

# **UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA**

## **MapServer y su aplicación a SIG**



**Víctor H. González Jaramillo**

e-mail: [vhgonzalez@utpl.edu.ec](mailto:vhgonzalez@utpl.edu.ec)

**Universidad Técnica Particular de Loja**

[www.utpl.edu.ec](http://www.utpl.edu.ec)

Telf. 593 07 2 570275 ext. 2613

**Equipo de Sistemas de Información Geográfica**

<http://serversig.utpl.edu.ec/sigutpl/index.php>

**Loja – Ecuador / Junio –2005**

## **Objetivos**

### **Objetivo general:**

Tener pleno conocimiento de la utilización del servidor de mapas MapServer versión 4.0 y su aplicación a los Sistemas de Información Geográfica.

### **Objetivos específicos:**

- Conocer la estructura del servidor, como se configura y los componentes fundamentales del mismo.
- Introducirnos en el entorno de los mapas georeferenciados y el conocimiento del archivo de configuración Mapfile.
- Iniciación en la interface de usuario y la utilización de los archivos templates en formato HTML.
- Conocer las capacidades que brinda el servidor de mapas MapServer para realizar consultas temáticas a partir de los mapas georeferenciados.



# MapServer y su aplicación a SIG

## Índice de contenidos

### INTRODUCCIÓN

Cap 1.	Estructura de una aplicación orientada al Web.	2
	1.1 Instalación del Servidor Apache 1.3.24	3
	1.2 MapServer	6
	1.2.1. Partes de una aplicación MapServer	6
	1.2.2. Primeros pasos para la instalación	8
	1.2.3. Utilidades de MapServer	10

### VISUALIZACIÓN DE MAPAS GEOREFERENCIADOS UTILIZACIÓN DE MAPSERVER 4.0

Cap 2.	Mapas georeferenciados y el Mapfile	12
	Ejemplo 2.1. Un mapa con un Layer	12
	Ejemplo 2.2. Un mapa con dos Layers	16
	Ejemplo 2.3. Usando clases para hacer un mapa visible	18
	Ejemplo 2.4. Etiquetado de Layers	20
	Ejemplo 2.5. Adhiriendo un Layer Raster	24
	Ejemplo 2.6. Diferentes formatos de salida	29
	Ejemplo 2.7. Diferencia entre modo map y browse	32
Cap 3.	Interface de Usuario y Templates HTML	36
	Ejemplo 3.1. Controles Pan y Zoom	36
	Ejemplo 3.2. Control de layers	41
	Ejemplo 3.3. Adherir un Scalebar (barra de escala)	46
	Ejemplo 3.4. Adherir un mapa de referencia	51
	Ejemplo 3.5. Adherir una leyenda	56
Cap 4.	Query y Templates HTML	61
	Ejemplo 4.1. Class level – query template	61
	Ejemplo 4.2. Layer – header y footer template	67
	Ejemplo 4.3. Mapas con header y footer template	73
	Ejemplo 4.4. El objeto Querymap	75
Bibliografía		79



## INTRODUCCIÓN

### 1. Estructura de una aplicación orientada al Web.

Para armar una estructura en cuanto a tecnología se refiere se deberán tener instalados y configurados los siguientes componentes de software:

1. Instalación y configuración de un servidor Web
2. Instalación y configuración del servidor de mapas MapServer 4.0

Como se puede ver en el siguiente esquema, el sistema se encuentra orientado al Web y debe funcionar en la red (Fig. 1. Estructura del Sistema SIG).

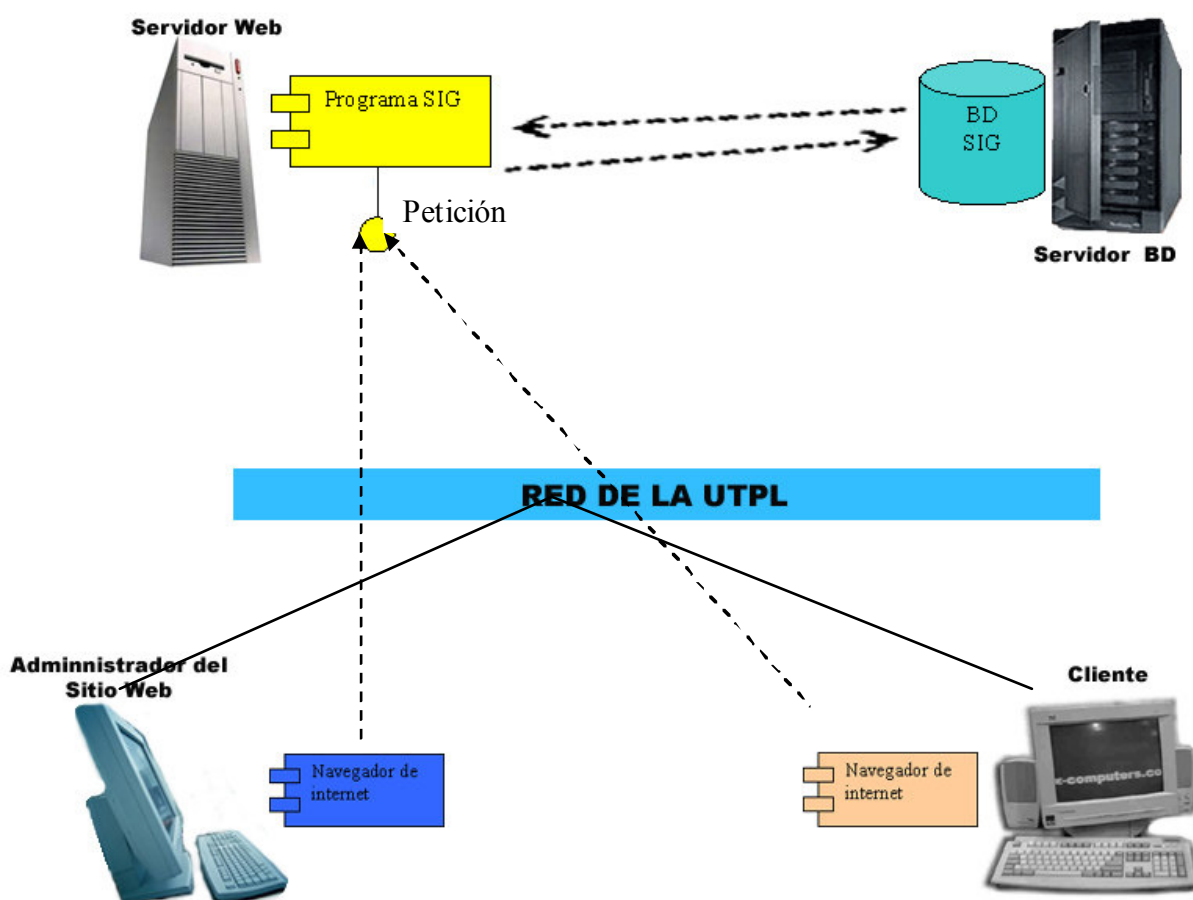


Figura 1. Estructura del Sistema SIG



Una vez que se ha explicado la arquitectura de la aplicación se procede con la instalación del software necesario para poder levantar un servidor Web, que servirá para alojar las páginas y los mapas, recibir las peticiones de los clientes y poder dar una respuesta adecuada a dichas peticiones.

### 1.1 Instalación del servidor Apache 1.3.24

Para levantar el servidor Web se ha escogido el Servidor Apache, este es servidor multiplataforma, que se puede usar en Windows y Linux como sistemas operativos de referencia ya que son unos de los más difundidos en la actualidad.

Sobre la versión del servidor se ha considerado conveniente usar una versión estable y que esté vigente actualmente.

Como pasos para la instalación podemos seguir los siguientes:

- Conseguir una versión del software Apache 1.3.24. Si no tiene el instalador se lo puede conseguir en la siguiente dirección Web:

<http://serverwatch.internet.com/>

La instalación se la realiza como cualquier software siguiendo las indicaciones que aparecen en el cuadro de diálogo de instalación (Fig. 1.1. Instalación Servidor Apache).

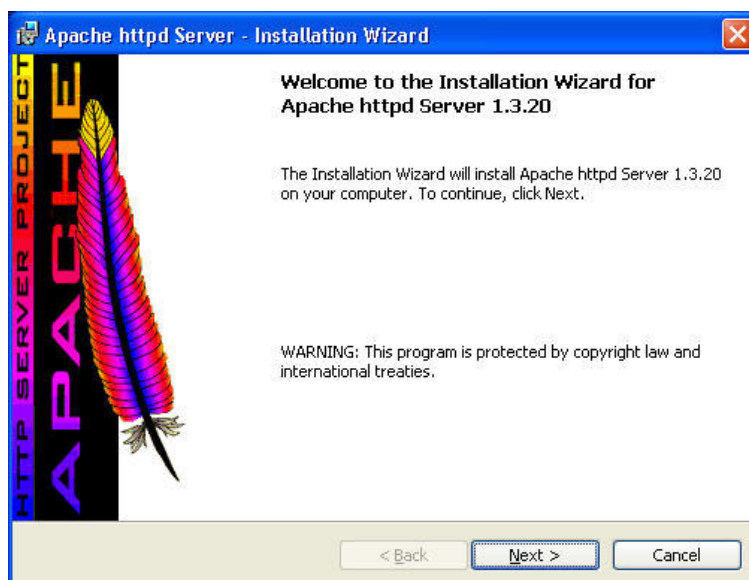


Fig. 1.1. Instalación servidor Apache



- En la Figura 1.2. (Instalación Servidor Apache) se indica algunos de los parámetros esenciales de la configuración del servidor Web:
  - Network Domain o dominio de red si lo tenemos
  - Nombre del servidor el cual se lo debe colocar en minúsculas o en su defecto la dirección IP del PC o computador.
  - E-mail del administrador del Servidor Web
  - Como parámetro final se escoge la forma de funcionamiento, en este caso se escogerá “Run when started manually, only for me”.

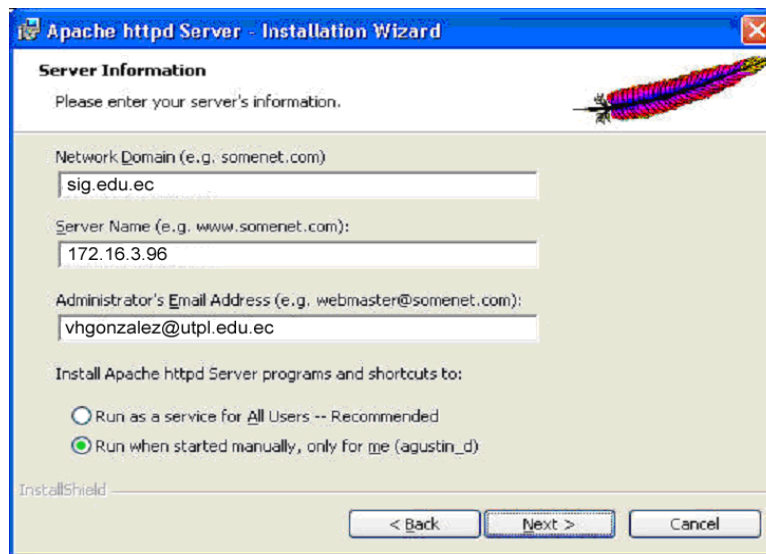


Fig 1.2. Instalación Servidor Apache

- Se continúa con la instalación y se escoge el tipo completo de instalación.
- Como paso siguiente en la instalación se deja el fólder por defecto para alojamiento de los archivos.
- Finalmente se termina la instalación y se tiene el servidor Web listo para su funcionamiento (Fig. 1.2.3. Instalación servidor apache).

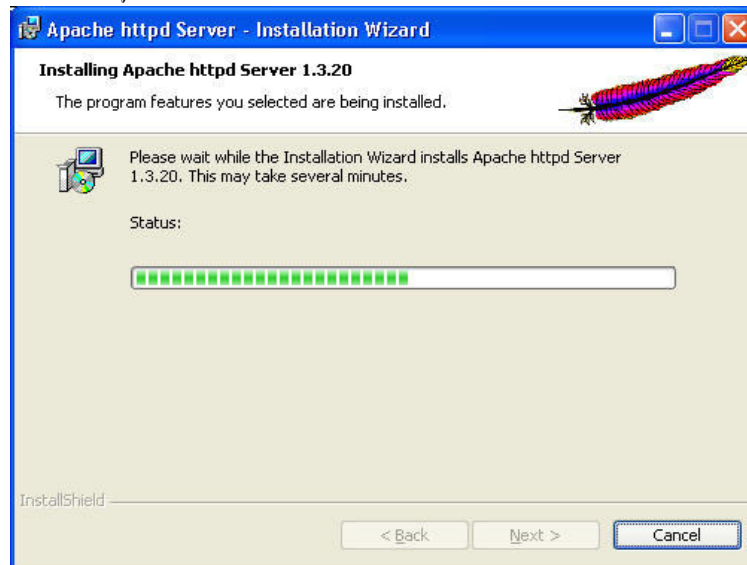


Fig. 1.3. Instalación Servidor Apache

Para comprobar que el servidor Web que acabamos de instalar se encuentra funcionando correctamente, deberemos hacer lo siguiente:

- Buscar en el Menú de Inicio -> Barra de programas -> Apache http Server y levantar el servicio del Servidor Apache
- Abrir una ventana en un navegador (ejemplo Internet Explorer o Netscape Navigator) y colocar en la barra de direcciones ya sea el nombre del PC o su dirección IP, lo cual será el equivalente para realizar la prueba en la Intranet.

En la pantalla del navegador deberá aparecer la pantalla de bienvenida del Servidor Apache (Fig. 1.4 Servicio del Servidor Apache).

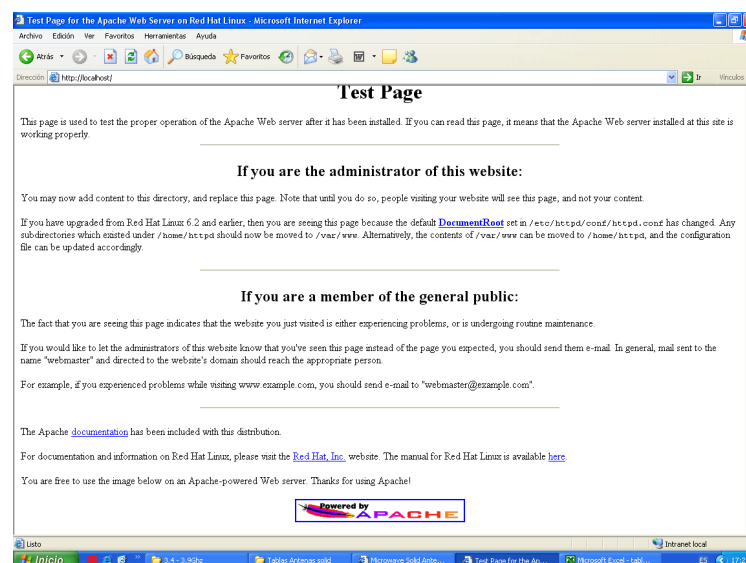


Fig. 1.4 Servicio del Servidor Apache



## 1.2. MapServer 4.0

### 1.2.1. Partes de una aplicación MapServer.

MapServer generalmente funciona como una aplicación **CGI** (**CGI** es una norma para establecer comunicación entre un servidor Web y un programa, de tal modo que este último pueda interactuar con Internet) y corre dentro de un servidor http. Este argumento es verdadero a menos que usted este trabajando una aplicación más complicada con Mapscript y un acceso a la API de MapServer.

El CGI de MapServer utiliza los siguientes recursos:

- ✓ Un servidor http como Apache o Internet Information Server.
- ✓ Software MapServer
- ✓ Un archivo de inicialización que active la primera vista de la aplicación de MapServer (opcional).
- ✓ Un archivo Mapfile que controle lo que MapServer hace con los datos.
- ✓ Un Template File que controle la aplicación de MapServer en la ventana del browser (navegador de Internet).
- ✓ Una fuente de datos SIG.

MapServer es normalmente instalado en el directorio cgi-bin del servidor http, y la información o fuente de datos SIG es almacenada en el directorio de documentos del servidor http.

La lógica de la aplicación MapServer es demostrada a continuación en el diagrama de flujo (Fig. 1.2.1. Lógica de la aplicación MapServer).

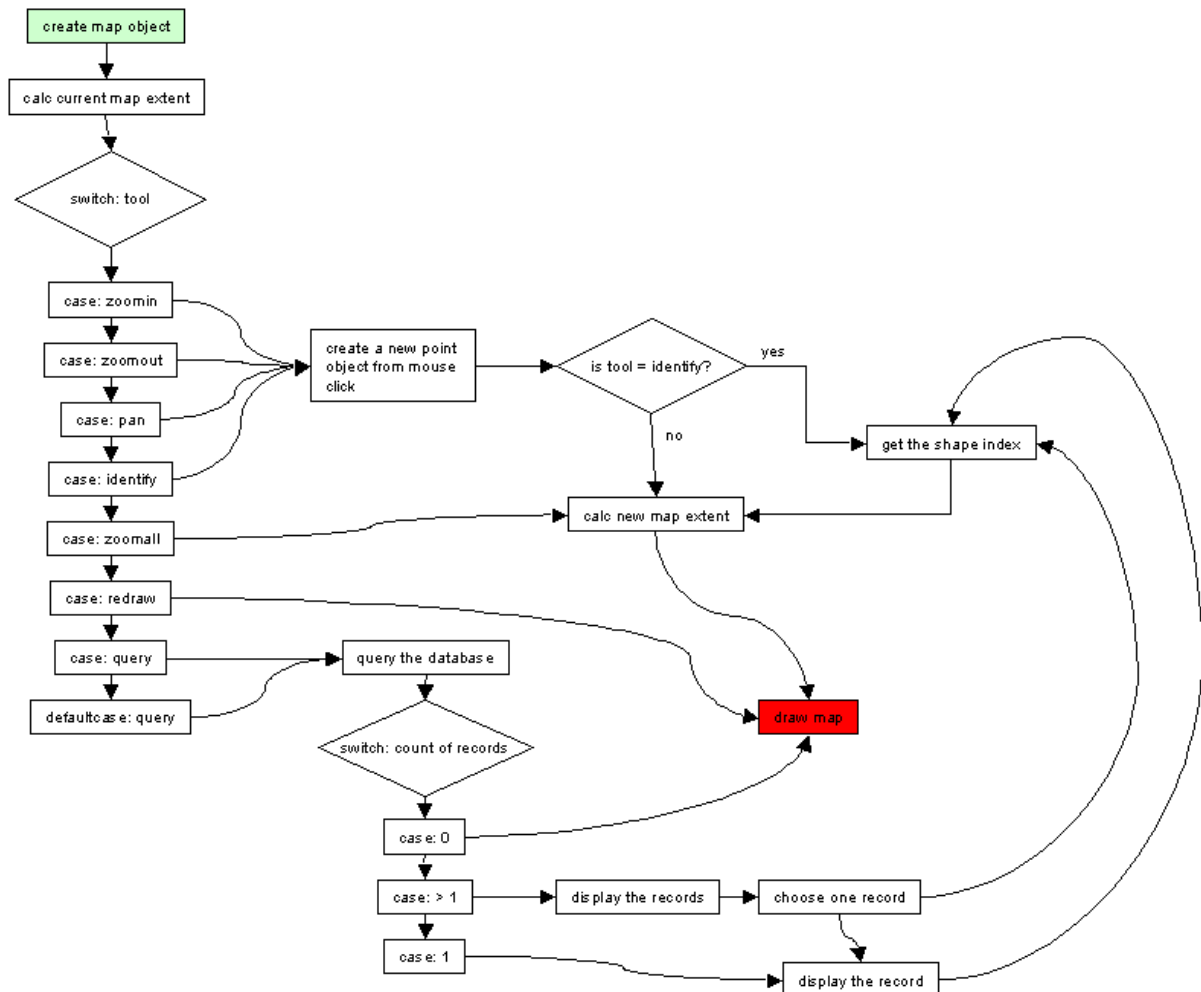


Fig. 1.2.1 Lógica de la aplicación MapServer

### 1.2.1.1. El archivo de inicialización

Este archivo puede ser parte de otro archivo html, pero por simplicidad puede ser un archivo separado. El archivo de inicialización se usa para enviar una consulta inicial al servidor http que retorna un resultado del servidor de mapas. MapServer está sin estado, este es iniciado y ejecuta una consulta cada vez que esta es recibida, para esto el archivo de inicialización es requerido, para pasar una variedad de parámetros a la aplicación (hidden). El archivo de inicialización es por lo general una archivo html, cuya extensión es .htm o .html. Alternativamente, se puede construir un hiperlink al servidor MapServer. Este puede pasar los parámetros básicos requeridos por la Aplicación CGI MapServer.



### 1.2.1.2. El Mapfile

El Mapfile define los datos a ser usados en la aplicación, muestra y consulta de parámetros. El Mapfile también contiene información acerca de cómo se debe dibujar el mapa, la leyenda y el resultado de realizar una consulta. El Mapfile tiene normalmente una extensión .map.

### 1.2.1.3. El Template File

El archivo Mapfile controla como las salidas de mapas y las leyendas de MapServer se deben presentar en la página html, este opera como cualquier otro archivo html a excepción de que contiene etiquetas que pueden ser modificadas en el CGI de MapServer. El Template File permite al autor del mapa colocar la posición de presentación del mapa, la leyenda y determina que vías son disponibles para que el usuario interactúe con la aplicación MapServer (browse, query, zoom, etc.). Para producir el documento html que se envía al browser MapServer usa palabras clave en el archivo Template y las reemplaza con información que se encuentra en la fuente de datos SIG. Cuando un Template File es usado para crear un archivo html, este es almacenado generalmente con extensión .html.

### 1.2.1.4. El Conjunto de Datos SIG

El CGI MapServer usa archivos de formato ESRI – shapefile como formato vector por default, en formato raster se puede utilizar algunos otros formatos, dependiendo de cómo MapServer es compilado. Por default, MapServer soporta archivos geoTiff y archivos Tiff. Otros formatos de datos pueden ser usados, pero esto es un poco más complicado. El conjunto de datos SIG puede ser ubicado en un directorio, el cual es referenciado en el Mapfile. Los Shapefiles pueden ser optimizados para uso con MapServer con las utilidades shptree y sortshp.

## 1.2.2 Primeros pasos

Lo primero es conocer como se debe instalar MapServer. Si usted no tiene los permisos correspondientes, el administrador del sistema podría ayudarle con los mismos. Para sistemas Windows, los archivos binarios pre-compilados son disponibles a través de la dirección Web <http://mapserver.gis.umn.edu/win32binaries.html>. Aquí se tiene las instrucciones correspondientes para la compilación y la instalación, las cuales están disponibles para los siguientes sistemas operativos:

- Windows
- Unix
- Cygwin en Win32



### 1.2.2.1. MapServer CGI y programas utilitarios para Windows

Los binarios incluidos con la distribución MapServer 4.0 son probados para trabajar en Windows XP, 2000, NT versión 4 y Windows 98. MapServer no es probado en Windows Me.

El archivo de nuestro interés es mapserv.exe. Los otros archivos ejecutables son los archivos de utilidades del MapServer, las utilidades de MapServer serán revisados en los contenidos posteriores (1.2.3. Utilidades de MapServer).

### 1.2.2.2. Requerimientos

- Sistema operativo Windows.
- Servidor Web: MS IIS, Apache Server, Netscape Server, etc.

Como paso previo a la instalación de MapServer en el sistema operativo Windows se debe tener los binarios, los cuales pueden ser descargados de la dirección Web <http://mapserver.gis.umn.edu/dload.html>.

Del paquete comprimido se extraen los archivos y se los coloca en el disco duro en C:\MapServer.

### 1.2.2.3. Instalación de las utilidades de MapServer

Las utilidades de MapServer pueden ser extraídas en cualquier directorio (ejemplo: C:\MapServer). Si usted desea acceder a estas utilidades desde cualquier directorio, usted necesita adherir en su directorio de instalación el PATH de la variable de entorno. Alternativamente, usted puede extraer las utilidades en su directorio Windows (C:\Windows o C:\WINNT).

### 1.2.2.4. Instalación del CGI MapServer

1. Para instalar el programa CGI MapServer, copie mapserv.exe en el directorio "bin" del CGI-BIN del servidor Web. (En Microsoft IIS, El directorio es "C:\inetpub\Scripts" o "C:\inetpub\wwwroot\cgi-bin". En apache v1.3.x, este se encuentra usualmente en "C:\Program Files\Apache Group\Apache\cgi-bin".)

2. Copie el directorio "Proj" en "C:\". (Su directorio debe quedar de la siguiente manera "C:\Proj".).

3. Copie los archivos del directorio "lib" en C:\WINNT\System o C:\WINDOWS\System32. Alternativamente, usted puede actualizar el PATH del sistema para incluir el path del directorio "lib" (ejemplo: C:\MapServer\lib).



### 1.2.2.5. Testeo del CGI MapServer

Para testear que el programa CGI está trabajando, digite `mapserv.exe` en la línea de comando. Usted puede visualizar el siguiente mensaje:

This script can only be used to decode form results and should be initiated as a CGI process via a httpd server.

Usted puede visualizar un popup indicando que un DLL (como `bgd.dll`) no ha sido encontrado. Usted podría necesitar copiar todos los DLL requeridos (ejemplo `bgd.dll`, y `proa.dll`) para el mismo directorio que el programa `mapserv.exe` (`c:\Windows`)

Ahora digite `mapserv -v` en la línea de comando para obtener el siguiente mensaje:

```
MapServer version 4.4.0-beta3 OUTPUT=GIF OUTPUT=PNG OUTPUT=JPEG
OUTPUT=WBMP
SUPPORTS=PROJ SUPPORTS=FREETYPE SUPPORTS=WMS_SERVER
INPUT=SHAPEFILE
DEBUG=MSDEBUG
```

Esto indica que formatos de datos y otras opciones son soportados por `mapserv.exe`. Asumiendo que usted tiene su Web Server levantado, copie `mapserv.exe`

### 1.2.3. Utilidades de MapServer.

Las utilidades de MapServer pueden residir en cualquier directorio. Estas utilidades son incluidas con la distribución de MapServer.

#### **shp2img**

Propósito: Crear un mapa de un Mapfile. La salida de este es PNG o GIF dependiendo que versión de la librería GD que es usada.

Sintaxis: `shp2img -m [map file] -o [image] -t -l [layers]`

#### **legend**

Propósito: Crear una leyenda de un Mapfile. La salida es PNG o GIF dependiendo de que versión de la librería GD sea usada.

Sintaxis: `legend [map file] [output image]`

#### **scalebar**

Propósito: Crear un scalebar (barra de escala) para un Mapfile. Las salidas de esto pueden ser PNG o GIF dependiendo de la versión de la librería GD usada.

Sintaxis: `scalebar [map file] [output image]`



### **sortshp**

Propósito: Ordenar un shapefile en una columna basado en un orden ascendente o descendente. Soporta columnas integer, double, y tipos de columna string. Usado para priorizar formar al renderizar o etiquetar.

Sintaxis: sortshp [infile] [outfile] [item] [ascending|descending]

### **sym2img**

Propósito: Crear un graphic dump de un symbol file. La salida puede ser PNG o GIF dependiendo de la versión de la librería GD.

Sintaxis: sym2img [symbolfile] [outfile]

### **shptree**

Propósito: Crear un quadtree- basado en un índice especial de un shapefile. La profundidad del árbol es calculada para cada nodo del árbol (quadtree cell) conteniendo 8 shapes. No use el valor por default para archivos de puntos, un valor entre 6 y 10 parece ser el adecuado para trabajar. Su medida puede variar y usted necesitara hacer algunas pruebas con estas.

Sintaxis: shptree [shpfile] [depth]

### **tile4ms**

Propósito: Crear un cuadro índice en el shape file para trabajar con MapServer y las características de TILEINDEX. El programa crea un shapefile de forma rectangular para la extensión de todos los shapefiles listados en el [metafile] (un shapefile nombrado por línea) y la Base de Datos con el nombre de archivo para cada tile shape en una columna llamada LOCATION y que es requerido por mapserv.exe.

Sintaxis: tile4ms [metafile] [tilefile] [-tile-path-only

-tile-path- es solo una etiqueta opcional que especifica que solo el path de el shape file podría ser almacenado en el etiqueta LOCATION.



## VISUALIZACIÓN DE MAPAS UTILIZACIÓN DE MAPSERVER 4.0

### 2. Mapas georeferenciados y el Mapfile

La idea general es tomar un shapefile (archivo del software ArcView) y visualizarlo en un browser mediante el servidor de mapas MapServer.

#### Ejemplo 2.1. Mapa con un Layer Simple



MapServer puede crear una imagen y almacenarla en un directorio local o ser enviada directamente para ser presentada en un Web browser, como en este ejemplo.

Usted puede ver esto entrando a la dirección URL: <http://utpl.edu.ec/sig/curso/practical/>

Este URL puede ser descompuesto en tres partes:

- La primera parte, <http://utpl.edu.ec/sig/curso/cgi-bin/mapserv40?>, llama el programa CGI de MapServer. Si usted invoca este cgi puede obtener el siguiente mensaje:

No query information to decode. QUERY\_STRING is set, but empty.

- La segunda parte está compuesta por la cadena de consulta. El parámetro "map=c:/Archivos de programa /Apache Group /Apache /htdocs /curso/ cap1/ ejemplo1 /lojamap.map" dice al programa CGI MapServer (mapserv40) que Mapfile debe procesar/analizar.



- El caracter ampersand (&) es usado por el programa CGI para separar el parámetro que fue pasado. El siguiente parámetro pasado es “mode=browse”. Este dice a “mapserv40” que es lo que debe hacer con la salida del Mapfile, en este caso se le dice a “mapserv40.exe” que coloque la imagen directamente en el Web browser (el cliente), creando primeramente una imagen temporal en el servidor. El la variable de MapServer “modo” toma otros valores “map”. Por ejemplo, si usted usa “mode=browse”, MapServer podría colocar la imagen en un directorio temporal dentro del servidor.

El archivo Mapfile debe lucir de la siguiente manera:

```
MAP                                     # Especificación del objeto MAP
NAME loja                               # Nombre del objeto MAP
IMAGETYPE PNG                           # Definición del tipo de imagen
                                         # Extensión georeferenciada del mapa
EXTENT 476041.48 9438571.50 1186767.15 10161286.94
SIZE 400 300                             # Tamaño de presentación del mapa
SHAPEPATH "data"                         # Fuente de origen de los datos
IMAGECOLOR 255 255 255                   # Color de relleno de la imagen
TEMPLATEPATTERN "loja"                   # Definición de archive template
UNITS METERS                              # Definición de unidades de medida

WEB                                       # Definición del objeto WEB
TEMPLATE 'ejemplo2.1.html'
IMAGEPATH 'c:/Archivos de programa/Apache Group/ Apache/ htdocs/
curso/ cap1/ practical/temp/'
IMAGEURL 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap1/practical/temp/'
END

PROJECTION                               # definición de proyección para MAP
"proj=laea"
"ellps=clrk66"
"lat_0=45"
"lon_0=-100"
END

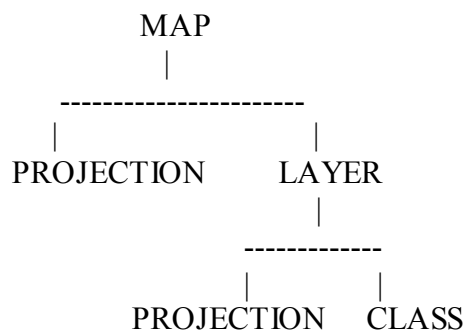
# Inicio de la definición del layer
LAYER                                     # Definición Layer Mapa Ecuador
NAME states
DATA cuador2
STATUS DEFAULT
TYPE POLYGON
PROJECTION                               # Definición de proyección para LAYER
"init=epsg:2163"
END
CLASS
COLOR 232 232 232
OUTLINECOLOR 32 32 32
END
END                                       # Fin de la definición de LAYER

END                                       # Fin del archivo MAP
```



Acorde a la Página de referencia del MapFile, este es el mecanismo de configuración básica que se usa para MapServer. El Mapfile está hecho de “objetos” y cada objeto posee palabras claves y otros objetos. Este tiene una estructura jerárquica como la de que algunos objetos están por encima de otros, en la cima de esta jerarquía se encuentra el objeto MAP, todos los demás objetos están bajo este. Aquí se encuentran objetos como el objeto PROJECTION que pueden pertenecer al objeto MAP o al objeto LAYER (que a su vez pertenece al objeto MAP). Cada objeto comienza con nombre del objeto y termina en “END”.

Si revisamos el archivo .map encontraremos la siguiente estructura:



Echemos una Mirada a las palabras clave (parámetros) dentro del objeto MAP:

### IMAGETYPE

La palabra clave IMAGETYPE es usada para definir que formato de imagen podría usar el programa CGI MapServer para la salida de las imágenes. En este caso se usa el color indexado PNG (similar al GIF). Este podría ser GIF, si se compila la librería GD con soporte para GIF, WBMP, o JPEG. También se puede especificar otros formatos de salida (PDF, SWF, GeoTIFF) asumiendo que se ha compilado el soporte para los mismos y que el OUTPUTFORMAT es de este tipo.

### EXTENT

Este parámetro especifica las dimensiones de salida del mapa. Este necesita ser en las mismas unidades de los datos. En este caso nuestra unidad de salida son los metros. Para extraer los valores de la extensión, usted puede usar ArcView u otro software SIG.

### SIZE

Este es el tamaño de la imagen (el mapa) que el MapServer puede generar, en pixels. Así el mapa es de 400 pixels de ancho por 300 pixels de alto.

### SHAPEPATH

Esta es la ruta para los datos de sus layers. Usted puede proveer rutas absolutas (path i.e. “/us/Project/data” o “C:/projects/data”) o rutas relativas para la ubicación de sus mapfile (como en el ejemplo). Esta ruta no debe ser accesible al Web. No tiene que



entrar directamente al Web, solo tiene el usuario que hacerlo correr en el servidor, y los datos pueden ser leídos en el SHAPEPATH.

### **IMAGECOLOR**

Este es el color de background del mapa. Los valores son RGB y valores como 255 Red, 255 Green y 255 Black resultan en un back ground de color blanco.

Veremos como lucen los parámetros del objeto LAYER:

### **LAYER**

Marca el inicio de un objeto LAYER dentro de un objeto MAP. Usted puede especificar los layers que desee y el límite para esto es de 100 por default. Usted puede cambiar este parámetro para lo cual hay que recompilar el CGI MapServer.

### **NAME**

Este es el identificador del nombre de cada uno de los layers especificados.

### **DATA**

El nombre del dato (shapefile en este caso). MapServer soporta formatos vectoriales y otros shapefiles que ESRI usa de ORG library (parte del GDAL software).

### **TYPE**

¿Qué tipo de dato es este? Si este es un dato vectorial, usted puede especificar si este es un POLYGON, LINE (usted usa LINE si sus datos técnicamente son POLYLINE), o un POINT. Usted también puede especificar RASTER o ANNOTATION.

### **STATUS**

Los layer pueden ser colocados en ON u OFF basados en su STATUS. DEFAULT es siempre en ON o visible. ON u OFF funciona cuando el nombre del LAYER es pasado como parte del parámetro del query string.

Veremos como lucen los parámetros del objeto CLASS:

### **CLASS**

Marca el inicio de un objeto CLASS dentro de un objeto LAYER. Usted puede especificar muchas clases en un layer y usted está limitado por default a 50. Se tendrá que recompilar el CGI MapServer para cambiar este valor.

### **COLOR**

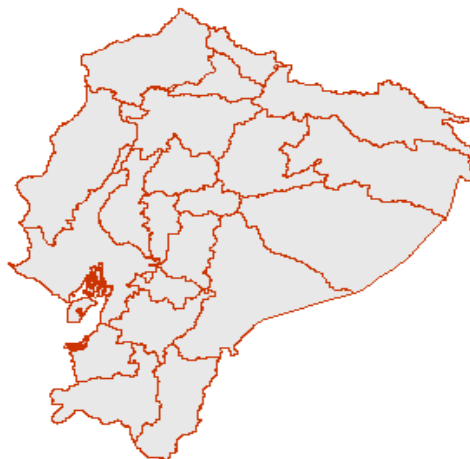
Este es el color de relleno del polígono. En caso de que el TYPE sea LINE, este es el color de línea. Los valores son en formato RGB.

### **OUTLINECOLOR**

Este es el color de línea de salida de los polígonos. Este valor está dado en RGB.



## Ejemplo 2.2. Mapa con dos Layers



En el primer ejemplo, esta imagen fue generada por un link al código del tag <img> para el URL: <http://utpl.edu.ec/sig/curso/practica2/>. Así es como el resto de los ejemplos de esta sección trabaja.

Usted puede darse cuenta que el mapa es casi similar que en el primer ejemplo. Si, pero el Mapfile es diferente por que ahora tendremos la definición de dos layers.

Así es como luce el archivo Mapfile:

```
MAP                                     # Definición del objeto MAP
  NAME loja
  IMAGETYPE PNG
  EXTENT 476041.48 9438571.50 1186767.15 10161286.94
  SIZE 400 300
  SHAPEPATH "data"
  IMAGECOLOR 255 255 255
  TEMPLATEPATTERN "loja"
  UNITS METERS

WEB                                     # Definición del objeto WEB
  TEMPLATE 'ejemplo2.2.html'
  IMAGEPATH 'c:/Archivos de programa/Apache
  Group/Apache/htdocs/curso/cap1/practica2/temp/'
  IMAGEURL 'c:/Archivos de programa/Apache
  Group/Apache/htdocs/curso/cap1/practica2/temp/'
END

PROJECTION                             # Definición del objeto PROJECTION
  "proj=laea"
  "ellps=clrk66"
  "lat_0=45"
  "lon_0=-100"
END
```



```
# Inicio de la definición de los LAYER
LAYER                                # Inicio del primer LAYER
  NAME states
  DATA cuador2
  STATUS DEFAULT
  TYPE POLYGON
  PROJECTION
    "init=epsg:2163"
  END
  CLASS
    COLOR 232 232 232

  END
END                                # Fin del primer LAYER

LAYER                                # Inicio del segundo LAYER
  NAME provincias_lineas
  DATA cuador2
  STATUS DEFAULT
  TYPE LINE
  PROJECTION
    "init=epsg:2163"
  END
  CLASS
    COLOR 204 51 0
  END
END                                # Fin del segundo LAYER

# Fin de la definición de los Layers

END                                # Fin del objeto MAP
```

El Mapfile es también similar al primer ejemplo excepto que aquí existen dos objetos LAYER. El primero que contiene un mapa de la República del Ecuador, este es una capa de tipo polígono, a la cual se le quitará el objeto OUTLINECOLOR. La segunda es también una capa de tipo polígono, pero se especifica que se usará como objeto línea (line) y se sobrepone a la primera, con lo cual logramos mostrar una sola imagen o un solo mapa. La diferencia ahora radica principalmente en el color del borde del objeto, que ha sido definido ahora por el segundo layer.



## Ejemplo 2.3. Despliegado de Clases en un Layer



Por usar atributos de información en los datos, nosotros podemos crear un Mapfile como este.

Así es como luce el archivo del Mapfile:

```
MAP                                     # Inicio del objeto MAP
  NAME loja
  IMAGETYPE PNG
  EXTENT 476041.48 9438571.50 1186767.15 10161286.94
  SIZE 400 300
  SHAPEPATH "data"
  IMAGECOLOR 255 255 255
  TEMPLATEPATTERN "loja"
  UNITS METERS

  WEB
    TEMPLATE 'ejemplo2.3.html'
    IMAGEPATH 'c:/Archivos de programa/Apache
  Group/Apache/htdocs/curso/cap1/practica3/temp/'
    IMAGEURL 'c:/Archivos de programa/Apache
  Group/Apache/htdocs/curso/cap1/practica3/temp/'
  END

  PROJECTION
    "proj=laea"
    "ellps=clrk66"
    "lat_0=45"
    "lon_0=-100"
  END

  # Inicio de la definición de layers
  LAYER                                 # El layer de Ecuador inicia
    NAME states
```



```

DATA cuador2
STATUS DEFAULT
TYPE POLYGON
PROJECTION
  "init=epsg:2163"
END
CLASSITEM "NOM_PROV"
CLASS
  EXPRESSION 'LOJA'
  COLOR 32 32 32
END
CLASS
  COLOR 232 232 232
END
END                                     # El layer de Ecuador termina

LAYER                                     # El layer de provincia inicia
NAME provincia_linea
DATA cuador2
STATUS DEFAULT
TYPE LINE
PROJECTION
  "init=epsg:2163"
END
  CLASSITEM "NOM_PROV"                 # Especificación del campo de BD
CLASS                                   # Definición del objeto CLASS
  EXPRESSION 'LOJA'
  COLOR 204 51 0
END                                     # Definición del objeto CLASS
END                                     # Fin del layer de provincia

# Termina la definición de los layers

END                                     # Fin del objeto MAP

```

El Mapfile es todavía similar a los dos primeros ejemplos pero con unos pequeños parámetros adicionales en los objetos LAYER y CLASS. Observemos los parámetros adicionales:

### CLASSITEM

Esta palabra clave es usada para especificar que atributos se usan para separar un objeto de la clase. En el ejemplo el atributo es "NOM\_PROV". Si usted abre el archivo de la base de datos asociada con el shapefile del layer, usted podrá ver aquí una columna (atributo) llamado "NOM\_PROV" y si buscamos entre la información almacenada podrá identificar un registro que contenga 'LOJA' en el campo "NOM\_PROV".

### EXPRESSION

Por cada clase, nosotros especificamos que valores de atributos usamos. Esta es la forma simple del valor EXPRESSION. EXPRESSION puede ser más complejo incluso que esto. Usted puede revisar el material referente al Mapfile para ver que es lo que usted puede hacer con EXPRESSION.



## Ejemplo 2.4. Etiquetando el mapa



Se puede también adherir labels al mapa, lo cual es indispensable para identificar objetos geográficos dentro de un mapa.

Así es como deberá lucir el archivo Map file:

```
MAP
NAME loja
IMAGETYPE PNG
EXTENT 476041.48 9438571.50 1186767.15 10161286.94
SIZE 800 600
SHAPEPATH "data"
IMAGECOLOR 255 255 255
UNITS METERS
FONTSET "fonts/fonts.list"

WEB
TEMPLATE 'ejemplo2.4.html'
IMAGEPATH 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap1/practica4/temp/'
IMAGEURL 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap1/practica4/temp/'
END

PROJECTION
"proj=laea"
"ellps=clrk66"
"lat_0=45"
"lon_0=-100"
END

LAYER # El primer layer inicia
```



**Universidad Técnica Particular de Loja**  
*La Universidad Católica de Loja*



Sistemas de Información Geográfica

```

NAME states
DATA cuador2
STATUS DEFAULT
TYPE POLYGON

PROJECTION
  "init=epsg:2163"
END

CLASSITEM "Nom_prov"
LABELITEM "Nom_prov"
CLASS

  COLOR 232 232 232
  EXPRESSION ./
  LABEL
    COLOR 0 0 0
    SHADOWCOLOR 218 218 218
    SHADOWSIZE 2 2
    TYPE BITMAP
    SIZE GIANT
    POSITION CC
    PARTIALS FALSE
    BUFFER 2
  END
END
END                                     # El primer layer termina

LAYER                                     # inicio del segundo layer
NAME provincias_line
DATA cuador2
STATUS DEFAULT
TYPE LINE
PROJECTION
  "init=epsg:2163"
END
  CLASSITEM "NOM_PROV"
CLASS
  EXPRESSION 'LOJA'
  COLOR 204 51 0
END
END                                     # Fin del primer layer
# Fin de la definición de Layers

END # Fin del objeto Map

```

Aquí se introduce algunos parámetros aparte del objeto LABEL:

### FONTSET

Aquí se especifica la ruta completa de nuestro archivo truetype fontlist (lista de fuentes). Este archivo lista cada una de las fuentes disponibles. Ver el archivo de documentación de fonset para más información. FONSET es un parámetro del objeto MAP.



## **LABELITEM**

Este especifica que atributos de datos se usa para el etiquetado, en este caso “Nom\_prov”. LABELITEM es un parámetro del objeto LAYER.

## **LABEL**

Marca el inicio del objeto LABEL. El objeto label puede ser usado bajo otros objetos (Ejemplo: El objeto SCALEBAR).

## **COLOR**

En el objeto LABEL, COLOR especifica el color de la etiqueta de texto.

## **SHADOWCOLOR**

Este especifica el color de la sombra de la etiqueta de texto.

## **SHADOWSIZE**

Especifica el tamaño de la sombra. El valor correspondiente para las variables X e Y en pixeles. Para “2 2” quiere decir dos pixels de ancho y dos pixels de alto.

## **TYPE**

En el objeto LABEL, TYPE especifica que tipo de fuente se va a usar. Nosotros podemos escoger entre TRUETYPE o BITMAP (el constructor en fuentes). Nosotros escogeremos BITMAP.

## **FONT**

Si usted especifica TYPE como TRUETYPE, usted necesita especificar que fuente usará. El valor de esta es el “alias” en el archivo de lista de fuentes.

## **SIZE**

Si usa Fuentes truetype, el valor del tamaño es en pixels. Si es bitmap, usted puede escribir “small” o “large”.

## **ANTIALIAS**

Este convierte el antialias truetype de ON en OFF. Recuerde el valor no es ON u OFF pero es TRUE o FALSE.

## **POSITION**

Donde la posición de la etiqueta de texto es en relación a los puntos del label. Los valores son una combinación de posiciones verticales y horizontales. Usted puede escoger lo siguiente para una alineación vertical: C para el centro, L para la izquierda y R para la derecha. Para la alineación de la etiqueta de texto en el centro del label ID usted debe usar el valor “CC” (center-center). O si usted quiere por abajo del ID, usted debe de usar el “LL”. Otra vía es que el MapServer decida la mejor posición de sus etiquetas. Para esto usted debe de estar usando el valor “AUTO”.



### **PARTIALS**

Indica a MapServer donde generar las etiquetas incompletas o no. Por default aquí no se generan fragmentos de una etiqueta de texto. Los valores de esta es TRUE o FALSE.

### **MINDISTANCE**

Esta es la mínima distancia en pixels entre etiquetas duplicadas. Usted puede incrementar este valor y observar que sucede con dicho cambio.

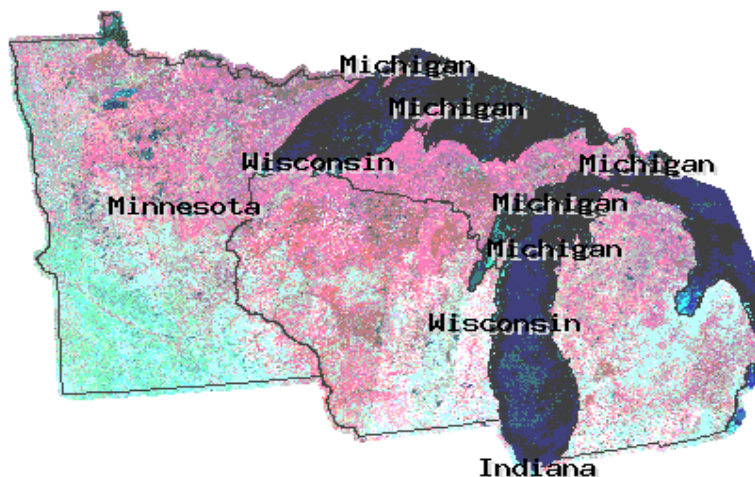
### **BUFFER**

La cobertura (en pixels) de cada etiqueta. Esta es usada para reforzar la legibilidad. Un BUFFER de 4 pixels asegura que ninguna otra etiqueta se dibujará dentro de los cuatro pixels. Cambie los valores para visualizar que sucede.

Usted puede también crear etiquetas separadas de un layer POLYGON. Usted puede hacer este dato de tipo ANNOTATION.



## Ejemplo 2.5. Adherir un Layer Raster.



MapServer puede utilizar datos raster como vectoriales. A través del uso de la librería GDAL, MapServer puede introducir y mostrar múltiples formatos raster. En versiones anteriores a la 4.0, las entradas raster eran limitadas a un layer simple, escala de grises o color indexado de imágenes. MapServer ahora soporta imágenes RGB y multispectral (multi-layer). Este ejemplo muestra como seleccionar un layer para mostrarlo.

Sin embargo MapServer 4.0 ahora usa la librería GD2 para generar la salida de imágenes, este ahora soporta RGB (24 bits o color verdadero). Así como con 8-bits (color indexado o escala de grises) PNG, usted ahora también puede utilizar PNG24 (color verdadero) para la salida. Este ejemplo usa PNG24 como IMAGETYPE. Como con entradas de RGB, hay un desarrollo notable cuando usamos PNG24.

Cuando usted revise el Mapfile, usted podrá ver que el nuevo objeto LAYER se coloca después de la definición del layer POLYGON. MapServer muestra los layers en orden inverso. El primer layer definido es dibujado en el fondo del mapa. Así si nosotros dibujamos la capa “Mapa del Ecuador” que es un POLYGON, esta estaría en el fondo. Desde que la capa raster se dibuja sobre esta, nosotros no la podríamos ver. Por eso la primera capa debe tener el STATUS de OFF. El estado del layer LINE es definido bajo el layer raster para lo cual se lo dibuja por encima (usted puede ver esto). Esto es porque nosotros separamos la capa LINE de la capa POLYGON. Las etiquetas se deben dibujar sobre todo.

Así es como luce el archivo Mapfile:



```
MAP
NAME ejemplo
IMAGETYPE PNG
EXTENT 201621.496941 -294488.285333 1425518.020722 498254.511514
SIZE 400 300
SHAPEPATH "data"
IMAGECOLOR 255 255 255
UNITS METERS
FONTSET "fonts/fonts.list"

WEB
  TEMPLATE 'ejemplo2.5.html'
  IMAGEPATH 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap1/practica5/temp/'
  IMAGEURL 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap1/practica5/temp/'
  END

PROJECTION
  "init=epsg:2163"
END

SYMBOL
  NAME 'line'
  TYPE ELLIPSE
  POINTS 1 1 END
  FILLED TRUE
END

# Inicio de la definición de los layers
LAYER # inicio del primer layer
  NAME states
  DATA states_ug1
  STATUS OFF
  TYPE POLYGON
  PROJECTION
    "init=epsg:4326"
  END
  CLASSITEM "CLASS"
  CLASS
    EXPRESSION 'land'
    COLOR 232 232 232
  END
END # layer Polígono STATES termina aquí

LAYER # Layer Modo RASTER
  NAME modis
  # Especificación de la fuente de datos para la imagen raster
  DATA "raster/mod09a12003161_ug1_ll_8bit.tif"
  STATUS DEFAULT
  TYPE RASTER
  PROCESSING "BANDS=1,2,3"
  OFFSITE 71 74 65
```



```

PROJECTION
  "init=epsg:4326"
END
END # modo raster termina aquí

LAYER                                # Layer de estado lineas
  NAME                                states_line
  DATA                                states_uql
  STATUS                                DEFAULT
  TYPE                                LINE

  PROJECTION
    "init=epsg:4326"
  END

  CLASSITEM                            "CLASS"
  CLASS
    EXPRESSION 'land'
    SYMBOL      'line'
    COLOR       64 64 64
    SIZE        1
  END
END                                     # Layer de estado líneas termina aquí

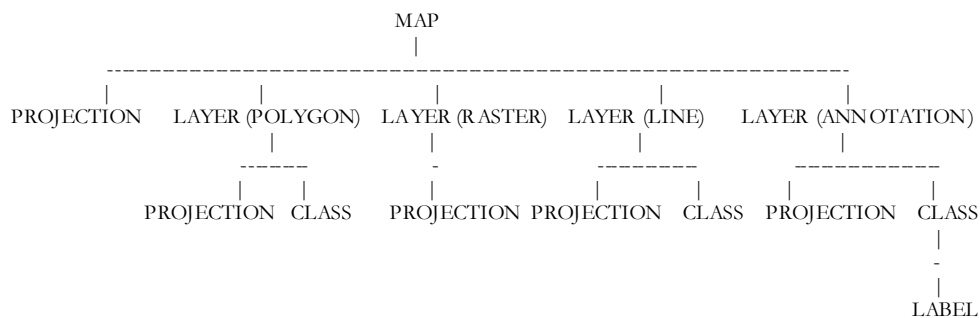
LAYER                                # Etiquetado del layer estado empieza
  NAME states_label
  DATA states_uql
  STATUS DEFAULT
  TYPE ANNOTATION

  PROJECTION
    "init=epsg:4326"
  END

  CLASSITEM                            "CLASS"
  LABELITEM                             "STATE"
  CLASS
    COLOR 232 232 232
    EXPRESSION ./
    LABEL
      COLOR 0 0 0
      SHADOWCOLOR 218 218 218
      SHADOWSIZE 2 2
      TYPE BITMAP
      SIZE GIANT
      POSITION CC
      PARTIALS FALSE
      BUFFER 2
    END
  END
END                                     # Fin de etiquetado de layer state
# Fin de la definición de LAYER
END                                     # Fin del objeto MAP

```

Esta es la estructura jerárquica de los objetos del Mapfile:



El archivo Mapfile ahora está empezando a ser muy extenso y se está usando solo dos archivos de datos. A continuación se verá los nuevos parámetros introducidos en el Mapfile:

### IMAGETYPE

Esto no es nuevo pero el valor “PNG24” si lo es. PNG24 es la versión del formato PNG de 24-bits o color verdadero. A cambio, en vez de estar limitados para 256 combinaciones de colores para formatos de salida, MapServer ahora tiene millones de colores.

### SYMBOLSET

Definición del path de la definición del archivo de símbolo. El archivo símbolo en este archivo es referenciado por el parámetro SYMBOL en el objeto CLASS.

**DATA** “raster/mod09a12003161\_ugl\_ll\_8bit.tif”

En el nuevo objeto incluido en LAYER, el parámetro DATA direcciona a una imagen GeoTIFF. Como fuentes de datos vectoriales, MapServer soporta múltiples formatos de archivos raster. Este soporte es logrado a través de la librería GDAL

### TYPE RASTER

Cuando se usan datos raster (imagenes) se deberá utilizar el valor RASTER del parámetro TYPE, como opuesto al POLYGON, LINE o PUNTO valores de datos vectoriales y ANNOTATION para identificación de etiquetas.

**PROCESSING** "BANDS=1,2,3"

Este parámetro del objeto LAYER es nuevo en MapServer 4.0. La palabra clave PROCESSING tiene muchos valores pero en este caso se está usando la selección de una banda multispectral en la imagen a mostrar. Estos valores en el string o cadena de datos que es pasado a la librería GDAL.

### OFFSITE

Este parámetro indica a MapServer los valores de pixel para renderizar como background (o ignorer). Usted puede obtener los valores de los pixels usando processing



**Universidad Técnica Particular de Loja**  
*La Universidad Católica de Loja*



Sistemas de Información Geográfica

de imagen o manipulación de programas de imagen (ejemplo Imagine, Photoshop, Gimp)

Para comparar la velocidad de creación de mapas cuando se usa imágenes RGB con una imagen de color indexado, reemplace las siguientes líneas en el Map file:

```
DATA "raster/mod09a12003161_ugl_ll_8bit.tif"  
STATUS DEFAULT  
TYPE RASTER  
PROCESSING "BANDS=1,2,3"  
OFFSITE 71 74 65
```

Con estas:

```
DATA "raster/mod09a12003161_ugl_ll_idx.tif"  
STATUS DEFAULT  
TYPE RASTER  
OFFSITE 70 74 66
```

También, trate de cambiar el IMAGETYPE por PNG24 to PNG.



## Ejemplo 2.6. Diferentes formatos de salida para los mapas.

Aparentemente el mapa presentado es igual al anterior, la diferencia radica en el formato de salida que se ha codificado en el archivo Mapfile.



Dependiendo del formato que usted escoja, la imagen no podría mostrarse en el browser. Si el link falla para la visualización en su browser, haga clic derecho en gráfico anterior y guardar el link como el formato que usted especifico en su Mapfile.

Así es como debe lucir el archivo Mapfile:

```
MAP
  NAME loja
  IMAGETYPE png
  EXTENT 476041.48 9438571.50 1186767.15 10161286.94
  SIZE 800 600
  SHAPEPATH "data"
  IMAGECOLOR 255 255 255
  TEMPLATEPATTERN "loja"
  UNITS METERS
  FONTSET "fonts/fonts.list"

  WEB
    TEMPLATE 'ejemplo2.6.html'
    IMAGEPATH 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap1/practica6/temp/'
    IMAGEURL 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap1/practica6/temp/'
  END
```



# Definición de los formatos de salida

```
OUTPUTFORMAT
  NAME png                                # Especificación del formato PNG
  DRIVER "GD/PNG"
  MIMETYPE "image/png"
  IMAGEMODE PC256
  EXTENSION ".png"
END
OUTPUTFORMAT
  NAME png24                              # Especificación del formato PNG24
  DRIVER "GD/PNG"
  MIMETYPE "image/png"
  IMAGEMODE RGBA
  EXTENSION ".png"
END
OUTPUTFORMAT
  NAME jpeg                                # Especificación del formato jpeg
  DRIVER "GD/JPEG"
  FORMATOPTION "QUALITY=75"
  MIMETYPE "image/jpeg"
  IMAGEMODE RGB
  EXTENSION ".jpg"
END
OUTPUTFORMAT
  NAME GTiff                              # Especificación del formato GTiff
  DRIVER "GDAL/GTiff"
  MIMETYPE "image/tiff"
  IMAGEMODE RGB
  EXTENSION ".tif"
END
OUTPUTFORMAT
  NAME swf                                # Especificación del formato
  #Shocwave de flash swf
  DRIVER "SWF"
  MIMETYPE "application/x-shockwave-flash"
  EXTENSION ".swf"
  IMAGEMODE PC256
  FORMATOPTION "OUTPUT_MOVIE=SINGLE"
END
PROJECTION
  "proj=laea"
  "ellps=clrk66"
  "lat_0=45"
  "lon_0=-100"
END
LAYER                                     # Definición del primer layer
  NAME states
  DATA cuador2
  STATUS DEFAULT
  TYPE POLYGON
PROJECTION
```



```

"init=epsg:2163"
END

CLASSITEM "Nom_prov"
LABELITEM "Nom_prov"
CLASS

    COLOR 232 232 232
    EXPRESSION ./
    LABEL
        COLOR 0 0 0
        SHADOWCOLOR 218 218 218
        SHADOWSIZE 2 2
        TYPE BITMAP
        SIZE GIANT
        POSITION CC
        PARTIALS FALSE
        BUFFER 2
    END
END
END # Definición del primer layer termina

LAYER # Definición del segundo layer
NAME states_line
DATA cuador2
STATUS DEFAULT
TYPE LINE
PROJECTION
    "init=epsg:2163"
END
CLASSITEM "NOM_PROV"
CLASS
    EXPRESSION 'LOJA'
    COLOR 204 51 0
END
END # Definición del segundo layer termina
# Fin de la definición de Layers

END # Fin del objeto MAP

```

El archivo Mapfile ahora contiene un nuevo objeto, OUPFORMAT. Este objeto es definido con el objeto MAP y es usado en conjunto con la palabra clave IMAGETYPE. Dependiendo de la compilación de librerías que se realizó con MapServer, usted podrá escoger algunos formatos de salida para MapServer. La librería GD provee PNG (8 y 24 bits), JPG, y WBMP. La librería GDAL, una fuente de los formatos de MapServer, puede también proveer salidas para PNG, JPEG, TIFF/GeoTIFF, y otros formatos raster. La librería MING provee formatos de salida Shocwave/flash.

Por favor consulte la referencia del objeto OUTPUTFORMAT en: <http://mapserver.gis.umn.edu/doc40/mapfile-reference.html#outputformat>



## Ejemplo 2.7. Diferencia entre el modo MAP y el modo BROWSE



El mapa anteriormente creado fue realizado en modo “map”. Este es un mapa estático o simplemente una imagen generada con el CGI MapServer que es visualizada en el navegador.

El mapa que se presenta a continuación es generado en modo “browse”.





Ambos mapas usan la misma definición del archivo Mapfile. La diferencia es que el segundo mapa (el creado dinámicamente) es realizado en un formulario para interactividad. Si usted observa como esta página es enlazada con la página anterior, usted puede ver que no es el mismo que el ejemplo previo. Esta página es llamada en términos de MapServer, un template HTML.

Eche una Mirada al bloque “form” contenida en esta página:

```
<!--Inicio del FORM MAPSERVER -->
<form name="mapserv" method="GET" action="/cgi-bin/mapserv40">
  <!-- HIDDEN MAPSERVER CGI VARIABLES -->
  <input type="hidden" name="map" value="[map]">
  <input type="hidden" name="imgext" value="[mapext]">
  <input type="hidden" name="imgxy" value="199.5 149.5">
  <input type="hidden" name="zoom" value="1">
  <input type="hidden" name="mode" value="browse">

  <div align="center">
    <table border="1" cellpadding="0" cellspacing="0">
      <tr><td>
        <!-- THE El mapa interactivo es creado dinámicamente --
        <input type="image" name="img" src="[img]" width="400"
height="300">
      </td></tr>
    </table>
  </div>
</form>
```

Este bloque es ejecutado por el programa CGI MapServer (“/cgi-bin/mapserver40.exe”) cada vez que el usuario hace clic sobre el mapa. El mapa es actualizado por una entrada “input”, representado en la línea:

```
<input type="image" name="img" src="[img]" width="400" height="300">
```

Los ítems en los corchetes ([map], [maptext], y [img]) son conocidos como tags de MapServer- estas son variables CGI del MapServer y son reemplazadas por el programa CGI MapServer cuando este es recargado. El tag [map] es un lugar de directorio para la ruta del Mapfile y es reemplazado con “/curso/practica7/lojamap.map” cuando MapServer esta corriendo. El tag [maptext] es reemplazado con la extensión actual del mapa “476041.48 9438571.50 1186767.15 10161286.94”, y el tag [img] es reemplazado con el path de la imagen que crea el programa CGI MapServer, “/tmp/Ext1.9\_111565969721930.png”.

Las variables de cabecera “mode” con el valor “browse” dice al programa CGI que este necesita crear y colocar en el directorio “tmp” una imagen. Esta imagen es referenciada como [img] y es la que usted puede ver en el browser.



Ahora observemos el archivo Map file:

```
MAP
  NAME loja
  IMAGETYPE PNG
  EXTENT 476041.48 9438571.50 1186767.15 10161286.94
  SIZE 800 600
  SHAPEPATH "data"
  IMAGECOLOR 255 255 255
  UNITS METERS
  FONTSET "fonts/fonts.list"

  WEB
    TEMPLATE 'ejemplo2.7.html'
    IMAGEPATH 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap1/practica7/temp/'
    IMAGEURL 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap1/practica7/temp/'
    END

  PROJECTION
    "proj=laea"
    "ellps=clrk66"
    "lat_0=45"
    "lon_0=-100"
    END

  LAYER                                # definición del primer layer
    NAME states
    DATA cuador2
    STATUS DEFAULT
    TYPE POLYGON

    PROJECTION
      "init=epsg:2163"
    END

    CLASSITEM "Nom_prov"
    LABELITEM "Nom_prov"
    CLASS
      COLOR 232 232 232
      EXPRESSION ./
      LABEL
        COLOR 0 0 0
        SHADOWCOLOR 218 218 218
        SHADOWSIZE 2 2
        TYPE BITMAP
        SIZE GIANT
        POSITION CC
        PARTIALS FALSE
        BUFFER 2
      END
    END

  END                                # definición del primer layer termina
```



```
LAYER                                # Definición del segundo layer
  NAME states_line
  DATA cuador2
  STATUS DEFAULT
  TYPE LINE
  PROJECTION
    "init=epsg:2163"
  END
  CLASSITEM "NOM_PROV"
  CLASS
    EXPRESSION 'LOJA'
    COLOR 204 51 0
  END
END                                    # Definición del segundo layer termina

# Fin de la definición de Layers

END # Fin del objeto Map
```

Hay solo una cosa que se adhiere al archivo Mapfile.:

**TEMPLATE 'template.html'**

Este parámetro indica al MapServer que use la página “template.html” como el archivo template. MapServer puede procesar este archivo y reemplazar los tags que encuentre luego de enviar esta al Web browser. De esta manera se puede recuperar la información que se encuentra almacenada en las tablas relacionadas con cada una de las capas temáticas a las que se hace referencia. De esta manera es como trabajan las dos secciones siguientes.



### 3: Interface de Usuario and HTML Templates

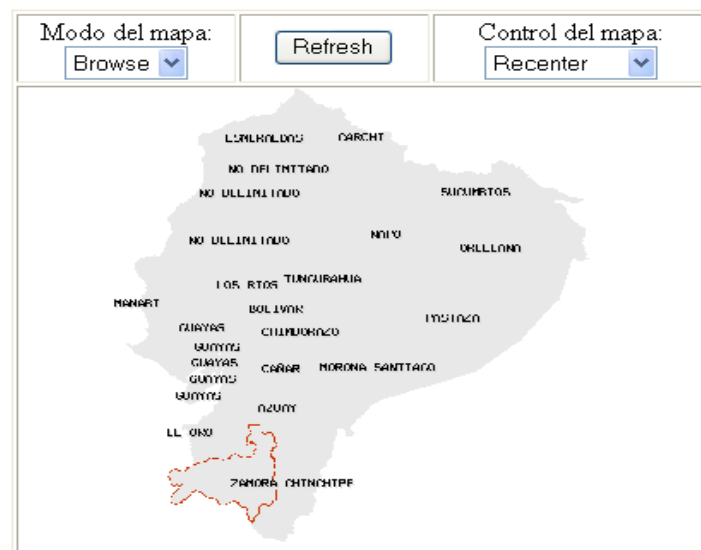
Hasta ahora solo se ha visto el archivo Mapfile al crear el mapa. En la creación de aplicaciones de mapas orientados al Web, la intención es hacer mapas que puedan ser cambiados interactivamente por el usuario (de la aplicación). Esto significa que se puede cambiar el contenido de (o la información en) el mapa. Para lograr esta interactividad, se usa los archivos templates HTML.

Un template HTML de MapServer es esencialmente un archivo HTML con unos tags específicos de MapServer. Los tags de MapServer son encerrados en corchetes []. Cuando el programa CGI MapServer procesa una aplicación, este primero analiza el archivo Mapfile y produce las salidas necesarias. Algunas de estas salidas podrían necesitar ser escritas en el archivo template HTML lo cual también puede estar especificado en el Mapfile (o en un archivo HTML de inicialización separado). El Programa CGI podría reemplazar todos los tags en el template HTML con los valores apropiados y luego enviar estos al Web browser. Si usted esta viendo directamente un template HTML en un Web browser, usted no verá ninguna imagen y podría ver una imagen en blanco u otra.

Así es como se introducirá a construir una interfaz interactiva para la aplicación.

#### Ejemplo 3.1: Zoom y controles Pan

Principio del formulario



Fin del formulario



Como con los mapas en la sección anterior, se inicializará la aplicación llamando a MapServer (“/cgi-bin/mapserv40.exe”) y enviando el path del Mapfile y otros parámetros (“map=/curso/practican/practican.map&mode=browse”).

Luego se usará el modo “browse” para instanciar “map”. El modo browse indica a MapServer que debe crear un mapa (una imagen) en nuestro directorio “/tmp/”. El path y el nombre de la imagen es referenciado por MapServer con la etiqueta “img”. Para cuando MapServer analice el archivo template HTML este podría reemplazar el tag con el path de la imagen correcta “/tmp/imagen.png”.

Estas son algunas de las cosas que usted puede hacer con el mapa. Primero, usted puede hacer clic en cualquier parte del mapa y el mapa se puede redibujar y tener centrado el punto donde se ha hecho clic, lo cual corresponde al panning, si usted da clic en el recuadro del campo “Map Control”, usted puede seleccionar entre los valores “zoom in” o “zoom out”. Si usted selecciona “Zoom In 2x”, y hace clic en cualquier parte del mapa, el mapa se redibujará, con zoom in en el punto que usted hizo clic centrado. Lo opuesto podría pasar si usted hace zoom out. Usted puede usar el botón de redibujado o refresh para actualizar el mapa, esto lo hace más sensible cuando se hace zoom in u out.

Trate de cambiar en el “MapMode” de “browse” a “map”. ¿Qué pasa cuando usted hace clic en refresh?. Recuerde que cuando usa “mode=map”. MapServer retorna a un mapa estático. Este ignora el template HTML y solo genera el mapa.

Puede mirar el Mapfile. Este será similar al Mapfile en la sección anterior y no es adherido ningún layer. Este no tiene ninguna cosa nueva para su explicación pero si usted se interesa en algunas palabras claves nuevas, por favor debe consultar la referencia de Mapfile.

Ahora, puede ver el archivo template HTML, usted notará que este es esencialmente un formulario que llama al programa CGI MapServer. Como este es un formulario, usted puede usar radio buttons y check boxes, también drop down boxes para implementar su interface.

### Ejemplo 3.1 Mapfile

```
MAP
  NAME loja
  IMAGETYPE PNG
  EXTENT 476041.48 9438571.50 1186767.15 10161286.94
  SIZE 400 300
  SHAPEPATH "data"
  IMAGECOLOR 255 255 255
  UNITS METERS
  FONTSET "fonts/fonts.list"

  WEB
    TEMPLATE 'ejemplo3.1.html'
```



**Universidad Técnica Particular de Loja**  
*La Universidad Católica de Loja*



**Sistemas de Información Geográfica**

```
IMAGEPATH 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap2/practical/temp/'
IMAGEURL 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap2/practical/temp/'
END
PROJECTION
  "proj=laea"
  "ellps=clrk66"
  "lat_0=45"
  "lon_0=-100"
END

LAYER # Definición del primer layer
NAME states
DATA cuador2
STATUS DEFAULT
TYPE POLYGON

PROJECTION
  "init=epsg:2163"
END

CLASSITEM "Nom_prov"
LABELITEM "Nom_prov"
CLASS
  COLOR 232 232 232
  EXPRESSION ./
  LABEL
    COLOR 0 0 0
    SHADOWCOLOR 218 218 218
    SHADOWSIZE 2 2
    TYPE BITMAP
    SIZE GIANT
    POSITION CC
    PARTIALS FALSE
    BUFFER 2
  END
END # Definición del primer layer termina

LAYER # Definición del segundo layer
NAME states_line
DATA cuador2
STATUS DEFAULT
TYPE LINE
PROJECTION
  "init=epsg:2163"
END
CLASSITEM "NOM_PROV"
CLASS
  EXPRESSION 'LOJA'
  COLOR 204 51 0
END # Definición del segundo layer termina
# Fin de la definición de Layers
END # Fin del objeto MAP
```



### Ejemplo 3.1 HTML Template

```

        <!--Inicio del archive Template HTML -->
<head>
<title>Tutorial MapServer 4.0</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-
1">
</head>

<body bgcolor="#FFFFFF" text="#000000">
<table width="601" align="center">
<tr>
    <td width="593">
        <h3 align="center">Ejemplo 3.1: Controles Pan y Zoom</h3>

        <!-- Inicio del Form de MapServer -->
        <form name="mapserv" method="GET" action="/cgi-bin/mapserv40.exe">
            <!-- MAPSERVER CGI VARIABLES -->
            <input type="hidden" name="map" value="[map]">
            <input type="hidden" name="imgext" value="[mapext]">
            <input type="hidden" name="imgxy" value="199.5 149.5">
            <table width="400" border="0" align="center">
                <tr>
                    <td>
                        <table width="400" border="1">
                            <tr>
                                <td>

                                    <!-- Especificación del modo del mapa -->

                                    <div align="center">Modo del mapa:<br>
                                        <select name="mode">
                                            <option value="browse">Browse</option>
                                            <option value="map">Map</option>
                                        </select>
                                    </div>
                                </td>
                                <td>
                                    <!-- Boton de submit -->
                                    <div align="center">
                                        <input type="submit" name="submit"
value="Refresh">
                                    </div>
                                </td>
                            </tr>
                        </table>
                    </td>
                </tr>
            </table>

            <!-- Controles ZOOM/PAN -->
            <div align="center">Control del mapa: <br>
            <select name="zoom">
                <option value="4" [zoom_4_select]>Zoom In
4x</option>
                <option value="3" [zoom_3_select]>Zoom In
3x</option>

```



```

                <option value="2" [zoom_2_select]>Zoom In
2x</option>
                <option value="1"
[zoom_1_select]>Recenter</option>
                <option value="-2" [zoom_-2_select]>Zoom Out
2x</option>
                <option value="-3" [zoom_-3_select]>Zoom Out
3x</option>
                <option value="-4" [zoom_-4_select]>Zoom Out
4x</option>
                </select>
            </div>
        </td>
    </tr>
    <tr>
        <td colspan="3" align="center" valign="top">
            <!-- Mostrar la imagen creada por el MapServer -->
            <input type="image" name="img" src="[img]" width="400"
height="300" border="0">
            </td>
        </tr>
    </table>
</td>
</tr>
</table>
</form>
<hr>
</td>
</tr>
</table>
</body>
</html>

```

La parte interesante de este código son los tags, ejemplo [zoom\_1\_selected]. Cuando MapServer muestra un tag como este, este podría evaluar el valor de ("1") y si este es seleccionado o no. Si este es seleccionado, MapServer podría marcar esta opción como "selected". De nuevo en el código del ejemplo "view", la opción "recenter" puede ser seleccionada (para cuando enviemos nuestro form otra vez, el valor de "zoom" podría ser 1).

El último tag es "[img]". Se puede notar que este tag es parte del objeto de entrada en el form. El tag "[img]" podría ser reemplazado con el path completo de la imagen generada por MapServer



## Ejemplo 3.2. Control de Layer

Principio del formulario

Selección de Layers: <input checked="" type="checkbox"/> Ríos <input type="checkbox"/> Carreteras <input checked="" type="checkbox"/> Ciudades		
Modo del mapa: <input type="text" value="Browse"/> ▾	<input type="button" value="Refresh"/>	Controles del mapa: <input type="text" value="Recenter"/> ▾

Fin del formulario

Regresando a mapas con layers y características estándares de aplicaciones de mapeo en Web. Estas son algunas maneras de lograr usar los objetos de los form. Usted puede usar el drop box/menú, check boxes, y/o radio buttons. En el ejemplo usted puede ver como implementar la selección de un layer usando check boxes y drop boxes.

Si usted revisa el Mapfile, usted podrá darse cuenta que el STATUS de el layer está cambiado a OFF a excepción de el polígono que sirve de background. El background esta siempre visible incluso cuando cualquier otro layer esta con STATUS en ON. El usuario de la aplicación puede tener control sobre cualquier layer para colocarlo a este en ON u OFF.

Usted puede entender como los layer pasan de ON a OFF en MapServer, si usted examina el código fuente del template HTML.



### Ejemplo 3.2 Mapfile



```
MAP
  NAME loja
  IMAGETYPE PNG
  EXTENT 541436.29 9472971.17 718110.56 9637083.74
  SIZE 400 300
  SHAPEPATH "data"
  IMAGECOLOR 255 255 255
  UNITS METERS
  FONTSET "fonts/fonts.list"

  WEB
    TEMPLATE 'ejemplo3.2.html'
    IMAGEPATH 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap2/practica2/temp/'
    IMAGEURL 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap2/practica2/temp/'
    END

  SYMBOL
    NAME 'circle'
    TYPE ELLIPSE
    POINTS 1 1 END
    FILLED TRUE
  END

  LAYER                                # Definición del primer layer Loja
    NAME loja
    DATA p_loja
    STATUS DEFAULT
    TYPE POLYGON

    CLASSITEM "Nom_prov"
    LABELITEM "Nom_prov"
    CLASS

    COLOR 232 232 232
    EXPRESSION ./
    LABEL
      COLOR 0 0 0
      SHADOWCOLOR 218 218 218
      SHADOWSIZE 2 2
      TYPE BITMAP
      SIZE GIANT
      POSITION CC
      PARTIALS FALSE
      BUFFER 2
    END
  END
END                                     # Definición del primer layer termina
```



```
LAYER                                     # Definición del Segundo layer ciudades
NAME ciudades
DATA ciudadesloja
STATUS OFF
TYPE POLYGON
CLASSITEM ciudad
CLASS
    NAME "Ciudades principales "
    EXPRESSION ./
    STYLE
        OUTLINECOLOR 251 7 13
        COLOR 251 7 13
    END
    TEMPLATE "templates/ciudad.html"
END
END                                     # Definición del segundo layer termina

LAYER                                     # Definición del tercer layer vias
NAME vias
DATA vias_t
STATUS OFF
TYPE LINE

CLASS
    NAME 'Ríos'
    STYLE
        COLOR 153 102 000
        SYMBOL 'circle'
        SIZE 1
    END
END
END                                     # Definición del tercer layer termina

LAYER                                     # Definición del cuarto Layer rios
NAME rios
DATA riOsfinales
STATUS OFF
TYPE LINE
CLASS
    NAME 'Ríos'
    STYLE
        COLOR 59 106 242
        SYMBOL 'circle'
        SIZE 1
    END
END
END                                     # Definición del cuarto Layer termina

# Fin de la definición de Layers

END # Fin del objeto MAP
```



### Ejemplo 3.2 Template HTML

```

<html>
<head>
<title>Tutorial MapServer 4.0</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

<body bgcolor="#FFFFFF" text="#000000">
<table width="600" align="center">
<tr>
  <td> <h3 align="center">Ejemplo 3.2: Control de Layers</h3>
  <!-- Inicio del form MapServer -->
  <form name="mapserv" method="GET" action="/cgi-bin/mapserv40">

      <!-- Variables ocultas de MapServer -->
      <input type="hidden" name="map" value="[map]">
      <input type="hidden" name="imgext" value="[mapext]">
      <input type="hidden" name="imgxy" value="199.5 149.5">
      <table width="600" border="0" align="center">
      <tr>
      <td>
        <table width="457" height="441" border="1" align="center">
          <tr>
            <td height="61" colspan="3">Selección de
Layers:<br>

              <!-- Especificamos layer vectors -->
              <input type="checkbox" name="layer" value="rios"
[rios_check]>
                Rios&nbsp;
              <input type="checkbox" name="layer" value="vias"
[vias_check]>
                Carreteras&nbsp;
              <input type="checkbox" name="layer" value="ciudades"
[ciudades_check]>
                Ciudades<br>
            </td>
          </tr>
          <tr>
            <td height="66">

              <!-- Especificar modo mapa -->
              <div align="center">Modo del mapa:<br>
                <select name="mode">
                  <option value="browse">Browse</option>
                  <option value="map">Map</option>
                </select>
              </div>
            </td>
          </tr>
        </table>
      </td>
    </tr>
  </table>

```



```

<!-- boton submit del form -->
<div align="center">
  <input type="submit" name="submit" value="Refresh">
</div>
</td>
<td>

<!-- controles ZOOM/PAN -->
<div align="center">Controles del mapa: <br>
  <select name="zoom">
    <option value="4" [zoom_4_select]>Zoom In 4x</option>
    <option value="3" [zoom_3_select]>Zoom In 3x</option>
    <option value="2" [zoom_2_select]>Zoom In 2x</option>
    <option value="1" [zoom_1_select]>Recenter</option>
    <option value="-2" [zoom_-2_select]>Zoom Out
2x</option>
    <option value="-3" [zoom_-3_select]>Zoom Out
3x</option>
    <option value="-4" [zoom_-4_select]>Zoom Out
4x</option>
  </select>
</div>
</td>
</tr>
<tr>

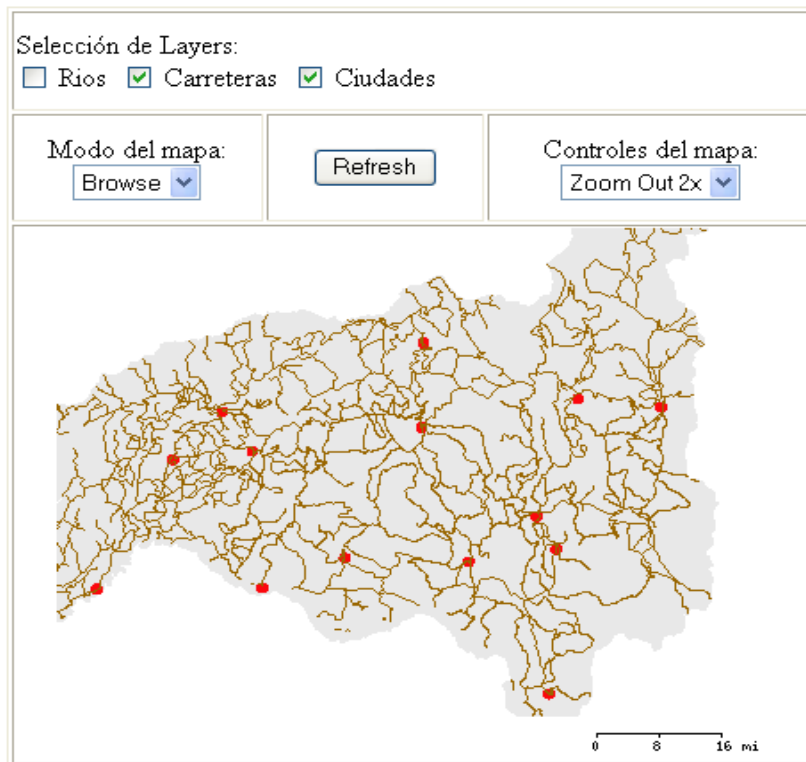
  <!-- Mostrar la imagen generada por el MapServer -->
  <td height="304" colspan="3" align="center" valign="top">
<input type="image" name="img" src="[img]" width="400" height="300"
border="0"></td>
</tr>
</table>
</td>
</tr>
</table>
</form>
</td></tr>
</table>
</body>
</html>

```



### Ejemplo 3.3. Adherir un Scalebar (barra de escala)

Principio del formulario



Fin del formulario

Como se puede ver en la imagen anterior, con la definición de una barra de escala (scalebar), se tendrá una referencia visual de la escala a la cual nos encontramos trabajando con el mapa georeferenciado visualizado en un web-browser. La barra de escala generada por el CGI MapServer varía automáticamente conforme el usuario interactúa con el mapa y aplica opciones como Zoom In u Out.

### Ejemplo 3.3 Mapfile

```
MAP
  NAME loja
  IMAGETYPE PNG
  EXTENT 541436.29 9472971.17 718110.56 9637083.74
  SIZE 400 300
  SHAPEPATH "data"
  IMAGECOLOR 255 255 255
  #TEMPLATEPATTERN "loja"
  UNITS METERS
  FONTSET "fonts/fonts.list"
```



```

WEB
  TEMPLATE 'ejemplo3.3.html'
  IMAGEPATH 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap2/practica3/temp/'
  IMAGEURL 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap2/practica3/temp/'
  END

SYMBOL
  NAME 'circle'
  TYPE ELLIPSE
  POINTS 1 1 END
  FILLED TRUE
  END

#
# Inicio de la barra de escala (scalebar)
#
SCALEBAR
  IMAGECOLOR 255 255 255
  LABEL
    COLOR 0 0 0
    SIZE TINY
  END
  STYLE 1
  SIZE 100 2
  COLOR 0 0 0
  UNITS MILES
  INTERVALS 2
  TRANSPARENT FALSE
  STATUS ON
  END

LAYER                                     # Definición del primer layer
  NAME loja
  DATA p_loja
  STATUS DEFAULT
  TYPE POLYGON

  CLASSITEM "Nom_prov"
  LABELITEM "Nom_prov"
  LABELMAXSCALE 500000
  CLASS
    COLOR 232 232 232
    EXPRESSION ./
    LABEL                                     # Definición del objeto LABEL
      COLOR 0 0 0
      SHADOWCOLOR 218 218 218
      SHADOWSIZE 2 2
      TYPE BITMAP
      SIZE GIANT
      POSITION CC
      PARTIALS FALSE
      BUFFER 2
    END
  END

```



```
END
END # Definición del primer layer termina

LAYER # Definición del Segundo layer
NAME ciudades
DATA ciudadesloja
STATUS OFF
TYPE POLYGON
CLASSITEM ciudad
CLASS
    NAME "Ciudades principales "
    EXPRESSION ./
    STYLE
        OUTLINECOLOR 251 7 13
        COLOR 251 7 13
    END
    TEMPLATE "templates/ciudad.html"
END
END # Definición del segundo layer termina

LAYER # Definición del tercer layer
NAME vias
DATA vias_t
STATUS OFF
TYPE LINE

CLASS
    NAME 'Ríos'
    STYLE
        COLOR 153 102 000
        SYMBOL 'circle'
        SIZE 1
    END
END
END # Definición del tercer layer

LAYER # Definición del cuarto layer
NAME rios
DATA riosfinales
STATUS OFF
TYPE LINE
CLASS
    NAME 'Ríos'
    STYLE
        COLOR 59 106 242
        SYMBOL 'circle'
        SIZE 1
    END
END
END # Definición del cuarto layer

# Fin de la definición de Layers

END # Fin del objeto Map
```





```

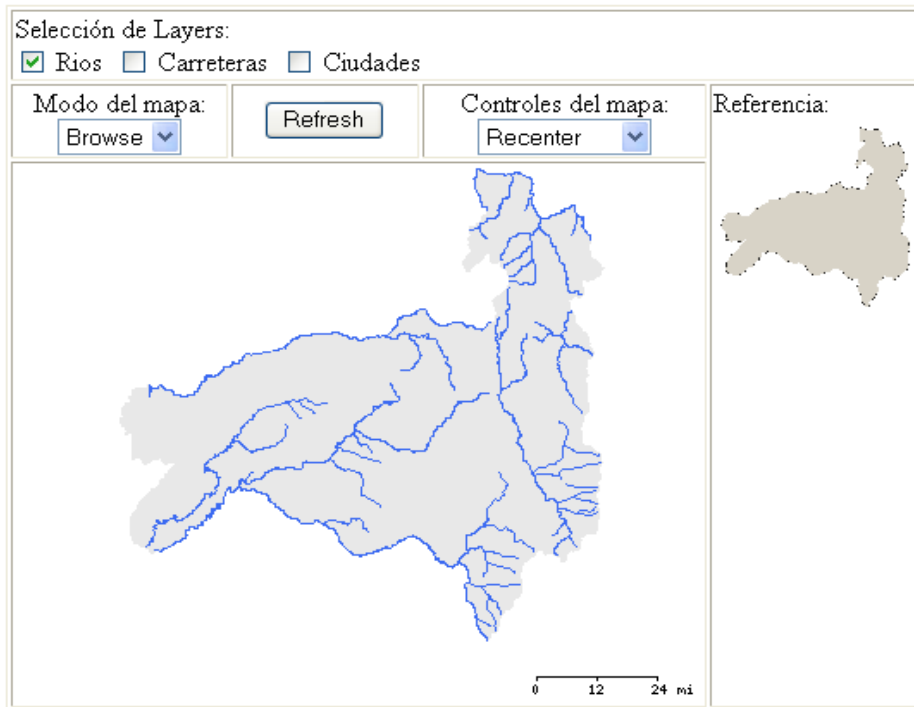
                <!-- boton submit del form -->
                <div align="center">
                    <input type="submit" name="submit" value="Refresh">
                </div>
            </td>
            <td>
                <!-- controles ZOOM/PAN -->
                <div align="center">Controles del mapa: <br>
                    <select name="zoom">
                        <option value="4" [zoom_4_select]>Zoom In 4x</option>
                        <option value="3" [zoom_3_select]>Zoom In 3x</option>
                        <option value="2" [zoom_2_select]>Zoom In 2x</option>
                        <option value="1" [zoom_1_select]>Recenter</option>
                        <option value="-2" [zoom_-2_select]>Zoom Out
2x</option>
                        <option value="-3" [zoom_-3_select]>Zoom Out
3x</option>
                        <option value="-4" [zoom_-4_select]>Zoom Out
4x</option>
                    </select>
                </div>
            </td>
        </tr>
        <tr>
            <td colspan="3" align="center" valign="top">
                <input type="image" name="img" src="[img]" width="400"
height="300" border="0">
                <table border="0" width="400" align="center">
                    <!-- Mostrar el SCALE BAR -->
                    <tr><td align="right"></td></tr>
                </table>
            </td>
        </tr>
    </table>
</td>
</tr>
</table>
</form>
</td></tr>
</table>
</body>
</html>

```



### Ejemplo 3.4. Adherir una Reference Map (mapa de referencia)

Principio del formulario



Fin del formulario

En el ejemplo anterior se adiciona un mapa de referencia (reference map), lo cual da mayor control a la aplicación, pues mediante esta imagen el usuario puede hacer clic sobre esta y situarse en la parte de interés del mapa.

### Ejemplo 3.4 Mapfile

```
MAP
  NAME loja
  IMAGETYPE PNG
  EXTENT 541436.29 9472971.17 718110.56 9637083.74
  SIZE 400 300
  SHAPEPATH "data"
  IMAGECOLOR 255 255 255
  #TEMPLATEPATTERN "loja"
  UNITS METERS
  FONTSET "fonts/fonts.list"

  WEB
    TEMPLATE 'ejemplo3.4.html'
    IMAGEPATH 'c:/Archivos de programa/Apache
  Group/Apache/htdocs/cap2/practica4/temp/'
```



**Universidad Técnica Particular de Loja**  
*La Universidad Católica de Loja*



**Sistemas de Información Geográfica**

```
IMAGEURL 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap2/practica4/temp/'
END
```

```
SYMBOL # Definición de un objeto SIMBOL
  NAME 'circle'
  TYPE ELLIPSE
  POINTS 1 1 END
  FILLED TRUE
END
```

```
SCALEBAR # Inicio de scalebar
  IMAGECOLOR 255 255 255
  LABEL
    COLOR 0 0 0
    SIZE TINY
  END
  STYLE 1
  SIZE 100 2
  COLOR 0 0 0
  UNITS MILES
  INTERVALS 2
  TRANSPARENT FALSE
  STATUS ON
END
```

```
REFERENCE # Inicio del mapa de referencia
  IMAGE graphics/reference.png
  EXTENT 550223.39 9463572.36 736295.50 9648226.34 #extensión en la
cual se presenta
  SIZE 120 120
  STATUS ON
  MINBOXSIZE 5
  MAXBOXSIZE 100
  COLOR 120 0 0
  OUTLINECOLOR 0 0 0
  MARKERSIZE 8
  MARKER 'star'
END
```

```
LAYER # Definición del primer layer inicia
  NAME loja
  DATA p_loja
  STATUS DEFAULT
  TYPE POLYGON

  CLASSITEM "Nom_prov"
  LABELITEM "Nom_prov"
  LABELMAXSCALE 500000
  CLASS

  COLOR 232 232 232
  EXPRESSION ./
  LABEL
    COLOR 0 0 0
```



Universidad Técnica Particular de Loja  
La Universidad Católica de Loja



Sistemas de Información Geográfica

```
SHADOWCOLOR 218 218 218
SHADOWSIZE 2 2
TYPE BITMAP
SIZE GIANT
POSITION CC
PARTIALS FALSE
BUFFER 2
END
END
END # Definición del primer layer termina

LAYER # Definición del Segundo layer
NAME ciudades
DATA ciudadesloja
STATUS OFF
TYPE POLYGON
CLASSITEM ciudad
CLASS
  NAME "Ciudades principales "
  EXPRESSION ./
  STYLE
    OUTLINECOLOR 251 7 13
    COLOR 251 7 13
  END
  TEMPLATE "templates/ciudad.html"
END
END # Definición del segundo layer termina

LAYER # Definición del tercer layer
NAME vias
DATA vias_t
STATUS OFF
TYPE LINE

CLASS
  NAME 'Ríos'
  STYLE
    COLOR 153 102 000
    SYMBOL 'circle'
    SIZE 1
  END
END
END # Definición del tercer layer termina

LAYER # Definición del cuarto layer inicia
NAME rios
DATA riosfinales
STATUS OFF
TYPE LINE
CLASS
  NAME 'Ríos'
  STYLE
    COLOR 59 106 242
    SYMBOL 'circle'
    SIZE 1
  END
END
```





```

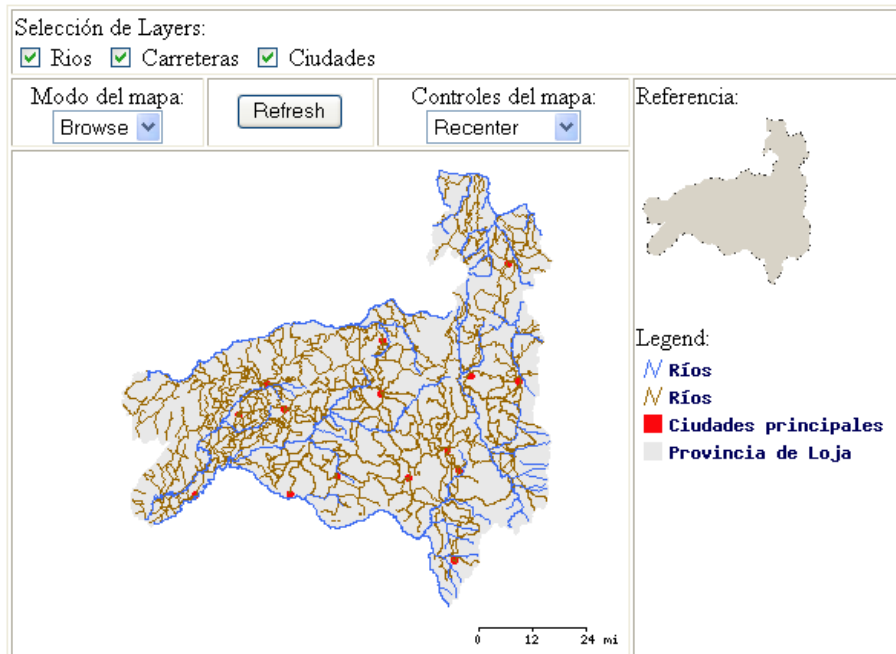
    </div>
  </td>
  <td>
    <!-- boton submit del form -->
    <div align="center">
      <input type="submit" name="submit" value="Refresh">
    </div>
  </td>
  <td>
    <!-- controles ZOOM/PAN -->
    <div align="center">Controles del mapa: <br>
    <select name="zoom">
      <option value="4" [zoom_4_select]>Zoom In 4x</option>
      <option value="3" [zoom_3_select]>Zoom In 3x</option>
      <option value="2" [zoom_2_select]>Zoom In 2x</option>
      <option value="1" [zoom_1_select]>Recenter</option>
      <option value="-2" [zoom_-2_select]>Zoom Out 2x</option>
      <option value="-3" [zoom_-3_select]>Zoom Out 3x</option>
      <option value="-4" [zoom_-4_select]>Zoom Out 4x</option>
    </select>
    </div>
  </td>
  <!-- Columna de referencia -->
  <td rowspan="2" valign="top">
    <p>Referencia:<br>
    <input type="image" name="ref" src="[ref]"
border="0"></p>
  </td>
</tr>
<tr>
  <!-- Mostrar la imagen creada por MapServer -->
  <td colspan="3" align="center" valign="top">
    <input type="image" name="img" src="[img]" width="400"
height="300" border="0">
    <table border="0" width="400" align="center">
      <!-- Mostrar el scalebar -->
      <tr><td align="right"></td></tr>
    </table>
  </td>
</tr>
</table>
</form>
</td></tr></table>
</body>
</html>

```



### Ejemplo 3.5. Adherir una Legend (Leyenda)

Principio del formulario



Fin del formulario

Para terminar esta sección se adiciona a la interfaz de usuario una leyenda, la cual indica gráficamente una descripción o significado de cada uno de los elementos dispuestos en el mapa.

### Ejemplo 3.5 Mapfile

```
MAP
  NAME loja
  IMAGETYPE PNG
  EXTENT 541436.29 9472971.17 718110.56 9637083.74
  SIZE 400 300
  SHAPEPATH "data"
  IMAGECOLOR 255 255 255
  #TEMPLATEPATTERN "loja"
  UNITS METERS
  FONTSET "fonts/fonts.list"

  WEB
    TEMPLATE 'ejemplo3.5.html'
    IMAGEPATH 'c:/Archivos de programa/Apache
  Group/Apache/htdocs/curso/cap2/practica5/temp/'
```



**Universidad Técnica Particular de Loja**  
*La Universidad Católica de Loja*



Sistemas de Información Geográfica

```
IMAGEURL 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap2/practica5/temp/'
END

SYMBOL
  NAME 'circle'
  TYPE ELLIPSE
  POINTS 1 1 END
  FILLED TRUE
END

# Inicio de scalebar
SCALEBAR
  IMAGECOLOR 255 255 255
  LABEL
    COLOR 0 0 0
    SIZE TINY
  END
  STYLE 1
  SIZE 100 2
  COLOR 0 0 0
  UNITS MILES
  INTERVALS 2
  TRANSPARENT FALSE
  STATUS ON
END

# Inicio de una leyenda
LEGEND
  KEYSIZE 12 12
  LABEL
    TYPE BITMAP
    SIZE MEDIUM
    COLOR 0 0 89
  END
  STATUS ON
END

# Inicio del mapa de referencia
REFERENCE
  IMAGE graphics/reference.png
  EXTENT 550223.39 9463572.36 736295.50 9648226.34 #extensión en la
cual se presenta
  SIZE 120 120
  STATUS ON
  MINBOXSIZE 5
  MAXBOXSIZE 100
  COLOR 120 0 0
  OUTLINECOLOR 0 0 0
  MARKERSIZE 8
  MARKER 'star'
END

LAYER # Definición del primer layer inicia
  NAME loja
  DATA p_loja
```



**Universidad Técnica Particular de Loja**  
*La Universidad Católica de Loja*



Sistemas de Información Geográfica

```
STATUS DEFAULT
TYPE POLYGON
CLASSITEM "Nom_prov"
LABELITEM "Nom_prov"
LABELMAXSCALE 500000
CLASS
  NAME 'Provincia de Loja'
  COLOR 232 232 232
  EXPRESSION ./
  LABEL
    COLOR 0 0 0
    SHADOWCOLOR 218 218 218
    SHADOWSIZE 2 2
    TYPE BITMAP
    SIZE GIANT
    POSITION CC
    PARTIALS FALSE
    BUFFER 2
  END
END
END # Definición del primer layer termina

LAYER # Definición del Segundo layer inicia
  NAME ciudades
  DATA ciudadesloja
  STATUS OFF
  TYPE POLYGON
  CLASSITEM ciudad
  CLASS
    NAME "Ciudades principales "
    EXPRESSION ./
    STYLE
      OUTLINECOLOR 251 7 13
      COLOR 251 7 13
    END
  TEMPLATE "templates/ciudad.html"
END
END # Definición del segundo layer termina

LAYER # Definición del tercer layer inicia
  NAME vias
  DATA vias_t
  STATUS OFF
  TYPE LINE
  CLASS
    NAME 'Ríos'
    STYLE
      COLOR 153 102 000
      SYMBOL 'circle'
      SIZE 1
    END
  END
END # Definición del tercerlayer termina

LAYER # Definición del cuarto layer inicia
  NAME rios
```





```

</tr>
<tr>
  <td>
    <!-- especificar modo de mapa -->
      <div align="center">Modo del mapa:<br>
      <select name="mode">
        <option value="browse" [browse_select]>Browse</option>
      </select>
      </div>
    </td>
    <!-- boton submit del form -->
      <div align="center">
      <input type="submit" name="submit" value="Refresh">
      </div>
    </td>
    <!-- controles ZOOM/PAN -->
      <div align="center">Controles del mapa: <br>
      <select name="zoom">
        <option value="4" [zoom_4_select]>Zoom In 4x</option>
        <option value="3" [zoom_3_select]>Zoom In 3x</option>
        <option value="2" [zoom_2_select]>Zoom In 2x</option>
        <option value="1" [zoom_1_select]>Recenter</option>
        <option value="-2" [zoom_-2_select]>Zoom Out 2x</option>
        <option value="-3" [zoom_-3_select]>Zoom Out 3x</option>
        <option value="-4" [zoom_-4_select]>Zoom Out 4x</option>
      </select>
      </div>
    </td>
    <!-- Columna de referencia -->
    <td rowspan="2" valign="top">
      <p>Referencia:<br>
      <input type="image" name="ref" src="[ref]"
border="0"></p>
      <p>Legend:<br>
      </p>
    </td>
  </tr>
  <!-- Mostrar la imagen creada por MapServer -->
  <td colspan="3" align="center" valign="top">
    <input type="image" name="img" src="[img]" width="400"
height="300" border="0">
    <table border="0" width="400" align="center">
      <!-- Mostrar el scalebar -->
      <tr><td align="right"></td></tr>
    </table>
  </td>
</tr>
</table>
</td>
</tr>
</table>
</form>
</td></tr></table>
</body>
</html>

```



## 4: Consultas y Templates HTML

En la sección tres, se usaron archivos template HTML para crear una interfaz de usuario para la aplicación Web. Se terminó con una aplicación donde los usuarios podían realizar acciones como: pan, zoom, y selección de layers. Lo que no podían hacer en esta sección es tener la capacidad de hacer consultas en la aplicación.

Los query templates son fragmentos de HTML que, cuando son puestos por Map Server, compaganan con el formulario de la página HTML.

### Ejemplo 4.1: Class-level Query Templates

Principio del formulario

The screenshot shows a web map interface with the following components:

- Seleccionar Layers:** Three checked checkboxes for "Ríos", "Vías", and "Ciudades".
- Modo del mapa:** A dropdown menu currently set to "Query Single Layer".
- Controles del mapa:** Includes a "Refresh" button and a "Recenter" dropdown menu.
- Referencia:** A small map outline of the province of Loja.
- Leyenda:** A legend with four items: "Ríos" (blue line), "Vías" (yellow line), "Ciudades principales" (red square), and "Provincia de Loja" (grey square).
- Map:** A main map area showing a network of rivers (blue), roads (yellow), and cities (red dots) within the province boundary (grey).

Fin del formulario

El mapa es similar a los anteriores, con la diferencia que en **Modo de mapa** se puede escoger la opción "Query single layer", lo cual nos da la funcionalidad de realizar consultas sobre el mapa. Como resultado de la consulta del objeto query se puede obtener lo siguiente, esto de acuerdo a la información contenida en la tabla de la base de datos correspondiente a la capa temática consultada:

1 CATAMAYO LOJA 3398301.000000



## Ejemplo 4.1 Mapfile

```
MAP
  NAME loja
  STATUS ON
  IMAGETYPE PNG
  EXTENT 541436.29 9472971.17 718110.56 9637083.74
  SIZE 400 300
  SHAPEPATH "data"
  IMAGECOLOR 255 255 255
  TEMPLATEPATTERN "loja"
  UNITS METERS
  FONTSET "fonts/fonts.list"

  WEB
    TEMPLATE 'ejemplo4.1.html'
    IMAGEPATH 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap3/practical/temp/'
    IMAGEURL 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap3/practical/temp/'
    MINSCALE 1000
    MAXSCALE 15000000

  END

SYMBOL
  NAME 'circle'
  TYPE ELLIPSE
  POINTS 1 1 END
  FILLED TRUE
END

# Inicio de scalebar
SCALEBAR
  IMAGECOLOR 255 255 255
  LABEL
    COLOR 0 0 0
    SIZE TINY
  END
  STYLE 1
  SIZE 100 2
  COLOR 0 0 0
  UNITS MILES
  INTERVALS 2
  TRANSPARENT FALSE
  STATUS ON
END

# Inicio de una leyenda
LEGEND
  KEYSIZE 12 12
  LABEL
    TYPE BITMAP
    SIZE MEDIUM
```



Universidad Técnica Particular de Loja  
La Universidad Católica de Loja



Sistemas de Información Geográfica

```
COLOR 0 0 89
END
STATUS ON
END

# Inicio del mapa de referencia
REFERENCE
IMAGE graphics/reference.png
EXTENT 550223.39 9463572.36 736295.50 9648226.34 #extensión en la
cual se presenta
SIZE 120 120
STATUS ON
MINBOXSIZE 5
MAXBOXSIZE 100
COLOR 120 0 0
OUTLINECOLOR 0 0 0
MARKERSIZE 8
MARKER 'star'
END

LAYER                                # Definición del primer layer
NAME loja
DATA p_loja
STATUS DEFAULT
TYPE POLYGON

CLASSITEM "Nom_prov"
LABELITEM "Nom_prov"
LABELMAXSCALE 500000
CLASS
NAME 'Provincia de Loja'
COLOR 232 232 232
EXPRESSION ./
LABEL
COLOR 0 0 0
SHADOWCOLOR 218 218 218
SHADOWSIZE 2 2
TYPE BITMAP
SIZE GIANT
POSITION CC
PARTIALS FALSE
BUFFER 2
END
END

END                                # Definición del primer layer termina

LAYER                                # Definición del Segundo layer
NAME ciudades
DATA ciudadesloja
STATUS OFF
TYPE POLYGON
CLASSITEM "ciudad"
CLASS
NAME "Ciudades principales "
TEMPLATE 'templates/ciudades_query.html'
EXPRESSION ./
```



Universidad Técnica Particular de Loja  
La Universidad Católica de Loja



Sistemas de Información Geográfica

```
STYLE
    OUTLINECOLOR 251 7 13
    COLOR 251 7 13
END
END
END # Definición del segundo layer termina

LAYER # Definición del tercer layer
NAME vias
DATA vias_t
STATUS OFF
TYPE LINE
CLASSITEM "VIAS_"
CLASS
    NAME 'Vias'
    TEMPLATE 'templates/vias_query.html'
    STYLE
        COLOR 153 102 000
        SYMBOL 'circle'
        SIZE 1
    END
END
END # Dedfinición del tercer layer termina

LAYER # Definición del cuarto layer
NAME rios
DATA riosfinales
STATUS OFF
TYPE LINE
CLASSITEM "nombres"
CLASS
    NAME 'Ríos'
    TEMPLATE 'templates/rios_query.html'
    STYLE
        COLOR 59 106 242
        SYMBOL 'circle'
        SIZE 1
    END
END
END # Definición del cuarto layer termina

# Fin de la definición de Layers

END # Fin del objeto MAP
```



## Ejemplo 4.1 Template HTML

```

<html>
<head>
<title>Tutorial MapServer 4.0</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-
1">
</head>

<body bgcolor="#FFFFFF" text="#000000">
<h3 align="center">Ejemplo 4.1: Class-level Query Templates</h3>

        <!-- INICIO DEL FORM DE MAPSERVER -->
<form name="mapserv" method="GET" action="/cgi-bin/mapserv40">

<table width="600" border="0" align="center">
  <tr>
    <td>
      <table width="600" border="1">
        <tr>
          <td colspan="4">Seleccionar Layers:<br>

            <!-- ESPECIFICACIÓN DE LAYER RASTER -->
            <input type="checkbox" name="layer" value="rios"
[rios_check]>
            Ríos&nbsp;
            <input type="checkbox" name="layer" value="vias"
[vias_check]>
            Vías&nbsp;
            <input type="checkbox" name="layer" value="ciudades"
[ciudades_check]>
            Ciudades&nbsp;
          </td>
        </tr>
        <tr>
          <td>

            <!-- ESPECIFICACIÓN DEL MODO DEL MAPA -->
            <div align="center">Modo del mapa:<br>
            <select name="mode">
              <option value="browse" [browse_select]>Browse </option>
              <option value="query" [query_select]>Query Single
Layer</option>
              <option value="nquery" [nquery_select]>Query Multiple
Layers</option>
            </select>
            </div>
          </td>
        </tr>
        <tr>
          <td>

            <!-- BOTON SUBMIT DEL FORM -->
            <div align="center">
            <input type="submit" name="submit" value="Refresh">
            </div>
          </td>
        </tr>
      </table>
    </td>
  </tr>
</table>

```



<td>

**<!-- CONTROLES ZOOM/PAN -->**

```

    <div align="center">Controles del mapa: <br>
    <select name="zoom">
      <option value="4" [zoom_4_select]>Zoom In 4x</option>
      <option value="3" [zoom_3_select]>Zoom In 3x</option>
      <option value="2" [zoom_2_select]>Zoom In 2x</option>
      <option value="1" [zoom_1_select]>Recenter</option>
      <option value="-2" [zoom_-2_select]>Zoom Out 2x</option>
      <option value="-3" [zoom_-3_select]>Zoom Out 3x</option>
      <option value="-4" [zoom_-4_select]>Zoom Out 4x</option>
    </select>
    </div>
  </td>

```

**<!-- COLUMNA DE REFERENCIA Y LEYENDA -->**

```

  <td rowspan="2" valign="top">
    <p>Referencia:<br>
    </p>
    <p>Leyenda:<br>
     </p>
  </td>
</tr>
<tr>

```

**<!-- MOSTRAR LA IMAGEN GENERADA POR MAPSERVER -->**

```

  <td colspan="3" align="center" valign="top">
    <input type="image" name="img" src="[img]"
    width="[mapwidth]" height="[mapheight]" border="0" alt="This is the
    map">
  </td>
</tr>
</table>
</td>
</tr>
</table>

```

**<!-- VARIABLES CGI OCULTAS -->**

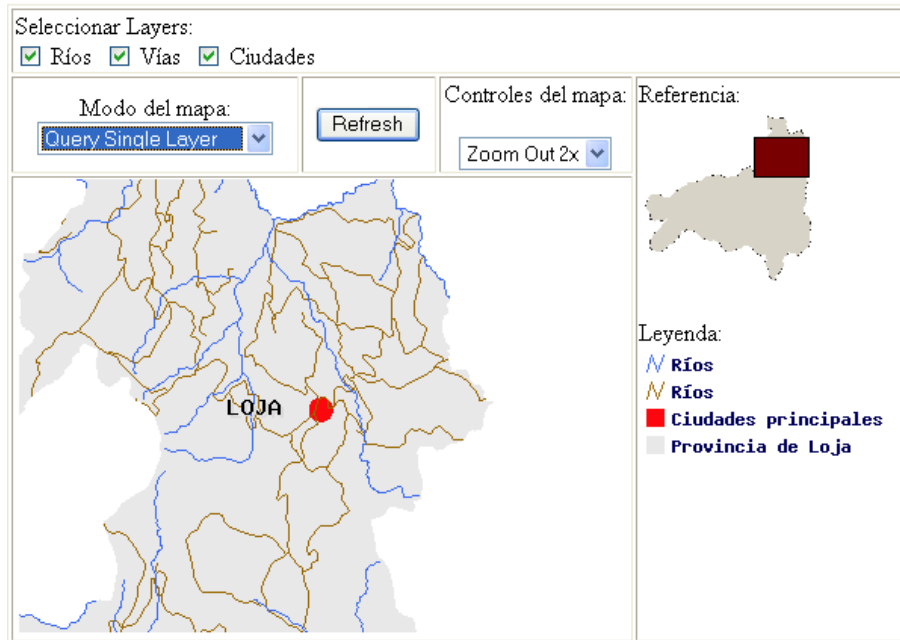
```

<input type="hidden" name="imgxy" value="[center]">
<input type="hidden" name="imgext" value="[mapext]">
<input type="hidden" name="map" value="[map]">
<input type="hidden" name="root" value="[root]">
<input type="hidden" name="savequery" value="true">
</form>
</body>
</html>

```



## Ejemplo 4.2. Layer – Header y Footer Template



Este es el resultado de la aplicación a la cual se le ha aumentado los respectivos header y footer templates, y como resultado de la consulta se llega a obtener parámetros como los siguientes:

Layer: CIUDAD

	CIUDAD	PROVINCIA	ÁREA
1	SARAGURO	LOJA	3398301.000000

## Ejemplo 4.2 Mapfile

```
MAP
  NAME loja
  STATUS ON
  IMAGETYPE PNG
  EXTENT 541436.29 9472971.17 718110.56 9637083.74
  SIZE 400 300
  SHAPEPATH "data"
  IMAGECOLOR 255 255 255
  TEMPLATEPATTERN "loja"
  UNITS METERS
  FONTSET "fonts/fonts.list"
```



**Universidad Técnica Particular de Loja**  
*La Universidad Católica de Loja*



Sistemas de Información Geográfica

```
WEB
  TEMPLATE 'ejemplo4.2.html'
  IMAGEPATH 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap3/practica2/temp/'
  IMAGEURL 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap3/practica2/temp/'
  MINSCALE 1000
  MAXSCALE 15000000

END

SYMBOL
  NAME 'circle'
  TYPE ELLIPSE
  POINTS 1 1 END
  FILLED TRUE
END

# Inicio de scalebar
SCALEBAR
  IMAGECOLOR 255 255 255
  LABEL
    COLOR 0 0 0
    SIZE TINY
  END
  STYLE 1
  SIZE 100 2
  COLOR 0 0 0
  UNITS MILES
  INTERVALS 2
  TRANSPARENT FALSE
  STATUS ON
END

# Inicio de una leyenda
LEGEND
  KEYSIZE 12 12
  LABEL
    TYPE BITMAP
    SIZE MEDIUM
    COLOR 0 0 89
  END
  STATUS ON
END

# Inicio del mapa de referencia
REFERENCE
  IMAGE graphics/reference.png
  EXTENT 550223.39 9463572.36 736295.50 9648226.34 #extensión en la
cual se presenta
  SIZE 120 120
  STATUS ON
  MINBOXSIZE 5
  MAXBOXSIZE 100
  COLOR 120 0 0
  OUTLINECOLOR 0 0 0
```



**Universidad Técnica Particular de Loja**  
*La Universidad Católica de Loja*



**Sistemas de Información Geográfica**

```

MARKERSIZE 8
MARKER 'star'
END

LAYER                                # Definición del primer layer
NAME loja
DATA p_loja
STATUS DEFAULT
TYPE POLYGON

CLASSITEM "Nom_prov"
LABELITEM "Nom_prov"
LABELMAXSCALE 500000
CLASS
NAME 'Provincia de Loja'
COLOR 232 232 232
EXPRESSION ./
LABEL
COLOR 0 0 0
SHADOWCOLOR 218 218 218
SHADOWSIZE 2 2
TYPE BITMAP
SIZE GIANT
POSITION CC
PARTIALS FALSE
BUFFER 2
END
END
END                                # Definición del primer layer termina

LAYER                                # Definición del Segundo layer
NAME ciudades
DATA ciudadesloja
STATUS OFF
TYPE POLYGON
CLASSITEM ciudad
HEADER 'templates/ciudades_header.html'
FOOTER 'templates/ciudades_footer.html'
CLASS
NAME "Ciudades principales "
TEMPLATE 'templates/ciudades_query.html'
EXPRESSION ./
STYLE
OUTLINECOLOR 251 7 13
COLOR 251 7 13
END
END                                # Definición del segundo layer termina

LAYER                                # Definición del tercer layer
NAME vias
DATA vias_t
STATUS OFF
TYPE LINE
CLASS

```



```

NAME 'Ríos'
TEMPLATE 'templates/vias_query.html'
STYLE
  COLOR 153 102 000
  SYMBOL 'circle'
  SIZE 1
END
END
END # Definición del tercer layert termina

LAYER # Definición del cuarto layer
NAME rios
DATA riosfinales
STATUS OFF
TYPE LINE
CLASS
  NAME 'Ríos'
  TEMPLATE 'templates/rios_query.html'
  STYLE
    COLOR 59 106 242
    SYMBOL 'circle'
    SIZE 1
  END
END
END # Definición del cuarto layer termina

# Fin de la definición de Layers

END # Fin del objeto MAP

```

## Ejemplo 4.2 Template HTML

```

<html>
<head>
<title>Tutorial MapServer 4.0</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

<body bgcolor="#FFFFFF" text="#000000">
<h3 align="center">Ejemplo 4.2: Layer - especific header y footer
template</h3>

      <!-- INICIO DEL FORM DE MAPSERVER -->
<form name="mapserv" method="GET" action="/cgi-bin/mapserv40">

<table width="600" border="0" align="center">
  <tr>
    <td>
      <table width="600" border="1">
        <tr>
          <td colspan="4">Seleccionar Layers:<br>

      <!-- ESPECIFICACIÓN DE LAYER RASTER -->

```



```

        <input type="checkbox" name="layer" value="rios"
[rios_check]>
        Ríos&nbsp;
        <input type="checkbox" name="layer" value="vias"
[vias_check]>
        Vías&nbsp;
        <input type="checkbox" name="layer" value="ciudades"
[ciudades_check]>
        Ciudades&nbsp;
    </td>
</tr>
<tr>
    <td>

        <!-- ESPECIFICACIÓN DEL MODO DEL MAPA -->
        <div align="center">Modo del mapa:<br>
        <select name="mode">
            <option value="browse" [browse_select]>Browse </option>
            <option value="query" [query_select]>Query Single
Layer</option>
            <option value="nquery" [nquery_select]>Query Multiple
Layers</option>
        </select>
        </div>
    </td>
    <td>

        <!-- BOTON SUBMIT DEL FORM -->
        <div align="center">
            <input type="submit" name="submit" value="Refresh">
        </div>
    </td>
    <td>

        <!-- CONTROLES ZOOM/PAN -->
        <div align="center">Controles del mapa: <br>
        <select name="zoom">
            <option value="4" [zoom_4_select]>Zoom In 4x</option>
            <option value="3" [zoom_3_select]>Zoom In 3x</option>
            <option value="2" [zoom_2_select]>Zoom In 2x</option>
            <option value="1" [zoom_1_select]>Recenter</option>
            <option value="-2" [zoom_-2_select]>Zoom Out 2x</option>
            <option value="-3" [zoom_-3_select]>Zoom Out 3x</option>
            <option value="-4" [zoom_-4_select]>Zoom Out 4x</option>
        </select>
        </div>
    </td>

        <!-- COLUMNA DE REFERENCIA Y LEYENDA -->
    <td rowspan="2" valign="top">
        <p>Referencia:<br>
        </p>
        <p>Leyenda:<br>
         </p>
    </td>
</tr>

```



Universidad Técnica Particular de Loja  
La Universidad Católica de Loja



Sistemas de Información Geográfica

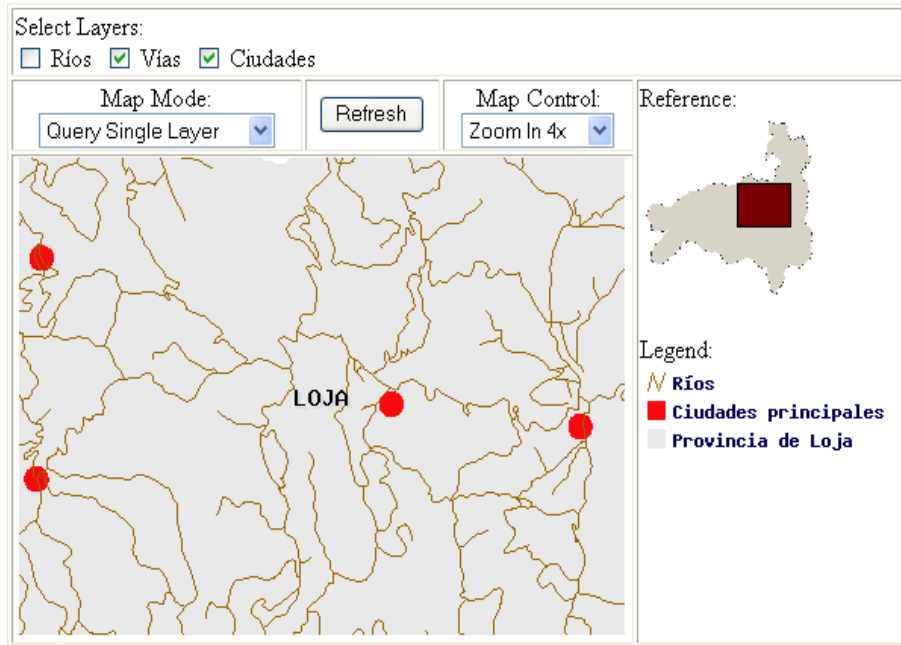
```
<tr>

    <!-- MOSTRAR LA IMAGEN GENERADA POR MAPSERVER -->
    <td colspan="3" align="center" valign="top">
        <input type="image" name="img" src="[img]"
width="[mapwidth]" height="[mapheight]" border="0" alt="This is the
map">
        </td>
    </tr>
</table>
</td>
</tr>
</table>

    <!-- VARIABLES CGI OCULTAS -->
<input type="hidden" name="imgxy" value="[center]">
<input type="hidden" name="imgext" value="[mapext]">
<input type="hidden" name="map" value="[map]">
<input type="hidden" name="root" value="[root]">
<input type="hidden" name="savequery" value="true">
</form>
</body>
</html>
```



### Ejemplo 4.3. Mapas con Header y Footer Template



Como resultado de una consulta realizada en el mapa se obtiene lo siguiente:

```

Search Window: 648965.984592 9538342.677050 703801.869142 9579370.819550
Query Point: 244, 155 (Image coordinates)
               682415.874167, 9558172.945925 (Map coordinates)
Query Metrics: Number of layers with results=1
               Total number of results=1
    
```

Layer: CIUDAD

	CIUDAD	PROVINCIA	ÁREA
1	CATAMAYO	LOJA	3398301.000000

A continuación se muestra el resultado de la programación del archivo Mapfile, en el archivo template HTML no se han realizado cambios, por lo cual no será necesario volverlo a estudiar por el momento.



## Ejemplo 4.3 Mapfile

```
MAP
  NAME loja
  STATUS ON
  IMAGETYPE PNG
  EXTENT 541436.29 9472971.17 718110.56 9637083.74
  SIZE 400 300
  SHAPEPATH "data"
  IMAGECOLOR 255 255 255
  TEMPLATEPATTERN "loja"
  UNITS METERS
  FONTSET "fonts/fonts.list"

  WEB
    HEADER 'templates/header.html' # Especificación de archivo header
    TEMPLATE 'ejemplo4.3.html'
    FOOTER 'templates/footer.html'
    IMAGEPATH 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap3/practica3/temp/'
    IMAGEURL 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap3/practica3/temp/'
    MINSCALE 1000
    MAXSCALE 15000000
  END
```

# El parámetro HEADER y su configuración es la parte importante de este ejemplo, puede abrir el archivo que se incluye en el ejemplo y ver su configuración.

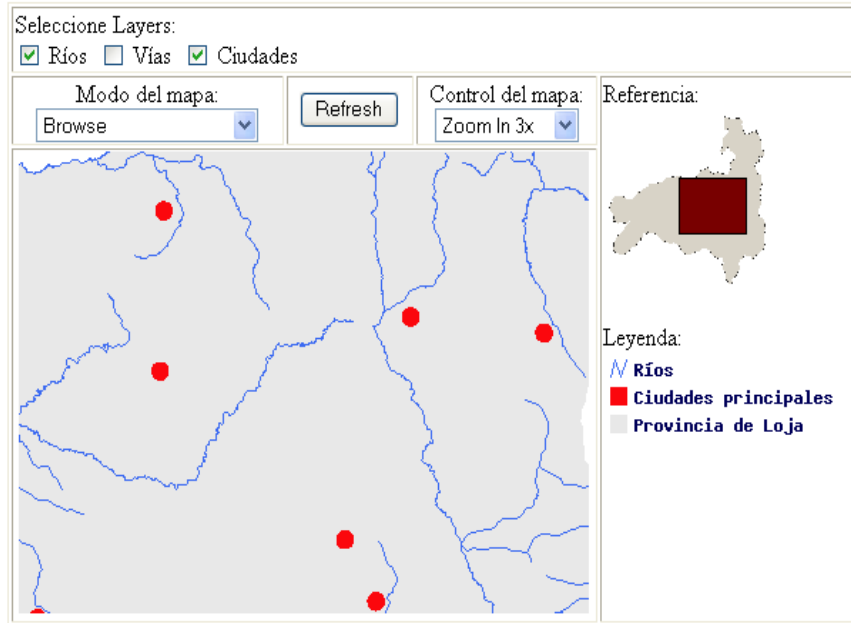
```
LAYER # States polygon layer begins here
  NAME ciudades
  DATA ciudadesloja
  STATUS OFF
  TYPE POLYGON
  CLASSITEM ciudad
  HEADER 'templates/ciudades_header.html'
  FOOTER 'templates/ciudades_footer.html'
  CLASS
    NAME "Ciudades principales "
    TEMPLATE 'templates/ciudades_query.html'
    EXPRESSION ./
    STYLE
      OUTLINECOLOR 251 7 13
      COLOR 251 7 13
    END
  END
END # Fin de la definición de LAYER

# Fin de la definición de Layers

END # Fin del objeto Map
```



## Ejemplo 4.4. El Query Map



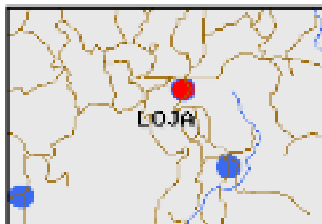
Como resultado del ejemplo veremos el resultado de la consulta en pantalla, los detalles de la visualización del archivo se los verán a continuación.

```
Ventana de búsqueda: 629110.623234 9518765.929125 686326.565509 9538300.000375
Posición de consulta: 217, 107 (Coordenadas de Imagen)
673984.856918, 9531983.314963 (Coordenadas de Mapa)
Métricas de consulta: Número de Layers con resultados=1
Número total de resultados=1
```

Layer: CIUDAD

CIUDAD	PROVINCIA	ÁREA
1 GONZANAMA	LOJA	3598301.000000

Ejemplo de Query Map



Querymap standard



## Ejemplo 4.4 Mapfile

```
MAP
NAME loja
STATUS ON
IMAGETYPE PNG24
EXTENT 541436.29 9472971.17 718110.56 9637083.74
SIZE 400 300
SHAPEPATH "data"
IMAGECOLOR 255 255 255
TEMPLATEPATTERN "loja"
UNITS METERS
FONTSET "fonts/fonts.list"

WEB
HEADER 'templates/header.html'
TEMPLATE 'ejemplo4.4.html'
FOOTER 'templates/footer_querymap.html'
IMAGEPATH 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap3/practica4/temp/'
IMAGEURL 'c:/Archivos de programa/Apache
Group/Apache/htdocs/curso/cap3/practica4/temp/'
MINSCALE 1000
MAXSCALE 15000000

END

SYMBOL
NAME 'circle'
TYPE ELLIPSE
POINTS 1 1 END
FILLED TRUE
END

# Inicio de scalebar
SCALEBAR
IMAGECOLOR 255 255 255
LABEL
COLOR 0 0 0
SIZE TINY
END
STYLE 1
SIZE 100 2
COLOR 0 0 0
UNITS MILES
INTERVALS 2
TRANSPARENT FALSE
STATUS ON
END

# inicio de una legend
LEGEND
KEYSIZE 12 12
LABEL
```



**Universidad Técnica Particular de Loja**  
*La Universidad Católica de Loja*



Sistemas de Información Geográfica

```
TYPE BITMAP
SIZE MEDIUM
COLOR 0 0 89
END
STATUS ON
END

# Inicio del mapa de referencia
REFERENCE
IMAGE graphics/reference.png
EXTENT 550223.39 9463572.36 736295.50 9648226.34 #extensión en la
cual se presenta
SIZE 120 120
STATUS ON
MINBOXSIZE 5
MAXBOXSIZE 100
COLOR 120 0 0
OUTLINECOLOR 0 0 0
MARKERSIZE 8
MARKER 'star'
END
LAYER                                # Definición del primer layer
NAME loja
DATA p_loja
STATUS DEFAULT
TYPE POLYGON

CLASSITEM "Nom_prov"
LABELITEM "Nom_prov"
LABELMAXSCALE 500000
CLASS
NAME 'Provincia de Loja'
COLOR 232 232 232
EXPRESSION ./
LABEL
COLOR 0 0 0
SHADOWCOLOR 218 218 218
SHADOWSIZE 2 2
TYPE BITMAP
SIZE GIANT
POSITION CC
PARTIALS FALSE
BUFFER 2
END
END
END                                # Definición del primer layer termina

LAYER                                # Definición del Segundo Layer
NAME ciudades
DATA ciudadesloja
STATUS OFF
TYPE POLYGON
CLASSITEM ciudad
HEADER 'templates/ciudades_header.html'
FOOTER 'templates/ciudades_footer.html'
CLASS
```



```
        NAME "Ciudades principales "
        TEMPLATE 'templates/ciudades_query.html'
        EXPRESSION ./
        STYLE
            OUTLINECOLOR 251 7 13
            COLOR 251 7 13
        END
    END
END # Definición del segundo layer finaliza

LAYER # Definición del tercer layer inicia
    NAME vias
    DATA vias_t
    STATUS OFF
    TYPE LINE

    CLASS
        NAME 'Ríos'
        TEMPLATE 'templates/vias_query.html'
        STYLE
            COLOR 153 102 000
            SYMBOL 'circle'
            SIZE 1
        END
    END
END # Definición del tercer layer termina

LAYER # Definición del cuarto layer
    NAME rios
    DATA riosfinales
    STATUS OFF
    TYPE LINE
    CLASS
        NAME 'Ríos'
        TEMPLATE 'templates/rios_query.html'
        STYLE
            COLOR 59 106 242
            SYMBOL 'circle'
            SIZE 1
        END
    END
END # Definición del cuarto layer finaliza

# Fin de la definición de Layers
END # Fin del objeto MAP
```



Universidad Técnica Particular de Loja  
*La Universidad Católica de Loja*



Sistemas de Información Geográfica

## Bibliografía:

- <http://serverwatch.internet.com/> Binarios instalación Servidor Apache 1.3.24.
- <http://mapserver.gis.umn.edu/dload.html> Dirección Descarga de binarios MapServer 4.0
- <http://terrasip.gis.umn.edu/projects/tutorial/> Tutorial MapServer 4.0
- <http://utpl.edu.ec/sig> Equipo de Sistemas de Información Geográfica. – UPSI – Universidad Técnica Particular de Loja.