivo de índices y una tabla de dBase. En el archivo principal se tiene acceso directo a los registros de longitud variable donde cada registro es un “shape” o un objeto geométrico con una lista de sus vértices.

En el archivo de índices cada registro es un apuntador al registro correspondiente en el archivo principal.

La tabla de dBase almacena los atributos de los objetos con un registro por objeto.

La figura 4.2 muestra la correspondencia entre el archivo de índices (shx) y el archivo principal (shp).

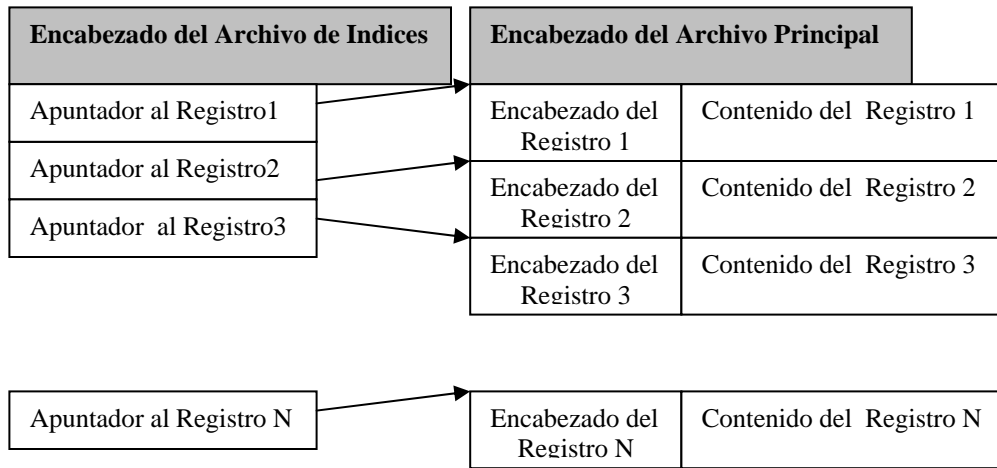


Figura 4.2 Relación entre los archivos “Shp” y “Shx”.

4.5 Nombres de los Shapefiles

Todos los archivos siguen la convención de 8 caracteres de nombre mas 3 de extensión. El archivo principal, el archivo de índices y el archivo de dBase tienen el mismo nombre.

El nombre del archivo comienza con un carácter alfanumérico (a-Z,0-9,-,_,), seguido de cero a siete caracteres más. La extensión del archivo principal es “.shp”, del archivo de índices es “.shx” y de la tabla de dBase es “.dbf”.

Ejemplo:

Archivo principal: Udla.shp

Archivo de índices: Udla.shx

Archivo de dBase: Udla.dbf

4.6 Tipos de datos numéricos de un shapefile

Un shapefile solamente maneja dos tipos de datos numéricos

1. Entero: número entero con signo de 32 bits (4 bytes)

2. Doble: número de punto flotante con 64 bits (8 bytes) de doble precisión con signo.

4.7 Tipos de ordenamiento de Bytes empleados en un shapefile

En un shapefile se mezclan dos tipos de ordenamiento de bytes:

1. LittleIndian: de derecha a izquierda es el orden de los bytes. Este método de ordenación de bytes se emplea principalmente en sistemas operativos como DEC OSF/1, DEC OpenVMS, MS-DOS y Windows NT.
2. BigIndian: de izquierda a derecha es el orden de los bytes. Se usa en muchos sistemas Unix incluyendo Sun, Hewlett-Packard, IBM y en Data General AviiON.

Ejemplo: el nombre de un dominio o la dirección de un email se encuentra en forma

LittleIndian: www.udlap.mx ⇐

Ahora bien si lo quisieramos en forma BigIndian sería así

BigIndian: 140.148.1.1 ⇒

4.8 Organización de los archivos shape

El archivo principal (.shp) consta de un encabezado de archivo, encabezado de registro y el registro, el cual es de longitud variable según el tipo de “shape”. Como se muestra en la figura 4.3 El encabezado de archivo es de tamaño fijo 100 bytes o 50 palabras de 2 bytes.

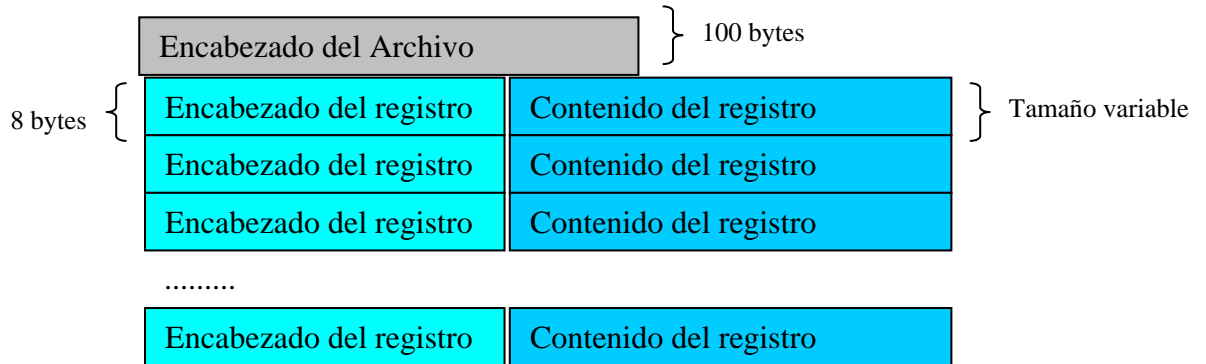


Figura 4.3 Organización del archivo principal

La organización del encabezado principal es el que se muestra en la tabla 4.2

El campo posición de la tabla, es el byte del archivo donde se encuentra. En todos los archivos “shape” solamente hay dos tipos de datos numéricos, dobles o enteros, lo que varía entre ellos es el orden del byte más significativo, ya sea BigIndian o LittleIndian.

Todos los campos descriptivos del Encabezado del Archivo Principal, Encabezado de Registro y Contenido del Registro del Archivo principal son del tipo LittleIndian, los demás números del archivo son tipo BigIndian.

Posición	Campo	Valor	Tipo	Orden Bytes
Byte 0	CódigoArchivo	9994	Entero	Big
Byte 4	Sin uso	0	Entero	Big
Byte 8	Sin uso	0	Entero	Big
Byte 12	Sin uso	0	Entero	Big
Byte 16	Sin uso	0	Entero	Big
Byte 20	Sin uso	0	Entero	Big
Byte 24	LongitudArchivo	File Length	Entero	Big

Posición	Campo	Valor	Tipo	Orden Bytes
Byte 28	Version	1000	Entero	Little
Byte 32	Shape Tipo	Shape Tipo	Entero	Little
Byte 36	Bounding Box	Xmin	Doble	Little
Byte 44	Bounding Box	Ymin	Doble	Little
Byte 52	Bounding Box	Xmax	Doble	Little
Byte 60	Bounding Box	Ymax	Doble	Little
Byte 68*	Bounding Box	Zmin	Doble	Little
Byte 76*	Bounding Box	Zmax	Doble	Little
Byte 84*	Bounding Box	Mmin	Doble	Little
Byte 92*	Bounding Box	Mmax	Doble	Little

Tabla 4.2 Campos del encabezado del Archivo Principal

La tabla 4.3 nos da la interpretación de los campos de la tabla 4.3

Posición	Campo	Descripción
Byte 0	CódigoArchivo	Código del Archivo
Byte 4 al Byte 20	Sin uso	Campos reservados para uso futuro
Byte 24	LongitudArchivo	Longitud del archivo en palabras de 2 bytes
Byte 28	Version	Version del archivo
Byte 32	Shape Tipo	Tipo de “Shape” del archivo, consultar la tabla 4.1
Byte 36 al Byte 92	Bounding Box	Es el rectángulo que rodea a una geometría determinada, es decir es el espacio suficiente para contener toda la geometría. El Bounding Box está definido por Xmin, Ymin, Xmax, Ymax. El Bounding Box del encabezado del archivo principal, es el rectángulo que contiene a todas las geometrías del archivo.

Tabla 4.3 Interpretación de los Campos del Encabezado Principal

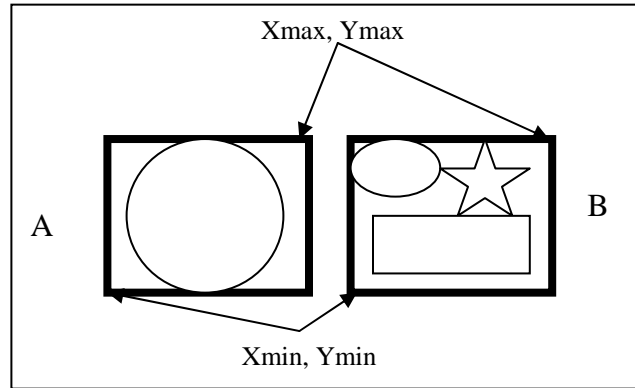


Figura 4.4. Bounding Box

Uno de los campos del archivo es el Bounding Box, el cual se maneja para una geometría (Figura 4.4 A) y también se maneja para varios objetos geométricos (Figura 4.4. B).

Temporalmente los “shapefiles” solamente tienen un tipo de geometría. En un futuro se tendrán en un archivo varios tipos de “shapes”.

El encabezado de cada registro almacena el número y tamaño de cada registro. Los encabezados de los registros tienen una longitud fija de 8 bytes.

Se toma como Byte 0 la posición donde comienza el encabezado del registro, el cual es de tipo entero y tiene un tamaño de 4 bytes. El campo de longitud del contenido es de tipo entero y nos indica el tamaño del registro “shape” así como se muestra en la tabla 4.4

Posición	Campo	Valor	Tipo	Orden Bytes
Byte 0	NúmeroRegistro	NúmeroRegistro	Entero	Big
Byte 4	TamañoRegistro	NúmeroRegistro	Entero	Big

Los números de registros comienzan en 1

Tabla 4.4 Descripción del encabezado de registros del archivo principal

El tamaño del registro está dado en palabras de 2 bytes, es decir, cada registro contribuye con (4 + tamaño del registro) palabras de 2 bytes.

Existen varios tipos de “shapes” como se mostró en la tabla 4.1. Según el tipo de objeto geométrico es el tamaño del contenido del registro. Es decir, los registros son de tamaño variable.

Por ejemplo, si el tipo de “shape” contiene un valor Nulo, nos indica que no existen datos geométricos para el archivo “shape”. El tipo Nulo se emplea principalmente durante la creación de los “shapefiles” y se les agrega información después de ser creados, su estructura se muestra en la tabla 4.5

Posición	Campo	Valor	Tipo	Número	Orden Bytes
Byte 0	Shape Tipo	0	Entero	1	Little

Tabla 4.5. Contenido del registro Null Shape (Tipo 0)

Otro tipo de “shape” es el Punto, el cual consiste de un par de coordenadas de tipo doble en el orden X,Y.

La estructura del punto se representa de la siguiente manera:

```
{
    Doble      X      //coordenada X
    Doble      Y      //coordenada Y
}
```

Posición	Campo	Valor	Tipo	Número	Orden Bytes
Byte 0	Shape Tipo	1	Entero	1	Little
Byte 4	X	X	Doble	1	Little
Byte 12	Y	Y	Doble	1	Little

Tabla 4.6 Contenido de un registro tipo Punto

La tabla 4.6 muestra el contenido del registro tipo punto.

A partir de los puntos se pueden generar MultiPuntos, estos se representan como un conjunto de puntos los cuales no están conectados en un orden específico.

La estructura del MultiPunto se representa como sigue:

```
{
    Doble[4]          Box    //Bounding Box (Xmin, Ymin, Xmax, Ymax)
    Entero           NumPuntos //Número de Puntos
    Punto[NumPuntos] Puntos  //Los Puntos del conjunto
}
```

Posición	Campo	Valor	Tipo	Número	Orden Bytes
Byte 0	Shape Tipo	8	Entero	1	Little
Byte 4	Box	Box	Doble	4	Little
Byte 36	NumPuntos	NumPuntos	Entero	1	Little
Byte 40	Puntos	Puntos	Punto	NumPuntos	Little

Tabla 4.7 Contenido del registro MultiPunto

La tabla 4.7 describe la estructura de un registro de multipuntos en un “shapefile”.

Otro “shape” importante lo constituye la PoliLinea, la cual es un conjunto ordenado de dos o más vértices. Cada par de vértices forma un segmento. El conjunto de segmentos pueden o no intersectarse.

La estructura de la polilínea es la siguiente:

```
{
    Doble[4]      Box    //Bounding Box (Xmin, Ymin, Xmax,
                    Ymax)
    Entero        NumSegmentos //Número de segmentos
    Entero        NumPuntos   //Número total de Puntos
    Entero [NumSegmentos] Parte //Indice al primer número de cada segmento
    Punto[NumPuntos] Puntos   //Los puntos del conjunto
}
```

La descripción de los campos de la Polilínea puede describirse de la siguiente manera:

- **Box:** Bounding Box de la PoliLinea almacenada en el siguiente orden Xmin, Ymin, Xmax, Ymax.
- **NumSegmentos:** Número de segmentos de la PoliLinea.
- **NumPuntos:** Número total de puntos para todas las partes.
- **Segmentos:** Arreglo de longitud NumSegmentos. Almacena para cada PoliLinea, el índice del primer punto en el arreglo de puntos (Punto[NumPuntos]).
- **Puntos:** Arreglo de longitud NumPuntos. Los puntos para cada parte de la PoliLinea son almacenados consecutivamente, después de los puntos de la parte uno siguen los de la parte dos y así sucesivamente.

La tabla 4.8 muestra la estructura de la forma en que se almacena un “shape” tipo polilínea.

Posición	Campo	Valor	Tipo	Número	Orden Bytes
Byte 0	Shape Tipo	3	Entero	1	Little
Byte 4	Box	Box	Doble	4	Little
Byte 36	NumPartes	NumPartes	Entero	1	Little
Byte 40	NumPuntos	NumPuntos	Entero	1	Little
Byte 44	Partes	Partes	Entero	NumPartes	Little
Byte X	Puntos	Puntos	Punto	NumPuntos	Little

Note: $X = 44 + 4 * \text{NumPartes}$

Tabla 4.8 Contenido del registro PoliLinea

El último “shape” que describiremos los constituyen los polígonos, un polígono consiste de uno o más anillos. Un anillo es una serie conectada de cuatro o más puntos que se cierran y no existe intersección entre ellos. Un polígono puede tener múltiples anillos externos. El orden de los vértices o su orientación indican de que lado del anillo es el interior del polígono. El vecindario a la derecha de un observador que camina siguiendo el orden de los vértices es el interior del polígono, los vértices de los anillos definen huecos cuando están en sentido contrario a las manecillas del reloj. Por consiguiente los vértices de un polígono sencillo van en el sentido de las manecillas del reloj. Esta especificación no admite puntos consecutivos con coordenadas idénticas. De otra forma se generarían longitudes o áreas de cero.

La estructura del Polígono y la descripción de los campos del polígono es exactamente la misma que de la PoliLinea.

Los requisitos que debe cumplir un “Shape” para ser un polígono son los siguientes:

- Los anillos son cerrados (el primer y último punto son los mismos)
- El orden de los anillos en el arreglo de puntos, no es significativo
- Los Polígonos almacenados en el “shapefile” deben ser limpios

Un Polígono limpio es aquel que no tiene intersecciones, es decir que los segmentos del Polígono no deben intersectarse con los segmentos de otros anillos o del mismo.

También un Polígono debe tener el lado correcto de la línea que lo define.

La tabla 4.9 nos muestra la estructura de los campos que componen el polígono:

Posición	Campo	Valor	Tipo	Número	Orden Bytes
Byte 0	Shape Tipo	3	Entero	1	Little
Byte 4	Box	Box	Doble	4	Little
Byte 36	NumPartes	NumPartes	Entero	1	Little
Byte 40	NumPuntos	NumPuntos	Entero	1	Little
Byte 44	Partes	Partes	Entero	NumPartes	Little
Byte X	Puntos	Puntos	Punto	NumPuntos	Little

Tabla 4.9 Estructura del registro polígono

4.9 Organización del Archivo de Índices

El archivo de índices (.shx) contiene un encabezado de 100 bytes seguido por encabezados de 8 bytes de longitud fija.

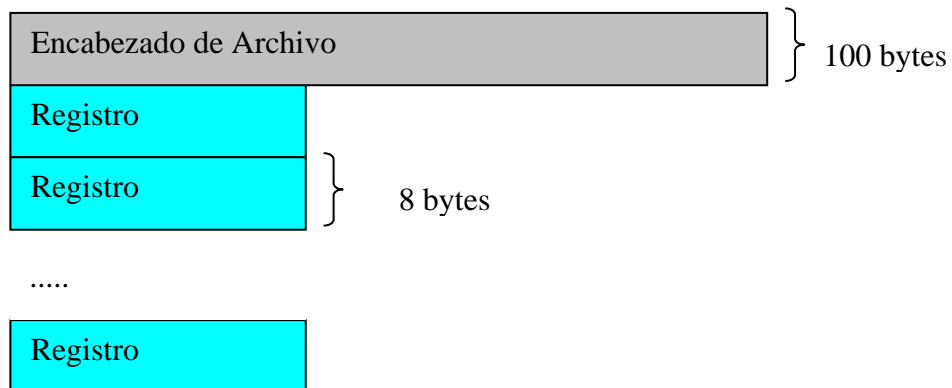


Figura 4.5 Organización del Archivo de Índices

El encabezado de este tipo de archivo tiene la misma estructura que el encabezado del archivo principal (.shx).

Cada registro guarda la posición donde comienza el registro en el archivo principal (Offset) y el tamaño del registro. Estos datos están dados en palabras de 2 bytes. El número de registro en el archivo de índices es el mismo número de registros que en el archivo principal. El tamaño del registro que se almacena en el archivo de índices es igual al tamaño del registro que se almacena en el archivo principal ya que estamos hablando del mismo registro. La estructura del registro del archivo de índices se muestra en la tabla 4.10

Posición	Campo	Valor	Tipo	Orden Bytes
Byte 0	Offset	Offset	Entero	Big
Byte 4	TamañoRegistro	TamañoRegistro	Entero	Big

Tabla 4.10 Registro del Archivo de Indices

Para almacenar la información descriptiva se utilizan los archivos dBase, estos archivos contienen las características de los atributos o las llaves de las tablas a través de las cuales se pueden efectuar joins. Este formato es el tipo estándar de cualquier archivo DBf basado en tablas para aplicaciones de Windows o D.O.S. Cualquier conjunto de campos presentados en la tabla deben de cumplir con los siguientes requisitos:

1. El nombre del archivo debe ser el mismo que el del archivo principal y que del archivo de índices, con excepción de la extensión la cual debe ser “.dbf”
2. La tabla debe contener un registro por “shape”
3. En el encabezado el valor del año debe ser desde 1900

4.10 Jerarquía de las clases de los Shapefiles

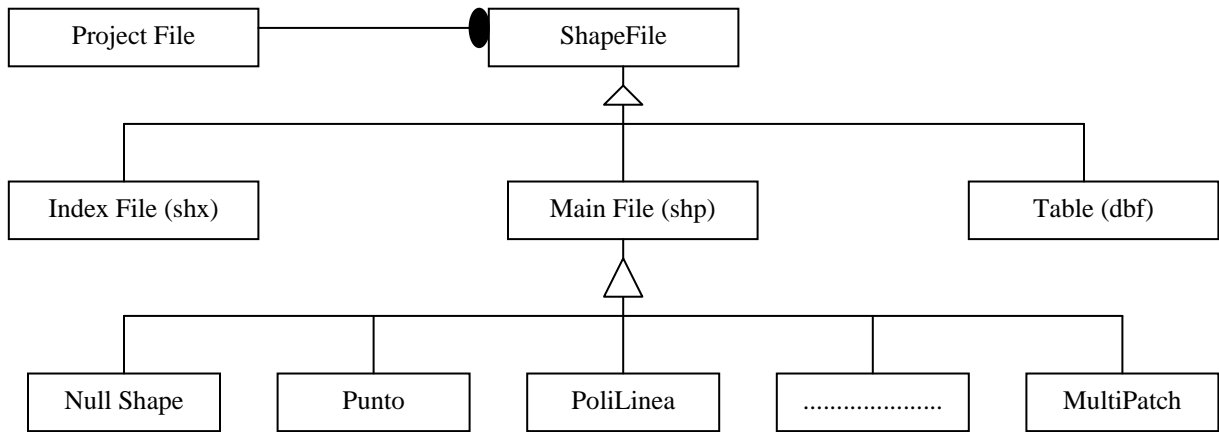


Figura No 4.6 Jerarquía de Clases de ArcView

Este esquema de clases se construyó a partir de las relaciones, y atributos de los elementos que conforman un proyecto de ArcView.

La intención por la cual creamos esta jerarquía es con el fin de ir obteniendo clases comunes o similares entre los “Shapefiles” y el estándar de OpenGIS. De todas las clases de la jerarquía, la única que no se ha comentado es el “Archivo de Proyecto”, el cual es un archivo que no está incluido en la especificación de los “Shapefiles”. Este archivo lo genera ArcView al guardar un proyecto y tiene las siguientes características:

- Es archivo de texto.
- Contiene información sobre las ventanas, documentos, botones y scripts que solamente son útiles para ArcView .
- Nombre, color, fecha, autor, escala y ubicación de los “Shapefiles”

Un archivo de proyectos tiene cero o más shapefiles, estos a su vez están compuestos de tres archivos, los cuales por su estructura pueden crecer en tipos de datos. De hecho la especificación que se tiene está adelantada a los tipos de datos que maneja el ArcView 3.1.

Características de los shapefiles
<ul style="list-style-type: none"> • Otros Sig pueden leerlos • Facilidad de exportación • Alto volumen de uso • Almacenamiento en archivos binarios • Extensible, su estructura le permite crecer • Espacio 2D y 3D

Tabla 4.11 Características de los “Shapefiles”

4.11 Conclusión

La especificación técnica de los “Shapefiles” acaba de ser liberada y esta se encuentra adelantada al software que actualmente se encuentra disponible en el mercado. Por estos días ESRI estará presentando su nuevo software al cual hace referencia la especificación. Los formatos de los “Shapefiles” están planeados para crecer y en un futuro un archivo principal tendrá diferentes “shapes” y no solamente uno como actualmente sucede.

Para nuestra implementación podemos pensar que nuestro formato debe ser abierto y que este preparado a crecer, al igual que lo maneja ArcView.

Tampoco podemos descartar el desarrollo de más exportadores ya que en el desarrollo del presente trabajo encontramos otras especificaciones como lo son el DGN de Intergraph,

DAT (formato nativo) y MIF (formato de intercambio) de MapInfo por solo mencionar los más conocidos.

Como mencionamos anteriormente el uso de los archivos “shape” o de cualquier otro SIG, es limitado por las aplicaciones existentes, de ahí la importancia del manejo de un estándar, de los estándares propuestos del que más información se puede conseguir es del estándar de OpenGIS, el cual se describe a fondo en el siguiente capítulo.