

**Universidad del Azuay**

**Facultad de Administración de Empresas**

**Escuela de Ingeniería de Sistemas**

**“Publicación de Cartografía Digital en la Web utilizando un  
Servidor de Mapas de Código Abierto”**

**Trabajo de Tesis previo a la obtención del título de  
Ingenieras de Sistemas**

**Autoras:**

**Ginna Antonieta Araujo Pacheco  
Jheimy Lorena Pacheco Niveló**

**Director: Ing. Pablo Esquivel**

**Cuenca, Ecuador**

**2006**

A mi familia por su guía, apoyo y comprensión a lo largo de mi vida. A mami que con su amor me enseñó perseverancia y valor. A Darwin por su cariño y ejemplo incomparables.

Ginna

A mis padres quienes son el mejor ejemplo de tenacidad,  
esfuerzo, superación y constancia que cualquier hijo pudiera tener.  
A mis hermanos por el cariño y apoyo incondicional;  
a Mateo y Sebastián la ternura viva.

Jheimy

## **Agradecimiento**

Dejamos constancia de nuestra gratitud al Ingeniero Pablo Esquivel por su colaboración en el desarrollo del proyecto; a los Ingenieros Omar Delgado y Paúl Ochoa por el apoyo y experiencia brindados. Agradecemos también a nuestros amigos por compartir con nosotros sus conocimientos.

Las autoras nos responsabilizamos por las ideas y el contenido vertidos en la presente tesis.

.....  
Ginna Antonieta Araujo Pacheco  
Nivelo

.....  
Jheimy Lorena Pacheco

## Índice de Contenidos

Dedicatoria .....	ii
Agradecimiento .....	iv
Hoja de Responsabilidad .....	v
Índice de Contenidos .....	vi
Resumen .....	ix
Abstract .....	x
Introducción .....	1
Capítulo 1: Introducción a los Conceptos Básicos relacionados con Sistemas de Información Geográfico y Servidores de Mapas .....	2
Conceptos de sistemas de información geográfico y servidores de mapas .....	2
Conceptos básicos relacionados con sistemas de información geográfica .....	4
Tipos de datos geográficos .....	4
Datos gráficos .....	4
Datos alfanuméricos .....	4
Modelos de datos .....	5
El modelo raster .....	5
El modelo vectorial .....	6
Escala de medida .....	7
Georeferenciación o geocodificación de datos geográficos .....	8
Sistema de proyección .....	8
Proyección cilíndrica .....	9
Proyección cónica .....	11
Proyección azimutal o plana .....	13
Sistema de proyección utilizado en el proyecto .....	16
Sistema de coordenadas .....	17
Metadatos .....	18
Principales usos de los metadatos .....	18
Diferentes estándares .....	18
Estructura de los metadatos .....	19
Conceptos de Servidores de Mapas .....	20
Contexto y fundamento .....	20
Servicios de catálogo .....	21
Interoperatividad de los servicios .....	22
Servidores de mapas .....	24
Cartografía en la Red (Web Mapping) .....	26
Conclusiones .....	28
Capítulo II: Evaluación de Diferentes Herramientas para la Publicación de Mapas .....	29

ALOV Map .....	29
Implementación del sistema ALOV Map .....	31
GeoServNet .....	32
Implementación del sistema GeoServNet .....	34
MapServer .....	37
Implementación del sistema MapServer .....	40
Conclusiones .....	42
 Capítulo III: Configuración, Compilación e Instalación de MapServer .....	 43
Componentes de una aplicación con MapServer .....	43
El Archivo de Inicialización .....	43
El mapfile .....	44
El archivo plantilla .....	44
El set de datos SIG .....	44
Descripción de las librerías utilizadas por MapServer .....	45
Librerías necesarias para una instalación básica de MapServer .....	45
Librerías adicionales .....	48
Librerías utilizadas por el instalador FGS .....	50
Instalación de MapServer .....	53
Instalación de MapServer-4.2.3 .....	53
Instalación de MapServer 4.6.2 con el paquete FGS .....	62
Conclusiones .....	62
 Capítulo IV: Desarrollo de la Aplicación Modelo .....	 63
Estructura de los componentes de una aplicación básica con MapServer .....	64
Conceptos del archivo mapfile .....	65
Sintaxis del archivo mapfile .....	66
Archivo de inicialización HTML .....	83
Plantilla HTML .....	83
Conjunto de datos SIG .....	85
Formas de Consultas con MapServer .....	86
Tipos de Consulta .....	87
Aplicación con Consultas .....	89
MapFile .....	89
Archivo de inicialización .....	92
Plantilla HTML .....	93
Plantillas de Consulta .....	93
Conclusiones .....	97
 Capítulo V: Implementación de la Interfaz para configurar el Mapfile .....	 99

Interfaz de Registro de Archivos <i>Shape</i> .....	99
Interfaz de Configuración del MapFile .....	105
Conclusiones .....	114
Capítulo VI: Pruebas y Depuración de la Aplicación .....	115
Configuración de MapServer .....	115
Aplicación realizada en MapServer .....	116
Interfaz para grabar el archivo de configuración .....	128
Enlace de ayuda .....	129
Conclusiones .....	130
Conclusiones .....	132
Recomendaciones .....	133
Referencia Bibliográfica .....	135
Referencias Capítulo I .....	135
Referencias Capítulo II .....	137
Referencias Capítulo III .....	139
Referencias Capítulo IV .....	142
Referencias del Glosario .....	143
Glosario .....	148
Diccionario de Siglas .....	160
Anexo 1: pendientes.map	

## Resumen

Con nuestro proyecto de tesis demostramos que es posible publicar cartografía digital en la *web* sin la necesidad de una herramienta de licenciamiento privado. Para esto, realizamos el análisis de tres herramientas de código abierto: ALOV Map, GeoServNet y MapServer; luego de evaluarlas, elegimos MapServer por las prestaciones y escalabilidad que ofrece.

Damos a conocer la funcionalidad de la herramienta escogida publicando un mapa de la cuenca Río Paute en Internet, dicha información fue proporcionada por el Instituto Ecuatoriano de Régimen Seccional (IERSE); y facilitamos la configuración de su archivo base a través de una aplicación en la cual el usuario proporciona datos propios del mapa a publicar.

## **Abstract**

This thesis demonstrates the possibility to make public digital cartography on the web with GPL software. For this purpose we analysed three open source tools: ALOV Map, GeoServNet and MapServer, after evaluated them, we chose MapServer because of its scalability.

We let to know the functionality of the chosen tool publishing on Internet a map, the information was provided by Instituto de Estudios de Régimen Seccional del Ecuador (IERSE). Also, we developed an application for configuring the basic file, so the user enters map properties for its publication.

## Introducción

Debido al avance de cada una de las ciencias y al considerable volumen de información que cada una de éstas ha generado a través del tiempo, es práctico e incluso necesario vincular esta información con un punto sobre la Tierra, refiriéndonos así a un Sistema de Información Geográfica (SIG) como una tecnología empleada para resolver problemas territoriales. La característica principal de un SIG es su capacidad de análisis y generación de nueva información a partir de un conjunto previo de datos mediante su manipulación y reelaboración.

La generación de esta información en la actualidad no es completamente útil si no está accesible para la mayor cantidad de usuarios posibles. Una forma de solucionar este inconveniente es publicar los resultados de los estudios en la red Internet a través de un servidor de mapas, logrando así incrementar el potencial de un SIG ya que la información se encuentra vinculada y puede ser compartida a diferentes niveles.

En base a lo expuesto anteriormente, con nuestro estudio se plantea implementar un servidor de mapas utilizando una herramienta de código abierto, en el que los usuarios tengan la capacidad de acceder a la información cartográfica digital y a sus metadatos relacionados.

El proyecto demostrará las utilidades y potencialidades mediante la investigación, análisis y estudio de un servidor de cartografía en la *web* el cual será escogido por las características que brinde para publicar la información.

Posteriormente se procederá a implementar una aplicación que permita al usuario proporcionar al servidor la información del mapa a publicar, que para nuestro caso será un mapa de la cuenca del Río Paute facilitado por el Instituto de Estudios de Régimen Seccional del Ecuador (IERSE) perteneciente a la Universidad del Azuay .

La utilización del *software* de código abierto, permite la disminución en gran parte de los gastos efectuados en la implementación de nuevas tecnologías, ofreciendo características similares a las de *software* con licenciamiento cerrado.

## CAPÍTULO I

### **Introducción a los Conceptos Básicos relacionados con Sistemas de Información Geográfico y Servidores de Mapas**

Desde siempre, el buen manejo de la información ha sido pieza clave para el avance de los pueblos debido a que ésta ha sido fundamental para la toma de decisiones. Al estar esta información relacionada con un lugar en la Tierra, es imprescindible conocer y aplicar los términos correctos que se emplean en el idioma de los Sistemas de Información Geográfico.

Este capítulo tiene por objeto en primer lugar, presentar conceptos básicos de SIG referidos a: tipos de datos que se emplean, modelos de datos, principales sistemas de proyecciones y de coordenadas, metadatos, entre otros; en segundo lugar, se presentan conceptos relacionados con servidores de mapas tales como: servicios de catálogo, interoperatividad, servidores de mapas propiamente dichos y cartografía en la *web*.

Los conceptos mencionados anteriormente nos servirán como una base para adentrarnos en los siguientes capítulos sin dificultades y tener una visión global de la importancia de disponer de una herramienta para publicar información cartográfica digital en la Internet.

#### **Conceptos de sistemas de información geográfico y servidores de mapas**

A lo largo del tiempo los mapas han sido utilizados como herramientas útiles para las personas dedicadas a la geografía, pero la cartografía convencional tiene sus limitaciones en cuanto a la cantidad de información que puede almacenar y la resolución con la que ésta puede ser representada. Además, su actualización no es sencilla y la información tiene que ser retroalimentada manualmente.

El desarrollo de la informática ha permitido la transformación del manejo de la información espacial. Inicialmente facilitó la creación de cartografía haciéndola más

rápida y fácil; pero luego se dio énfasis al análisis e interpretación de los datos geográficos, apoyados en herramientas como el SIG que es considerado como una aplicación de la tecnología para resolver problemas territoriales.

El *National Center for Geographic Information and Analysis, N.C.G.I.A.* define al SIG como "Un sistema de *hardware*, *software* y procedimientos diseñados para facilitar la obtención, gestión, manipulación, análisis, modelación y salida de datos espacialmente referenciados, para resolver problemas complejos de planificación y gestión".

La definición del diccionario de la *Association for Geographic Information (AGI)* y el Departamento de Geografía de la Universidad de Edimburgo lo explica como: "un sistema de cómputo para obtener, almacenar, integrar, manipular, analizar y representar datos relativos a la superficie terrestre". [AA01]

Las áreas de aplicación práctica de un SIG son muy amplias: desde el inventario de los recursos naturales y humanos, hasta el control y gestión de los datos catastrales así como de grandes instalaciones: telefonía, alcantarillado, redes de comunicación, etc. Un SIG es necesario en cualquier circunstancia donde se requiera manejar información espacial.

El mapa tradicional es una representación analógica o continua de la realidad, por lo tanto, no está adaptado para ser procesado por un ordenador que utiliza datos en formato digital (discreto). Por ello el primer paso que es necesario realizar para introducir los datos en un SIG, es su conversión al formato digital.

Para lograr una representación digital correcta de los datos espaciales se debe tomar en cuenta dos componentes: la geocodificación de los datos y la descripción en términos digitales de las características espaciales. La primera consiste en un procedimiento "mediante el cual un objeto geográfico recibe directa o indirectamente una etiqueta que identifica su posición espacial con respecto a algún punto común o marco de referencia" (Goodchild, 1984).

En un segundo lugar, se debe realizar una descripción de la posición geométrica de cada objeto y de las relaciones espaciales (la topología) que mantiene con los

restantes objetos geográficos existentes en la realidad. Para llevar a cabo esta última labor es imprescindible una abstracción y simplificación de todos los elementos existentes, es decir, crear un modelo de datos (representación simplificada de la realidad) de los objetos a representar digitalmente (Peuquet, 1984 y 1988). Existen tipos de modelos de datos como: vectorial, raster.

## **1.- Conceptos básicos relacionados con sistemas de información geográfica**

Durante el desarrollo de nuestro proyecto estaremos abordando terminología básica ligada a los SIG por lo que damos a conocer las definiciones necesarias para su entendimiento.

### **Tipos de datos geográficos**

Los datos en un SIG pueden ser clasificados en: gráficos y alfanuméricos.

**Datos gráficos.-** Son descripciones digitales de las entidades del plano. Suelen incluir las coordenadas, reglas y símbolos que definen los elementos cartográficos en un mapa.

El SIG utiliza esos datos para generar un mapa o representación gráfica en una pantalla de ordenador o bien sobre papel. Para la representación de datos gráficos se utilizan tres tipos básicos de entidades: nodos o puntos, líneas y polígonos.

**Datos alfanuméricos.-** Son descripciones de las características de las entidades gráficas. Generalmente son almacenados en formatos convencionales para este tipo de información.

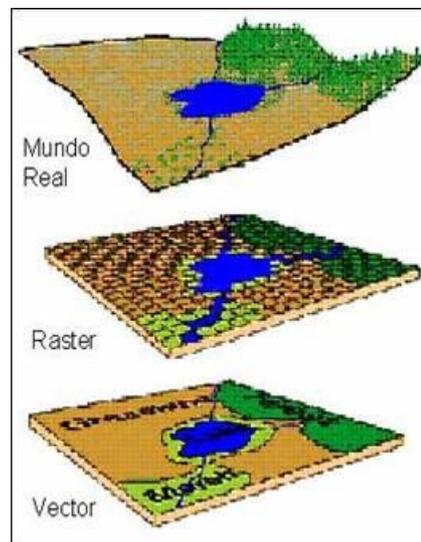
La información alfanumérica y gráfica se encuentra completamente integrada, siendo esta integración, junto con la capacidad de gestión de ambos tipos de datos, lo que caracteriza a los SIG. <sup>[AA01]</sup>

## Modelos de datos

Las entidades del mundo real pueden ser abstraídas o representadas de diferentes formas: como puntos, líneas, áreas (abstracción geométrica o cartográfica), como imágenes (por ejemplo fotografías) o como etiquetas (por ejemplo una dirección).

Las abstracciones de los objetos del mundo real deben ser representadas. Estas representaciones pueden ser en formato vectorial, formato raster, como entidades topológicas (nodos, polígonos), por símbolos o por textos. Ver figura 1.

Figura 1: Representación de modelos de datos



[http://gis.sopde.es/cursosgis/DHTML/que\\_2\\_2.html](http://gis.sopde.es/cursosgis/DHTML/que_2_2.html), Título: ¿Qué es un Sistema de Información Geográfica?

**El modelo raster** funciona a través de una retícula que permite asociar datos a una imagen; es decir, se pueden relacionar paquetes de información a los píxeles de una imagen digitalizada.

En el modelo raster el espacio es discretizado en pequeños rectángulos o cuadrados, de forma que el tamaño que tienen estos elementos es fundamental y determina la resolución. Utiliza un único elemento muy similar al punto, el píxel. Una malla de puntos de forma cuadrada o rectangular que contiene valores numéricos representa

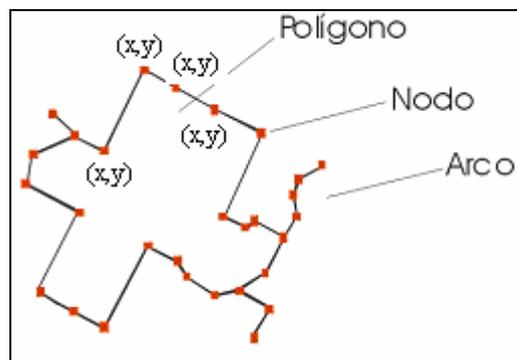
las entidades cartográficas y sus atributos a la vez. Los modelos lógicos menos complejos son los basados en el modelo conceptual raster, en buena medida porque la georeferenciación y la topología son implícitas a la posición - columna y fila - del pixel en la malla. Cada atributo temático es almacenado en una capa propia. La separación entre datos cartográficos y datos temáticos no existe, pues cada capa representa un único tema y cada celda contiene un único dato numérico.

**El modelo vectorial** se basa en tres elementos básicos:

- El nodo: es la unidad básica para representar entidades con posición pero sin dimensión.
- La línea o el arco: representa entidades de una dimensión y está restringido a línea recta en algunas implementaciones.
- El polígono o área: se utiliza para representar las entidades bidimensionales. Algunos autores añaden una cuarta, el volumen.

La información sobre puntos, líneas y polígonos se almacena como una colección de coordenadas  $x, y$ . La ubicación de una característica puntual, pueden describirse con un sólo punto  $x, y$ . Las características lineales, pueden almacenarse como un conjunto de puntos de coordenadas  $x, y$ . Las características poligonales, pueden almacenarse como un circuito cerrado de coordenadas. Figura 2.

Figura 2: Componentes del modelo vectorial



[http://gis.sopde.es/cursosgis/DHTML/que\\_2\\_2.html](http://gis.sopde.es/cursosgis/DHTML/que_2_2.html), Título: ¿Qué es un Sistema de Información Geográfica?

## Escalas de medida

La medición pone en relación dos mundos diferentes; un físico, en el cual el hombre puede realizar observaciones, con otro mundo formalizado y teórico, donde sólo se manejan relaciones abstractas y lógicas (Castro, 1978, p. 24; Castro, 1987). Los hechos de la realidad son muy diferentes y variados haciéndose imposible representar numéricamente sus cualidades. Por ello, se ha hecho necesario formular varias escalas de medida, las cuales difieren en el número de relaciones matemáticas que es posible establecer entre las modalidades de una característica real. <sup>[AA02]</sup>

Se pueden distinguir los siguientes tipos de escalas o niveles de medida:

1. Escala nominal.- Es de carácter cualitativo. Determina si un objeto pertenece o no a un grupo con ciertas características definidas.
2. Escala ordinal.- A más de determinar si un objeto es igual o diferente que un conjunto (escala nominal), establece si es mayor o menor que éste.
3. Escala de intervalos.- Es posible establecer tres relaciones matemáticas: igualdad/desigualdad (nominal), mayor/menor que (ordinal) y la capacidad de asignar valores numéricos a las distancias o diferencias entre dos modalidades.
4. Escala de razón.- Representación gráfica proporcional en función de cierto parámetro. Representa la relación entre la distancia de dos puntos de la Tierra y la distancia de los puntos que se corresponden con ellos en el mapa. La escala numérica se representa en cifras, como por ejemplo: 1:100 000, lo que indica que una unidad medida en el mapa (por ejemplo 1 cm) representa 100.000 de las mismas unidades en la superficie terrestre. <sup>[AA03]</sup>

## **Georeferenciación o geocodificación de datos geográficos**

La georeferenciación se puede definir como aquel proceso mediante el cual se identifica una posición en la superficie terrestre. Existen dos tipos de georeferenciación:

- **Georeferenciación directa**

Se basa en el uso de un sistema de coordenadas establecido para un determinado sistema de proyección.

- **Georeferenciación indirecta o discreta**

Su fundamento es asociar al elemento que se representa una clave o índice, normalmente con significado administrativo (dirección, código postal, etc.), que puede ser usada para la determinación de una posición, naturalmente con una precisión no siempre equivalente a la obtenida con georeferenciación directa. La virtud de este sistema es el poder aprovechar de forma inmediata la gran cantidad de información disponible con georeferenciación directa.

[AA04]

## **Sistema de proyección**

Los sistemas de proyección están pensados para resolver el problema de proyectar la superficie curva de la tierra en un sistema plano. Aunque todo sistema de proyección distorsiona la realidad, podemos mantener sin distorsión el área (proyecciones equivalentes), las distancias (equidistantes) o los ángulos (conformes).

Una proyección geográfica es un sistema ordenado que traslada desde la superficie curva de la Tierra la red de meridianos y paralelos sobre una superficie plana. Se representa gráficamente en forma de malla. La única forma de evitar los problemas de proyección es usar un globo, pero en la mayoría de las ocasiones sería demasiado grande para que resultase útil.

Una buena proyección debe tener dos características: que conserve las áreas y que conserve los ángulos. Esto no es posible, por lo que hay que buscar soluciones intermedias. Cuando una proyección conserva los ángulos de los contornos decimos que es ortomórfica o conforme, pero estas proyecciones no conservan las áreas.

Dependiendo de cuál sea el punto que consideremos como centro del mapa distinguimos entre proyecciones polares, cuyo centro es uno de los polos; ecuatoriales cuyo centro es la intersección entre la línea del ecuador y un meridiano; y oblicuas o inclinadas, cuyo centro es cualquier otro punto.

Existen tres tipos principales de sistemas de proyecciones

### **Proyección cilíndrica**

Es la proyección que revolucionó a la cartografía. En ella se proyecta el globo terrestre sobre un cilindro. Es una de las más utilizadas aun cuando por lo general en forma modificada, debido a las grandes distorsiones que ofrece en las zonas de latitud elevada, cosa que impide apreciar en sus verdaderas proporciones a las regiones polares. Ver figura 3.

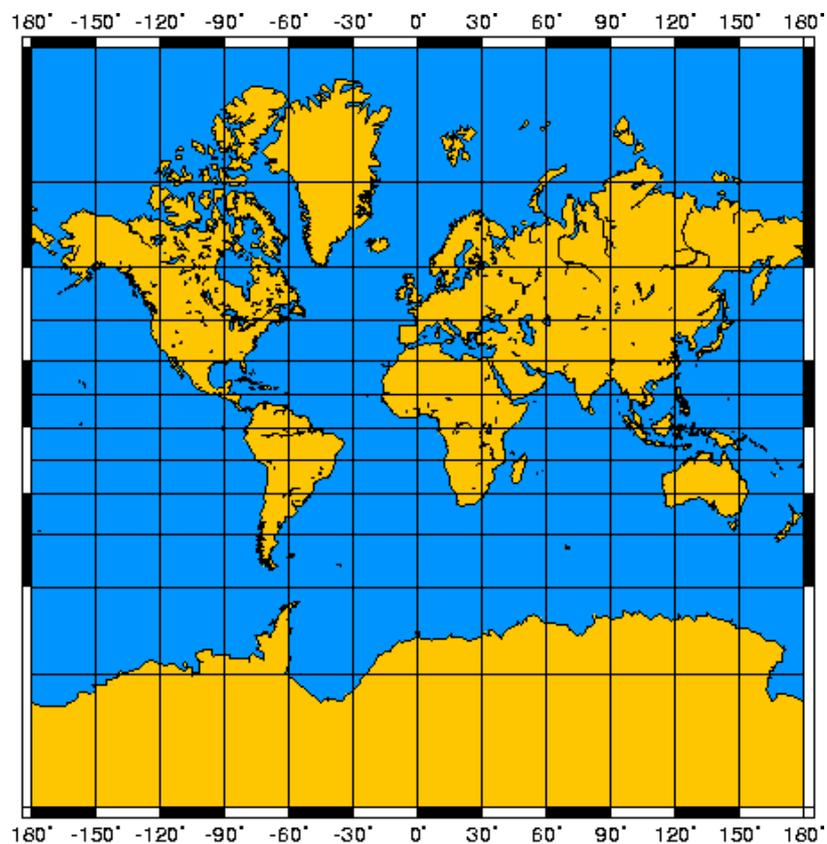
- **Proyección de Mercator**

La característica más destacable de esta proyección es que tanto los meridianos como los paralelos son líneas rectas y se cortan perpendicularmente. Los meridianos son líneas rectas paralelas entre sí dispuestas verticalmente a la misma distancia unos de otros. Los paralelos son líneas rectas paralelas entre sí dispuestas horizontalmente pero aumentando la escala a medida que nos alejamos de Ecuador. Este aumento de escala hace que no sea posible representar en el mapa las latitudes por encima de los 80°. [AA05]

El éxito de la proyección de Mercator se debe a que cualquier línea recta que se trace marca el rumbo real, con lo cual se puede navegar siguiendo con la brújula el ángulo que se marca en el mapa. A esta línea de rumbo se llama loxodrómica

La proyección Mercator, a pesar de su gran distorsión, es muy empleada porque los ángulos, las líneas, direcciones y distancias aparecen como líneas rectas y pueden medirse directamente.

Figura 3: Proyección de Mercator

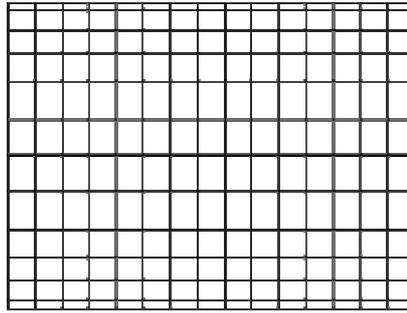


[http://es.wikipedia.org/wiki/Proyecci%C3%B3n\\_de\\_Mercator](http://es.wikipedia.org/wiki/Proyecci%C3%B3n_de_Mercator), Título: Proyección de Mercator

- Proyección de Peter

En ella los paralelos y los meridianos son sustituidos por una cuadrícula de 10 grados decimales. La proyección refleja correctamente las áreas de los países. <sup>[AA06]</sup> Ver figura 4

Figura 4: Proyección de Peter



[http://es.wikipedia.org/wiki/Proyecci%C3%B3n\\_de\\_Peter](http://es.wikipedia.org/wiki/Proyecci%C3%B3n_de_Peter), Título: Proyección de Peters

### **Proyección cónica** <sup>[AA07]</sup>

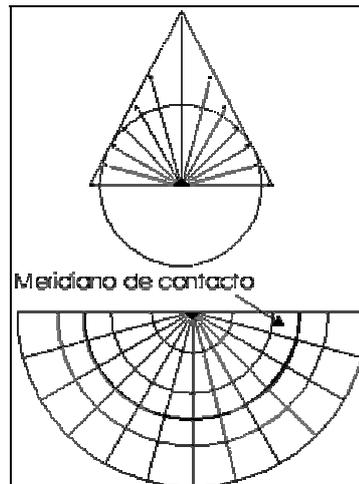
La proyección se hace trasladando la información de la esfera a un cono, tomando como punto focal uno de los polos. Hay una distorsión asimétrica que también afecta a las zonas polares, pero ofrece mayor precisión en el hemisferio que corresponde al polo que se haya tomado como foco.

- Proyección cónica simple. Ver figura 5

La proyección simple puede tener uno o dos paralelos de referencia. Si tiene un paralelo de referencia, la malla de meridianos y paralelos se dibuja proyectándolos sobre el cono suponiendo un foco de luz que se encuentra en el centro del globo. El resultado es un mapa semicircular en el que los meridianos son líneas rectas dispuestas radialmente y los paralelos arcos de círculos concéntricos. La escala aumenta a medida que nos alejamos del paralelo de contacto entre el cono y la esfera.

Si tiene dos paralelos de referencia el cono secante corta el globo a medida que nos alejamos de ellos la escala aumenta pero en la región comprendida entre los dos paralelos la escala disminuye.

Figura 5: Proyección cónica simple



<http://club.telepolis.com/geografo/general/pconica.htm#simple>, Título: Proyecciones cónicas

- Proyección conforme de Lambert

Sobre la base de la proyección simple con dos meridianos de referencia Lambert ajustó matemáticamente la distancia entre paralelos para crear un mapa conforme. Como los meridianos son líneas rectas y los paralelos arcos de círculo concéntricos las diferentes hojas encajan perfectamente.

- Proyección cónica múltiple

Esta proyección consiste en utilizar no un cono, sino varios superpuestos. El resultado es un mapa dividido en franjas. El único meridiano que tendrá la misma escala es el central, que aparece como una línea recta. Los demás meridianos son curvas, y la escala aumenta con la distancia. También el ecuador es una línea recta, perpendicular al meridiano central. Los demás paralelos son arcos concéntricos.

Esta proyección no es conforme ni conserva las áreas, pero en la zona central las variaciones de escala son mínimas.

## **Proyección azimutal o plana** <sup>[AA08]</sup>

En este caso se proyecta una porción de la Tierra sobre un disco plano tangente al globo en un punto seleccionado, obteniéndose la visión que se lograría ya sea desde el centro de la Tierra o desde un punto del espacio exterior. Si la proyección es del primer tipo se llama proyección gnomónica; si del segundo, ortográfica. Estas proyecciones ofrecen una mayor distorsión cuanto mayor sea a su vez la distancia al punto tangencial de la esfera y del plano.

- **Proyección ortográfica**

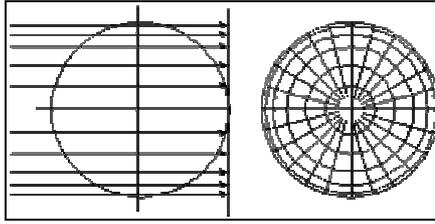
La proyección ortográfica se obtiene cuando consideramos que el foco de luz procede de una fuente muy lejana. Su aspecto es el de una fotografía de la Tierra. Figura 6.

La proyección polar se caracteriza por que todos los meridianos son líneas rectas y la distancia entre paralelos disminuye según nos alejamos del centro. La distancia entre paralelos o meridianos depende de la escala así que cuando disminuye la distancia disminuye la escala y cuando aumenta la distancia aumenta la escala.

La proyección ecuatorial se caracteriza porque los paralelos son líneas rectas. También es una línea recta el meridiano central. A medida que nos alejamos del centro la escala disminuye. Los meridianos tienen forma de arco.

Con esta proyección sólo se puede representar de una vez un hemisferio. Este sistema ni conserva las áreas ni es conforme. La escala es mayor en el centro que en el borde.

Figura 6: Proyección ortográfica



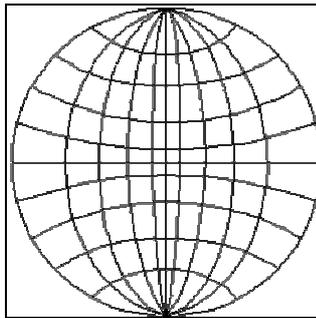
[http://club.telepolis.com/geografo/general/pcenital.htm#ortografica,](http://club.telepolis.com/geografo/general/pcenital.htm#ortografica)  
Proyecciones cenitales

Título:

- Proyección estereográfica

En la proyección estereográfica consideramos que el foco de luz está en los antípodas (lugar opuesto). La superficie que puede representar es mayor que un hemisferio. El rasgo más característico es que la escala aumenta a medida que nos alejamos del centro. Figura 7. En su proyección polar los meridianos son líneas rectas. En la proyección ecuatorial sólo son líneas rectas el ecuador y el meridiano central. Esta es una de las proyecciones conformes que existen.

Figura 7: Proyección estereográfica



[http://club.telepolis.com/geografo/general/estecua.htm,](http://club.telepolis.com/geografo/general/estecua.htm)  
estereográfica ecuatorial

Título: Proyección

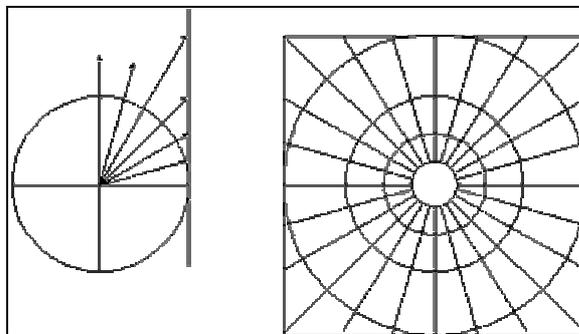
- Proyección gnomónica

La proyección gnomónica resulta de colocar el foco de luz en el centro de la Tierra. La escala aumenta rápidamente del centro al exterior. Figura 8.

En la proyección polar todos los meridianos son líneas rectas y se disponen radialmente, en la proyección ecuatorial son líneas rectas el ecuador y los meridianos, que se disponen verticalmente, en la proyección oblicua son líneas rectas el ecuador y los meridianos.

Esta proyección se caracteriza por que toda línea recta es un círculo máximo, y por lo tanto el camino más corto entre dos puntos de la Tierra. Con este sistema no se puede representar un hemisferio completo.

Figura 8: Proyección gnomónica



<http://club.telepolis.com/geografo/general/gnomopolar.htm>, Título: Proyección gnomónica polar.

- Proyección azimutal

La proyección acimutal dispone la red de meridianos y paralelos de manera equidistante de forma deliberada. Se puede ampliar el mapa hasta representar todo el globo.

- Proyección azimutal de Lambert

La proyección de Lambert conserva deliberadamente las áreas. Es una proyección conforme. La escala disminuye a medida que nos acercamos al borde exterior, pero en menor medida que en la proyección ortográfica. Este sistema es muy adecuado para trazar mapas de pequeña escala.

## **Sistema de proyección utilizado en el proyecto**

La cartografía empleada en la aplicación de nuestro proyecto, utiliza el sistema UTM (Universal Transversal Mercator), que se obtiene proyectando sobre un cilindro cuya directriz es un meridiano terrestre (a lo largo del cual la distorsión es nula). En este caso, la georeferencia se expresa mediante un identificador de zona y dos coordenadas (x, y) en metros, según los ejes E-O y N-S respectivamente. Este sistema es el que se usa en la mayoría de los organismos cartográficos nacionales e internacionales, así como en el que se proporcionan habitualmente los datos de imágenes de satélites. Sin embargo, los problemas se presentan cuando es necesario trabajar con datos de dos zonas diferentes.

En el sistema UTM, se divide a la Tierra en 60 zonas limitadas por meridianos, cubriendo cada zona seis grados de longitud. La división en zonas es la siguiente: Zona 1 De 180° a 174°W, con el meridiano tangente en 177° W, Zona 2 De 174° W a 168° W con el meridiano tangente en 171° W. Etc... Zona 30 De 6° W a 0° meridiano de Greenwich. Zona 31 Desde el meridiano de Greenwich hasta los 6°E y la zona 60 va desde 174°E a 180°.

La proyección UTM tiene la ventaja es que sus magnitudes se expresan en metros, en vez de medidas angulares cuya dimensión lineal puede variar, además de que ningún punto está alejado del meridiano central de su zona, por lo que las distorsiones son pequeñas. Pero esto se consigue al coste de la discontinuidad: un punto en el límite de la zona se proyecta en dos puntos distintos, salvo que se encuentre en el ecuador. Una línea que una dos puntos de entre zonas contiguas no es continua salvo que cruce por el ecuador.

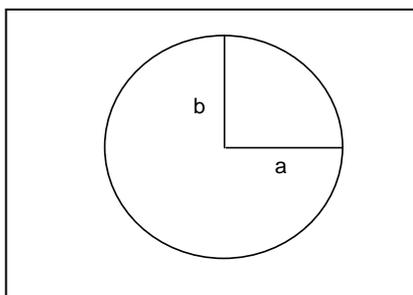
Para evitar estas discontinuidades, a veces se extienden las zonas, para que el meridiano tangente sea el mismo. Esto permite mapas continuos casi compatibles con los estándares. Sin embargo, en los límites de esas zonas, las distorsiones son mayores que en las zonas estándares. <sup>[AA18]</sup>

## Sistema de coordenadas

Existen diversos tipos de sistemas de referencia, los cuales permiten ubicarse espacialmente. La cartografía es una ciencia que permite, a través de proyecciones, representar el geoide (superficie de la tierra) en una superficie plana.

Toda proyección está definida por dos elementos fundamentales; *el elipsoide de referencia*, el cual nos indica los parámetros matemáticos que se utilizaron en su definición (figura 9), y el *datum*, que es el punto donde la tierra (geoide) se hace tangente al elipsoide.<sup>[AA09]</sup>

Figura 9: Elipsoide



Tutorial de prácticas ArcGIS, Autor: Ing. Paúl Ochoa A. p. 8

a: semieje mayor,

b: semieje menor,

f: achatamiento que es la diferencia entre los dos semiejes expresado en fracción decimal  $f = (a-b)/a$

Un *datum* tiene asociado uno y sólo un elipsoide. Por el contrario, un elipsoide puede ser usado en la definición de muchos *datum*.

El *datum* que se utilizó en Ecuador hasta hace poco fue el “*Provisional South American 1956*” (PSAD56 también conocido como SAM56) cuyo elipsoide es el internacional de Hayford, tiene como punto de origen a un sitio denominado “La Canoa” en Venezuela y sus parámetros son:  $a=6378388$  y  $1/f=297$ <sup>[AA19]</sup>. El *Datum* más utilizado actualmente, impulsado sobre todo por el uso de los GPS (*Global*

*Position System*), es el *World Geodetic System* de 1984 (WGS84) cuyo elipsoide es el internacional 1984, y tiene como punto de origen al centro de gravedad de la tierra, y sus parámetros son:  $a = 6378137.0\text{m}$ ;  $b = 6356752.3\text{m}$  ;  $1/f = 298.257223563$  <sup>[AA10]</sup>

## **Metadatos** <sup>[AA11]</sup> y <sup>[AA12]</sup>

Un metadato es una información que describe la calidad, el contenido, distribución, actualidad y referencia espacial de un conjunto de datos. Es "información sobre información" o "datos sobre los datos".

### **Principales usos de los metadatos**

- Organizar y mantener el conjunto de datos de una organización.
- Proporcionar información para catálogos de datos y centros de distribución de metadatos (*clearinghouses*).
- Proveer información necesaria para interpretar y procesar datos transferidos por otra organización.

### **Diferentes estándares**

Las comunidades de usuarios han definido diferentes estándares de metadatos para satisfacer sus necesidades:

- *Government Information Locator Service* (GILS): define información gubernamental.
- *Federal Geographic Data Committee* (FGDC): describe datos geo-espaciales.
- *U. S. Machine Readable Catalog* (USMARC): define catálogos de fichas bibliográficas.
- *Dublin Core*: define metadatos asociados con páginas web.
- *Consortium for the Interchange of Museum Information* (CIMI): define los metadatos asociados con información de museos.

## Estructura de los metadatos

Los metadatos pueden ser mínimos o detallados según el nivel de información que brinden de un conjunto de datos documentado.

El **metadato detallado** se compone de nueve secciones. Siete de éstas son secciones independientes y las dos restantes son secciones de soporte que cobijan elementos comunes de las siete básicas.

El **metadato mínimo** se restringe sólo a los componentes más importantes e involucra sólo tres secciones del detallado con algunos de sus elementos: identificación, calidad y distribución.

En la tabla No.1 se describen las secciones que componen la estructura de los metadatos.

-  Secciones Obligatorias
-  Secciones condicionales

Tabla 1: Estructura de los metadatos

<b>Identificación</b>	Información básica sobre el conjunto de datos (título, autoría, propósito, resumen, temática, localización, área incluido, restricciones etc).
<b>Calidad</b>	Evaluación general de la calidad de un conjunto de datos: precisión, a qué nivel están completos los datos.
<b>Representación espacial</b>	Información sobre los mecanismos empleados para representar espacialmente el conjunto de datos: vector, raster, punto.
<b>Referencia espacial</b>	Descripción del marco de referencia para las coordenadas del conjunto de datos y los medios de codificación:

<b>Entidades y atributos</b>	<p>proyección, datum, sistemas de coordenadas, etc.</p> <p>Información sobre los objetos geográficos involucrados y sus atributos: información acerca de entidades, atributos, dominio de valores de los atributos, etc.</p>
<b>Distribución</b>	<p>Datos del distribuidor y medios para obtener el conjunto de datos: distribuidor, formatos, medios, estatus, precio, etc.</p>
<b>Referencia del metadato</b>	<p>Actualidad de la información del metadato y de sus responsables: nivel de actualización, institución o persona responsable, etc.</p>
<b>Citación</b>	<p>Datos de soporte sobre las referencias citadas dentro del conjunto de datos</p>
<b>Contacto</b>	<p>Información de soporte sobre personas y organizaciones asociadas al conjunto de datos</p>

<http://gemini.udistrital.edu.co/comunidad/profesores/rfranco/metadatos.htm>, Título: Sistemas de Información Geográfica

## 2.- Conceptos de Servidores de Mapas <sup>[AA13]</sup>

Esta sección documenta conceptos simples de cartografía en la red, así como herramientas que permiten la visualización de información geoespacial proveniente de varias organizaciones y servidores a través de la *World Wide Web*.

### Contexto y fundamento

El crecimiento de Internet y específicamente de la *World Wide Web* ha creado expectativas en lo que se refiere al acceso a información geoespacial en la Red por

parte del navegante. La cartografía en la Red incluye la presentación de mapas de uso general para exhibir lugares y accidentes geográficos, como también más sofisticadas herramientas cartográficas, interactivas e individualizadas. La intención de la cartografía en la Red es representar información espacial rápida y fácil para los usuarios, requiriéndose para ello solamente la habilidad para leer mapas.

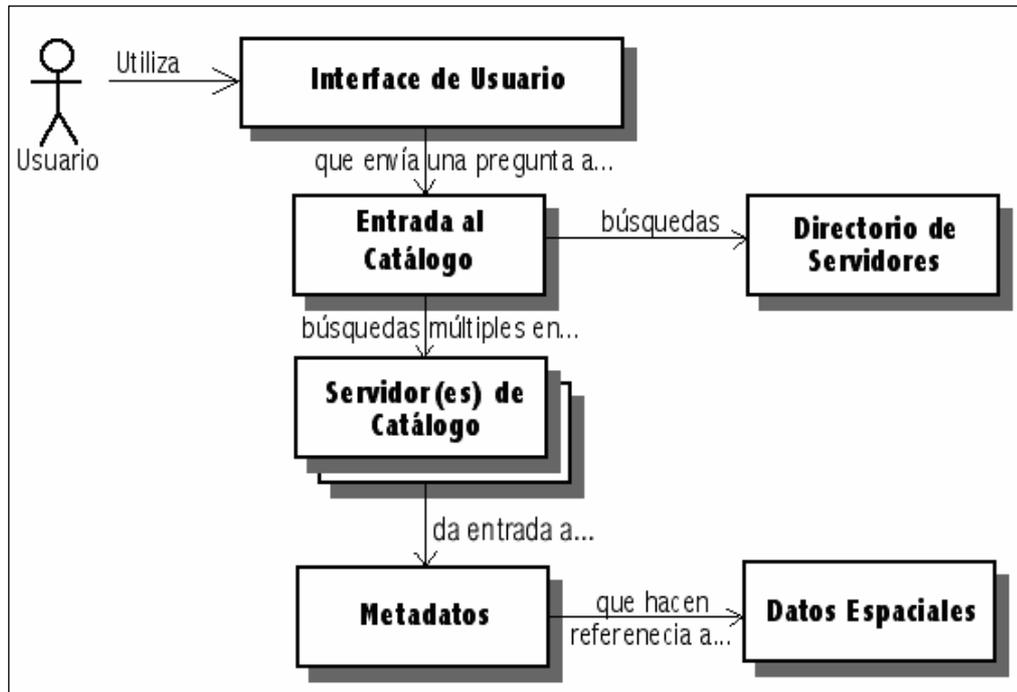
Se pueden descubrir los servicios de cartografía en la Red a través de guías *online* que sirven datos espaciales (por medio de metadatos) e información de servicios (Servicios de Catálogo OGC). En efecto, los servicios de cartografía en Red se utilizan con frecuencia para asistir a los usuarios en sistemas de búsqueda espacial, mostrando el contexto geográfico y la amplitud de los datos relevantes frente a los datos de referencia de mapas básicos.

### **Servicios de catálogo**

Al servicio que soporta el descubrimiento y acceso a la información geoespacial se le conoce de diferentes maneras dentro de la comunidad geoespacial; así se conoce como “Servicios de Catálogo” según la definición que da el *OpenGIS Consortium* (OGC); la *Australian Spatial Data Infrastructure* le llama “Directorio de Datos Espaciales” y “*Clearinghouse*” que se entiende como “Agencia de Distribución” es el nombre dado por el FGDC de EEUU. Aunque tengan nombres diferentes, el objetivo es el mismo: descubrir datos geoespaciales a través de las propiedades descritas por sus metadatos.

Un proveedor de datos debe tener domiciliados los metadatos en alguno de los nodos accesibles de la red. El servidor de catálogos, ante una pregunta por parte de un usuario a través de un simple interfaz, remite una interrogación mediante un mensaje XML al resto de los servidores de catálogos disponibles, los cuales contestan mediante mensajes XML que pueden presentarse al usuario (figura 10). Los servidores de catálogo pueden corresponder a fabricantes diferentes pero si cumplen la especificación OGC CAT v1.1.1 podrán entender la pregunta que se les hace y responderla adecuadamente. <sup>[AA14]</sup>

Figura 10: Diagrama de interacción que muestra el uso básico de servicios de catálogo



[http://cndg.clearinghouse.gub.uy/documentos/capitulo04\\_1.aspx.htm](http://cndg.clearinghouse.gub.uy/documentos/capitulo04_1.aspx.htm) Título: Catálogo de datos geoespaciales / Localización de los datos

### Interoperatividad de los servicios <sup>[AA15]</sup> y <sup>[AA16]</sup>

La cartografía de red ejecutada como conjunto de sistemas patentados, funciona bien mientras que cada persona con la que se trate dentro y fuera de la organización utilice el mismo software de propiedad. Debido a esta obvia limitación particular, se implanta el concepto de interoperatividad, el cual formalmente se define como la capacidad para comunicar, ejecutar programas o transferir datos entre varias unidades funcionales de forma que un usuario necesite pocos conocimientos de las características de estas unidades. Para que dos sistemas diferentes puedan comunicarse e intercambiar información primero deben anunciar su existencia y su voluntad para el intercambio y, segundo, deben utilizar una semántica adecuada para resolver los problemas técnicos que puedan presentarse.

En este contexto, se denominan *Servicios Web (Web Services)* a un conjunto de tecnologías basadas en la interoperabilidad y que cumplen una serie de opciones: son abiertas, neutras con respecto a la plataforma y explotan la arquitectura de la *web*. Están pensados para crear servicios distribuidos, que funcionen de forma autónoma y que deben comunicarse o colaborar entre ellos.

Esta es la estrategia que está utilizando OGC, una organización sin ánimo de lucro, fundada en 1994 y dedicada a la promoción de nuevas aproximaciones técnicas y comerciales para geoprocesamiento abierto e interoperable, fundada por las más importantes entidades industriales, gubernamentales y académicas. Su objetivo es conseguir acuerdos sobre interfaces de *software* estándar abiertos que permitan la interoperación entre sistemas de geoprocesamiento de vendedores diferentes y de diferentes tipos (GIS/SIG, teledetección, cartografía automática, gestión de instalaciones, etc.)

En la actualidad existe un buen número de especificaciones OGC definiendo *Servicios Web*, tales como:

*Servicios de Mapas en Web – OGC WMS*: Especifica el comportamiento que debe tener un servidor de mapas para ser considerado estándar. Se trata de una aplicación sin estados, que responde de una forma u otra según sea la petición recibida. Tanto peticiones como respuestas se realizan en código XML. Un servidor de mapas de tipo WMS funciona de la siguiente manera: dada una petición de mapa en formato XML, genera un nuevo mapa (en una imagen), y devuelve código XML indicando dónde está esa imagen. La respuesta a peticiones de tipo “identificar” es código XML con los datos devueltos. Además de esto, un servidor WMS y, en general, todo servidor W?S, debe responder a la petición *GetCapabilities*. Con esta petición se pretende devolver un fichero XML que permita al cliente saber qué puede pedir a ese servidor.

*Servicios de Features en Web – OGC WFS*: El objetivo de la especificación de la interfaz *Web Feature Server (WFS)* es describir operaciones de manipulación de datos sobre objetos definidos en la especificación *Simple Features* de modo que los

clientes y servidores pueden comunicarse a nivel de objeto. Esto permite enviar objetos (vectores) a los clientes, e incluso la edición de los mismos.

Servicios de *Coverages en Web* – OGC WCS: Extiende la interface *Web Map Server* (WMS) para proporcionar acceso a coberturas que representan valores o propiedades de emplazamientos geográficos en lugar de los mapas generador por WMS (imágenes).

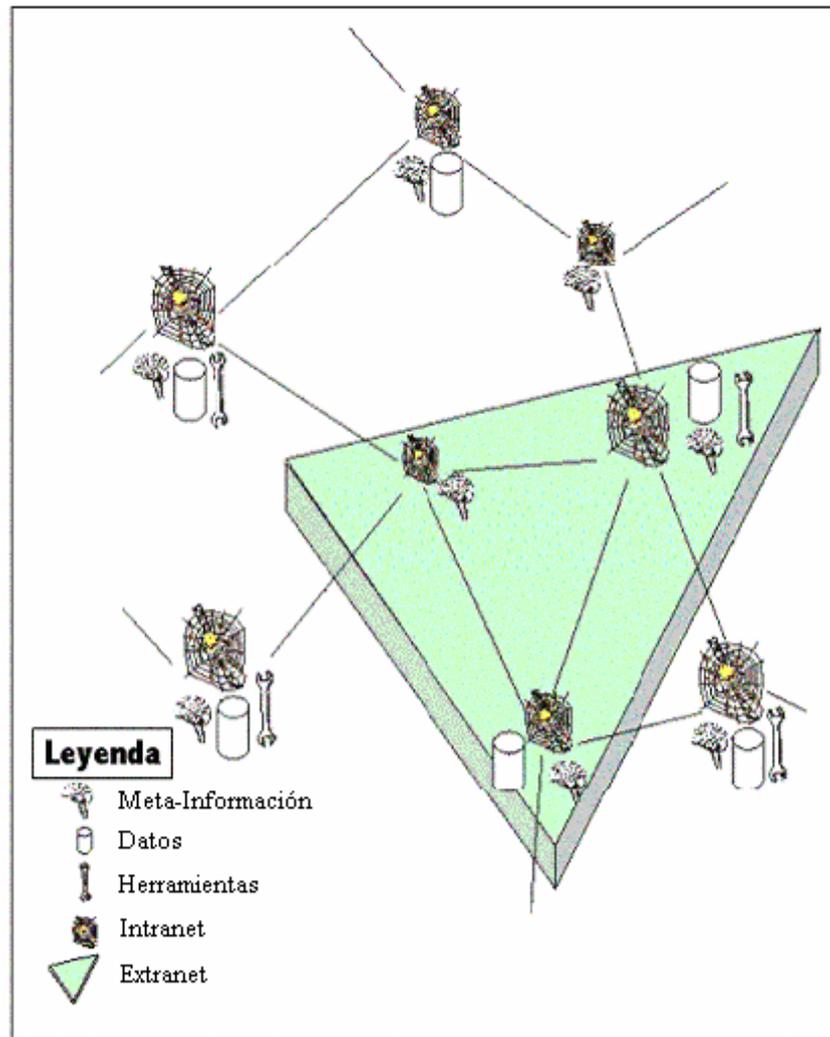
Servicios de *Gazetteer* (GAZ): Es conocido también como servicio de Nomenclátor, permite realizar búsquedas de topónimos con varios criterios (nombre exacto, nombre incluido, empezando por, en una zona, sólo topónimos de una clase, etc) y devuelve su situación geográfica.

Servicios de Catálogo – OGC CAT: Define una interfaz común que permite que diversas aplicaciones conformes a esta especificación realicen búsquedas, naveguen y consulten contra servidores de catálogos de información geográfica potencialmente heterogéneos.

### **Servidores de mapas** <sup>[AA13]</sup> y <sup>[AA17]</sup>

Para que llegue a tener éxito el concepto de Cartografía en la Red, tiene que establecerse una serie casi global, verdaderamente interconectada, de servidores de mapas, a través del uso de protocolos comunes, bien sea en un escenario de intranet, extranet o internet. La figura 11 da una noción de red de servidores de ese tipo. Los servidores que respalden la cartografía "online" serán registrados en un Sistema de Servicio de Catálogo.

Figura 11: Figuración de la red de Servidores de Cartografía en Red



<http://cndg.clearinghouse.gub.uy/documentos/recetario/capitulo05.aspx>

Título: Visualización de datos geospaciales / Web Mapping

A manera de introducción a las ejecuciones de los Servidores de Mapas en la Red, WMS, lo que sigue es un extracto de la especificación WMS v1.0:

Un servidor de Mapas puede hacer tres cosas:

1. Producir un mapa (como ilustración, como una serie de elementos gráficos o como un conjunto empaquetado de datos de características geográficas).
2. Responder a preguntas básicas sobre el contenido del mapa, y
3. Decirle a otros programas qué mapas puede producir y cuáles de ellos pueden ser cuestionados adicionalmente.

En primer lugar un navegante típico puede pedir al Servidor de Mapas que haga estas cosas, enviando peticiones en forma de URL (*Uniform Resource Locators*) (Localizadores Uniformes de Recursos). El contenido de tales URLs depende de cuál de las tres tareas se pide. Todos los URLs incluyen una especificación de Tecnología de Cartografía en la Red, con número de versión y un parámetro de tipo de petición. Además,

1. Para producir un mapa, los parámetros URL indican de qué porción de la Tierra se trata, el sistema de coordenadas que se va a usar, el tipo o tipos de información que han de aparecer, el formato de salida deseado y quizá su tamaño, estilo de presentación u otros parámetros.
2. Para interrogar el contenido del mapa, los parámetros URL indican qué mapa se está interrogando y qué localización dentro del mapa es de interés.
3. Para preguntar al Servidor de Mapas sobre sus posesiones, los parámetros URL incluyen un tipo de petición de capacidades.

### **Cartografía en la Red (Web Mapping)** <sup>[AA13] y [AA17]</sup>

Las especificaciones del *Web Map Server* ofrecen una manera de permitir la superposición visual de información geográfica compleja y distribuida (mapas) simultáneamente en Internet. Además, otras especificaciones del OGC permitirán compartir servicios de geoprosesamiento, tales como transformación de coordenadas en la WWW.

Cartografía en la Red (*Web Mapping*) se refiere, como mínimo, a las siguientes acciones:

- Un cliente hace peticiones a uno o más Registros de Servicio (basados en la Especificación de los Servicios de Catálogo del *OpenGIS*) de descubrir URLs de Servidores de Mapas en la Red que contengan la información deseada.
- Los Registros de Servicio devuelven URLs y también información sobre métodos por medio de los cuales se puede ganar acceso a la información descubierta en cada URL.

- El cliente localiza uno o más servidores que contienen la información deseada y recurre a ellos simultáneamente.
- Como ha sido ordenado por el cliente, cada servidor de Mapas gana acceso a la información pedida y la da de la manera idónea para ser representada como una o más capas en un mapa compuesto de muchas capas.
- Los Servidores de mapas proveen al cliente (o clientes) la información preparada para ser representada y éstos lo hacen. Los clientes pueden representar la información proveniente de muchas fuentes en una sola ventana.

Hoy en día el WMS 1.0 define tres principales interfaces que soportan la cartografía de Red

- *GetCapabilities*: investiga las capacidades del servidor de mapas interrogado mediante un mensaje XML. El servidor le devuelve la información mediante otro mensaje XML, es decir, explica lo que un servidor de mapas puede hacer (para que los integradores sepan qué pedir).
- *GetMap*: conociendo las capacidades del servidor, requiere un mapa mediante un mensaje XML y el servidor interrogado devuelve un mapa en formato ráster (PNG, JPEG, GIF). Estos mapas pueden superponerse al definir colores transparentes. Especifica los parámetros de petición de mapas que permite a servidores múltiples producir diferentes capas de mapas para un único cliente
- *GetFeatureInfo*: sobre el mapa devuelto se puede interrogar al servidor remoto sobre información asociada a algún elemento (que se puede seleccionar, por ejemplo, mediante un clic sobre un píxel del elemento). Tanto la pregunta como la respuesta se vuelven a realizar mediante mensajes XML.

Estas interfaces procuran un alto nivel de abstracción que esconde dificultades en el escenario de la Cartografía en la Red. Éstas incluyen encontrar servidores de almacén de datos remotos, pedirles datos en estructuras específicamente definidas, adjuntar símbolos inteligentemente, cambiar sistemas de coordenadas y devolver información preparada y representada para el cliente -todo en cuestión de segundos-.

## **Conclusiones**

Como hemos visto a lo largo de este capítulo, la capacidad para organizar y acceder a la información geográfica a través de Internet tiene una demanda creciente no únicamente a nivel internacional sino también a nivel local, ya que cada día la necesidad de consulta y visualización de datos geográficos *on line* es un requisito para dar soporte a la sociedad en cualquier ámbito que ésta la solicite.

Los conceptos presentados anteriormente se apoyan en estándares y normas aceptadas a nivel mundial, todos los sistemas SIG para publicación de cartografía en Internet libres o pagados basan su funcionamiento en dichas normas debido a que todos tienden a brindar en mayor o menor grado una interconexión entre redes de trabajo.

## CAPÍTULO II

### Evaluación de Diferentes Herramientas para la Publicación de Mapas

En la actualidad la industria del software ofrece diversas alternativas de sistemas de código abierto para la publicación de cartografía digital en Internet. El presente capítulo tiene por objetivo investigar y analizar tres posibles herramientas para cumplir con esta tarea. Se describirá el origen, bases teóricas, funcionamiento y demás características de dichas herramientas que son las de mayor uso y difusión en el mercado por las facilidades que presentan cada una de ellas.

#### **ALOV Map** <sup>[BA01] y [BA02]</sup>

ALOV Map se desarrolló como un proyecto conjunto de *ALOV Software* y el *Archaeological Computing Laboratory* de la Universidad de Sydney, Australia. Empezó en noviembre del 2001 como una parte del trabajo en la versión de *Windows* para el software *TimeMap*® el cual se inició en 1999.

ALOV Map es una aplicación para la publicación en Internet de mapas raster y vectoriales. Soporta una arquitectura compleja de despliegue y tiene navegación ilimitada, permite trabajar con múltiples capas, mapas temáticos, atributos de datos y características como un hiperenlace. Desde Febrero del 2004 ALOV Map es distribuido bajo el logo *TMJava* el cual es una versión actualizada de ALOV Map, además provee nuevas características incluyendo filtrado temporal, control avanzado de capas, despliegue de *TimeMap MapSpaces*, y acceso a un índice central de registros de datos desarrollado por el *Electronic Cultural Atlas Initiative*.

ALOV dispone de dos versiones del producto: la versión del lado del cliente (una de las alternativas más sencillas para publicar cartografía en la *web*) y la versión del lado del servidor.

La versión del lado del cliente es un *applet* Java que se descarga junto con los datos en formato shape SHP o MIF, y que automáticamente despliega la cartografía en el navegador de cliente, dicho navegador tiene que ser compatible con Java. Por lo

tanto, no es necesario un servidor dedicado que procese la cartografía, sino que ésta se envía de forma automática al cliente y es en su máquina local donde se despliega y procesa por medio del *applet*. Se puede definir la apariencia del *applet* mediante un archivo esquema, además es posible crear y agregar componentes y controles al *applet* logrando así personalizarlo con un ambiente más amigable.

Esta versión es la más sencilla para añadir funcionalidades cartográficas a un sitio *web*, pues prácticamente no requiere programación alguna y el aspecto final conseguido es más que aceptable. El inconveniente está en que necesariamente hay que servir la totalidad de cada mapa a cada cliente, y no se puede seleccionar servir sólo las zonas en las que cada cliente está interesado; esto puede disparar el tráfico y consumir gran ancho de banda.

La versión del lado del servidor utiliza un Sistema Gestor de Bases de Datos (SGBD) que tenga implementada la *Simple Features Specification* (SFS) para SQL del OGC (como por ejemplo, PostGIS con PostgreSQL o las últimas versiones de MySQL) para guardar las características de los mapas y así tener un depósito avanzado para reducir el tráfico de la red; es decir, sobre este SGBD se almacenan los datos y el aplicativo se encarga de gestionar la cartografía a visualizar. Debido a que el lado del servidor está basado en los *servlets* de Java se puede integrar con cualquier servidor *web*.

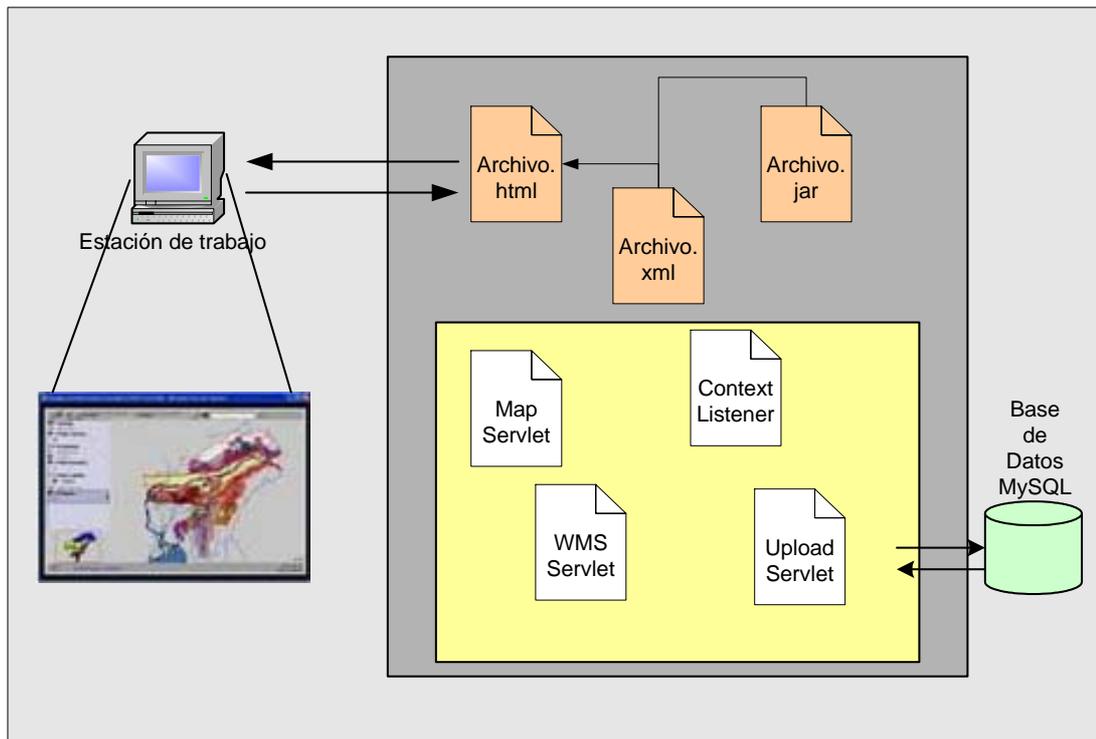
En el lado del servidor ALOV Map está implementado mediante los *servlets* de Java y filtros (programa que acepta un cierto tipo de datos como *input* (entrada), los transforma de alguna manera, y luego expide los datos transformados), estos dos últimos corren en el servidor propio de ALOV Map bajo Tomcat 4.x, pero el usuario puede emplear cualquier otro servidor que sea compatible con *Servlet 2.2*

La figura 1 ilustra la estructura que comprende el sistema de ALOV Map, el servidor web Java Tomcat y la base de datos relacional MySQL.

Los datos del mapa a transferir son controlados por un conjunto de *servlets* en el servidor web Tomcat y un *applet* del mapa en el lado del cliente en una petición vía HTTP que éste realiza. Los *servlets* que están disponibles en el contenedor son:

*MapServlet* que es el *servlet* principal, *ContextListener* el cual permite recargar todos los objetos en el servidor, *WMServlet* que implementa el *Web Map Service* y; *UploadServlet* sube los archivos shp y dbf a la base de datos SQL y crea los metadatos y registra los *datasets*.

Figura 1: GIS Internet con ambiente Java Cliente/Servidor



<http://www.gisdevelopment.net/technology/gis/ma03230pf.htm> Título: Implementing Internet GIS with Java based Client/Server Environment

### Implementación del sistema ALOV Map

A continuación se listan de manera sencilla los pasos para construir una aplicación para Internet basada en ALOV Map.

1. Crear las capas del mapa en formato shp y su respectiva tabla de atributos (dbf)
2. Crear la base de datos en MySQL
3. Crear la carpeta para los mapas en el directorio Webapps de Tomcat y colocar los archivos requeridos como las clases de Java que ALOV Map utilizará.

4. Llevar los datos de los mapas (archivos shp y dbf) a la base de datos de mapas en MySQL utilizando *UploadServlet*. Este crea un archivo XML automáticamente.
5. Editar el archivo XML con los parámetros requeridos.
6. Realizar las pruebas respectivas.

### **GeoServNet** <sup>[BA03], [BA04], [BA05], [BA06] y [BA07]</sup>

GeoServNet es un producto desarrollado por el Laboratorio *York Geospatial Information and Communication Technology* (GeoICT) que se estableció en el año 2001 para: el avance y desarrollo de información geoespacial, tecnologías de comunicación e integración de la geomática con las tecnologías ICT para aplicaciones espaciales. El Laboratorio GeoICT fue fundado por el programa *Canada Research Chair*, la Fundación Canadá para la Innovación, el Consorcio de Innovación de Ontario y la Universidad York.

GeoServNet es un visualizador espacial de alto rendimiento, que distribuye información geoespacial en 2D y 3D sobre una red. GSN tiene una Arquitectura Orientada a Servicios (*Services Oriented Architecture SOA*), la cual es escalable, transparente y distribuida. Utiliza tecnología Java y Java 3D y puede ser desplegada con cualquier plataforma. Con GSN, la información geoespacial puede ser accedida en cualquier momento y desde cualquier lugar. Tiene una estructura completa para la visualización en 3D y un análisis de datos (imágenes, mapas, datos de terreno y modelos 3D), y puede exportar modelos 3D a una variedad de formatos.

GeoServNet es un conjunto completo de componentes que permite a las organizaciones la construcción sofisticada de sistemas basados en servicios web que soportan visualización 3D. El sistema está comprendido de servidores y clientes poderosos para soportar interoperabilidad en un rango amplio de ambientes de información.

El sistema GeoServNet es una solución completa con tecnologías de cliente, servidor y administración de datos. A continuación se detallan algunas ventajas que ofrece el uso de un sistema GeoServNet:

- Provee extrema rapidez en la manipulación de conjuntos muy grandes de datos espaciales sobre un amplio rango de redes de trabajo.
- Provee verdaderas capacidades de visualización en 3D y puede usar la información desde una amplia variedad de fuentes de datos heterogéneas, rápidamente.
- Basada en los estándares OpenGIS, GeoServNet es fácilmente integrada en redes existentes basadas en estos estándares. También, la tecnología de servidor permite a las organizaciones construir y extender la disponibilidad de los datos espaciales dentro de sus organizaciones.
- GeoServNet es altamente escalable y fiable, soporta agrupamientos de servidores y dispone de tecnologías para soportar aplicaciones de empresas por niveles.
- Las tecnologías de visualización de GeoServNet pueden ser desplegadas en una amplia variedad de sistemas operativos y plataformas, desde servidores y estaciones de trabajo, a computadores portátiles y PDA's, permitiendo una organización entera y altos beneficios de las capacidades de visualización espacial en 2D y 3D.

El cliente GSN 3D está desarrollado para ser compatible con OGC para el acceso de datos geoespaciales, su despliegue, procesamiento y análisis para SIG abierto. Ofrece al usuario final una eficiente visualización, consultas y análisis de información geoespacial.

El Cliente GSN 3D es una herramienta de visualización basada en los protocolos WMS y WCS de la OGC. Incluye dos partes principales: la primera, es una simple herramienta de navegación para 2D que provee una interface visual basada en el protocolo WMS de OGC para definir el área y *features* requeridos. El otro es el visualizador inteligente para 3D que direcciona la información geoespacial de un área o *feature* dado. El visualizador 3D además provee una interfaz para operaciones con 3D incluido el análisis de datos. Todos los datos necesarios son retribuidos vía las interfaces estandarizadas por las especificaciones OGC. El visualizador 3D puede ser transplantado e inicializado por cualquier otro visualizador 2D que sea compatible con las especificaciones OGC.

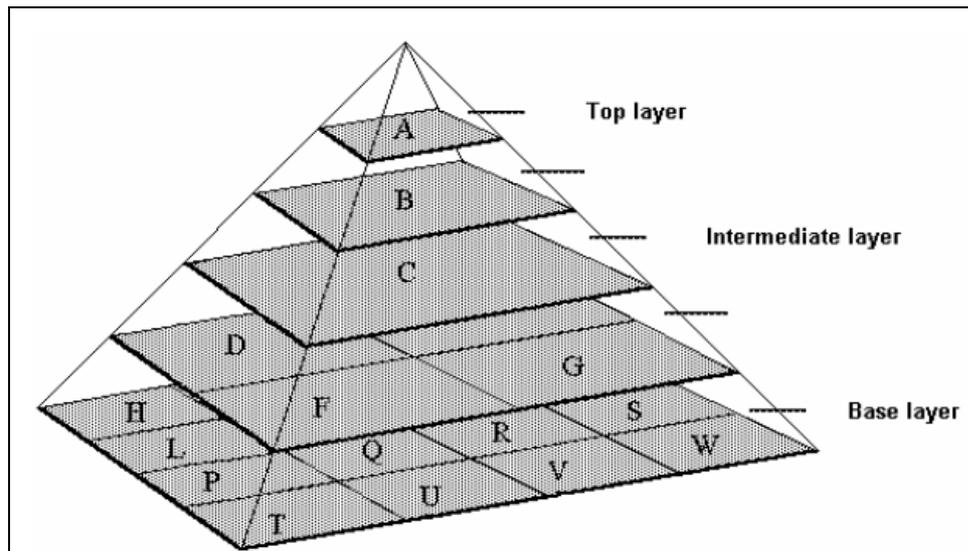
El cliente GSN 3D puede ser capaz de ejecutar las siguientes funciones:

- Operaciones con la vista en 3D: paneo, *roam*, alejamiento, acercamiento, rotar, volar a través de, traslado, etc.
- Operaciones de mejora con la vista en 3D: mostrar una capa, ocultar una capa, añadir o remover capas, etc.
- Operaciones de consulta: propiedades de las capas, etc.
- Operaciones de análisis: perfil, análisis de visibilidad, análisis de área visible y una simulación simple de inundación, etc.

### **Implementación del sistema GeoServNet**

Es necesario comprender el concepto de pirámide de imagen que GSN aplica para el manejo de las imágenes. La pirámide de imagen es una tecnología donde una imagen de alta resolución es creada y guardada en un formato de multi-resolución. Donde la resolución entera está en la base de la pirámide, y en la parte superior existe una menor resolución. La imagen es particionada para lograr este efecto.

Figura 2: Pirámide de Imagen



[http:// www.isprs.org/istanbul2004/comm7/papers/129.pdf](http://www.isprs.org/istanbul2004/comm7/papers/129.pdf), Título: Use Image Streaming Technologies To Present High Resolution Images On The Internet

Trabajando con la pirámide al lado del servidor, el cliente GSN será diseñado fácilmente. La solución para una ejecución alta y fácil es guardar una pirámide de imagen que sea igual a la pirámide que está en el lado del servidor.

GSN construye una pirámide de imagen al lado del servidor para imágenes con alta resolución. Es necesario notificar al cliente GSN de alguna información extra como: la resolución original de la imagen, el cuadro de propiedades de la imagen, el nivel superior e inferior de la pirámide. Al mismo tiempo, cada nivel de la pirámide de imagen necesita ser cortado en pedazos pequeños. En el sistema GSN, estos pedazos están representados por objetos de imágenes.

En segundo lugar, se necesita de una piscina de imágenes para almacenar las imágenes que son bajadas del servidor. Esta piscina actúa como el caché del navegador o del sistema de archivos de *Windows*. De hecho, este no es sólo un caché que sirve al cliente GSN, sino que es un servidor en el lado del cliente. En lugar de la comunicación con el servidor GSN, el cliente GSN envía la imagen pedida a esta piscina de imagen. La piscina de imagen transferirá y almacenará la imagen particionada al lado del cliente. Mientras el cliente pide imágenes, este buscará primero en la piscina de imagen, si la imagen está en la piscina, este enviará la

imagen particionada requerida al módulo de ejecución. De otra manera, éste reenviará la petición desde el módulo de ejecución al servidor si la imagen requerida no está disponible en la piscina.

Finalmente, un vistazo de imágenes necesita ser almacenado en el cliente GSN. Es importante generar un vistazo de imágenes de tal manera que los usuarios puedan familiarizarse con la gran escala de características de las imágenes. Un vistazo de imágenes puede ayudar a responder a la piscina de imágenes las peticiones de los usuarios, tan rápido como sea posible.

El cliente GSN está dividido en cuatro módulos y cada uno de ellos debe estar ligado. Los módulos corren concurrentemente pero sin interrupción y con una adecuada comunicación entre cada uno de ellos.

#### Módulo de ejecución

Este módulo tiene cuatro características: operaciones de respuesta para cada usuario desde teclado y ratón, dibujar los mapas o imágenes en la pantalla del usuario, todos los APIs relacionados con la presentación del mapa, y todos los APIs para el análisis del mapa.

#### Módulo de piscina de imágenes

Este módulo tiene tres características: procesa las peticiones del cliente, almacena las imágenes bajadas desde el servidor y el administrador de la piscina.

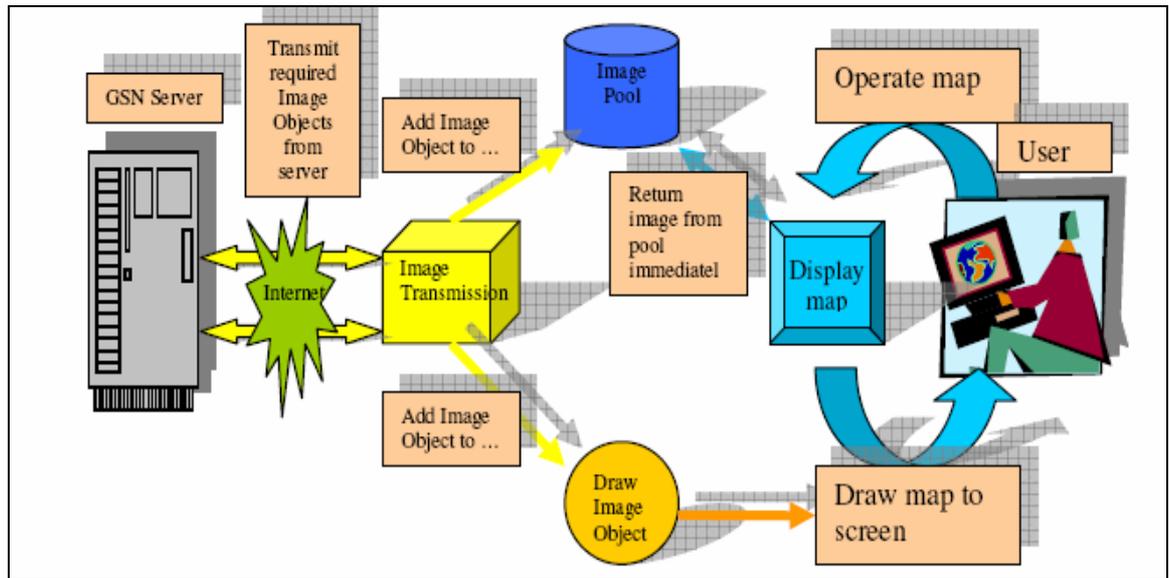
#### Módulo de dibujo del objeto imagen

Este dibujará el objeto imagen en la pantalla del usuario. Un objeto imagen hace referencia a los elementos de la pirámide de imagen.

#### Módulo de transmisión de imágenes

Este módulo comunica con el servidor GSN y consigue los objetos de imagen requeridos por el cliente GSN. Este puede transmitir objetos de imagen ya sea serial o concurrentemente. 30.

Figura 3: Estructura del Cliente GSN



[http:// www.isprs.org/istanbul2004/comm7/papers/129.pdf](http://www.isprs.org/istanbul2004/comm7/papers/129.pdf), Título: Use Image Streaming Technologies To Present High Resolution Images On The Internet

GeoServNet puede ser usado para una amplia serie de aplicaciones en organizaciones que trabajan con información espacial:

- Defensa, Inteligencia, Policía, y Seguridad de la Patria y Bienes raíces
- Facilidad y Dirección de Recursos
- Educación
- Manejo de desastres
- Planeamiento y valoración medioambiental
- Salud

### MapServer <sup>[BA08], [BA09] y [BA10]</sup>

MapServer fue desarrollado por la Universidad de Minnesota como herramienta para un proyecto de distribución de datos de gestión medioambiental por Internet. Actualmente, es mantenido por el proyecto *TerraSIP*, patrocinado por la NASA (*National Aeronautics and Space Administration*), en el que también trabaja la Universidad, y están involucrados más de veinte desarrolladores en su evolución.

MapServer no es un sistema SIG completo, ni pretende llegar a serlo. Es simplemente una herramienta que permite construir aplicaciones *web* interactivas que permitan la visualización y consulta de información geográfica en forma de mapas. Es una aplicación de carácter libre, distribuido bajo licencia GPL (*General Public License*). Está compuesto, entre otros, de paquetes de librerías gráficas y tipos de fuentes de código abierto como Shapelib, FreeType, Proj.4, GDAL/OGR; puede correr en múltiples plataformas como UNIX/Linux, Microsoft Windows e incluso en sistemas operativos para Mac.

La aplicación CGI (*Common Gateway Interface*) básica de MapServer provee varias características como:

- Formatos vectoriales soportados: ESRI shapefiles, PostGIS, ESRI ArcSDE y muchos otros vía la librería OGR.
- Formatos raster soportados: TIFF/GeoTIFF, EPPL7 y muchos otros a través de la librería GDAL.
- *quadtree* indexado espacial para archivos *shape*
- Completamente adaptable a las necesidades del usuario con plantillas de salida manejables.
- Selección de *feature* por item, valor, punto, área y otros aspectos.
- Soporte de varios tipos de fuentes.
- Elementos automáticos del mapa (barra de escala, referencia del mapa y leyenda).
- Construcción de mapas temáticos usando expresiones lógicas y regulares basadas en clases.
- Etiquetado de *features* incluyendo el control de colisión.
- Configuración de acceso rápida vía URLs
- Proyección rápida de mapas

Soporta varias especificaciones *web* establecidas por OGC: WMS (cliente/servidor), WFS no transaccional (cliente/servidor), WCS (servidor), WMC, SLD, GML y Codificación Filtrada.

Cualquier desarrollador puede crear sus propias aplicaciones utilizando el modelo de objetos de MapServer gracias a la biblioteca de componentes *MapScript*. El API de *MapScript* puede ser utilizado por lenguajes *script* como *PHP*, *Perl*, *Python* o incluso desde Java si se desarrollan los conectores JNI necesarios para acceder al API en C de *MapScript*.

MapServer crea imágenes de mapas desde la información espacial almacenada en formato digital. Puede representar más de veinte diferentes formatos de datos vectoriales. No toda la información desplegada en un mapa necesita estar en formato vectorial. Además, puede leer más de veinte formatos raster incluyendo *bitmaps* de *Windows*, GIF y JPEGs a través del paquete GDAL. A pesar de que MapServer entiende y puede representar estos tipos de raster, no hay manera de etiquetarlos con información espacial.

Puede operar en dos modos diferentes: CGI y *MapScript*. En el modo CGI, funciona en un ambiente de servidor web como un *script* CGI, es una manera fácil de preparar y producir una aplicación íntegra. En el modo *MapScript*, el API de MapServer es accesible desde *Perl*, *Python* o *PHP*. Además permite una aplicación flexible que puede tomar ventaja de las facilidades de plantilla de MapServer.

Este software está basado en plantillas. Cuando se ejecuta en respuesta a un requerimiento web, éste lee un archivo de configuración llamado *mapfile*, que describe las capas y otros componentes del mapa. Entonces, dibuja y guarda el mapa. Luego, lee uno o más archivos de plantilla de HTML que son identificados en el *mapfile*. Cada plantilla consiste en etiquetas convencionales de HTML y cadenas especiales sustituibles de MapServer. Estas cadenas son usadas, por ejemplo, para especificar las rutas de la imagen del mapa que ha sido creada por MapServer, para identificar qué capas están representadas, y para especificar el nivel de *zoom* y la dirección. Los valores actuales son sustituidos para estas cadenas y luego es enviado el flujo de datos al servidor web, el cual activa el mapa en el navegador. Cuando una petición cambia, y se hace clic en el botón *submit*, MapServer recibe la nueva petición desde el servidor web con los nuevos valores; entonces el ciclo comienza nuevamente.

MapServer ejecuta automáticamente varias tareas cuando genera un mapa. Etiqueta *features*, prevé la colisión entre etiquetas, provee el uso de fuentes *True Type*. El tamaño de las etiquetas puede ser ajustado o configurado a escala de acuerdo a la escala del mapa. Crea leyendas y barras de escala (configurables en el *mapfile*) y genera mapas de referencia. Un mapa de referencia muestra el contexto del mapa actualmente desplegado. Construye mapas mediante una pila de capas, es decir, una capa encima de otra. Cada capa despliega *features* seleccionados de un conjunto simple de datos. Los *features* a ser desplegados pueden ser seleccionados usando expresiones regulares de Unix, cadenas de comparación y expresiones lógicas. Por la similitud de datos y la similitud del estilo de parámetros (como escala, color y etiquetas), se puede pensar una capa como un tema. El despliegado de las capas está bajo control interactivo, permitiendo al usuario seleccionar la capa a ser representada. Mientras las capas no pueden ser generadas rápidamente, las capas vacías pueden ser pobladas con datos dinámicos y manipulados vía URL. Tiene capacidades de consulta poderosas y sofisticadas, pero en el modo CGI le falta las herramientas de análisis que provee un SIG verdadero.

Este resumen describe algunas características de MapServer y muestra por qué no es completamente un SIG: provee de herramientas de DBMS (*database management system*) no integradas, las capacidades analíticas son limitadas, no tiene herramientas para la georeferenciación.

### **Implementación del sistema MapServer**

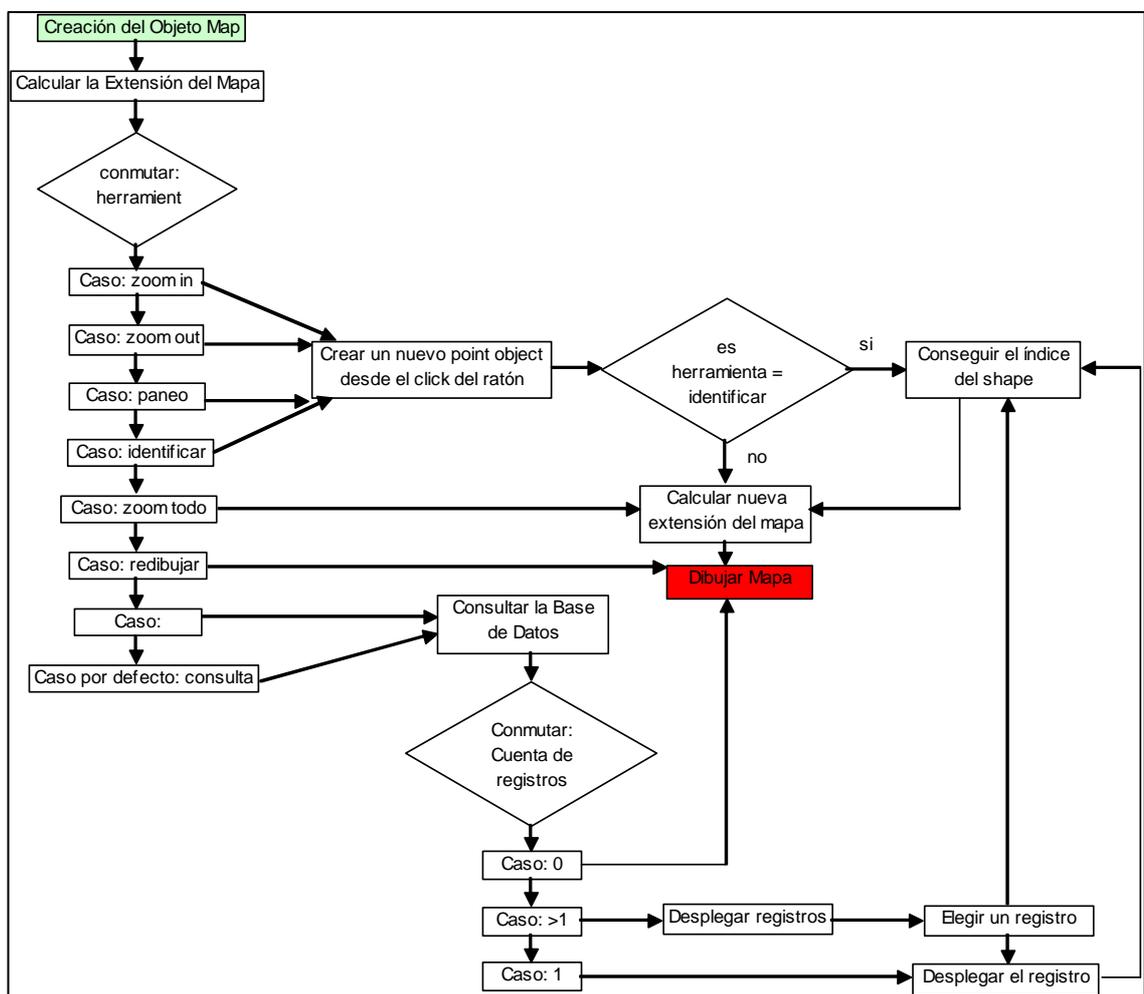
Las aplicaciones CGI con MapServer utilizan los siguientes recursos:

1. Un servidor HTTP como Apache o IIS (*Internet Information Server*),
2. El Programa MapServer,
3. Un archivo de inicialización que lance la primera vista de una aplicación con MapServer (opcional),
4. Un *Mapfile* que controle lo que MapServer hará con los datos,
5. Un archivo plantilla que controle la interfaz de usuario de la aplicación con MapServer en la ventana del explorador de Internet,
6. Un set de datos SIG.

MapServer normalmente se instala en el directorio cgi-bin del servidor HTTP, y sus archivos y los *sets* de datos están almacenados en el directorio de documentos de dicho servidor.

La lógica de una aplicación con este *software* se puede apreciar en el siguiente diagrama de flujo.

Figura 4: Lógica de una aplicación con MapServer



[http://mapserver.gis.umn.edu/doc/getstarted-howto\\_es.html](http://mapserver.gis.umn.edu/doc/getstarted-howto_es.html), Título: Comenzar Con Mapserver COMO

**Conclusiones:**

Una vez evaluadas las herramientas en estudio, hemos decidido que MapServer es la que más se ajusta a nuestros requerimientos por las características que ofrece: es multiplataforma, tiene mayor cantidad de software acoplable para incrementar su productividad, actualmente es el más desarrollado de los servidores de mapas de código abierto por tanto la información está más difundida siendo posible acceder a las experiencias de personas que han trabajado ya con este software. Al generar imágenes de la información requerida por el cliente, hace que la respuesta a la petición se haga en un menor tiempo sin ocupar tantos recursos del servidor lo que constituye una gran ventaja de este paquete.

Por el contrario, GeoServNet está considerado como un visualizador para cartografía digital, así que no se ajusta a las características buscadas. Sólo parte del software es libre y la información relacionada con este software es muy limitada y difícil de conseguir. Con AlovMap el inconveniente está en que necesariamente se debe enviar la totalidad del mapa al cliente y no sólo aquello que le interesa; lo que puede producir altos niveles de tráfico y consumir gran ancho de banda haciendo la transferencia de datos lenta y pesada.

## CAPÍTULO III

### Configuración, Compilación e Instalación de Mapserver

En este capítulo se describirán los componentes necesarios para desarrollar una aplicación con el software escogido, así como todas las librerías usadas por el mismo. Detallamos las formas de instalación de MapServer con cada una de sus librerías y con el paquete FGS.

#### 1.- Componentes de una aplicación con MapServer <sup>[CA01]</sup>

MapServer generalmente se ejecuta como una aplicación CGI en un Servidor HTTP. Esto será así a menos que se esté construyendo una aplicación más avanzada con MapScript, el cual accede directamente a la API de MapServer.

Las aplicaciones CGI con MapServer utilizan los siguientes recursos:

1. Un servidor HTTP como Apache o IIS (*Internet Information Server*),
2. El Programa MapServer,
3. Un archivo de inicialización que lance la primera vista de una aplicación con MapServer (opcional),
4. Un Mapfile que controle lo que MapServer hará con los datos,
5. Un archivo plantilla que controle la interfaz de usuario de la aplicación con MapServer en la ventana del explorador de internet,
6. Un set de datos SIG.

#### El Archivo de Inicialización

Este archivo puede ser parte de un archivo plantilla HTML, pero por simplicidad, también puede ser otro archivo, su extensión es .htm or .html. El archivo de inicialización utiliza un formulario para enviar una consulta inicial al servidor HTTP, que retorna un resultado desde MapServer, el que al ser dinámico, comienza y se

ejecuta cada vez que recibe una consulta, por lo tanto, dicho archivo sólo se requiere para pasar una serie de parámetros iniciales (ocultos) hacia la aplicación. Este pasaría los parámetros básicos requeridos por la aplicación con el CGI de MapServer.

### **El mapfile**

El mapfile define parámetros de los datos, el despliegue y las consultas que serán usados en una aplicación con MapServer; se puede hablar del *Mapfile* como un archivo de configuración de la aplicación que incluye información sobre cómo dibujar el mapa, la leyenda, y los mapas resultantes desde una consulta; normalmente tiene una extensión .map.

### **El archivo plantilla**

El archivo plantilla controla como saldrán los mapas y las leyendas desde MapServer hacia la página HTML; éste opera como cualquier otro archivo HTML excepto porque contiene celdas que pueden ser modificadas por el CGI de MapServer. El archivo plantilla permite colocar el mapa y la leyenda en una página, y determina la manera en que el usuario interactúa con MapServer (navegar, consultar, hacer *zoom*, etc.). MapServer usa el archivo plantilla y reemplaza las palabras clave de las celdas, con información de su estado actual o del estado del *set* de datos SIG, para generar el archivo HTML final que será enviado al explorador de Internet.

### **El set de datos SIG**

MapServer usa por defecto el formato vectorial *shape* de ESRI. Los datos raster pueden estar en diferentes formatos, dependiendo de como haya sido compilado MapServer. Por defecto, soporta archivos geoTiff y Tiff con archivos de georeferenciación (archivos *world*). Los archivos de datos deberían ser colocados en un directorio que esté referido en el *Mapfile*.

## **2.- Descripción de las librerías utilizadas por MapServer**

Para el correcto funcionamiento de MapServer es necesario configurar una serie de librerías tanto para una instalación básica como para una instalación con paquetes adicionales que fortalecen la utilidad de este *software*. A continuación se presentan las librerías mencionadas:

### **Librerías necesarias para una instalación básica de MapServer**

#### **GD**

Librería para la creación dinámica de imágenes. Crea imágenes PNG, JPEG y GIF entre otros formatos. GD está escrita en C y está disponible para Perl, PHP y otros lenguajes. Es usada para generar cuadros gráficos, gráficos, imágenes en miniatura. Las aplicaciones más comunes de GD, involucran el desarrollo de sitios *web*. <sup>[CA02]</sup>

#### **Freetype**

Es un *software* para fuentes que está diseñado para ser pequeño, eficiente, altamente personalizable, portable y capaz de producir resultados de alta calidad. Puede ser usado en librerías gráficas, servidores de visualización, herramientas de conversión de fuentes, herramientas generadoras de imágenes de texto, etc.

Es un servicio de fuentes y no provee APIs para ejecutar características de nivel superior, como diseño de texto o procesamiento gráfico (por ejemplo: color de texto, *rendering*, *hollowing*). Sin embargo, facilita estas tareas mediante una interfaz simple, fácil y uniforme para acceder al contenido de los archivos de fuentes independientemente de su formato. Provee una API para el fácil acceso a archivos de fuentes. Soporta formatos de fuentes escalables como TrueType.

Desde un modelo de salida dado, FreeType 2 es capaz de producir una alta calidad de *bitmap* monocromático, usando 256 niveles de colores de “gris”. Esto es mejor que los 5 niveles usados por *Windows 9x/98/NT/2000* o FreeType 1. Soporta el mapeo de caracteres definidos por *TrueType* y las especificaciones de *OpenType*.

FreeType 2 está escrito por la industria estándar ANSI C y debería compilarse en cualquier compilador C sin problemas. La librería no tiene ninguna dependencia externa y puede ser compilada e instalada por sí misma en cualquier clase de sistema. [CA03] y [CA04]

## **Libjpeg**

El paquete libjpeg contiene librerías que permiten la compresión de ficheros de imagen basándose en el estándar del *Joint Photographic Experts Group*. Es un algoritmo de compresión "con pérdidas".

Es una librería para programas de computación creada por el Grupo Independiente JPEG. Contiene funciones para manipulación de imágenes JPEG. Está escrito en su totalidad en el lenguaje de programación C. La versión actual de la librería es 6b, a pesar de que no ha sido actualizada desde 1998. [CA05]

## **Libpng**

PNG es un formato de compresión de gráficos sin patente desarrollado por Macromedia que se espera reemplace a GIF. PNG ofrece opciones avanzadas de gráfico, como el color de 48 bits.

Libpng es la librería oficial de referencia PNG (*Portable Network Graphics*) ya que soporta casi todas las características de éste formato, además la librería es expandible y ha sido probada ampliamente por más de nueve años. Esta librería requiere de la librería zlib 1.0.4 o superior. [CA06] y [CA07]

## **Zlib**

El paquete Zlib contiene la librería libz, utilizada por varios programas para realizar las funciones de compresión y descompresión de archivos.

Está diseñada para ser libre, de propósito general. Puede ser usada en cualquier computadora con cualquier hardware y sistema operativo. El formato de los datos zlib son portables entre varias plataformas. Al contrario del método de compresión LZW utilizado por Unix y en el formato de imagen GIF que expande los archivos al doble o el triple en casos extremos, el método utilizado por actualmente zlib esencialmente nunca expande los datos. El espacio en memoria ocupado por zlib es independiente de los datos de entrada y puede ser reducido, si fuese necesario, en la compresión. <sup>[CA08]</sup>

## **GDAL**

### *Geospatial Data Abstraction Library*

Es una librería de traducción de formatos de datos geospaciales raster. Está bajo una licencia de código abierto. Como una librería presenta un modelo de datos abstractos a la aplicación llamada para todos los formatos que soporte. <sup>[CA09]</sup>

## **PROJ4**

PROJ 4 es una librería para el manejo de conversión de proyección de la cartografía. Recomendado para el soporte WMS <sup>[CA10]</sup>

## **Libcurl**

Es una librería de transferencia URL del lado del cliente, soporta FTP, FTPS, HTTP, HTTPS, GOPHER, TELNET, DICT, FILE Y LDAP. Es requerida para el soporte WMS. <sup>[CA10]</sup>

## **Ming:**

Ming es una librería de C para generar SWF (*Shock Wave Flash*) el cual permite emplear formato de película para animaciones, además posee un conjunto de instrucciones para utilizar librerías de C++ y populares lenguajes *scripting* como PHP, Perl, Python. <sup>[CA11]</sup>

## Librerías adicionales

### Libmccrypt

MCrypt es un reemplazo para los viejos paquetes y comandos de encriptación, con extensiones. Permite a desarrolladores un amplio rango de funciones de encriptación, sin hacer cambios drásticos en sus códigos. Permite a los usuarios encriptar archivos o flujos de datos (*data streams*) sin ser criptógrafos. Permite tener código ordenado en el computador.

El acompañante de MCrypt es Libmccrypt, el cual contiene sus mismas funciones de encriptación, y provee un mecanismo estandarizado de acceso a dichas funciones. Libmccrypt es una librería que implementa todos los algoritmos y modos encontrados en mccrypt. Está actualmente en desarrollo. Al contrario de la mayoría de librerías de encriptación, libmccrypt no tiene funciones como: generación randómica de números, funciones *hash*, intercambio de clave, etc. Libmccrypt solo implementa una interfaz para acceder a los algoritmos de encriptación. <sup>[CA12]</sup> y <sup>[CA13]</sup>

### Libxml2

Es usada por una variedad de aplicaciones populares de código abierto, puede correr en diversos sistemas operativos sin problemas significativos. Esta librería permite manipular los archivos XML. Este incluye soporte para leer, modificar y escribir archivos XML y HTML. Tiene soporte DTDs que incluye un compilador y un validador para DTDs complejos, que comprueba el documento modificado ya sea en tiempo de compilación o luego. La salida puede ser un simple SAX (*Simple API for XML*) o un DOM (*Document Object Model*). En este caso se puede usar el constructor de implementaciones de XPath y XPointer para seleccionar subnodos o rangos. Está disponible un mecanismo flexible de entrada/salida, con módulos HTTP y FTP y combinados con una librería URI. No tiene ninguna dependencia extra. <sup>[CA14]</sup>

## **Libgeotiff**

Georeferencia imágenes Tiff (*Tagged Image File Format*). Librería para leer y escribir información de sistemas de coordenadas desde archivos GeoTiff. Incluye: archivos CSV (*comma-separated values*) para expandir códigos de sistemas de coordenada proyectados en proyecciones completas, definiciones y ejemplos de transformaciones en formas que puedan ser usados por la librería de proyección PROJ.4 <sup>[CA15]</sup>

## **Jasper**

Es un *kit* de herramientas de software para manipular datos de imágenes: representación, importación y exportación de éstas en numerosos formatos como JPEG-2000 JP2, JPEG, PNM, BMP, *Sun Rasterfile*, y PGX permite autodetección del formato de la imagen. <sup>[CA16]</sup>

## **Geos**

GEOS (*Geometry Engine – Open Source*) es un puerto de C++ de JTS (*Java Topology Suite*). Como tal, apunta a contener la funcionalidad completa de JTS en C++. Incluye todos los predicados de funciones espaciales de OpenGIS (características simples para SQL) y operadores espaciales, así como funciones específicas de la topología de JTS. <sup>[CA17]</sup>

## **Php MapScript**

El módulo PHP MapScript es un módulo de PHP que hace que las funciones y clases MapScript de MapServer estén disponibles para un ambiente PHP. <sup>[CA18]</sup>

## **Libtiff**

Libtiff es la librería sin costo y que trabaja en varios sistemas operativos; provee soporte para el formato TIFF (*Tag Image File Format*), el cual es un formato ampliamente usado para guardar datos de imágenes. <sup>[CA19]</sup>

## **BZIP2**

BZIP2 es una instrucción de Unix. Se trata de un programa que comprime y descomprime ficheros usando los algoritmos de compresión de *Burrows-Wheeler* y de codificación de *Huffman*. El porcentaje de compresión alcanzado depende del contenido del fichero a comprimir, pero por lo general es bastante mejor al de los compresores basados en el algoritmo LZ77/LZ78 tales como gzip, compress, winzip, pkzip, y otros. [CA20]

## **PostgreSQL**

PostgreSQL es un servidor de base de datos relacional libre, liberado bajo la licencia BSD. Es una alternativa a otros sistemas de bases de datos de código abierto (como MySQL, Firebird y MaxDB), así como sistemas propietarios como Oracle o DB2. [CA21]

## **TCL y TK:**

Tcl (*Tool Command Language*) es un lenguaje de programación interpretado y multiplataforma. Además es un lenguaje de comandos, cuyo intérprete recibe el nombre de tclsh (tclsh80 para Tcl/Tk 8.0), su principal característica es la facilidad para implementar funciones en C/C++.

Tk es el *ToolKit* que permite crear GUIs (Interface Gráfica de Usuario) en XWindow mediante Tcl. [CA22]

## **Librerías utilizadas por el instalador FGS** [CA23]

El instalador FGS (*Free GIS Suite*) es un archivo auto-extraíble que instala MapServer con PHP/MapScript y todas sus dependencias en un sistema Linux. Provee un ambiente autosuficiente con todos los requerimientos de software, incluyendo Apache y PHP, para correr aplicaciones de mapeo por Internet basadas en PHP/Mapscript. La versión de Apache que trae el instalador puede ser configurada para cualquier puerto, de manera que no interfiera con instalaciones previas de Apache u otros servidores que estén corriendo en el sistema.

El paquete básico FGS instala un ambiente *Web Server* pre-configurado que incluye los siguientes componentes:

- apache-base 1.3.33
- curl-lib 7.12.2
- expat-base 1.95.8
- freetype-lib 2.1.9
- gd-lib 2.0.33
- gdal-base 1.2.5
- jpeg-lib 6b
- libgeotiff-lib 1.2.2
- libiconv-base 1.9.1
- libpng-lib 1.2.6
- mapserver-base 4.4.1
- mapserver-php 4.4.1
- netcdf-lib 3.5.1
- openssl-lib 0.9.7d
- php-base 4.3.10
- postgresql-lib 7.4.6
- proj-lib 4.4.8
- proj4\_epsg42xxx-support 1.0.0
- tiff-lib 3.7.1
- unixODBC-base 2.2.10
- xerces\_c-base 2\_6\_0

A continuación se describen las librerías que no han sido mencionadas anteriormente y que constan en el paquete FGS.

### **Apache base**

Apache es un servidor *Web* poderoso, eficiente y disponible libremente. Es además, el servidor *web* más popular de Internet. <sup>[CA24]</sup>

### **Expat base**

Es una librería de compilación de XML escrita en C. Es un compilador orientado a flujo (*a stream-oriented parser*) en el cual una aplicación registra los encabezados que podrían ser encontrados por el compilador, en el documento XML. <sup>[CA25]</sup>

### **Libiconv**

Libiconv convierte un caracter codificado a otro a través de una conversión Unicode. Tiene soporte para transliteración, lo cual quiere decir que cuando un carácter no puede ser representado en el conjunto de caracteres deseado, es transformado a uno o

varios caracteres aproximadamente parecidos. Esto es útil cuando la aplicación necesita soporte para múltiples codificaciones de caracteres pero el sistema carece de las mismas. <sup>[CA26]</sup>

### **Netcdf-lib**

NetCDF (*Network Common Data Form*), es una interface de acceso de datos orientado a arreglos (vectores o matrices) y una librería que provee una interface de implementación. La librería netCDF además define una máquina independiente de formatos para representar datos científicos. La interfaz y la librería apoyan la creación, el acceso y el compartir datos científicos. El código fuente libre se puede conseguir en formato comprimido tar o zip. <sup>[CA27]</sup>

### **Openssl**

OpenSSL es una implementación de código abierto para los protocolos SSL (*Secure Sockets Layer*) y TLS (*Transport Layer Security*). La parte medular de la librería implementa funciones básicas de criptografía y provee varias funciones útiles. <sup>[CA28]</sup>

### **UnixODBC**

Es una solución *Open Database Connectivity* (ODBC) para Unix/Linux, completa, y abierta o libre. Los objetivos de este proyecto son desarrollar y promover unixODBC para ser un estándar definitivo para ODBC para las plataformas que no sean MS Windows. Incluye una interfaz gráfica para el usuario. <sup>[CA29]</sup>

### **Xerces**

Xerces es un conjunto de analizadores compatible con XML (*Extensible Markup Language*), dichos analizadores están disponibles para Java y C++. <sup>[CA30]</sup>

### **Mapserver-base**

Esta librería contiene el programa de MapServer ya descrito anteriormente.

## Mapserver-php

Esta librería contiene el módulo PHP con el cual MapServer puede ser ejecutado como PHP/MapScript descrito anteriormente

### 3.- Instalación de MapServer

Existen dos formas para la instalación de MapServer: la primera consiste en la instalación paso a paso de cada una de las librerías utilizadas por el software, y la segunda, se la realiza mediante la ejecución del paquete autoextraíble FGS el cual contiene un *script* de instalación con las librerías necesarias.

#### Instalación de MapServer-4.2.3 <sup>[CA31]</sup>

Se especificará los pasos a seguir para la instalación y configuración de los siguientes paquetes:

- j2sdk-1\_4\_2\_04-linux-i586-rpm.bin
- tiff-3.7.2.tar.gz
- libgeotiff-1.2.2.tar.gz
- ming-0.2a.tar.gz
- gd-2.0.28.tar.gz
- postgresql-7.4.5-2PGDG.i386.rpm
- geos-2.0.0.tar.bz2
- jasper-1.701.0.uuid.tar.gz
- gdal-1.2.3.tar.gz
- php-4.3.9.tar.bz2
- mapserver-4.2.3.tar.gz

Una vez instalado el sistema operativo Linux por defecto procedemos con los siguientes pasos:

#### **a) Instalación de j2sdk-1\_4\_2\_04-linux-i586-rpm.bin**

Dar permisos de ejecución

1.- `chmod +x j2sdk-1_4_2_04-linux-i586-rpm.bin`

2.- `./j2sdk-1_4_2_04-linux-i586-rpm.bin`

Aceptar el licenciamiento y después instalar

3.- `rpm -ivh j2sdk-1_4_2_04-linux-i586.rpm`

Por defecto este paquete se instala en `/usr/java/j2sdk1.4.2_04`, por lo cual tendremos que definir una variable global llamada `JAVA_HOME` que se dirija a `/usr/java/j2sdk1.4.2_04`, para esto editamos el `/etc/profile`, así cada usuario contará con esta variable.

4.- `vi /etc/profile`

Añadir estas líneas y modificar la sentencia `export` añadiendo la variable `JAVA_HOME`

Java

5.- `export JAVA_HOME=/usr/java/j2sdk1.4.2_04`

6.- `PATH=$PATH:${JAVA_HOME}/bin`

También ponemos disponibles las librerías de java editando el `/etc/ld.so.conf` y añadimos la siguiente línea al final

7.- `vi /etc/ld.so.conf`

8.- `/usr/java/j2sdk1.4.2_04/lib`

Comprobamos la configuración realizada

9.- `java -version`

10.- `ldconfig`

En caso de no haber realizado la instalación de Linux con todos sus paquetes será necesario agregar:

- freetype
- libpng
- libtiff
- zlib
- libjpeg

- unixODBC
- mx
- tk
- tcl
- libxml

**b) Instalación de freetype-2.1.9-1**

- freetype
- freetype-devel

freetype-2.1.9-1.i386.rpm

rpm -ivh freetype-devel-2.1.9-1.i386.rpm

**c) Instalación de libpng**

- libpng
- libpng-devel

libpng-1.2.7-1

rpm -ivh libpng-devel-1.2.7-1.i386.rpm

**d) Instalación de libtiff**

- libtiff
- libtiff-devel

libtiff-3.6.1-7

rpm -ivh libtiff-devel-3.6.1-7.i386.rpm

**e) Instalación de zlib**

- zlib
- zlib-devel

zlib-1.2.1.2-1.i386.rpm

zlib-devel-1.2.1.2-1.i386.rpm

**f) Instalación de libjpeg**

- libjpeg
- libjpeg-devel

libjpeg-6b-33

rpm -ivh libjpeg-devel-6b-33.i386.rpm

#### **g) Instalación de unixODBC**

- unixODBC
- unixODBC-devel

rpm -ivh unixODBC-2.2.9-1

rpm -ivh unixODBC-devel-2.2.9-1.i386.rpm

#### **h) Instalación de mx**

- mx

rpm -ivh mx-2.0.5-3.i386.rpm

#### **i) Instalación de bzip2**

- bzip2
- bzip2-devel

bzip2-1.0.2-13 i386.rpm

bzip2-devel-1.0.2-13 i386.rpm

#### **j) Instalación de libmcrypt**

- libmcrypt
- libmcrypt-devel

libmcrypt-2.5.7-3.i386.rpm

libmcrypt-devel-2.5.7-3.i386.rpm

#### **k) Instalación de tk**

- tk
- tk-devel

tk-8.4.7-2 i386.rpm

rpm -ivh tix-8.1.4.98.i386.rpm

rpm -ivh fontconfig-devel-2.2.3-7.i386.rpm

rpm -ivh xorg-x11-6.8.1-23.EL

rpm -ivh xorg-x11-devel-6.8.1-23.EL.i386.rpm

rpm -ivh tk-devel-8.4.7-2.i386.rpm

### **l) Instalación de tcl**

- tcl
- tcl-devel

tcl-8.4.7-2 i386.rpm

rpm -ivh tcl-devel-8.4.7-2.i386.rpm

### **m) Instalación libxml2-2.6.16-6**

- libxml2
- libxml2-devel

libxml2-2.6.16-6 i386.rpm

libxml2-devel-2.6.16-6 i386.rpm

**n)** Poner disponibles las librerías que se encuentran en /usr/local/lib /etc/ld.so.conf y añadimos la siguiente línea al final.

```
vi /etc/ld.so.conf
```

```
/usr/local/lib
```

```
ldconfig
```

### **o) Instalación de proj**

```
rpm -ivh proj-4.4.8-1.i386.rpm
```

### **p) Instalación de tiff**

- tiff

```
tar -zxvf tiff-3.7.2.tar.gz
```

```
cd tiff-3.7.2
```

```
./configure --prefix=/usr/local
```

```
make
```

```
make install
```

### **q) Instalación de libgeotiff**

```
tar -zxvf libgeotiff-1.2.2.tar.gz
```

```
cd libgeotiff-1.2.2/
```

```
./configure --prefix=/usr/local --with-libtiff=/usr/local
```

```
make
make install
```

### **r) Instalación de ming**

```
tar -zxvf ming-0.2a.tgz
cd ming-0.2a/
```

Editar el Makefile que se encuentra en ese directorio y modificar la variable PREFIX  
iv Makefile

```
PREFIX = /usr/local
```

Documentar la línea PREFIX=/usr

```
# PREFIX=/usr
```

Compilar e instalar

```
make
make static
make install
```

### **s) Instalación de gd**

```
tar -zxvf gd-2.0.28.tar.gz
cd gd-2.0.28/
./configure --prefix=/usr/local
make
make install
```

### **t) Instalación de PostgreSQL**

```
rpm -ivh postgresql-libs-7.4.6-1.RHEL4.2.i386.rpm
rpm -ivh postgresql-7.4.5-2PGDG.i686.rpm
tar -jxvf /opt/postgresql/src/postgresql-7.4.5.tar.bz2
tar -jxvf /opt/postgresql/src/postgresql-base-7.4.5.tar.bz2
tar -jxvf /opt/postgresql/src/postgresql-docs-7.4.5.tar.bz2
tar -jxvf /opt/postgresql/src/postgresql-opt-7.4.5.tar.bz2
tar -jxvf /opt/postgresql/src/postgresql-test-7.4.5.tar.bz2
Copiar este header, necesario para krb5
cp /usr/include/et/com_err.h /usr/include/
cd postgresql-7.4.5/
```

```
./configure --exec-prefix=/usr --bindir=/usr/bin --sbindir=/usr/sbin
--sysconfdir=/etc --datadir=/usr/share/postgresql --includedir=/usr/include
--libdir=/usr/lib --libexecdir=/usr/libexec --localstatedir=/var
--mandir=/usr/share/man --disable-rpath --with-perl --with-tcl
--with-tclconfig=/usr/lib --with-krb5 --with-tk --with-python --with-openssl
--with-pam --with-heimdal --enable-nls --enable-thread-safety
--with-docdir=/usr/share/doc
```

NOTA: "No hacer make ni make install"

#### **u) Instalación de geos**

```
tar -jxvf geos-2.0.0.tar.bz2
cd geos-2.0.0/
./configure --prefix=/usr/local
make
make install
ldconfig
```

#### **v) Instalación de Jasper**

```
tar -zxvf jasper-1.701.0.uuid.tar.gz
cd jasper-1.701.0.uuid/
./configure --prefix=/usr/local
make
make install
ldconfig
```

#### **w) Instalación de gdal**

```
tar -zxvf gdal-1.2.3.tar.gz
cd gdal-1.2.3/
./configure --prefix=/usr/local --with-libtiff=/usr/local
--with-libgeotiff=/usr/local --with-png --with-pg --with-ogr --with-odbc
--with-jasper=/usr/local --with-static-proj4=/usr
--with-geos=/usr/local/bin/geos-config --with-gif --with-threads
make
make ogr-all
```

make install

ldconfig

### **x) Instalación de PHP**

```
tar -jxvf php-4.3.9.tar.bz2
```

```
cd php-4.3.9/
```

```
./configure --enable-shared --with-regex=system --with-jpeg-dir=/usr  
--with-png-dir=/usr --with-zlib --with-gd --with-freetype-dir=/usr  
--enable-force-cgi-redirect --enable-pic --enable-dbase --with-mysql  
--with-openssl --with-ming --with-bz2 --enable-ftp  
--with-kerberos --with-java --with-mcrypt --with-ming=/usr --with-ncurses  
--with-tiff-dir=/usr --with-unixODBC=shared,/usr --enable-memory-limit  
--enable-bcmath --enable-shmop --enable-calendar --enable-dbx --enable-dio  
--enable-mcal --enable-mbstring=shared --enable-mbstr-enc-trans --enable-mbregex  
--enable-wddx --enable-fastcgi --enable-zend-multibyte --with-pear  
--with-config-file-path=/usr/share/php439 --prefix=/usr/share/php439
```

```
make
```

```
make install
```

```
cp /usr/share/php439/bin/php /var/www/cgi-bin/
```

```
cp php.ini-dist /usr/share/php439/php.ini
```

```
ln -s /usr/share/php439/php.ini /etc/php.ini
```

```
cd /usr/share/php439/
```

```
ln -s lib/php/extensions/no-debug-non-zts-20020429/ extensions
```

Añadir la ruta donde instalamos el PHP, a la variable de ambiente PATH en /etc/profile.

```
vi /etc/profile
```

Añadir las siguientes líneas al /etc/profile para PHP

```
export PHP_HOME=/usr/share/php439
```

```
PATH=$PATH:${PHP_HOME}/bin
```

Terminaremos de configurar PHP después de compilar MapServer

### **y) Instalación de MapServer**

```
tar -zxvf mapserver-4.2.3.tar.gz
```

```
cd mapserver-4.2.3/
```

```
./configure --without-tiff --with-eppl --with-jpeg=/usr --with-threads
```

```
--with-proj=/usr --with-gdal=/usr/local/bin/gdal-config
--with-ogr=/usr/local/bin/gdal-config --with-ming --with-php=/usr/share/php439
--with-gd=/usr/local --with-freetype --with-wmsclient --with-wfs
--with-wfscient --with-curl-config=/usr/bin/curl-config
--enable-debug --disable-ignore-missing-data --enable-runpath
make
cp -f mapserv scalebar legend /var/www/cgi-bin/
cp -f mapsript/php3/php_mapscript.so /usr/share/php439/extensions/
service httpd restart
```

### **z) Configuración de PHP para que funcione php\_mapscript**

Procedemos a configurar /etc/php.ini para que cargue al inicio la librería php\_mapscript.so, añadir en la parte de carga de los módulos la siguiente línea:

```
vi /etc/php.ini
extension=php_mapscript.so
```

Modificamos la variable extension\_dir para que sea la siguiente:

```
vi /etc/php.ini
extension_dir = "/usr/share/php439/extensions"
```

Configurar apache (httpd daemon) para que funcione con PHP como CGI

Nota: Añadir estas líneas en sus respectivos sitios dentro del archivo /etc/httpd/conf/httpd.conf

Debemos configurar el archivo /etc/httpd/conf/httpd.conf para añadir las siguientes líneas:

```
vi /etc/httpd/conf/httpd.conf
    AddType application/x-httpd-php-cgi .php .php4 .phtml
    Action application/x-httpd-php-cgi /cgi-bin/php
```

Luego reiniciamos el servicio de apache (httpd)

```
service httpd restart
```

Crear una página php con lo siguiente

```
vi /var/www/html/phpinfo.php
<?php
    phpinfo();
?>
```

chequear la ruta en el servidor <http://localhost/phpinfo.php> y verificar el soporte de `php_mapscript`

### **Instalación de MapServer 4.6.2 con el paquete FGS** <sup>[CA32]</sup>

1. Descargar y ejecutar el archivo autoextraíble en cualquier directorio del sitio *web*:  
[http://www.maptools.org/dl/fgs/self-installers/fgs-mapserver\\_phpmapscript\\_4.6.2-linux-i386.bin](http://www.maptools.org/dl/fgs/self-installers/fgs-mapserver_phpmapscript_4.6.2-linux-i386.bin)
2. Ingrese la ruta en la cual se instalará el paquete.
3. Especificar el puerto que utilizará para correr la aplicación
4. Instalación completa.

### **Conclusiones:**

Hemos presentado dos alternativas de instalación de MapServer, cada una de ellas debe ser escogida de acuerdo a las necesidades de cada usuario. El paquete FGS disponible para la plataforma Linux, nos permite realizar una instalación rápida y sencilla, con una configuración preestablecida. Para realizar una instalación personalizada el usuario deberá instalar el *software* paso a paso, ya que de esta forma es posible instalar y agregar librerías con configuraciones específicas. Además permite controlar características como: ampliar el número de capas que MapServer puede mostrar en la aplicación cuyo valor por defecto es de cien, escoger el módulo de base de datos a utilizar, la utilización de librerías para presentaciones dinámicas, entre otras opciones.

## CAPÍTULO IV

### Desarrollo de la Aplicación Modelo

En el presente capítulo se pretende dar a conocer a detalle la estructura y sintaxis de los archivos necesarios para desarrollar una aplicación con MapServer en modo CGI, se explicarán las opciones para consultas ofrecidas por el software, se presentarán algunos ejemplos del código empleado y de los resultados obtenidos.

Es necesario conocer que MapServer funciona de dos maneras: utilizando el modo CGI nos permite hacer uso de las funciones ya programadas y, accediendo a su API mediante varios lenguajes de programación es posible adicionar nuevas funciones para diversos requerimientos.

Una aplicación desarrollada con MapServer en modo CGI consta de cuatro componentes: el mapfile, el archivo de inicialización HTML, los archivos plantilla HTML y un conjunto de datos SIG.

En este modo, MapServer haciendo uso de sus componentes, funciona de la siguiente manera: el archivo de inicialización es un archivo convencional HTML que tiene la información de inicialización codificado dentro de las variables del formulario. Casi todos los valores utilizados por MapServer pueden ser seteados en este archivo. Cuando este archivo es invocado por el navegador, una variable del formulario es usada para especificar el nombre de un mapfile (.map). El archivo lee el mapfile para localizar fuentes, símbolos, plantillas y datos espaciales. El mapfile además especifica el tamaño del mapa resultante, su extensión geográfica, y el formato del mapa: GIF, JPEG, o PNG. Una vez leído el mapfile, MapServer despliega en el navegador una o más imágenes: el mapa, la leyenda, barra de escala, una referencia del mapa, entre otras. El software guarda las imágenes generadas en un directorio especificado en el mapfile.

MapServer necesita poner el mapa y sus elementos asociados en un formato de página web. El programa no crea por sí solo el archivo HTML, más bien, examina una plantilla HTML para la sustitución de cadenas. Mediante la sustitución de cadenas, las

palabras clave propias de MapServer pueden tomar valores como: rutas de archivos de referencia, detalles de la geometría del mapa, especificaciones de los *layers*, factores de *zoom*, etc. Además pueden tomar valores actuales de las variables CGI tales como: tamaño de la imagen, nombre del *mapfile*, extensión del mapa, etc. Una vez realizada la sustitución de cadenas con los valores apropiados, MapServer retorna un HTML modificado como respuesta a la petición del navegador.

Para un mejor entendimiento de la aplicación, dividiremos este capítulo en dos secciones. La primera indicará la sintaxis necesaria para un ejemplo básico de cómo desplegar los mapas con MapServer; y la siguiente, incluirá formas de consultas para la aplicación desarrollada.

### 1.- Estructura de los componentes de una aplicación básica con MapServer

A continuación se detalla la estructura y sintaxis empleada en los diferentes archivos utilizados para desplegar el mapa de Pendientes en Porcentaje de la Cuenca del Río Paute, el mismo que consta de *layers* definidos en el sistema de coordenadas WGS84 y sus escalas se muestran en la Tabla 1:

Tabla 1: Escalas de *layers* del mapa de Pendientes en Porcentaje de la Cuenca del Río Paute

Layer	Escala
Iluminacion.tif	1 : 25 000
p10r62_2001.tif	1 : 25 000
pendiente_por.tif	1 : 25 000
Limite_CRP_25MIL_WGS84_MDT.shp	1 : 25 000
Cantones_CRP_250MIL_WGS84.shp	1 : 250 000
Rios_Principales_CRP_25MIL_WGS84.shp	1 : 25 000
Ciudades_CRP_25MIL_WGS84.shp	1 : 25 000
Localidades_CRP_25MIL_WGS84.shp	1 : 25 000

#### MapFile

El *mapfile* es un archivo creado con un editor de texto y almacenado con la extensión *.map*, que define una colección de objetos de mapeo como: barra de escala, leyenda, colores, nombre del mapa, nombre de los *layers*, etc., que juntos determinan la

aparición y comportamiento del mapa que se desplegará en el navegador. Un *mapfile* tiene una estructura jerárquica que será descrita más adelante. La definición del *mapfile* consiste en valores asignados a palabras clave propias de MapServer.

Algunos valores son listas de ítems separados por espacios en blanco, que deben estar encerradas por comillas simples o dobles. MapServer no es sensible a caracteres en mayúscula o minúscula pero algunos métodos de acceso a base de datos lo son.

Existe un máximo de 100 *layers* por *mapfile*. Este valor puede ser cambiado editando el valor *MS\_MAXLAYERS* del archivo *map.h* y recompilándolo.

Las rutas de los archivos pueden ser absolutas o relativas para la localización del *mapfile*.

Para marcar una línea de comentario se utiliza el signo de número (#)

Los atributos son nombrados utilizando la siguiente sintaxis: [ATRIBUTENAME], el nombre del atributo incluido entre los corchetes es sensible a mayúsculas y minúsculas.

### **Conceptos del archivo *mapfile***

El *mapfile* consta de tres objetos principales o niveles de jerarquía:

- *Map Object*.- Que tiene las características generales del mapa,
- *Layer Object*.- Para describir las características de cada *layer* (selección de *features* dibujados a una misma escala, provenientes de un conjunto de datos espaciales simple) que se desplegará en el mapa; y
- *Class Object*.- Los *features* son clasificados de acuerdo con algún criterio y deben ser dibujados de diferente manera según su respectivo criterio. Una clase es el conjunto de *features* que corresponde a un mismo criterio. MapServer usa el objeto *class* para seleccionar los *features* y la forma de desplegarlos. Un *layer* necesita al menos una clase pero si no está definido un criterio, todos los *features* de ese *layer* son incluidos en una clase por defecto.

Cada objeto tiene sus propias etiquetas o palabras clave, es por eso que hablaremos de etiquetas a nivel de mapa, a nivel de *layer* y, a nivel de clase.

### Sintaxis del archivo *mapfile*

En el nivel más alto de la jerarquía se encuentra el objeto *MAP*, el resto de objetos pertenecen a éste. La declaración de un objeto comienza con su nombre y termina con la palabra “*END*”.

Dentro de las etiquetas a nivel del *Map Object* tenemos:

MAP

```
STATUS ON
EXTENT 682225.979526 9638766.160761 782783.083523 9746164.346537
SIZE 640 480
UNITS METERS
SHAPEPATH "pendientes_porcentaje"
IMAGECOLOR 189 202 222
IMAGETYPE PNG
SYMBOLSET "/opt/fgs/www/htdocs/pendientes_porcentaje/symbols/symbols35.sym"
FONTSET "/opt/fgs/www/htdocs/pendientes_porcentaje/fonts/fonts.list"
```

*MAP*: Es el inicio del *mapfile*, indica la declaración del *Map Object*.

MAP

Luego de esta etiqueta, se detallan varias características del mapa como:

*EXTENT*: es la extensión del mapa que está definido por sus coordenadas: esquina inferior izquierda y la esquina superior derecha. Es necesario utilizar las mismas unidades especificadas con el parámetro *UNITS*. El parámetro *extent* puede obtenerse con herramientas como ArcView.

```
#extensión de la capa Límite_CRP_25MIL_WGS84_MDT
EXTENT 682225.979526 9638766.160761 782783.083523 9746164.346537
```

*SIZE*: es el ancho y la altura en píxeles de la imagen del mapa que MapServer genera. En este caso, el mapa tiene 640 píxeles de ancho por 480 píxeles de alto.

```
SIZE 640 480
```

*UNITS*: especifica las unidades de distancia del mapa. Este parámetro afecta al cálculo de la escala y a la barra de escala. Puede estar en metros (*meters*), millas (*miles*), grados decimales (*dd decimal degrees*).

```
UNITS METERS
```

*SHAPEPATH*: Indica la ruta absoluta o relativa del directorio en el cual se encuentra depositada la información de los archivos shape. Este valor es preasignado al conjunto de datos especificado en cada *layer* por el parámetro *DATA*.

```
SHAPEPATH "pendientes_porcentaje"
```

*IMAGECOLOR*: Es el color del fondo del mapa. Los valores están en el formato RGB (*Red, Green, Blue*), puede ir en un rango de 0 a 255.

```
IMAGECOLOR 189 202 222111
```

*IMAGETYPE*: Define el formato para la imagen de salida. En este caso utilizamos PNG. Podría ser GIF, si la librería GD fue compilada con soporte para GIF, WBMP, JPEP, entre otros.

```
IMAGETYPE PNG
```

*SYMBOLSET*: Indica la ruta del archivo de definición de símbolos utilizados en el mapa.

```
SYMBOLSET "/opt/fgs/www/htdocs/pendientes_porcentaje/symbols/symbols35.sym"
```

*FONTSET*: Especifica la ruta al archivo que contiene las fuentes que serán utilizadas para etiquetar los *layers*.

```
FONTSET "/opt/fgs/www/htdocs/pendientes_porcentaje/fonts/fonts.list"
```

MapServer no tiene forma de cambiar el tamaño por defecto de las líneas que es un píxel de ancho. Mas, usando símbolos escalables es posible mediante la etiqueta *SIZE* a nivel de clase, manipular el tamaño de los símbolos. Un símbolo está definido a nivel de

mapa, está disponible para todas las clases y en todos los *layers*. Existen símbolos como: vector, elipse, pixmap y *truetype*.

- Vector: Es una serie de puntos que describen el símbolo de la línea de salida.
- Elipse: Usa la misma sintaxis que el símbolo vector pero es interpretado de manera diferente.
- Pixmap: Usa imágenes PNG o GIF como símbolos.
- *Truetype*: Usa caracteres de las fuentes *TrueType* como símbolos.

```
#Símbolo para dibujar líneas gruesas
SYMBOL
  NAME "BigLine"
  TYPE ELLIPSE
  POINTS 1 1 END
END

#Símbolo para dibujar puntos
SYMBOL
  NAME "Circle"
  FILLED true
  TYPE ellipse
  POINTS 1 1 END
END
```

*SYMBOL*: Inicio de la etiqueta symbol

*NAME*: Especifica el nombre del símbolo

*FILLED*: Indica si se rellena el símbolo (*true*) o no (*off*)

*TYPE*: Especifica el tipo de símbolo a usar.

*POINTS*: Especifica una serie de pares de coordenadas que representa los vértices del símbolo. En nuestro caso, indica  $x_1=1$ ,  $y_1=1$  hasta el final.

*END*: Fin de la etiqueta *symbol*

Las siguientes líneas describen el *Web Object* que especifica la interfaz web, incluyendo rutas, URLs, archivos plantilla, y otros detalles que afectan la manera en que la aplicación responde a las peticiones del usuario.

```

WEB
  TEMPLATE
  '/opt/fgs/www/htdocs/pendientes_porcentaje/pendientes.html '
  IMAGEPATH '/opt/fgs/www/htdocs/pendientes_porcentaje/data/tmp/ '
  IMAGEURL  '/pendientes_porcentaje/data/tmp/ '
END

```

*TEMPLATE*: Especifica la ubicación física de la plantilla HTML principal utilizada para desplegar el mapa.

```

TEMPLATE  '/opt/fgs/www/htdocs/pendientes_porcentaje/pendientes.html '

```

*IMAGEPATH*: Especifica la ruta al directorio en dónde las imágenes y los archivos temporales son escritos.

```

IMAGEPATH '/opt/fgs/www/htdocs/pendientes_porcentaje/data/tmp/ '

```

*IMAGEURL*: Define el URL base que apunta al directorio en dónde se generan las imágenes escritas.

```

IMAGEURL  '/opt/fgs/www/htdocs/pendientes_porcentaje/data/tmp/ '

```

Inicio de *Projection Object* que especifica la proyección utilizada para desplegar o describir los datos espaciales. La cadena que va entre comillas. En nuestro caso utilizamos el código EPSG definido por *European Petroleum Survey Group (EPSG)*.

EPSG:4326 - WGS84 (World Geodesic Datum)<sup>[DA01]</sup>

```

PROJECTION
  "init=epsg:4326"
END

```

Inicio de *Reference Map Object*. Un mapa de referencia es una imagen del mapa en formato GIF, JPEG, PNG, etc.; que muestra la extensión inicial del mapa mientras va identificando con un recuadrado la zona que el usuario está explorando. En la Figura 1 se presenta un ejemplo de un mapa de referencia.

```

REFERENCE
  IMAGE
  '/opt/fgs/www/htdocs/pendientes_porcentaje/graficos/limite_referencia.jpg'

```

```
SIZE 345 369
EXTENT 682225.979526 9638766.160761 782783.083523 9746164.346537
STATUS ON
COLOR -1 -1 -1
OUTLINECOLOR 255 0 0
END
```

Se detallan las líneas dentro de éste objeto:

*IMAGE*: Especifica la ruta al archivo que contiene la imagen de referencia.

```
IMAGE
'/opt/fgs/www/htdocs/pendientes_porcentaje/graficos/limite_referencia.jpg'
```

*SIZE*: Define el ancho y el alto del mapa de referencia en píxeles. En nuestro caso la imagen tiene 345 píxeles de ancho por 369 píxeles de alto.

```
SIZE 345 369
```

*EXTENT*: Especifica la extensión espacial del mapa de referencia en las unidades del mapa.

```
EXTENT 682225.979526 9638766.160761 782783.083523 9746164.346537
```

*STATUS*: Esta etiqueta puede tener los valores *on*, *off* o *embed*. *On* indica que el mapa va a ser generado, de lo contrario se utiliza *off*, y *embed* genera la imagen de referencia y la embebe dentro del mapa.

```
STATUS ON
```

*COLOR*: Especifica el color utilizado para dibujar el cuadro de referencia. Se utiliza el código RGB para su definición, los números negativos indican transparencia.

```
COLOR -1 -1 -1
```

*OUTLINECOLOR*: Especifica el color de la línea del borde del cuadro de referencia. Se utiliza el código RGB que va en rangos de 0 a 255.

```
OUTLINECOLOR 255 0 0
```

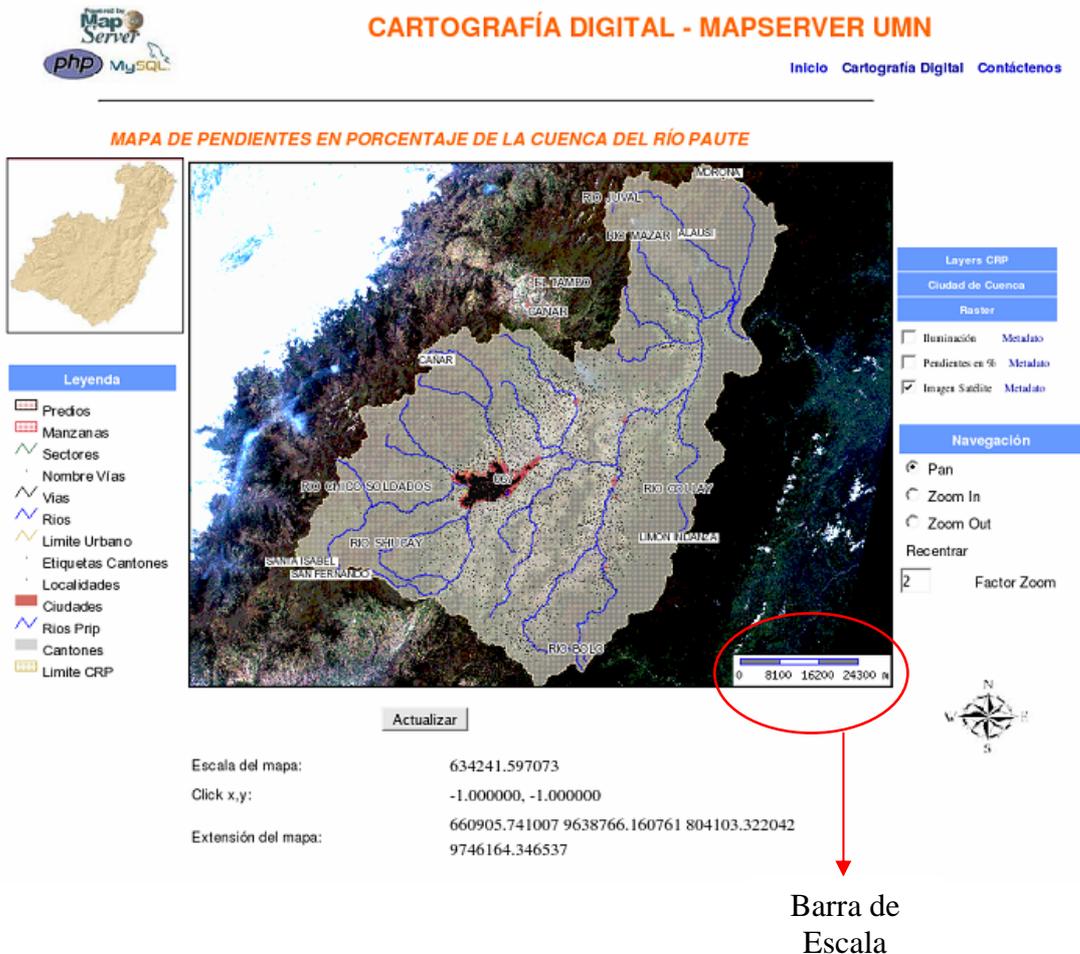


```

SIZE 144 5
STYLE 0
UNITS meters
BACKGROUNDCOLOR 255 255 255
COLOR 128 128 128
OUTLINECOLOR 0 0 255
TRANSPARENT off
END

```

Figura 2: Barra de Escala



**LABEL:** Marca el inicio del *Label Object*, define la cadena o símbolo de texto utilizados para etiquetar un *feature*.

LABEL

**COLOR:** Especifica el color usado para desplegar las etiquetas de texto. Usa el código RGB.

COLOR 0 0 0

*ANTI\_ALIAS*: Indica que para el etiquetado se utilizarán símbolos de texto. Puede tener el valor de true o false.

`ANTI_ALIAS true`

*SIZE*: Especifica el tamaño de la etiqueta de texto. El tamaño de las fuentes *TrueType* es especificado en píxeles con un valor entero y el de los símbolos de texto, es especificado por valores como *tiny*, *small*, *medium*, *large* o *giant*.

`SIZE small`

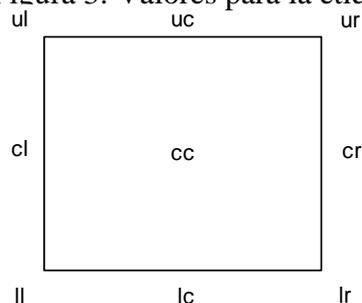
*POSITION*: Especifica la posición de la etiqueta con respecto a un punto de la etiqueta. Los puntos y polígonos pueden utilizar las ocho posiciones detalladas en la Tabla 2, pero no la posición de centrado, cc. Las líneas pueden usar solo uc o lc. Si este parámetro está puesto en auto, entonces MapServer se encarga de desplegar las etiquetas cuidando que no se produzca colisión. Por defecto este valor está en lc.

Tabla 2: Valores para la etiqueta *position*

Abreviatura	Palabra	Significado
ul	upper left	superior izquierdo
uc	upper center	superior centrado
ur	upper right	superior derecho
cl	center left	centrado izquierdo
cc	center center	centrado
cr	center right	centrado derecho
ll	lower left	inferior izquierdo
lc	lower center	inferior centrado
lr	lower right	inferior derecho

Para facilitar el entendimiento de estas posiciones, se presenta la Figura 3

Figura 3: Valores para la etiqueta *position*



*END*: Final del objeto *label*

END

*POSITION*: Especifica la posición en la imagen del mapa de una barra de escala embebida. Puede recibir valores como: ul, uc, ur, ll, lc, lr.

POSITION lr

*INTERVALS*: Especifica el número de intervalos mostrados en la barra de escala.

INTERVALS 3

*STATUS*: Especifica si la imagen de la barra de escala será creada (on) o no (off), o embebida a la imagen del mapa.

STATUS embed

*SIZE*: Especifica el tamaño en píxeles de la barra de escala, no incluye las etiquetas.

SIZE 144 5

*STYLE*: Especifica el estilo de la barra de escala. Puede tomar los valores 0 ó 1.

STYLE 0

*UNITS*: Especifica las unidades de la barra de escala. Si las unidades de la barra de escala y las del mapa son diferentes, la conversión se hace automáticamente.

UNITS meters

*BACKGROUND**COLOR*: Especifica el valor del fondo de la barra de escala.

BACKGROUND

COLOR 255 255 255

*COLOR*: Especifica el color del frente de la barra de escala.

COLOR 128 128 128

*OUTLINECOLOR*: Especifica el color usado en las líneas de intervalo.

```
OUTLINECOLOR 0 0 255
```

*TRANSPARENT*: Hace transparente el color de la imagen de la barra de escala.

```
TRANSPARENT off
```

*END*: Fin del objeto de la barra de escala.

```
END
```

Inicio del *Legend Object*. Los elementos de esta etiqueta, determinan la apariencia y localización de la leyenda del mapa. Solo las clases con nombre (etiqueta *NAME*) son incorporadas en la leyenda. Está asociada con los símbolos establecidos. En la Figura 4 se presenta un ejemplo de leyenda en un mapa.

```
LEGEND
  STATUS on
  IMAGECOLOR 230 230 230
  LABEL
    TYPE truetype
    FONT "arial"
    COLOR 0 0 0
    SIZE 10
    ANTIALIAS true
  END
END
```

*LEGEND*: Inicio del objeto leyenda

```
LEGEND
```

*STATUS*: Especifica si la imagen de una leyenda será creado (*on*) o no (*off*) o embebida dentro de la imagen del mapa.

```
STATUS on
```

*IMAGECOLOR*: Especifica el color del fondo de la imagen de la leyenda.

```
IMAGECOLOR 230 230 230
```

A este nivel, aparecen nuevas etiquetas utilizadas dentro del objeto *LABEL*. Sólo se describirán aquellas que no se han mencionado con anterioridad.

*TYPE*: Permite escoger si se utilizarán fuentes *TrueType* o símbolos de texto bitmapped.

TYPE truetype

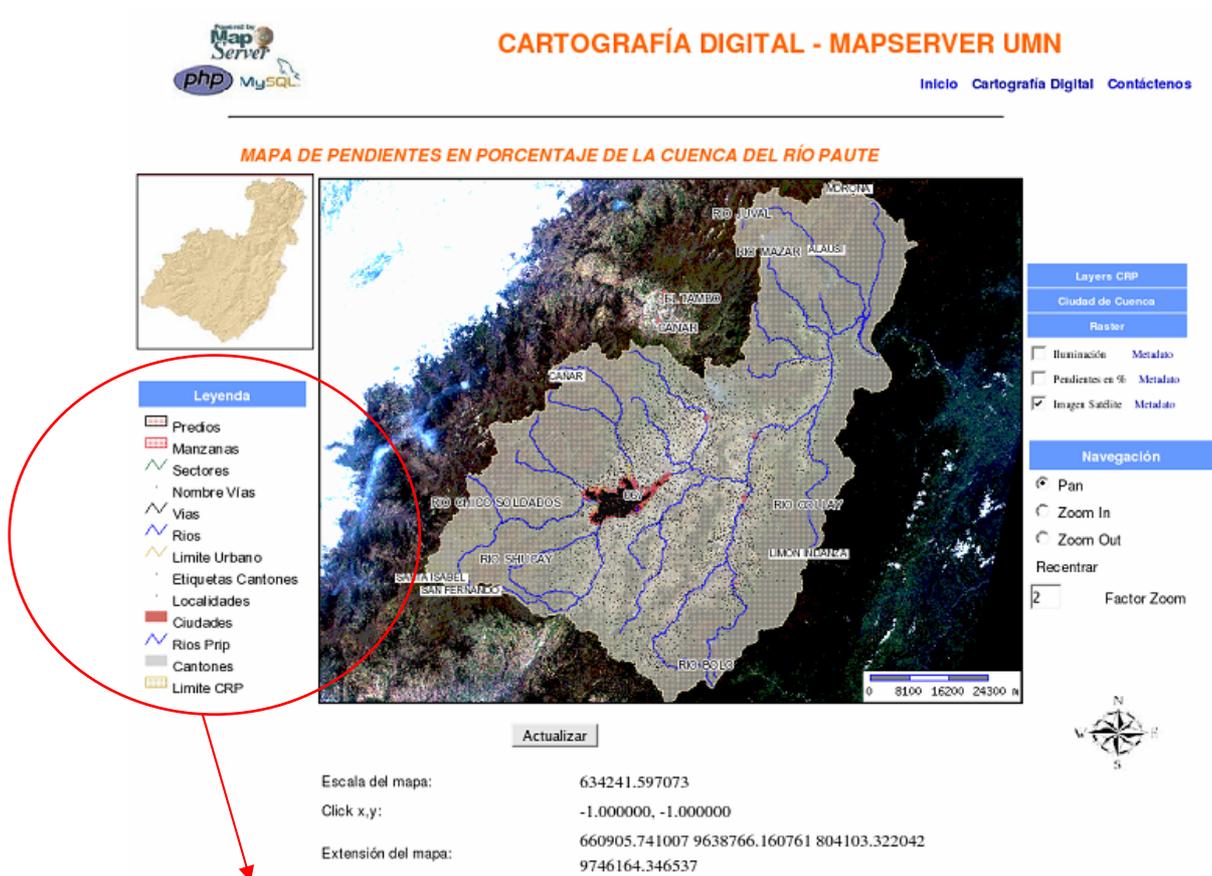
*FONT*: Especifica el alias de la fuente usada para etiquetar un *feature*.

FONT "arial"

*END*: Final del objeto de leyenda.

END

Figura 4: Leyenda



Barra de Escala

En las líneas siguientes se realiza la declaración de los *layers* que se presentarán en el mapa. Se utiliza el *LAYER OBJECT* y sus elementos para este propósito. Los *layers* son desplegados de un modo LIFO (*last in first out*)

El código que se presenta a continuación es utilizado para la definición de capas de tipo raster:

Layer *ilu\_25MIL* que tiene el raster de iluminación de la Cuenca del Río Paute.

```
LAYER
  NAME "iluminacion"
  DATA
"/opt/fgs/www/htdocs/pendientes_porcentaje/raster/ilu_25mil.tif"
  STATUS ON
  TYPE RASTER
  PROCESSING "BANDS=1,2,3"
  OFFSITE 71 74 65
  PROJECTION
    "init=epsg:4326"
  END
END
```

*LAYER*: Inicio del *Layer Object*

LAYER

*NAME*: Especifica el nombre del *layer*, puede tener como máximo 20 caracteres. Este nombre es usado como valor de una variable CGI para permitir que el *layer* pueda ser activado o desactivado interactivamente.

```
NAME "iluminacion"
```

*DATA*: Especifica la ruta al archivo shape relativo a la localización especificada por la etiqueta *SHAPEFILE* o relativo a la ubicación del *mapfile*. No se coloca la extensión del archivo shape (.shp), sólo el nombre. Cuando se trata de un raster, se hace referencia a un archivo .tif.

```
DATA "/opt/fgs/www/htdocs/pendientes_porcentaje/raster/ilu_25mil.tif"
```

*STATUS*: Especifica si un *layer* es desplegado o no (*on* u *off* respectivamente). Puede tomar el valor de *default*, lo que indica que el *layer* siempre se desplegará.

```
STATUS ON
```

*TYPE*: Especifica si el *layer* es raster o vectorial (*polygon, line, point, annotations*).

```
TYPE RASTER
```

*PROCESSING*: Especifica una directiva de proceso para un *layer*. El valor asignado a esta etiqueta, selecciona que bandas de una imagen multiespectral desplegar.

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

*OFFSITE*: Especifica el color de transparencia para los *layers* raster.

```
OFFSITE 71 74 65
```

*END*: Final de la declaración del *layer*.

```
END
```

*Layer pendiente\_por*: es una capa tipo raster que muestra un análisis de las pendientes en porcentaje de la Cuenca del Río Paute. Para este *layer* se ha utilizado un código similar al descrito en el *layer* *ilu\_25MIL*.

```
LAYER
  NAME "pendiente"
  DATA "/opt/fgs/www/htdocs/pendientes_porcentaje/raster/pendiente_por.tif"
  STATUS ON
  TYPE RASTER
  PROCESSING "BANDS=1,2,3"
  OFFSITE 71 74 65
  PROJECTION
    "init=epsg:4326"
  END
END
```

A continuación se presenta el código para trabajar con *layers* vectoriales.

*Layer Limite\_CRP\_25MIL\_WGS84\_MDT*: Determina la zona de la Cuenca del Río Paute. Se explicarán las etiquetas que no han sido mencionadas anteriormente.

```
LAYER
  NAME "limite"
  DATA "Limite_CRP_25MIL_WGS84_MDT"
  STATUS on
  TYPE polygon
  PROJECTION
```

```

    "init=epsg:4326"
  END
CLASS
  NAME "Limite CRP"
  STYLE
    SYMBOL "BigLine"
    COLOR      212 192 100
    OUTLINECOLOR 0 0 0
  END
END #fin de class
END #fin de layer

```

**CLASS:** Marca el inicio del *CLASS OBJECT*. Cada *layer* debe tener una o más clases.

**CLASS:** Inicio del objeto *class*.

```
CLASS
```

**NAME:** Especifica el nombre de la clase a usar en la leyenda. Si el nombre no es especificado, el *feature* se dibujará pero no será incluido en la leyenda.

```
NAME "Limite CRP"
```

**STYLE:** Marca el inicio del *STYLE OBJECT*. Este objeto determina cómo los símbolos serán desplegados. Una clase puede contener múltiples objetos de estilo, los cuales son aplicados en secuencia.

**STYLE:** Inicio del objeto de estilo.

```
STYLE
```

**SYMBOL:** Especifica el símbolo usado para dibujar *features*, identificado por nombre o número definidos en la definición de *SYMBOL* a nivel del mapa.

```
SYMBOL "BigLine"
```

**COLOR:** Especifica el color usado para dibujar los *features*. Se utiliza el código RGB.

```
COLOR 212 192 100
```

**OUTLINECOLOR:** Especifica el color de línea a dibujar, se utiliza sólo para polígonos.

```
OUTLINECOLOR 0 0 0
```

*Layer Rios\_Principales\_CRP\_25MIL\_WGS84*, contiene información sobre los ríos principales pertenecientes a la Cuenca del Río Paute.

```
LAYER
  NAME "rios"
  DATA "Rios_Principales_CRP_25MIL_WGS84" # nombre del shape
  STATUS on
  TYPE line
  LABELCACHE on
  LABELITEM "RIO" #campo del archivo .dbf por el cual se van a
  etiquetar los features
  PROJECTION
    "init=epsg:4326"
  END
  CLASS
    NAME "Rios Principales"
    LABEL
      TYPE truetype
      FONT "arial"
      SIZE 8
      OUTLINECOLOR 255 255 255
      COLOR 0 0 0
      POSITION auto
    END # of label
  END #end of class
END #end of layer
```

*LABELCACHE*: Cambia el cache de etiquetas a *on* u *off*. Si toma como valor *off*, las etiquetas son dibujadas cuando los *features* son dibujados. De otra manera, son almacenadas y dibujadas después que todos los *layers* hayan sido dibujados.

```
LABELCACHE on
```

*LABELITEM*: Especifica el campo del archivo .dbf por el cual se van a etiquetar los *features*. En la Figura 5 presentamos parte del archivo *Rios\_Principales\_CRP\_25MIL.dbf*, el atributo por el cual se etiquetarán los ríos es “RIO”.

```
LABELITEM "RIO"
```

Figura 5: Fragmento del archivo Rios\_Principales.dbf

	A	B
1	RIO	LONGITUD
2	PAUTE	10802,6
3	RIO PULPITO (PALMIRA)	1840,5
4	RIO BURGAY	41206,1
5	RIO CHICO SOLDADOS	15086,0
6	RIO CHULCO	21472,9
7	RIO COLLAY	39241,5
8	RIO CUENCA	26076,7
9	RIO DELEG	28747,6
10	RIO IRQUIS	21845,4
11	RIO LLAVIRCAY	10795,3
12	RIO MACHANGARA	48074,1
13	RIO MALUAY	17823,4
14	RIO MAZAR	30243,3
15	RIO PAUTE	65157,1
16	RIO SHUCAY	23445,2
17	RIO TAITA CHUGO	13440,5
18	RIO TARQUI	52690,0
19	RIO TOMBAMBA	40530,6

*Layer Ciudades\_CRP\_25MIL\_WGS84*, contiene información de las ciudades de la Cuenca del Río Paute.

```
LAYER
  NAME "ciudades"
  DATA "Ciudades_CRP_25MIL_WGS84"
  STATUS on
  TYPE polygon
  LABELCACHE on
  LABELITEM "CIUDAD"
  PROJECTION
    "init=epsg:4326"
  END
  CLASS
    NAME "Ciudades"
    STYLE
      COLOR 212 105 97
      OUTLINECOLOR 0 0 0
    END
    LABEL
      TYPE truetype
      FONT "arial"
      SIZE 8
      OUTLINECOLOR 255 255 255
      COLOR 0 0 0
      POSITION lr
    END # of label
  END #end of class
END #end of layer
```

*Layer* Localidades\_CRP\_25MIL\_WGS84, contiene información sobre las localidades pertenecientes a la Cuenca del Río Paute.

```
LAYER
NAME "localidades"
DATA "Localidades_CRP_25MIL_WGS84"
STATUS on
TYPE point
LABELCACHE on
LABELITEM "NOMBRE"
PROJECTION
    "init=epsg:4326"
END
CLASS
    NAME "Localidades"
    STYLE
        SYMBOL "Circle"
        SIZE 1
        COLOR 0 0 0
        OUTLINECOLOR 0 0 0
    END
    LABEL
        TYPE truetype
        FONT "arial"
        SIZE 8
        OUTLINECOLOR 255 255 255
        COLOR 0 0 0
        POSITION lc
    END # of label
END #end of class
END #end of layer
```

*Layer* de Anotaciones para Cantones\_CRP\_250MIL\_WGS84, permite etiquetar los Cantones de la CRP independientemente del *éste layer*, es decir, es un *layer* de nombres.

```
LAYER
NAME "cantones"
DATA "Cantones_CRP_250MIL_WGS84"
STATUS default
TYPE annotation
LABELCACHE on
LABELITEM "NOMBRE"
CLASS
    STYLE
        COLOR 0 0 0
    END
    LABEL
        TYPE truetype
        FONT "arial"
        SIZE 7
        COLOR 0 0 0
        BACKGROUNDCOLOR 255 255 255
        MINDISTANCE 50
        POSITION cc
        ANTIALIAS true
```

```
        END # of label
    END #end of class
END #end of layer

END #mapfile
```

Este es el código escrito en el *mapfile*, es necesario terminar con la palabra *END* para indicar el final del objeto map.

```
END #mapfile
```

### **Archivo de inicialización HTML**

MapServer utiliza archivos HTML a los que adiciona sus etiquetas propias, para desplegar los mapas. Estas etiquetas, llamadas variables CGI, van cerradas entre corchetes. Al procesar una petición, MapServer analiza el *mapfile*, genera los mapas y sustituye las etiquetas que están entre corchetes con los valores que tiene en memoria, antes de enviarlos al cliente. El archivo de inicialización especifica el programa CGI que va a correr,

```
<input type="hidden" name="program" value="/cgi-bin/mapserv">
```

el nombre del *mapfile* del cual se tomará información para la configuración,

```
<input type="hidden" name="map"
value="/opt/fgs/www/htdocs/pendientes_porcentaje/pendientes.map">
```

un factor de zoom inicial

```
<input type="hidden" name="zoomsize" size=2 value=2>
```

y los *layers* que se desplegarán cuando el archivo sea invocado por primera vez.

```
<input type="hidden" name="layers" value="localidades ciudades rios limite
iluminacion pendiente">
```

### **Plantilla HTML**

Mediante esta plantilla, el usuario a través del navegador puede interactuar con MapServer, cambiando los valores de las variables CGI. En la Figura 6 se muestra la plantilla obtenida al sustituir dichas variables.

Las etiquetas [program] y [img] son sustituidas por /cgi-bin/mapserv y por el nombre de la imagen del mapa que MapServer ha generado, respectivamente.

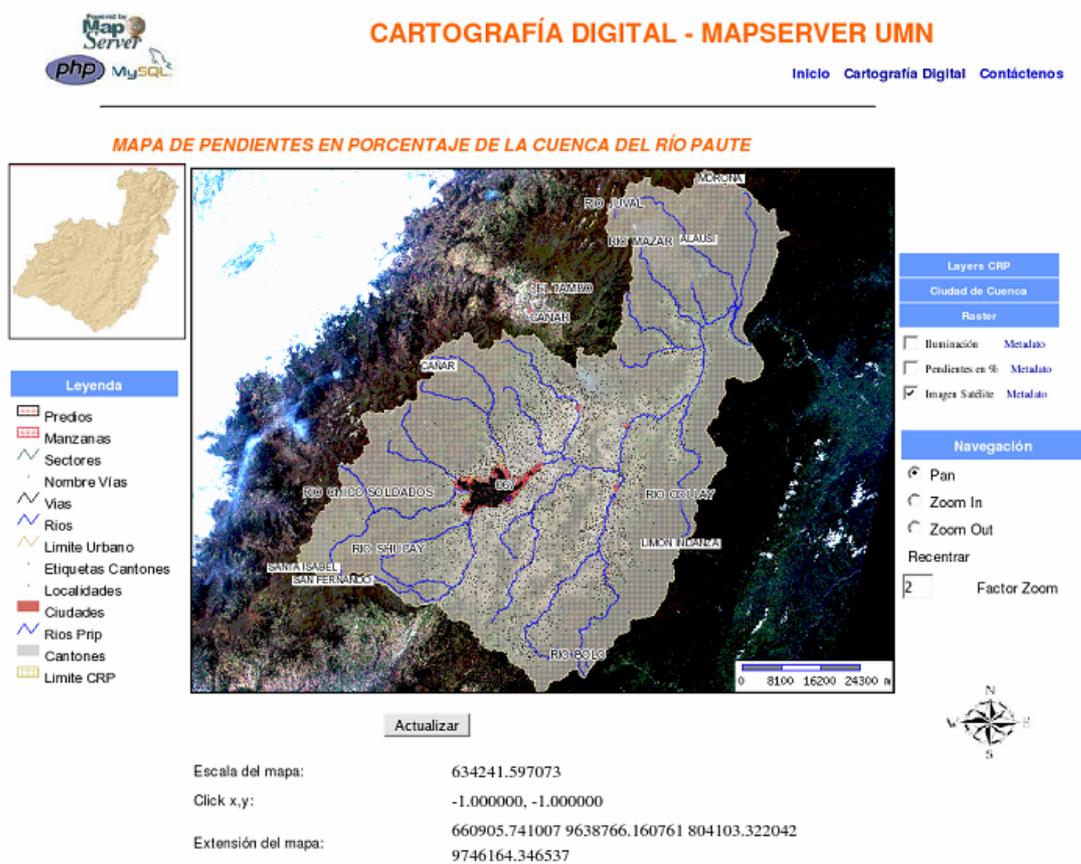
Los diferentes componentes del mapa como: [ref] será reemplazado por la imagen del mapa de referencia, [scale] indicará la escala a la que está la imagen del mapa, [mapx] y [mapy] darán las coordenadas del punto en donde se dio un clic con el ratón, [mapext] especifica la extensión del mapa y, [legend] desplegará la leyenda de los *layers* que se muestran.

Para proveer controles de *zoom* se utilizan definen los controles y sus etiquetas. La variable *zoomdir* determina la dirección del *zoom*; si esta variable toma el valor de 1, significa que se hará un *zoom in* de acuerdo al valor que tenga la variable *zoomsize*, si toma el valor de -1, se hará un *zoom out* y si está en 0, significa que estará en la misma escala que la última escala desplegada.

Para asociar el nombre de cada *layer* se utiliza la variable “*layer*” especificada en el *mapfile*. Si un *layer* va a ser desplegado, el *checkbox* que se presenta se marca, por tanto da la oportunidad al usuario de quitar o agregar al mapa la información de los diferentes *layers* presentados.

Por último, para que MapServer pueda mantener el estado de la aplicación, se realizan las sustituciones de características como: la extensión del mapa consiguiendo las coordenadas inferior izquierda y superior derecha de las esquinas del *bounding box* lo que permite obtener los valores de un *zoom* o un paneo que haya sido realizado; además, tiene la misma información que fue seteada con el archivo de inicialización: *mapfile* utilizado, programa cgi que corre.

Figura 6: Plantilla HTML para el mapa de Pendientes en Porcentaje



### Conjunto de Datos SIG

Una forma de incrementar la potencialidad de MapServer, es mediante la utilización de una base de datos. Para ello es necesario acceder al API del software, con la utilización de un lenguaje de programación como PHP y una base de datos como MySQL es posible cumplir este objetivo.

Para demostrar esta utilidad, hemos estructurado una base de datos a pequeña escala con información de la ciudad de Cuenca sobre predios, manzanas, sectores y propietarios de predios.

El objetivo consiste en utilizar las herramientas antes mencionadas para realizar consultas sobre ciertos predios de la manzana comprendida entre las calles Simón Bolívar, Luis Cordero, Mariscal Sucre y Antonio Borrero; estas consultas serán descritas más adelante.

Para acceder al API de MapServer, PHP utiliza funciones tales como:

- *getLayerByName* ( *string name* ): Retorna una referencia a la capa nombrada en *string name*.
- *AddFeature* (*shapeobj*): Agrega un nuevo *feature* a un *layer*.
- *MS\_ON* y *MS\_OFF*: Son valores booleanos de control lógico para habilitar o deshabilitar *layers*.
- *SetXY*(*float x*, *float y*): Setea una coordenada espacial, y opcionalmente, mide valores *simultáneamente*.
- *ZoomPoint*(*int zoomfactor*, *pointObj imgpoint*, *int width*, *int height*, *rectObj extent*, *rectObj maxextent*): Método del objeto mapa, hace un zoom según el *zoomfactor* a *imgpoint* en unidades de pixel dentro de la imagen de dimensiones según *height* y *width* dimensions y que está georeferenciada por *extent*. El zooming puede ser restringido a un *maxextent* máximo.
- *draw*(): Dibuja el mapa, procesando *layers* de acuerdo a su estatus y orden definido, retorna un objeto imagen.
- *drawReferenceMap*(): Dibuja un mapa de referencia, devuelve un objeto imagen.

## 2.- Formas de Consultas con MapServer

MapServer, trabaja en dos modos: en modo *browser* con el cual brinda capacidades para desplegar y explorar mapas mediante el paneo, *zoom*, etc.; y el modo *query* que ofrece poderosas facilidades para consultas. Soporta consultas espaciales la cuales seleccionan *features* basados en una localidad; y consultas de atributos que seleccionan *features* basados en los valores de los atributos.

Para trabajar con consultas, MapServer adiciona componentes extras tanto en el *mapfile* y en las plantillas HTML dando lugar a los siguientes conceptos:

*Query templates.*- Cuando los resultados de una consulta son retornados, MapServer los inserta en una o más plantillas HTML y luego los presenta en el navegador. MapServer debe estructurar las plantillas de salida, de tal manera que sean lo más

generalizadas posible y que satisfagan todos los modos de consulta ofrecidos. Para realizar esto, MapServer especifica una jerarquía de plantillas de consulta que pueden ser definidos a nivel de mapa, de *layer* y de clase.

A nivel de mapa, una plantilla de consulta es usada para presentar un resumen de una consulta completa (como el número total de coincidencias y número de *layers* con coincidencias). En el caso de ser definidos a nivel de *layer*, se presenta información específica de los *layers*; y al nivel de clase, se presenta atributos de *features* individuales.

*Query maps*.- Es un mapa que presenta los resultados de una consulta resaltando los *features* seleccionados. Estos mapas deben ser especificados en el *mapfile* por el objeto *Querymap*; si éste no está especificado, los resultados de las consultas serán tabulares.

## **Tipos de Consulta**

MapServer ofrece varios modos de consulta, entre ellos tenemos:

Modo Consulta (*Query Mode*).- Ejecuta a partir de un punto en el mapa, es considerada la consulta espacial más simple. Las coordenadas en píxeles de un punto en el mapa, donde el usuario da un clic, son devueltas a MapServer como valores de las variables CGI *img.x* e *img.y*. Estas coordenadas son usadas para buscar en cada *layer*, hasta encontrar el *feature* más cercano al clic, dentro de una tolerancia (máxima distancia entre el clic del mapa y un *feature* aceptado como coincidente) especificada. Este tipo de consulta devuelve tan solo la primera coincidencia encontrada. La tolerancia se determina con la etiqueta *TOLERANCE* a nivel de *layer* en el *mapfile*.

Modo Consulta N (*NQuery Mode*).- Trabaja de manera similar que el modo consulta. Este tipo de consulta devuelve todas las coincidencias que estén dentro la distancia de tolerancia especificada.

Modo consulta por ítem (*ItemQuery Mode*).- Ejecuta consultas simples por atributo, busca las coincidencias de texto en las tablas de atributos (archivos *.dbf*) y retorna la primera coincidencia encontrada. El valor de tolerancia es ignorado en este tipo de

consulta. La expresión de búsqueda está especificada con la variable CGI *qstring*. Es posible especificar el nombre de un atributo para limitar la búsqueda a una sola columna de lo contrario todos los atributos son incluidos, este valor está relacionado con la variable CGI *qitem*. De igual manera se puede restringir los *layers* que intervienen en la búsqueda con la variable *qlayer*.

Modo consulta por ítem N (*ItemNQuery Mode*).- Es igual que el modo de consulta por ítem pero, éste devuelve todas las coincidencias encontradas.

Modo consulta por feature (*FeatureQuery Mode*).- Ejecuta consultas espaciales que usan un *feature* de un *layer* para consultar otro *layer*. Sólo el primer *feature* coincidente de un *layer* escogido es seleccionado para usar en búsquedas espaciales subsecuentes. Todos los *features* encontrados dentro del *feature* seleccionado serán devueltos. El *layer* de selección está definida por la variable CGI *slayer* y debe ser de tipo polígono.

El valor asignado a la tolerancia es importante; si está en 0, sólo los *features* que estén dentro del polígono seleccionado son devueltos, caso contrario, la consulta devolverá los *features* coincidentes fuera de dicho polígono pero dentro de la tolerancia.

Modo consulta por feature N (*FeatureNQuery Mode*).- Es similar al *FeatureQuery*, pero retorna todos los *features* coincidentes desde el *layer* seleccionado.

Modo consulta por ítem y feature (*ItemFeatureQuery Mode*). - Este modo combina una búsqueda por atributo con una espacial y devuelve la primera coincidencia. Es similar al *FeatureQuery*, pero en lugar de usar un clic y valores de tolerancia para determinar las coincidencias del *layer* de selección, se realiza una búsqueda por atributo en éste *layer*. El *slayer* debe ser un *layer* tipo polígono, y el parámetro *qstring* contiene la expresión de búsqueda.

Modo consulta por ítem feature N (*ItemFeatureNQuery Mode*). - Es un modo similar al *ItemFeatureQuery*, pero todas las coincidencias de los *features* en el *slayer* son devueltas.

Modo consulta por índice (*IndexQuery Mode*).- Esta consulta recupera un *feature* basado en el índice del *shape*.

## Aplicación con Consultas

Una vez que hemos presentados los conceptos utilizados, damos a conocer la aplicación realizada. Se presenta el *mapfile*, pero explicamos únicamente el nuevo código adicionado.

### MapFile

Damos a conocer el código agregado al *mapfile* presentado en la sección anterior. Dentro del *Web Object*, constan tres nuevas palabras claves que son utilizadas para aplicaciones con consultas. *HEADER* especifica la ruta y el nombre del nivel más alto de la plantilla de consulta, el que es procesado antes de realizar cualquier otra cosa. *FOOTER*, especifica la ruta y nombre del nivel más bajo de la consulta; éste será procesado al final. Estos son solo fragmentos de código *HTML* y necesitan ser unidos con otros fragmentos para formar una página *web* completa.

*EMPTY* especifica el nombre de una página *web* completa que se utilizará cuando no se encuentren resultados de las consultas realizadas.

```
WEB
  HEADER
  "/opt/fgs/www/htdocs/pendientes_porcentaje/pendientes_web_header.html "
  FOOTER
  "/opt/fgs/www/htdocs/pendientes_porcentaje/pendientes_web_footer.html "
  EMPTY  "/opt/fgs/www/htdocs/pendientes_porcentaje/pendientes_vacio.html "
  TEMPLATE  '/opt/fgs/www/htdocs/pendientes_porcentaje/pendientes.html '
  IMAGEPATH  '/opt/fgs/www/htdocs/pendientes_porcentaje/data/tmp/ '
  IMAGEURL   '/pendientes_porcentaje/data/tmp/ '
END
```

Inicio de objeto *QueryMap*; especifica que el mapa a generar es un mapa de consulta. A este objeto pertenecen etiquetas como:

*STATUS*: Especifica si el mapa de consulta se desplegará (*on*) o no (*off*).

*STYLE*: Especifica como los *features* seleccionados son desplegados, de acuerdo con los siguientes valores.

- *Normal*.- Los *features* seleccionados son dibujados sin resaltarlos
- *Hilite*.- Los *features* seleccionados son dibujados en el color especificado por la etiqueta *COLOR* y los demás son dibujados normalmente.
- *SELECTED*.- Los *features* seleccionados son dibujados normalmente, mientras los otros *features* no son dibujados.

*COLOR*: Especifica el color utilizado para resaltar los *features* seleccionados.

*SIZE*: Especifica el tamaño en píxeles de la imagen del mapa de consulta.

```
QUERYMAP
  STATUS on
  STYLE hilite
  COLOR 255 255 0
  SIZE 320 260
END
```

Es necesario además, agregar nuevas sentencias en la definición de los layers. En nuestra aplicación utilizamos los layers *manzanas\_WGS84.shp* y *predios\_WGS84.shp* para la demostración de consultas.

```
#Layer: manzanas_WGS84
```

```
LAYER
  NAME "manzanas"
  DATA "manzanas_WGS84"
  STATUS on
  TYPE polygon
  PROJECTION
    "init=epsg:4326"
  END
  LABELCACHE on
  LABELITEM "MANZANA" #campo por el cual se va a etiquetar
  TRANSPARENCY 90
```

Las plantillas de consulta *HEADER* y *FOOTER* son especificados para presentar los resultados correspondientes a este *layer*. *TOLERANCE* y *TOLERANCEUNITS* definen cuan cerca de un *feature* debe darse un clic del ratón para que éste sea aceptado como una coincidencia en las consultas realizadas.

```

HEADER "/opt/fgs/www/htdocs/pendientes_porcentaje/
manzanas_header.html"
FOOTER "/opt/fgs/www/htdocs/pendientes_porcentaje/
manzanas_footer.html"
TOLERANCE 1
TOLERANCEUNITS meters

```

Se especifica la plantilla de consulta a nivel de clase mediante la etiqueta *TEMPLATE*.  
Será usada para presentar cada resultado.

```

CLASS
NAME "Manzanas"
TEMPLATE "/opt/fgs/www/htdocs/pendientes_porcentaje/
manzanas_query.html"
STYLE
SYMBOL "BigLine"
COLOR 234 74 86
OUTLINECOLOR 234 74 86
END
END

```

En el *layer* predios\_WGS84 se utiliza un código similar al explicado anteriormente.

```

LAYER
NAME "predios"
DATA "predios_WGS84"
STATUS on
TYPE polygon
PROJECTION
"init=epsg:4326"
END
TRANSPARENCY 90
HEADER "/opt/fgs/www/htdocs/pendientes_porcentaje/
predios_header.html"
FOOTER "/opt/fgs/www/htdocs/pendientes_porcentaje/
predios_footer.html"
TOLERANCE 1
TOLERANCEUNITS meters
CLASS
NAME "Predios"
TEMPLATE "/opt/fgs/www/htdocs/pendientes_porcentaje/
predios_query.html"
STYLE
SYMBOL "BigLine"
COLOR 212 105 97
OUTLINECOLOR 0 0 0
END
END #end of class
END #end of layer

```

En el Anexo 1 se presenta el código completo del archivo *mapfile* descrito en el capítulo.

## Archivo de inicialización

El archivo de inicialización es el mismo descrito anteriormente, pero es necesario agregar nuevas sentencias para el correcto funcionamiento con las consultas.

El siguiente código indica los parámetros de inicialización para MapServer que ya fueron especificados en la sección anterior.

Con las siguientes líneas vinculamos con los valores definidos en cada *layer* de consulta del mapfile. El valor de 100 asignado en la variable “*value*” de la plantilla se relaciona con la etiqueta *TOLERANCEUNITS* que determina el tipo de unidad (píxeles, metros, kilómetros, millas) de la tolerancia.

```
<input type="hidden" name=map_Ciudades_tolerancia size=4
      value=100>
<input type="hidden" name=map_Cantones_tolerancia size=4
      value=100>
```

La variable *slayer* especifica el *layer* para seleccionar cuando se ejecuta a *FeatureQuery* o *ItemFeatureQuery*. Este valor debe ser un asignado a un *layer* de tipo polígono. Además se deben inicializar en cero las variables CGI relacionadas con consultas.

La variable *mapshape* contiene una lista con las coordenadas de los vértices en las unidades del mapa, que definen el shape de la región de consulta; el primer par de coordenadas debe ser igual que el último par.

La variable *imgshape* se utiliza de manera semejante que *mapshape*, pero las coordenadas están expresadas en pixels. *Imgbox* contiene las coordenadas de los vértices izquierdo-superior y derecho-inferior de la región rectangular que serán utilizados como zona de consulta. Se debe realizar la inicialización de la expresión de consulta (*qstring*), del *layer* de consulta (*qlayer*: es el nombre del *layer* de consulta como el usado en el *mapfile*), del ítem de consulta (*qitem*: nombre del atributo consultado), del índice del *shape* (*shapeindex*) y *savequery*.

La variable *savequery* puede tomar valores de *true* o *false*; cuando toma el valor de *true*, MapServer almacena los resultados de la consulta en un archivo localizado en la ruta

indicada por la etiqueta *IMAGEPATH* en el *mapfile*. Este archivo tiene la *extensión* que y toma el nombre especificado en el *mapfile* por la etiqueta *name* a nivel del mapa.

## Plantilla HTML

La plantilla HTML para las consultas es la misma que se describió anteriormente; pero en la parte inferior de la plantilla se presenta dentro de un combo todos los modos de consulta para que el usuario pueda elegir la consulta de acuerdo a la información que requiera, además se presenta cuadros de texto en los que el usuario puede ir asignando los valores necesarios para que MapServer trabaje en los diferentes modos de consulta ofrecidos.

La siguiente Figura 7 muestra la sección referente a las consultas.

Figura 7: Sección de Consultas

Rios Prop  
Cantones  
Limite CRP

Actualizar

Escala del mapa:	3171.207989
Click x,y:	721626.549656, 9678438.603094
Extensión del mapa:	721268.555703 9678170.107630 721984.543609 9678707.098559

**CONSULTAS**

Modo del Mapa :

Layer de consulta	<input type="text"/>	Coordenadas imgbox	<input type="text"/>
Item de consulta	<input type="text"/>	Coordenadas imgshape	<input type="text"/>
Cadena de consulta	<input type="text"/>	Coordenadas mapshape	<input type="text"/>
Tolerancia Predios	<input type="text" value="100"/> Metros	Índice del shape	<input type="text"/>
Tolerancia Manzanas	<input type="text" value="100"/> Metros	Layer de selección por defecto: Manzanas	

## Plantillas de Consulta

Las plantillas de consulta proveen a MapServer la posibilidad de presentar los resultados de una consulta. También le informan qué *layers* son de consulta, puesto de

éstos tienen una plantilla a nivel de clase y de *layer*. Las plantillas de consultas son fragmentos de código HTML, que unidos forman una página web completa.

#### Plantillas a nivel de mapa

Estas plantillas están definidas dentro del objeto *WEB*. La Figura 8 muestra la plantilla a nivel de mapa.

Figura 8: Plantilla a nivel de mapa

**Resultados de Consultas**

Extensión de búsqueda:	721257.368392 9678254.012462 721973.356298 9678791.003392
Click point:	252,202 (Coordenadas de la imagen) 721539.288630,9678565.019709 (Coordenadas del mapa)
Resultado de la Consulta:	Capas con resultados=2 Número total de resultados=2

El detalle de los archivos plantilla a nivel de mapa presentamos a continuación:

#### Plantilla *Header*

Inicia con las primeras etiquetas de una página *web*, es utilizada por MapServer para presentar un resumen de los resultados de las consultas realizadas. Tiene información como la extensión del mapa, las coordenadas del punto en donde se dio un clic.

#### Plantilla *Footer*

Contiene las cadenas de sustitución y son embebidas en las etiquetas. Cuando el usuario da clic en el link, MapServer recarga la plantilla original de la aplicación con los valores almacenados mediante el botón para regresar a la página anterior que se presenta. Los espacios no están permitidos.

## Plantillas para el *layer* de consulta

El *layer* de consulta contiene tres plantillas: *Header* y *Footer* a nivel de *layer* y una plantilla a nivel de clase. La Figura 9 muestra la plantilla a nivel de *layer*.

Figura 9: Plantilla a nivel de mapa

Layer: Predios

	shpidx	PREDIO	MANZANA	SECTOR	AREA	PERIMETER
1	24978	009	010	01	1646.44220	164.99947

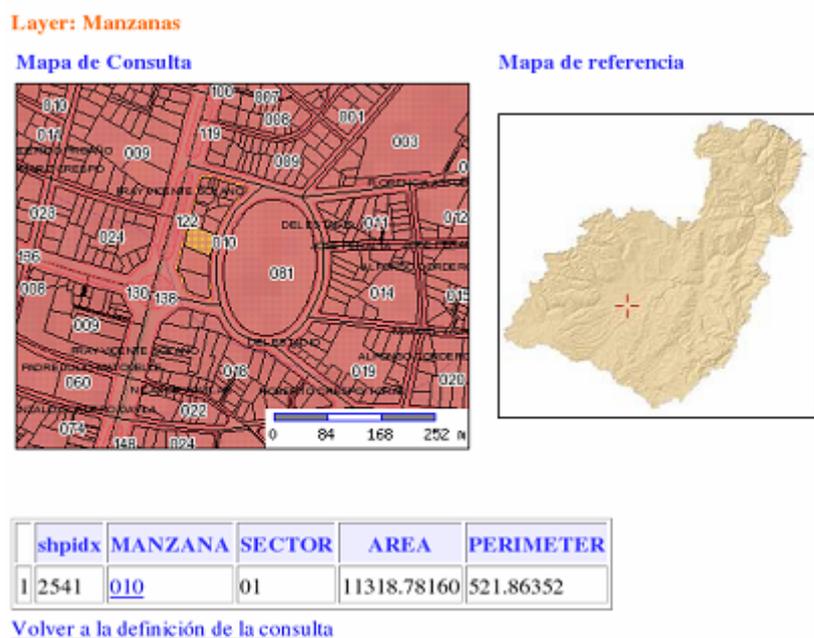
### Plantilla *Header* a nivel de *layer*

No comienza con las etiquetas de apertura HTML, es un fragmento de una página *Web*. Estructura algo de texto, prepara tablas, y hace referencia a dos imágenes con cadenas de sustitución: [img] que puede ser reemplazado por la imagen del mapa o con la imagen del mapa de consulta con *features* resaltados; y [ref]. Además inicia una tabla que especifica una fila con los títulos de las columnas, que son los atributos del *feature* consultado.

### Plantilla a nivel de clase

Una plantilla de consulta a nivel de clase es usada para procesar los *features* coincidentes. Consiste en una sola fila (ver Figura 10) que es adicionada a la tabla que fue abierta en la plantilla *Header* del *layer*.

Figura 10: Plantilla a nivel de clase



Cada resultado en un *layer* es asignado con un número secuencial empezando por 1. Este es referenciado por la cadena de sustitución [lrn], se muestra el índice del *shape*, las cadenas de sustitución en el código corresponden a los campos del archivo .dbf y son sustituidas por los atributos correspondientes al *feature* seleccionado.

Cuando se da clic en el *link* (Figura 11), MapServer ejecuta una consulta por índice (*indexquerymap*) y genera un mapa de consulta con el *feature* seleccionado resaltado. Al especificar *mapext = shape* significa que la extensión del mapa generado, será ajustada a la extensión del *feature* seleccionado.

Figura 11: Consulta por índice de link



#### Plantilla *Footer* a nivel de *layer*

Cuando se ha terminado de procesar toda la información consultada por el usuario, se hace uso de esta plantilla para cerrar la tabla abierta por la plantilla *Header* a nivel de *layer*. Consta únicamente de la etiqueta HTML que cierra tablas.

#### Plantillas para el *layer* de predios\_WGS84

Las plantillas para este *layer* de consulta son iguales que las mencionadas en el *layer* anterior, con la diferencia que en ésta plantilla no se da la posibilidad de generar un mapa de consulta.

La información y conceptos presentados están basados en las siguientes referencias: [DD01], [DD02], [DD03], [DD04], [DD05], [DD06] y [DD07]

#### **Conclusiones:**

MapServer brinda dos modos de uso para satisfacer los requerimientos de las distintas aplicaciones. El modo CGI permite un mapeo básico, es decir, visualización de layers, navegación, consultas. Para aplicaciones escalables se debe acceder a su API mediante lenguajes de programación como: Perl, Python, PHP; permite además hacer uso de

sistemas gestores de bases de datos como: Oracle, MySQL, PostGIS basado en PostgreSQL; lo que le convierte en un software potencial.

Estructurar los archivos que MapServer utiliza para desplegar los mapas no es tarea que se considere extremadamente complicada una vez entendido el funcionamiento de clases y objetos propios del archivo *mapfile*. De igual manera, las plantillas utilizan lenguaje básico HTML con cadenas de sustitución MapServer. Lo que requiere mayor cuidado son las aplicaciones que acceden al API, en nuestro caso la implementada con PHP.

MapServer proporciona consultas por atributo y espaciales pero la utilidad y el verdadero potencial, se ven reflejados en la calidad de información que se tenga en la base de datos o en los archivos *shape*.

## CAPÍTULO V

### Implementación de la Interfaz para configurar el Mapfile

Como hemos expuesto en el capítulo anterior, MapServer consta de cuatro componentes: el *mapfile*, el archivo de inicialización HTML, los archivos plantilla HTML y un conjunto de datos SIG; así mismo, hemos detallado cada uno de estos elementos.

El objetivo de este capítulo consiste en exponer la interfaz que el usuario empleará para configurar el archivo base de MapServer, es decir, el *mapfile*. La interfaz propuesta requiere ser manipulada por personal experimentado en SIG ya que la información a suministrarse es de tipo técnico.

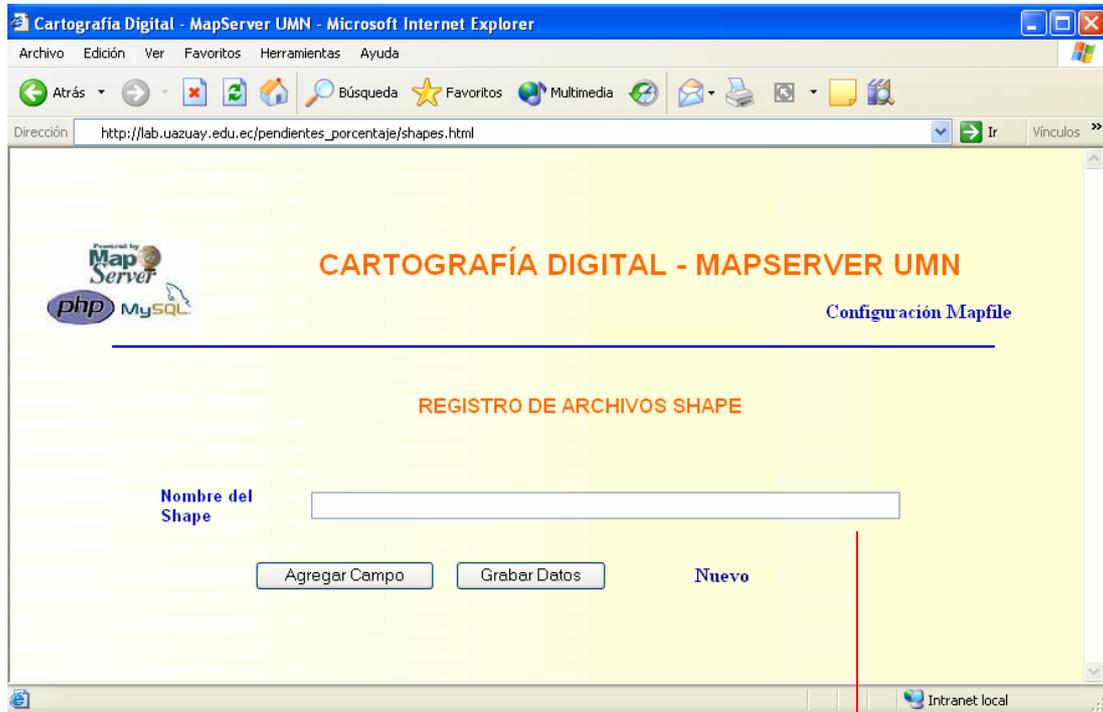
Se tienen dos interfaces que detallamos a continuación:

#### 1.- Interfaz de Registro de Archivos *Shape*

En esta interfaz no se requiere para la configuración del *mapfile*, es utilizada como apoyo para registrar los archivos *shape* y los campos de su *dbf* relacionado para posteriormente cargarlos en la aplicación que el usuario empleará para formar el *mapfile*.

En la Figura 1 se presenta la pantalla inicial para ingresar la información de los *shapes*, distinguimos el área de texto en el cual ingresamos el nombre del *shape* a registrar.

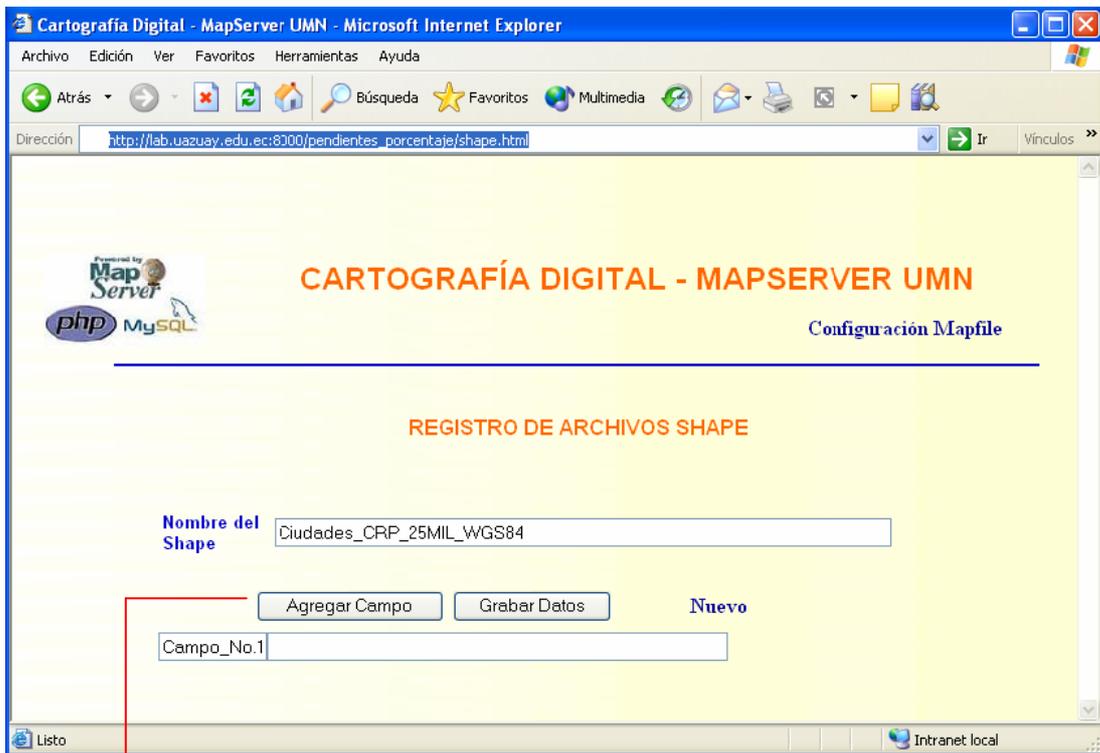
Figura 1: Pantalla inicial



Entrada de texto para registrar el nombre del *shape*

En la Figura 2 en esta pantalla, luego de ingresar el nombre del *shape* damos clic en el botón Agregar Campo para registrar las columnas de la tabla del *shape*, así se generan las entradas necesarias cada vez que se presione dicho botón.

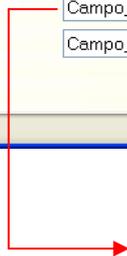
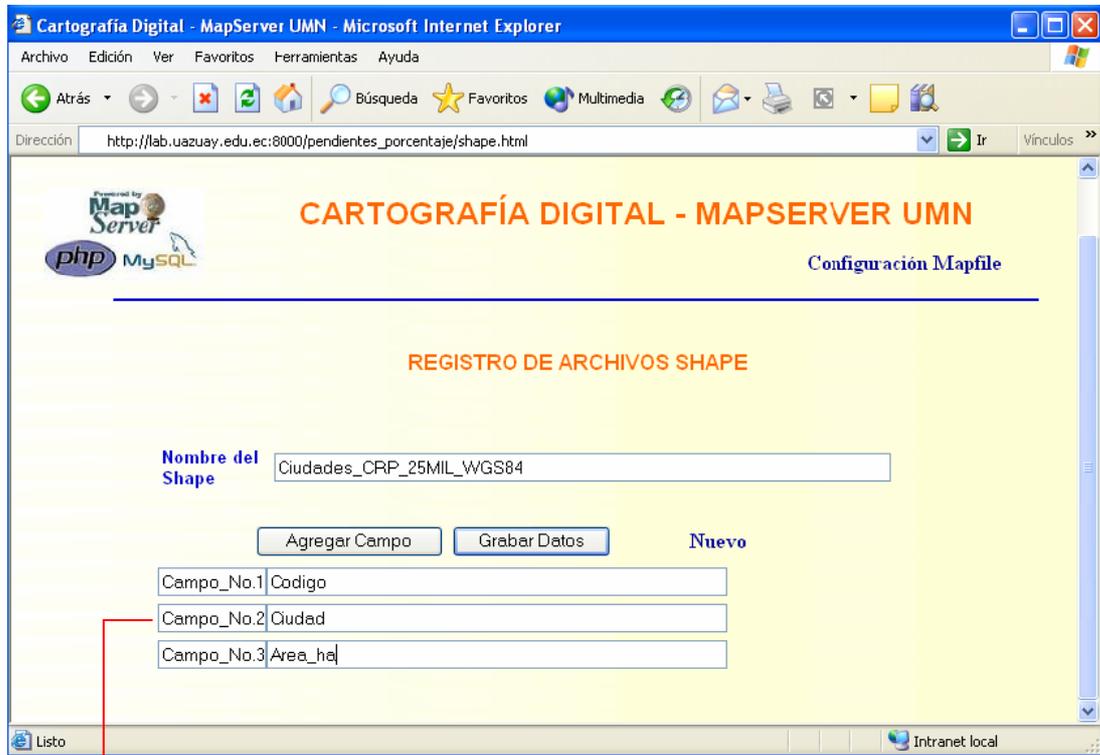
Figura 2: Ingreso de Campos del dbf del *shape*



Botón para agregar campos del *shape*

En la Figura 3 vemos cómo se agregan dinámicamente las entradas de texto para los campos a registrar.

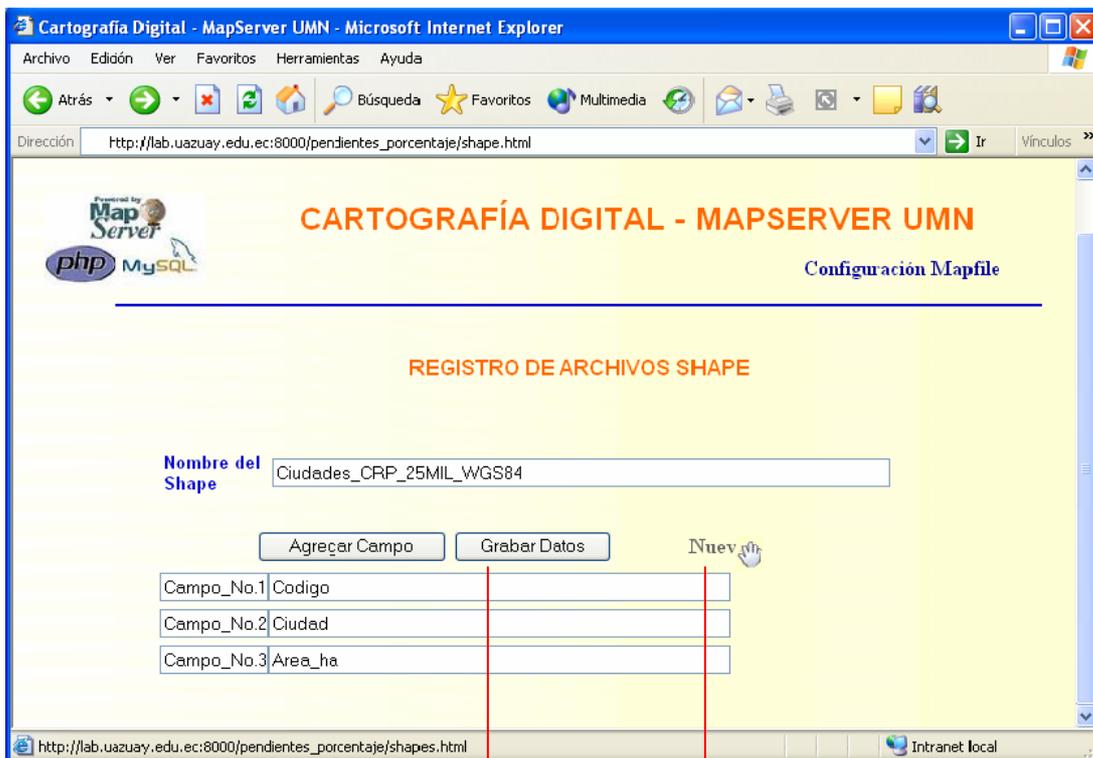
Figura 3: Entradas dinámicas



Entradas dinámicas

En la Figura 4 se indican el botón Grabar Datos, el cual registra los datos ingresados en un archivo de texto; y el *link* **Nuevo** que utilizamos para ingresar un nuevo *shape*.

Figura 4: Botón Grabar Datos y *link* Nuevo

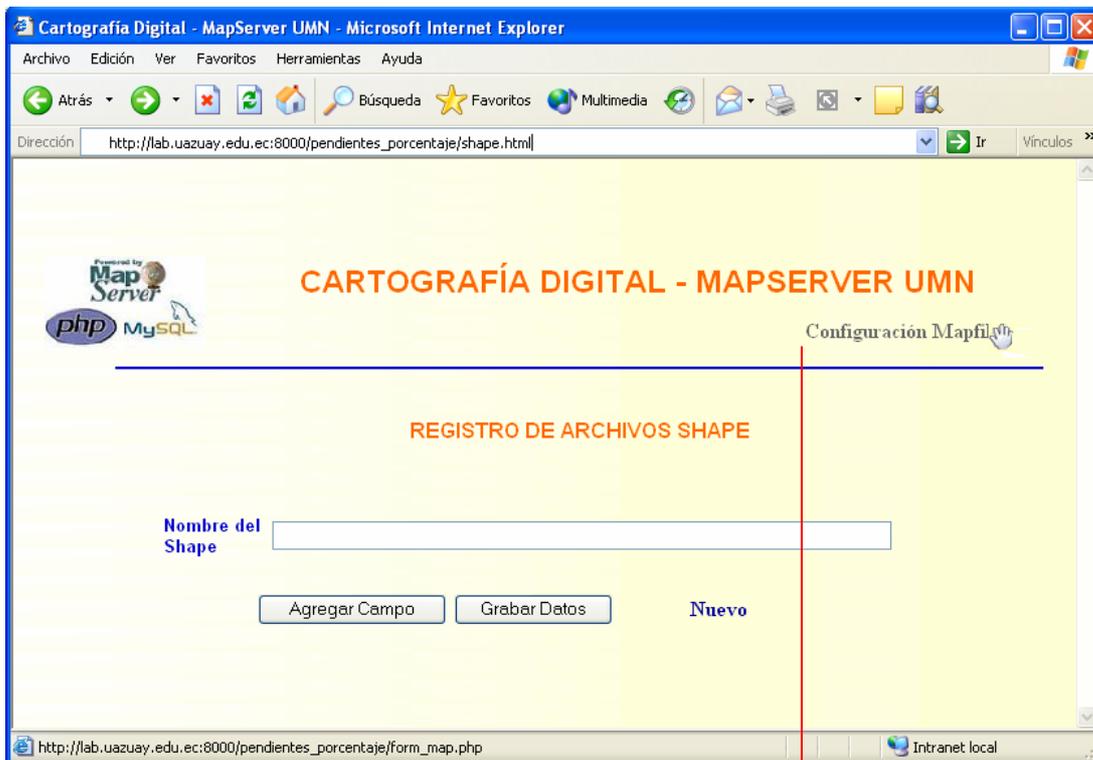


Botón para grabar los  
datos ingresados

*Link* para registrar un  
nuevo shape

La figura 5 señala el link **Configuración Mapfile**, ubicado en la esquina superior derecha de la página web, el mismo que nos permite enlazarnos a la página de configuración del archivo base de MapServer.

Figura 5: *Link* al archivo de configuración *MapFile*



*Link* para ir a la página de configuración del *Mapfile*

## 2.- Interfaz de Configuración del MapFile

El archivo de configuración de MapServer, mapfile, consta de varias etiquetas que pueden ser seteadas según los requerimientos y necesidades de la aplicación que servirá los mapas en Internet. A continuación describimos la interfaz desarrollada para hacer más amigable la configuración básica del archivo map, cabe resaltar que los términos que se emplean son técnicos y deben ser conocidos por el usuario que registre los datos, además cada uno de los campos fueron descritos ampliamente en el capítulo anterior de esta tesis.

La Figura 6 muestra la primera parte del archivo de configuración, donde los datos requeridos servirán para desplegar el mapa principal, el primer campo consiste en una breve descripción del archivo map, seguido se solicitan los valores de extensión del mapa los mismos que serán ingresados según coordenada inferior izquierda *minx*, *miny*; y coordenada superior derecha *maxx*, *maxy*; estos valores pueden ser negativos y con varios decimales según el sistema de proyección que se emplee.

Figura 6: Datos generales para el despliegue del mapa

The image shows a screenshot of a web browser window titled "Configuración de Archivo Map - Microsoft Internet Explorer". The address bar shows the URL "http://lab.uazuay.edu.ec:8000/pendientes\_porcentaje/form\_map.php". The page content includes the MapServer logo, "CARTOGRAFÍA DIGITAL - MAPSERVER UMN", and a "Registrar Shapes" link. The main section is titled "CONFIGURACIÓN DEL ARCHIVO MAP" and has a "General" sub-section. The "General" section contains several form fields: "Descripción" (a text area), "Extent" (four input fields for min x, min y, max x, and max y, which are circled in red), "Tamaño" (input fields for "Ancho ( Píxeles )" and "Alto ( Píxeles )" ), "Unidades" (a dropdown menu set to "Grados"), "Formato generado" (a dropdown menu set to "PNG"), and "Sistema de Proyección" (a dropdown menu set to "PSAD56"). A red arrow points from the circled "Extent" fields to the text "Coordenadas de la imagen principal" located below the form.

El tamaño de la imagen principal generada por la aplicación debe ingresarse en píxeles de ancho y de alto requeridos. Figura 7.

Figura 7 Tamaño de la imagen principal

Configuración de Archivo Map - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Atrás Búsqueda Favoritos Multimedia Ir Vínculos

Dirección http://lab.uazuay.edu.ec:8000/pendientes\_porcentaje/form\_map.php

Powered by Map Server php MySQL

## CARTOGRAFÍA DIGITAL - MAPSERVER UMN

Registrar Shapes

### CONFIGURACIÓN DEL ARCHIVO MAP

**General**

Descripción

Extent  min x  min y  
 max x  max y

Tamaño  Ancho ( Píxeles )  
 Alto ( Píxeles )

Unidades Grados

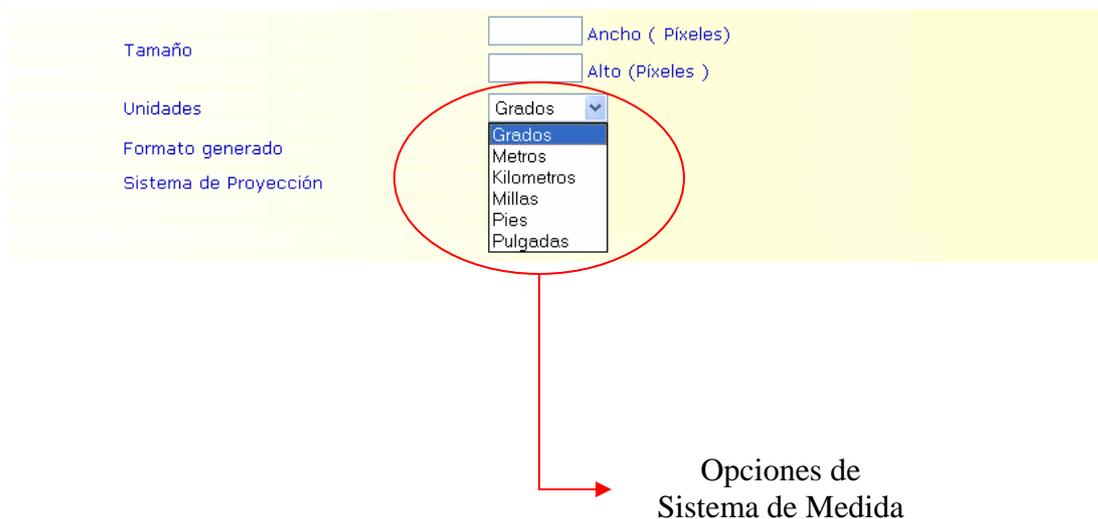
Formato generado PNG

Sistema de Proyección PSAD56

Tamaño de la imagen principal

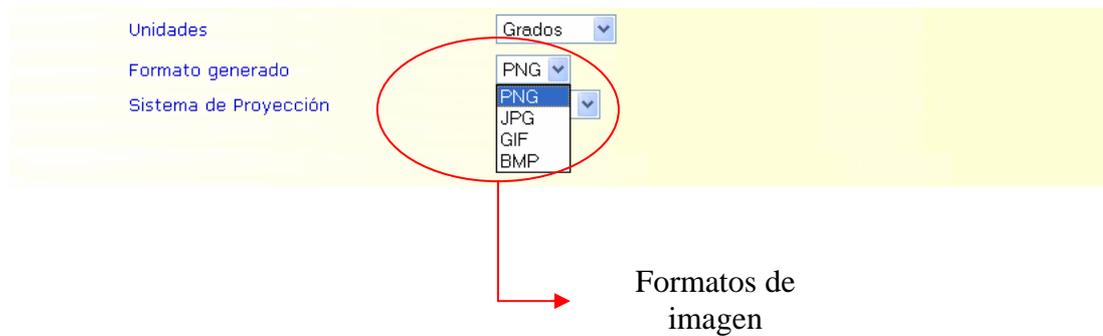
La Figura 8 muestra las unidades de configuración de la imagen principal, éstas unidades pueden elegirse entre: grados decimales, metros, kilómetros, millas, pies, o pulgadas; según el sistema de medida que se emplee.

Figura 8: Opciones de Sistema de Medida



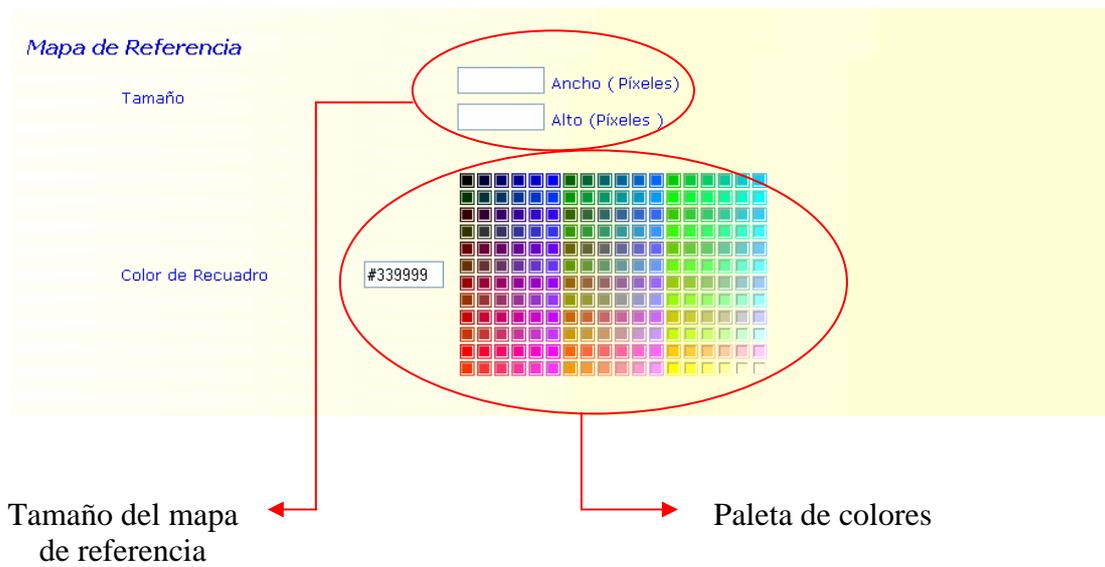
La Figura 9 muestra las opciones de imagen que MapServer puede generar dinámicamente para el mapa de salida.

Figura 9: Formatos de imagen de salida de MapServer



En la Figura 11 se indica el tamaño del mapa de referencia por ancho y alto en píxeles, también se observa la paleta de colores en la cual el usuario elige el color para el recuadro que marca, sobre el mapa de referencia, la ubicación del punto navegado en el mapa principal.

Figura 11: Configuración del Mapa de Referencia



La configuración de los valores de la barra de escala se muestran en la Figura 12 y 13. La primera muestra los intervalos en los cuales puede dividirse la escala gráfica, y la Figura 13 indica las unidades que se pueden elegir para las divisiones escalares tales como: metros, kilómetros, millas, pies y pulgadas.

Figura 12: Intervalos de la Escala Gráfica

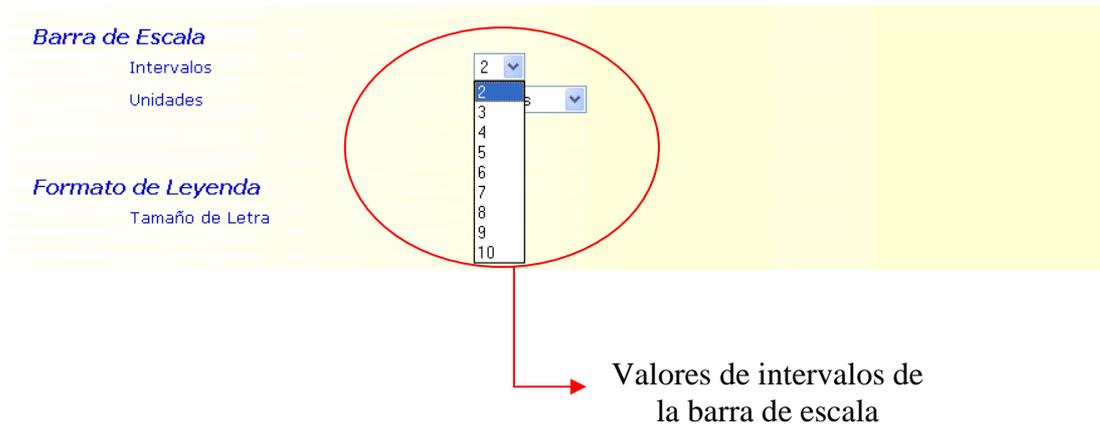
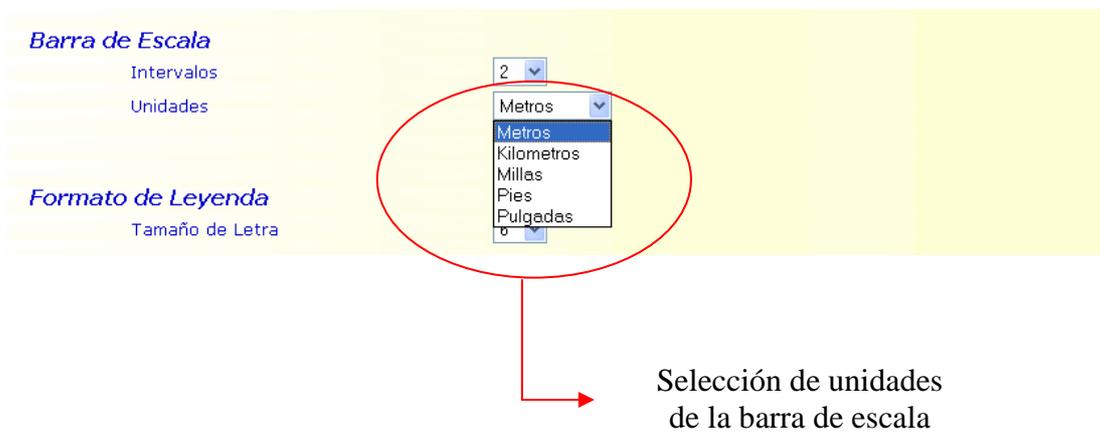
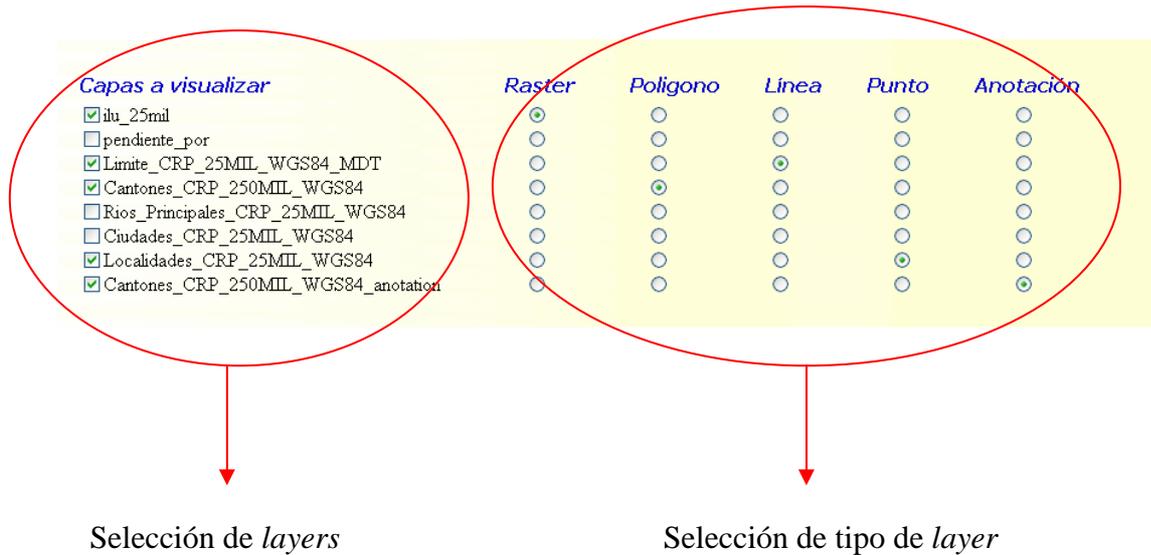


Figura 13: Unidades de la Escala Gráfica



La Figura 14 indica las capas que se pueden cargar en el *mapfile*, éstas capas corresponden a los *shapes* que estén registrados; al lado izquierdo se observan las opciones: Raster, Polígono, Línea, Punto y Anotación que deben corresponder al tipo de capa que esté seleccionado a su derecha.

Figura 14: Selección de *layers* y sus tipos



En la Figura 15 se indica el formato de consulta donde se eligen: el ancho y alto (en píxeles) de la imagen resultado de una consulta espacial; las dos capas de consulta y la paleta donde se escoge el color con el que se resaltará el *feature* consultado.

Figura 15: Formato de consulta

**Formato de Consulta**

Tamaño

Ancho ( Píxeles )

Alto ( Píxeles )

Capa de Consulta 1:

Etiquetar por:

Tolerancia:

Capa de Consulta 2:

Etiquetar por:

Tolerancia:

Color de Consulta

#99cccc

Layers de Consulta

Color para resaltar la consulta

Tamaño de la imagen de la consulta

Al escoger las capas de consulta (Figura 16), debemos escoger también el campo por el cual se va a etiquetar el layer consultado (figura 17), de igual manera se debe indicar las unidades de tolerancia (grados, metros, kilómetros, millas, pies, pulgadas, píxeles) para cada capa de consulta (Figura 18). El proceso es el mismo para las dos capas de consulta.

Figura 16: Selección de *layer* de consulta

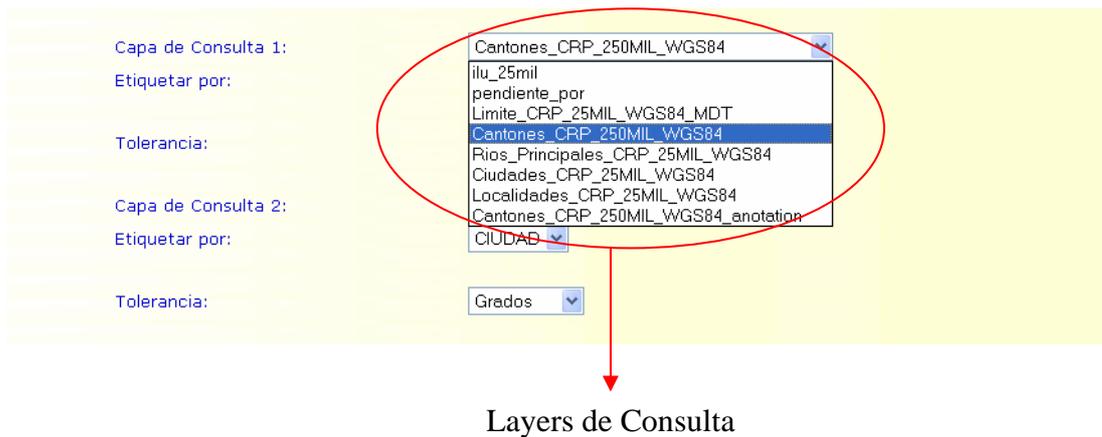


Figura 17: Selección de la etiqueta de consulta

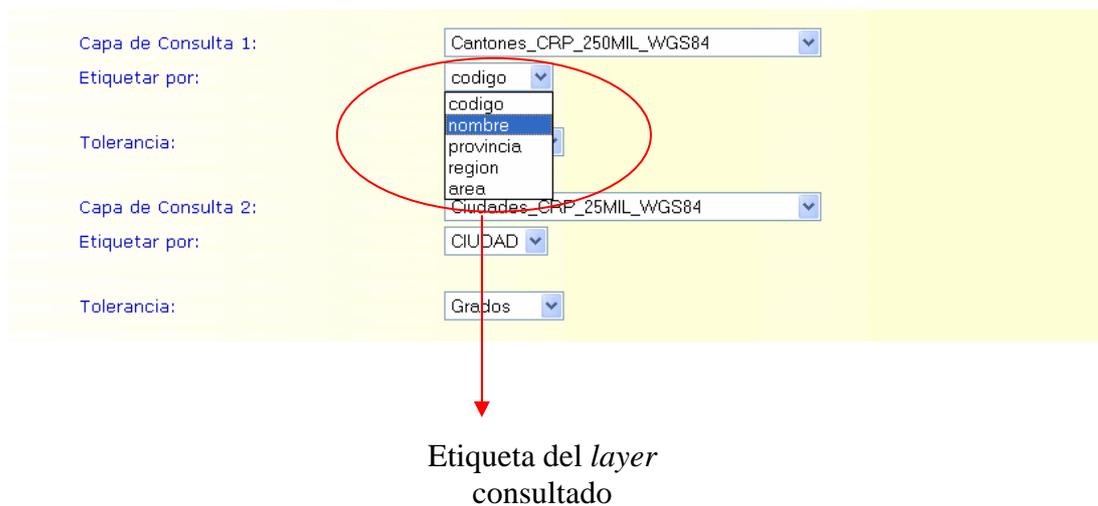
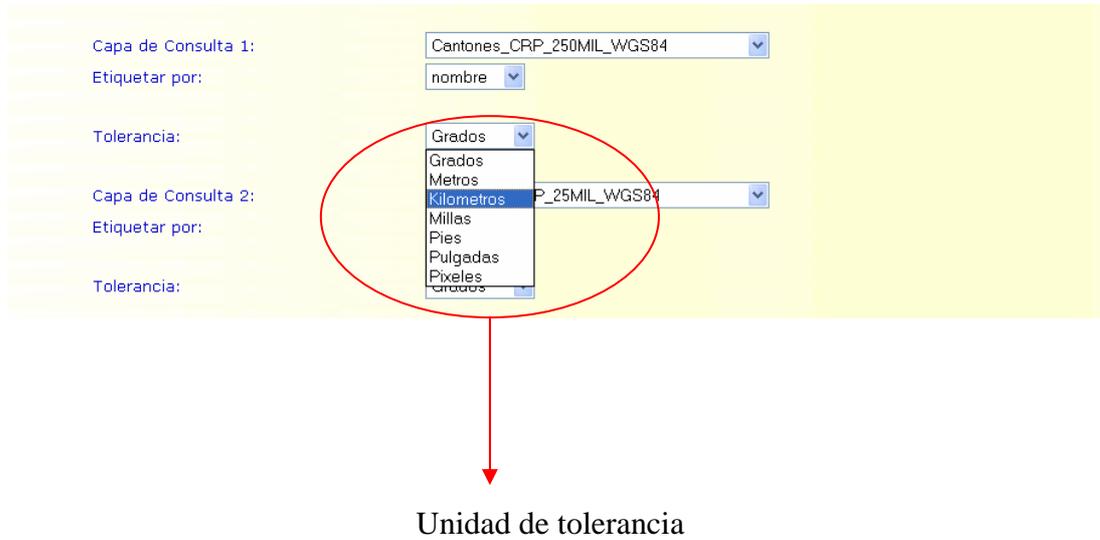


Figura 18: Selección de las unidades de tolerancia de consulta



## Conclusiones

La aplicación detallada en este capítulo muestra una forma intuitiva de configurar el archivo base de MapServer, a pesar de ser una interfaz muy sencilla se requiere conocimientos de SIG para proporcionar la información de manera correcta y no tener problemas al momento de subir el mapa al servidor. Los datos requeridos por la aplicación son los más básicos ya que existen muchos más parámetros que no son solicitados pero han sido seteados para comodidad del usuario, pero que para aplicaciones con más requerimientos que los elementales, deben ser tomados en cuenta ya que generan otros archivos que muestran información de forma dinámica.

## CAPÍTULO VI

### Pruebas y Depuración de la Aplicación

Para cualquier aplicación que se realice, ya sea utilizando un software existente o desarrollando uno nuevo, es necesario considerar un tiempo adecuado para probarla, lo que garantizará que no se produzcan futuras complicaciones. La etapa de pruebas y depuración se la ha realizado tanto en la fase de desarrollo como en la puesta en marcha.

Explicaremos las pruebas realizadas en las distintas fases de desarrollo de nuestro proyecto y e ilustraremos algunos ejemplos para un mejor entendimiento de su funcionamiento. Para esto, hemos dividido el proyecto en tres partes:

#### 1. Configuración de MapServer

La aplicación desarrollada en MapServer se encuentra corriendo en el servidor Apache de la Universidad del Azuay, con plataforma Red Hat Enterprise Linux 4.

A continuación se detallan las instalaciones y configuraciones realizadas:

La primera instalación que realizó de MapServer en el servidor fue con un paquete para Linux conocido como *EasyMapServer*, básicamente la instalación es igual que con el paquete FGS, pero ésta, instala las librerías necesarias solo para la visualización de la cartografía digital. Por esta razón fue necesario hacer uso de otras alternativas de instalación y configuración que permitan adicionar un mayor número de librerías para poder conseguir los objetivos planteados.

Instalación paquete por paquete:

La instalación de MapServer paquete por paquete que se explicó en el tercer capítulo de nuestro proyecto de tesis, fue posible realizarla en máquinas locales. Cuando se intentó instalar en el servidor, se originó un conflicto con la instalación ya existente de PHP.

Instalación con el paquete FGS:

En el servidor se encuentra la configuración de MapServer mediante el paquete *FGS*, con el cual el único parámetro que se debe tomar en cuenta, es el puerto con el que se trabajará. En nuestro caso estamos utilizando el puerto 8.000.

## **2. Aplicación realizada en MapServer**

La aplicación desarrollada consta de la publicación de dos mapas: Mapa CRP y Mapa de la ciudad de Cuenca:

En el primero desplegamos un mapa con *shapes* pertenecientes a la cuenca del río Paute; con esto demostramos las utilidades que brinda MapServer en el modo CGI como: desplegar archivos raster y los distintos tipos de datos vectoriales, activación y desactivación de *layers* mediante el manejo de *check box*, despliegue de una barra de escala, de la leyenda utilizada en el mapa; presenta opciones de navegación como paneo, *zoom in*, *zoom out*; estas opciones funcionan con un factor de ampliación o disminución, el mismo que debe ser un entero positivo no mayor a 40 esto varía según el navegador que se utilice. Estas características podrán ser utilizadas en el modo de navegación del mapa. Cada uno de los *layers* presenta un enlace a su metadato vinculado.

Para la activación y desactivación de los *layers* es necesario dar clic en el botón “Actualizar” para que se cumpla la petición requerida; únicamente los *layers* dibujados se verán reflejados en la leyenda. En el caso del paneo, el acercamiento y alejamiento aparecerá por defecto el valor de 2 en el factor de *zoom* y, de ser necesario podrá cambiar este valor, luego bastará con dar un clic sobre el mapa y se cumplirá la tarea. Los valores de la barra de escala también cambiarán dependiendo de cuán lejos o cerca esté desplegado el mapa. Se informará de las coordenadas del punto en dónde se da el clic y de la extensión del mapa.

Además damos a conocer diferentes formas de consulta por atributo y espaciales aplicadas para las manzanas y predios de la ciudad de Cuenca, cuya forma de ejecución se presenta a continuación:

La opción de consultas es posible eligiendo una ellas desde el combo de selección “Modo del Mapa” y seteando los diferentes valores que necesitan cada una de ellas. Si no se especifica nada en el campo *layer* de consulta, MapServer tomará por defecto el declarado como *slayer*.

El modo de Consulta, toma en cuenta el valor de la tolerancia del *layer* de predios, que por defecto estará en 100 metros. Luego de seteado este valor según las necesidades del usuario, se deberá dar un clic dentro de la ciudad de Cuenca y se presentará una pantalla con los resultados generados a partir de esta consulta.

Figura 1: Parámetros necesarios para el modo de Consulta

**CONSULTAS**

---

Modo del Mapa :

Layer de consulta

Item de consulta

Cadena de consulta

Tolerancia Predios  Metros

Tolerancia Manzanas  Metros

Índice del shape

Layer de selección por defecto: Manzanas

Figura 2: Resultado del modo Consulta

**Resultados de Consultas**

Extensión de búsqueda:	660905.741007 9638766.160761 804103.322042 9746164.346537
Click point:	272,296 (Coordenadas de la imagen) 721764.712947,9679935.465308 (Coordenadas del mapa)
Resultado de la Consulta:	Capas con resultados=1 Número total de resultados=1

**Layer: Predios**

	shpidx	PREDIO	MANZANA	SECTOR	AREA	PERIMETER
1	3700	005	010	02	3921.61300	278.61600

[Volver a la definición de la consulta](#)

El modo de Consulta N.- Para este modo de consulta se debe elegir la opción correspondiente, manejar el valor de la tolerancia de acuerdo con el valor que se requiera y dar un clic dentro de la ciudad de Cuenca. Se puede o no dar un *zoom in* a la Ciudad. Para el ejemplo, se toma en cuenta el valor de la tolerancia de predios por tanto como resultado mostrará el predio en dónde se dio clic y la manzana a la que pertenece. Si se aumentase el valor de la tolerancia, se generaría una lista de predios pertenecientes a esa manzana.

Figura 3 Parámetros necesario para el modo Consulta N

**CONSULTAS**

---

Modo del Mapa :

Layer de consulta  Tolerancia Predios  Metros

Item de consulta  Tolerancia Manzanas  Metros

Cadena de consulta  Índice del shape

Layer de selección por defecto: Manzanas

Figura 4: Resultado del modo Consulta N

**Resultados de Consultas**

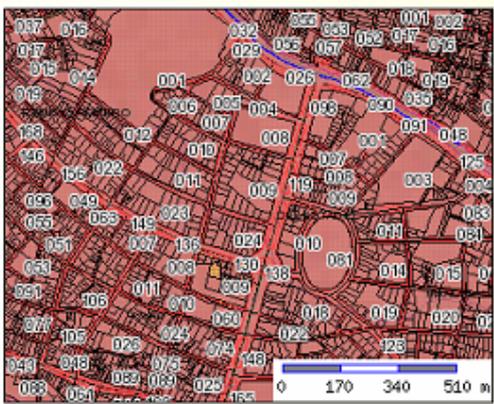
Extensión de búsqueda:	720713.105711 9678123.120923 722145.081521 9679197.102781
Click point:	281,324 (Coordenadas de la imagen) 721341.832590,9678472.165027 (Coordenadas del mapa)
Resultado de la Consulta:	Capas con resultados=2 Número total de resultados=2

**Layer: Predios**

	shpidx	PREDIO	MANZANA	SECTOR	AREA	PERIMETER
1	9232	015	009	03	1217.20870	145.55664

**Layer: Manzanas**

**Mapa de Consulta**



**Mapa de referencia**



	shpidx	MANZANA	SECTOR	AREA	PERIMETER
1	2644	009	03	21254.38535	813.01357

Modo consulta por ítem.- En este tipo de consulta la tolerancia no es relevante, el *layer* de consulta podrá ser el de manzanas o el de predios. El ítem de consulta es una columna del archivo .dbf perteneciente al *layer* de consulta escogido (este campo es sensitivo a mayúsculas y a minúsculas) y la cadena de consulta es un campo de dicha columna. Para ejecutar esta consulta, se debe dar un clic en el botón “Actualizar”; no importa si antes se realizó un *zoom* o no. En la figura 6.5 se presenta un fragmento del archivo .dbf de manzanas.

Figura 5: Fragmento del archivo Manzanas\_WGS84.dbf

AREA	PERIMETER	MA0707	MA0707	ID	ZONA	SECTOR	MANZANA	CLAVECATAS
19028,14455	676,12245	0	0	04	03	005		403005
5682,80575	338,75129	0	0	01	01	010		101010
1651,52945	171,02610	0	0	07	03	039		703039
7416,05608	346,58385	0	0	04	01	029		401029
395,25485	217,11926	0	0	05	04	077		504077
19295,17648	1088,96767	0	0	05	04	030		504030
24345,62900	892,70271	0	0	04	03	007		403007
12778,35110	463,83669	0	0	05	03	034		503034
1680,13100	168,59752	0	0	07	03	040		703040

Figura 6: Parámetros necesarios para el modo Consulta por ítem

**CONSULTAS**

---

Modo del Mapa :

Layer de consulta  Tolerancia Predios  Metros

Item de consulta  Tolerancia Manzanas  Metros

Cadena de consulta  Índice del shape

Layer de selección por defecto: Manzanas

El resultado obtenido, es la información correspondiente a la primera coincidencia de la cadena de consulta solicitada.

Figura 7: Resultado del modo Consulta por ítem



Modo consulta por ítem N.- Para esta forma de consulta los parámetros necesarios son los mismos que los empleados en la consulta anterior.

Figura 8: Parámetros necesarios para el modo Consulta por Ítem N

**CONSULTAS**

---

Modo del Mapa :

Layer de consulta  Tolerancia Predios  Metros

Ítem de consulta  Tolerancia Manzanas  Metros

Cadena de consulta  Índice del shape

Layer de selección por defecto: Manzanas

El resultado que genera esta consulta es una lista de todas las coincidencias encontradas con la cadena de consulta.

Figura 9: Resultado del modo Consulta por ítem N

	shpidx	MANZANA	SECTOR	AREA	PERIMETER
1	102	<u>001</u>	05	123794.96053	1639.86707
2	106	<u>001</u>	11	130297.75020	2245.93007
3	124	<u>001</u>	10	9077.57582	518.58353
4	150	<u>001</u>	08	2506.60045	300.54539
5	208	<u>001</u>	04	802604.74211	5042.75482

Modo consulta por *feature*.- En el ejemplo trabajaremos con el *layer* de manzanas, por tanto seteamos a 0 la tolerancia de predios y a 10.000 la tolerancia de manzanas. Es necesario dar un clic dentro de la ciudad de Cuenca para que se ejecute la consulta, haciendo o no un *zoom in* previo.

Figura 10: Parámetros necesarios para el modo Consulta por *Feature*

**CONSULTAS**

---

Modo del Mapa :

Layer de consulta

Item de consulta

Cadena de consulta

Tolerancia Predios  Metros

Tolerancia Manzanas  Metros

Índice del shape

Layer de selección por defecto: Manzanas

La consulta devolverá información de todos los predios pertenecientes a la manzana en donde se dio el clic.

Figura 11: Resultado del modo Consulta por *Feature*

**Layer: Predios**

	shpidx	PREDIO	MANZANA	SECTOR	AREA	PERIMETER
1	222	001	011	01	134.07232	46.75519
2	225	002	011	01	42.22045	26.05048
3	228	003	011	01	730.91775	116.88480
4	231	004	011	01	1354.63950	173.04801

**Layer: Manzanas**

	shpidx	MANZANA	SECTOR	AREA	PERIMETER
1	1308	011	01	14477.14530	478.40130

Modo consulta por *feature* N.- En este caso, a más de la información de la manzana en dónde se dio clic, aparecerán datos de las manzanas cercanas a ésta.

Figura 12: Parámetros necesarios para el modo Consulta por *Feature* N

**CONSULTAS**

---

Modo del Mapa :

Layer de consulta

Item de consulta

Cadena de consulta

Layer de selección por defecto: Manzanas

Tolerancia Predios  Metros

Tolerancia Manzanas  Metros

Índice del shape

Figura 13: Resultado del modo de Consulta por *Feature* N

**Layer: Predios**

	shpidx	PREDIO	MANZANA	SECTOR	AREA	PERIMETER
1	25000	005	014	01	452.61489	112.97125
2	25001	004	014	01	220.12191	62.18691
3	25002	003	014	01	807.91030	115.82026
4	25003	002	014	01	1072.65865	164.09242

**Layer: Manzanas**

**Mapa de Consulta**

**Mapa de referencia**

	shpidx	MANZANA	SECTOR	AREA	PERIMETER
1	2629	<u>014</u>	01	16858.81528	522.33776

Modo consulta por índice.- Todos los *shapes* tienen un índice o identificador, esta consulta permite pedir información de los *features* por este campo. El valor del índice del *shape* se puede ver en todas las consultas anteriores o en programas como *ArcView*. Se debe dar un clic en el botón Actualizar dando o no un *zoom in* previo.

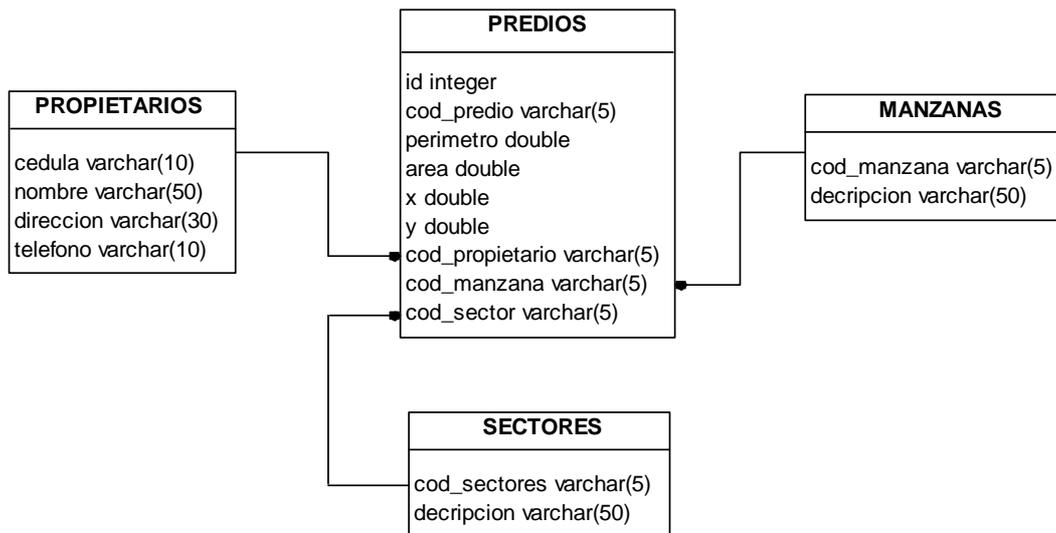
Figura 14: Parámetros necesarios para el modo Consulta por Índice

Figura 15: Resultado del modo Consulta por Índice

	shpidx	MANZANA	SECTOR	AREA	PERIMETER
1	5	<a href="#">084</a>	05	43960.86701	1244.29223

En el segundo mapa publicado utilizamos *shapes* de la ciudad de Cuenca, para esta demostración utilizamos el API de MapServer accediendo con *PHP-Mapscript*. Además hacemos uso de una base de datos en *MySQL* en la que tenemos relacionadas cuatro tablas: predios, manzanas, sectores y propietarios, de la siguiente manera:

Figura 16: Modelo entidad-relación de la base de datos utilizada



En este mapa se puede navegar, activar y desactivar los *layers*, es decir, tiene opciones semejantes al mapa anterior. La diferencia básica es que para las consultas en lugar de acceder a los archivos *.dbf*, consultamos los datos que están almacenados en la base de datos. Las consultas están realizadas sólo sobre predios de una manzana escogida para el ejemplo, por lo cual es necesario realizar un *zoom in* adecuado a esta manzana como se muestra en la siguiente figura.

Figura 17: Zoom a la manzana y predios utilizados para las consultas



Para las consultas se debe escoger el modo consulta, setear un valor adecuado para el radio de búsqueda tomando en cuenta que estamos trabajando en metros. Luego se debe dar un clic sobre el predio que se desea consultar y dependiendo del valor asignado al radio de consulta, aparecerá los datos del predio elegido, del predio elegido y de los más cercanos o de todos los predios de esta manzana.

Figura 18: Resultado al hacer la consulta a un predio con una tolerancia baja

CONSULTAS				
ID	Predio	Propietario	Manzana	Sector
1	001	Vinicio Paredes	Manzana 1	Sector A

Figura 19: Resultado de la consulta con una tolerancia mayor

CONSULTAS				
ID	Predio	Propietario	Manzana	Sector
1	001	Vinicio Paredes	Manzana 1	Sector A
3	003	Domingo Hernandez	Manzana 1	Sector A
6	006	Renata Paredes	Manzana 1	Sector A
16	016	Fernando Escandon	Manzana 1	Sector A

### 3. Interfaz para grabar el archivo de configuración

La interfaz está implementada utilizando *PHP*, *JavaScript* y *HTML*; facilitará configurar el *Mapfile* pidiendo información necesaria como sistema de proyección, escalas, *layers* a desplegar, etc. Para visualizar el mapa es necesario elegir el *shape* y su tipo correctamente y necesariamente éstos dos datos para que se registre en el *Mapfile*; además para que conste como *layer* de consulta debe estar seleccionado en la sección “Capas a visualizar” de la aplicación.

Figura 20: Selección de *layers* y sus tipos

Capas a visualizar	Raster	Poligono	Linea	Punto	Anotación
<input checked="" type="checkbox"/> ibu_25mil	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
<input type="checkbox"/> pendiente_por	<input type="radio"/>				
<input checked="" type="checkbox"/> Limite_CRP_25MIL_WGS84_MDT	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
<input checked="" type="checkbox"/> Cantones_CRP_250MIL_WGS84	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
<input type="checkbox"/> Rios_Principales_CRP_25MIL_WGS84	<input type="radio"/>				
<input type="checkbox"/> Ciudades_CRP_25MIL_WGS84	<input type="radio"/>				
<input checked="" type="checkbox"/> Localidades_CRP_25MIL_WGS84	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
<input checked="" type="checkbox"/> Cantones_CRP_250MIL_WGS84_annotacion	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

El resto de campos necesarios para armar el *Mapfile* son pedidos de una manera amigable y en su mayoría pueden ser elegidos desde combos con lo que evitamos posibles confusiones en el ingreso de datos. La persona encargada de llenar esta información, debe tener conocimientos relacionados con los archivos *.shp* pero no necesita estar enterado de las etiquetas manejadas por *MapServer*.

## Enlace de ayuda

El objetivo es presentar una pequeña explicación de las opciones presentadas en la página, para que un usuario que no esté familiarizado con los parámetros que se piden, pueda navegar de una mejor manera.

Figura 21: Enlaces disponibles en la página web publicada



En el enlace de ayuda presentamos la siguiente información:

## Cómo navegar y consultar con MapServer

---

- **Para activar o desactivar los layers:**
  - Escoger modo Navegación en el combo “Modo del Mapa”
  - Desplegar el menú del grupo de layers situado a la izquierda superior del mapa.
  - Activar o desactivar la casilla del layer.
  - Dar clic en el botón “Actualizar”
- **Paneo**
  - Escoger modo Navegación en el combo “Modo del Mapa”
  - Activar la opción de paneo
  - Ajustar el valor del factor de zoom según se desee.
  - Con el ratón sobre el mapa, arrastar el puntero en la dirección que se desee navegar.
- **Zoom in**
  - Escoger modo Navegación en el combo “Modo del Mapa”
  - Activar la opción de zoom in
  - Ajustar el valor del factor de zoom según se desee.
  - Dar clic sobre el mapa en el punto a donde se desea acercar
- **Zoom out**
  - Escoger modo Navegación en el combo “Modo del Mapa”
  - Activar la opción de zoom out
  - Ajustar el valor del factor de zoom según se desee.
  - Dar clic sobre en mapa en el punto de donde se desea alejar
- **Recentrar**
  - Escoger modo modo Navegación en el combo “Modo del Mapa”
  - Dar clic sobre la opción recentrar

- **Factor Zoom**
  - Valor del tamaño del cuadro de zoom
- **Clic x y**
  - Presenta las coordenadas del punto en el mapa en donde se dio clic
- **Consulta y Consulta N**
  - Escoger modo de Consulta o Consulta N en el combo “Modo del Mapa”
  - Setear los valores de tolerancia (máxima distancia entre el clic del mapa y un *feature* aceptado como coincidente)
  - Dar clic dentro de la ciudad de Cuenca.
- **Consulta por ítem y consulta por ítem N**
  - Escoger modo de consulta por ítem o por ítem N
  - Setear los valores de tolerancia a cero.
  - En layer de consulta poner Manzanas o Predios
  - En ítem de consulta poner un campo del dbf correspondiente al layer de consulta.
  - En la cadena de consulta poner un valor del ítem de consulta
  - Dar clic en el botón Actualizar
- **Consulta por feature y por feature N**
  - Escoger modo de consulta por feature y por feature N
  - Setear los valores de tolerancia según los requerimientos
  - Dar clic dentro de la ciudad de Cuenca o en el botón Actualizar
- **Consulta por índice**
  - Escoger el modo de consulta por índice
  - Poner el índice del shape que se quiere consultar.
  - Dar clic dentro de la ciudad de Cuenca o en el botón Actualizar

### **Conclusiones:**

La elección de las alternativas de instalación y configuración de MapServer presentadas, dependerá de las necesidades de cada usuario. De igual manera se ha presentado las opciones que presenta MapServer y con ejemplos detallamos su funcionamiento, esto facilitará el entendimiento para aquellas personas que no estén familiarizadas con el software. Para la utilización de las consultas, es necesario tener conocimientos mínimos de los conceptos explicados en capítulos anteriores y comprender la diferencia entre consulta espacial y por atributo para poder elegir el tipo de consulta con el cual se requiere trabajar.

El poder vincular la cartografía a una base de datos de una manera sencilla, es una de las grandes potencialidades de MapServer, pero el éxito de todo esto dependerá de la información que se tenga y del uso que se le dé.

La aplicación facilitará el proceso de publicación de la cartografía digital pero se requiere de personal preparado y con conocimientos adecuados para cumplir con esta tarea.

## Conclusiones

- Dentro de las herramientas que se evaluaron para Web Mapping se concluyó que la mejor es MapServer, por la cantidad de características y escalabilidad.
- Cuando se despliega información de tipo raster, el tiempo de respuesta del servidor puede incrementarse, esto dependerá del número de *layers* raster a servir. Los ejemplos de la tesis trabajan con tres *layers* tipo raster.
- La implementación de una aplicación Web Mapping con MapServer no tiene costo; ésta es la razón por la cual se encuentra muy difundida en países como España, Estados Unidos, y Canadá. En nuestro país se está comenzando con proyectos de este tipo siendo la Universidad del Azuay la primera en implementarlos con la presente tesis.
- Al empezar el desarrollo de nuestro proyecto de tesis, no existía mayor información sobre MapServer, pero conforme ha transcurrido el tiempo esto ha cambiado, inclusive se ha publicado un libro siendo éste una de las primeras obras de *Open Source* escrito, así podemos concluir que en la actualidad existe mayor variedad de información sobre la herramienta.
- Como la creación de archivos map requiere de conocimientos sobre el manejo de MapServer se concluyó que se debía realizar una aplicación *web* para la creación del mismo y de esta forma abstraer al usuario final del aprendizaje de las sentencias para crear este archivo. Esto ayudó en gran manera al uso y difusión de la herramienta.
- La herramienta de desarrollo de archivos map es un ejemplo básico de lo que se podría hacer, pero no es una aplicación con todas las características que se requiere, una herramienta completa podría ser un nuevo tema de tesis.

## Recomendaciones

- Una vez concluido nuestro estudio podemos recomendar que para el manejo e implementación de MapServer, basándonos en las características presentadas, es necesario tener conocimientos básicos de Red Hat Enterprise Linux como: manejo e instalación de de paquetes, activación de librerías, configuración de servicios, etc. Se recomienda también tener conocimientos previos de tipos de datos SIG, sistemas de proyección, sistemas de coordenadas, manejo de escalas, MySQL, PHP, JavaScript, HTML, y PHPMapscript para poder avanzar más rápidamente en la instalación y configuración de MapServer.
- La instalación del software, como hemos visto, se puede hacer de dos formas: paquete por paquete o por módulo FGS. En el primer caso el proceso es lento y tedioso pero ofrece mayores alternativas de configuración de paquetes. Por módulo FGS, el proceso es mucho más rápido pero el software se instala con las opciones por defecto. Lo recomendado es aplicar la primera alternativa, pero si se tiene un servidor configurado anteriormente y no es posible re-configurarlo (como fue nuestro caso), lo recomendable es instalar el paquete FGS ya que este coloca todo lo necesario independientemente de lo que se tenga ya el servidor.
- Las bondades de este tipo software como de cualquier otro son aprovechadas eficientemente cuando la información de la que se dispone es de calidad y está relacionada. Recomendamos que para incrementar sus potencialidades necesariamente se debe ligar el servidor de mapas a un lenguaje de programación para que MapServer no sea un simple visualizador sino una verdadera herramienta Web Mapping.
- Para publicar mapas en Internet no es necesario un servidor dedicado exclusivamente para este tipo de información, pero en caso de que el volumen de información sea considerable recomendamos contar con un servidor que únicamente maneje cartografía digital ya que esto se verá reflejado en los tiempos de respuesta. Los requerimientos mínimos, para ejecutar MapServer son: procesador Pentium IV de 1.7 MHz y 1 GB de RAM .La aplicación de la

presente tesis se ejecuta sobre un servidor IBM Intel Xeon de 2.8GHz con 4 GB de RAM, y con dos discos duros SCSI de 140 GB cada uno.

- Se recomienda utilizar la versión de MapServer superior o igual a la 4.6 ya que presenta mayor estabilidad para la implementación de base de datos espaciales. La que se empleó en los ejemplos de esta tesis fue el paquete FGS 4.6 con el módulo MySQL 4.3.11 y MapServer 4.6.

## REFERENCIA BIBLIOGRÁFICA

### Referencias Capítulo I

- [AA01] [http://gis.sopde.es/cursosgis/DHTML/que\\_2\\_2.html](http://gis.sopde.es/cursosgis/DHTML/que_2_2.html), Título: ¿Qué es un Sistema de Información Geográfica?, Fecha de ingreso: 20-05-2005
- [AA02] Libro: Sistemas de Información Geográfico, Joaquín Bosque Sendra, Segunda edición, Septiembre 1997, página 33
- [AA03] <http://usuarios.lycos.es/kinei/mapas.htm>, Título: Todo Geografía, Autor: Kinei – todogeografia.com, Fecha de ingreso: 06-11-2005
- [AA04] [http://gis.sopde.es/cursosgis/DHTML/que\\_2\\_2.html](http://gis.sopde.es/cursosgis/DHTML/que_2_2.html), Título: ¿Qué es un Sistema de Información Geográfica?, Fecha de ingreso: 20-05-2005
- [AA05] [http://es.wikipedia.org/wiki/Proyecci%C3%B3n\\_de\\_Mercator](http://es.wikipedia.org/wiki/Proyecci%C3%B3n_de_Mercator), Título: Proyección de Mercator, Autor: Wikipedia, Fecha de ingreso: 12-11-2005
- [AA06] [http://es.wikipedia.org/wiki/Proyecci%C3%B3n\\_de\\_Peter](http://es.wikipedia.org/wiki/Proyecci%C3%B3n_de_Peter), Título: Proyección de Peters, Autor: Wikipedia, Fecha de ingreso:12-11-2005
- [AA07] <http://club.telepolis.com/geografo/general/pconica.htm#simple>, Título: Proyecciones cónicas, Autor: Santiago Pastrana, Fecha de ingreso:12-11-2005
- [AA08] <http://club.telepolis.com/geografo/general/pcenital.htm#ortografica>, Título: Proyecciones cenitales, Autor: Santiago Pastrana, Fecha de ingreso:12-11-2005
- [AA09] [http://www.geoinformacion.com/ediciones/1998/septiembre/proyeccion\\_p.html](http://www.geoinformacion.com/ediciones/1998/septiembre/proyeccion_p.html), Título: Proyecciones cartográficas, Autor: Carlos Pattillo - Director, Centro de Percepción Remota, Pontificia Universidad Católica de Chile, Fecha de Ingreso: 02-11-2005

- [AA10] (Tutorial de prácticas ArcGIS, Autor: Ing. Paúl Ochoa A. p. 8).
- [AA11] <http://gemini.udistrital.edu.co/comunidad/profesores/rfranco/metadatos.htm>,  
Título: Sistemas de Información Geográfica, Autor: Rodolfo Franco  
rfranco@udistrital.edu.co, Fecha de ingreso: 02-11-2005
- [AA12] <http://mapserver.inegi.gob.mx/geografia/espanol/normatividad/metadatos/index.cfm?c=631>, Título: Metadatos de Datos Geográficos, Autor: Instituto Nacional de Estadística, Geografía e Informática, Fecha de ingreso: 05-05-2005
- [AA13] <http://cndg.clearinghouse.gub.uy/documentos/recetario/capitulo05.aspx>  
Título: Visualización de datos geoespaciales/Web Mapping, Fecha de ingreso: 10-10-2005
- [AA14] [http://cndg.clearinghouse.gub.uy/documentos/capitulo04\\_1.aspx.htm](http://cndg.clearinghouse.gub.uy/documentos/capitulo04_1.aspx.htm)  
Título: Catálogo de datos geoespaciales / Localización de los datos, Fecha de ingreso: 5-10-2005
- [AA15] <http://www.ub.es/geocrit/sn/sn-170-61.htm> Título: Infraestructura De Datos Espaciales (Ide). Definición Y Desarrollo Actual En España; Fecha de ingreso: 11-04-2005
- [AA16] <http://www.opengeospatial.org>, Título: Especificaciones OGC, Servicios y Componentes, Fecha de ingreso: 11-27-2005
- [AA17] <http://www.ub.es/geocrit/sn/sn-170-61.htm> Título: Infraestructura De Datos Espaciales (Ide). Definición Y Desarrollo Actual En España; Fecha de ingreso: 11-04-2005
- [AA18] [http://es.wikipedia.org/wiki/Coordenadas\\_UTM](http://es.wikipedia.org/wiki/Coordenadas_UTM), Título: Coordenadas UTM, Fecha de ingreso: 29-01-2006
- [AA19] [http://www.incom.cl/arch\\_pdf/downloads/semsig-ab2004/sgm-flores-1.pdf](http://www.incom.cl/arch_pdf/downloads/semsig-ab2004/sgm-flores-1.pdf),  
Título: Geodesia y Cartografía Municipal “Parámetros de Transformación y Actualización de Datos con Tecnología GPS” Autor: Daniel Flores R., Fecha de ingreso: 25-01-2006 .

## Referencias Capítulo II

[BA01] <http://www.alov.org/index.html> Título: ALOV Overview, Fecha de ingreso:10-09-2005

[BA02] <http://www.gisdevelopment.net/technology/gis/ma03230pf.htm> Título: Implementing Internet GIS with Java based Client/Server Environment, Fecha de ingreso: 22-11-2005

[BA03][http://www.geoict.net/GSN\\_main\\_ogc.htm](http://www.geoict.net/GSN_main_ogc.htm), Título: GeoServNet, Autor: Open GIS Consortium, Inc, Fecha de ingreso: 14-11-2005

[BA04]<http://www.isprs.org/istanbul2004/comm7/papers/129.pdf>, Título: Use Image Streaming Technologies To Present High Resolution Images On The Internet, Fecha de ingreso: 16-11-2005

[BA05]<http://www.iseis.org/EIA/abstract.asp?no=04080&sn=2>, Título: International Society For Environmental Information Sciences, Autor: ISEIS, Fecha de ingreso:, 14-11-2005

[BA06]<http://www.geotango.com/products/geoservnet.htm>, Título: GeoServNet, Autor: GeoTango, Inc, Fecha de ingreso: 16-11-2005

[BA07][www.ogcnetwork.org/docs/03-050r1.pdf](http://www.ogcnetwork.org/docs/03-050r1.pdf), Titulo: OpenGIS® Web Map Server Cookbook, Autor: Open GIS Consortium, Fecha de ingreso: 10-11-2005

[BA08][www.mapserver.gis.umn](http://www.mapserver.gis.umn), Título: MapServer Home Page, Fecha de ingreso: 14-03-2005.

[BA09] Traducido del libro: Título: Beginning MapServer Open Source GIS Development, Autor: Kropla Hill, Fecha de edición: Agosto-2004, Página: XXV, XXVI.

[BA10][http://mapserver.gis.umn.edu/doc/getstarted-howto\\_es.html](http://mapserver.gis.umn.edu/doc/getstarted-howto_es.html), Título: Comenzar  
Con Mapserver COMO, Fecha de ingreso: 25-04-2005

### Referencias Capítulo III

[CA01][http://mapserver.gis.umn.edu/doc/getstarted-howto\\_es.html](http://mapserver.gis.umn.edu/doc/getstarted-howto_es.html), Título: Comenzar Con Mapserver COMO, Fecha de ingreso: 25-04-2005

[CA02]<http://www.boutell.com>, Título: GD Graphics Library, Autor: Boutell.Com, Fecha de ingreso: 05-12-2005

[CA03]<http://www.linuxbase.org/LSBWiki/Freetype>, Título: Freetype, Autor : LSB Workgroup Wiki, Fecha de ingreso: 05-12-2005

[CA04]<http://freetype.sourceforge.net/freetype2/index.html>, Título: The FreeType Project, Fecha de ingreso: 05-12-2005

[CA05]<http://en.wikipedia.org/wiki/Libjpeg>, Título: Libjpeg - Wikipedia, the free encyclopedia, Fecha de ingreso: 05-12-2005

[CA06]<http://www.libpng.org/pub/png/libpng.html> Título: libpng, Fecha de ingreso: 2-12-2005

[CA07]<http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=es&q=define:PNG> Título: Definiciones de PNG en la web, Fecha de ingreso: 2-12-2005

[CA08]<http://www.zlib.net>, Título: zlib Home Site, Autor: Greg Roelofs, Fecha de ingreso: 02-12-2005

[CA09]<http://www.remotesensing.org/gdal>, Título: GDAL - Geospatial Data Abstraction Library, Fecha de ingreso: 02-12-2005

[CA10]Título: Beginning MapServer Open Source GIS Development, Autor: Kropla Hill, Fecha de edición: Agosto-2004, Página: 3

[CA11]<http://ming.sourceforge.net/> Título: Ming - a SWF output library and PHP module Autor: Fecha de ingreso: 30-10-2005

[CA12]<http://mccrypt.sourceforge.net>, Título: Mccrypt, Autor: John Smith, Fecha de ingreso: 02-12-2005

[CA13]<http://mccrypt.hellug.gr>, Título: Mccrypt, Autor: Nikos Mavroyanopoulos autor de las librerías, Fecha de ingreso: 02-12-2005

[CA14]<http://www.linuxbase.org/LSBWiki/DesktopWG>, Título: DesktopWG, Autor: LSB Workgroup Wiki, Fecha de ingreso: 02-12-2005

[CA15]<http://directory.fsf.org/libgeotiff.html>, Título: libgeotiff, Autor: Free Software Foundation, Inc., Fecha de ingreso: 05-12-2005

[CA16] [http://tonelli.sns.it/pub/wavelets/M\\_D\\_Adams/](http://tonelli.sns.it/pub/wavelets/M_D_Adams/) Título: Jasper Software Referente Manual Autor: Michael David Adams, Fecha de ingreso: 05-12-2005

[CA17] <http://geos.refractive.net/> Título: GEOS, Fecha de ingreso: 13-01-2006

[CA18]<http://mapserver.gis.umn.edu/doc46/phpmapscript-class-guide.html>, Título: MapServer PHP/MapScript Class Reference - Version 4.6, Autor: DM Solutions Group Inc, Fecha de ingreso: 11-12-2005

[CA19]<http://www.remotesensing.org/libtiff/> Título: LibTIFF – TIFF Library and Utilities Autor: Fecha de ingreso: 2-12-2005

[CA20]<http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=es&q=define:Bzip2> Título: Definiciones para Bzip2 en la web, Fecha de ingreso: 2-12-2005

[CA21]<http://www.google.com.ec/search?hl=es&lr=&ie=UTF8&oi=defmore&defl=es&q=define:PostgreSQL> , Título: Definiciones de PostgreSQL en la web, Fecha de ingreso: 2-12-2005

[CA22]<http://www.etsimo.uniovi.es/tcl/tutorial/cap1.html>, Título: Capítulo 1: Tcl/Tk, Fecha de ingreso: 2-12-2005

[CA23]<http://www.maptools.org/fgs/index.phtml>; Título: FGS.MapTools.org; Fecha de ingreso: 10-10-2005

[CA24]<http://rpm.pbone.net/index.php3/stat/4/idpl/2541641/com/apache-base-2.2.0-6.i386.rpm.html>, Título: RPM Search apache-base-2.2.0-6.i386.rpm, Autor: SourceForge™.Net, Fecha de ingreso: 17-12-2005

[CA25]<http://expat.sourceforge.net/>, Título: The Expat XML Parser, Autor: SourceForge™.Net, Fecha de ingreso: 04-01-2006

[CA26]<http://directory.fsf.org/localization/libiconv.html> Título: Libiconv - Converts between character encodings; Fecha de ingreso: 22-12-2005

[CA27]<http://rpm.pbone.net/index.php3/stat/4/idpl/1339818/com/netcdf-3.5.1-1.rh80.bio.i386.rpm.html>, Título: RPM Search netcdf-3.5.1-1.rh80.bio.i386.rpm, Autor: RPM Phone-net, Fecha de ingreso: 04-01-2006

[CA28]<http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=en&q=define:OpenSSL> Título: Definiciones de OpenSSL en la web en inglés, Fecha de ingreso: 22-2-2005

[CA29]<http://www.unixodbc.org/>, Título: The unixODBC Project home page, Fecha de ingreso: 27-12-2005

[CA30]<http://www.google.com/search?hl=en&lr=&oi=defmore&defl=en&q=define:Xerces> Título: Definitions of Xerces on the web; Fecha de ingreso: 30-12-2005

[CA31][http://www.christianyely.com/FedoraC2-Mapserver4.2.3-PostgreSQL7.4.5-PostGIS0.9.0\\_v13.txt](http://www.christianyely.com/FedoraC2-Mapserver4.2.3-PostgreSQL7.4.5-PostGIS0.9.0_v13.txt); Autor: Christian Gonzalez (christiangda@cantv.net); Ubicación: Caracas/Venezuela; Fecha de creación: 30/09/2004

[CA32]<http://www.maptools.org/fgs/index.phtml?page=install.html>; Título: FGS.MapTools.org; Fecha de ingreso: 15-10-2005

## Referencias Capítulo IV

[DD01] Título: Beginning MapServer Open Source GIS Development, Autor: Kropla Hill, Fecha de edición: Agosto-2004, Páginas: 15 - 166

[DD02] Título: Beginning MapServer Open Source GIS Development, Autor: Kropla Hill, Fecha de edición: Agosto-2004, Páginas: 207-290

[DD03] <http://hypnos.cbs.umn.edu/tutorial> Titulo: MapServer 4.6 Tutorial; Fecha de ingreso: 12-06-2005

[DD04] <http://www.apress.com> Título: APRESS; Fecha de ingreso: 01-05-2006

[DD05] <http://mapserver.gis.umn.edu/docs/reference/mapfile>; Título: Mapfile Referente; Fecha de ingreso: 10-10-2005

[DD06] <http://www.uazuay.edu.ec/estudios/sistemas/web.htm> Título: Programación Web; Fecha de ingreso: 20-02-2006

[DD07] [http://es.tldp.org/Manuales-LuCAS/manual\\_PHP/manual\\_PHP/mysql/crear\\_bd\\_mysql.htm](http://es.tldp.org/Manuales-LuCAS/manual_PHP/manual_PHP/mysql/crear_bd_mysql.htm); Título: Creación de una base de datos en MYSQL; Fecha de ingreso: 25-11-2005

## Referencias del Glosario de Conceptos Técnicos

- [GG01] <http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=es&q=define:API>, Título: define:API - Búsqueda en Google, Fecha de ingreso: 12-10-2005
- [GG02] <http://www.planetside.co.uk/terrigen/guide/es/files.html>, Título: Trabajando con Archivos, Fecha de ingreso: 16-01-2006
- [GG03] <http://www.esri.com/software/arcgis/arcscde/> Título : ArcSDE Advanced Spatial Data Server; Autor: ESRI; Fecha de acceso: 17-01-2005
- [GG04] [www.telecable.es/personales/carlosmg1/glosario\\_a.htm](http://www.telecable.es/personales/carlosmg1/glosario_a.htm), Título: define:ASCII - Búsqueda en Google, Fecha de ingreso: 27-12-2005
- [GG05] <http://es.wikipedia.org/wiki/CERN>, Título: CERN – Wikipedia, Fecha de ingreso: 4-11-2005
- [GG06] [http://portal.opengeospatial.org/files/?artifact\\_id=8340](http://portal.opengeospatial.org/files/?artifact_id=8340), Título: OpenGIS<sup>®</sup> Filter Encoding Implementation Specification, Autor: Open Geospatial Consortium, Inc (2005), Fecha de ingreso: 21-09-2005
- [GG07] <http://www.learningservices.gcal.ac.uk/it/staff/definitions.html>, Título: Streaming, Autor: Jupitermedia Corporation, Fecha de ingreso: 19-11-2005
- [GG08] <http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=en&q=define:dict> Título: Definiciones de DICT en la web; Autor: Fecha de ingreso : 18-01-2006
- [GG09] [www.microsoft.com/presspass/features/2000/01-03xmlglossary.mspx](http://www.microsoft.com/presspass/features/2000/01-03xmlglossary.mspx), Título: define:DTDS - Búsqueda en Google, Fecha de ingreso: 13-11-2005
- [GG10] [www.tierradenomadas.com/diccionario.phtml](http://www.tierradenomadas.com/diccionario.phtml), Título: define:Dom - Búsqueda en Google, Fecha de ingreso: 17-11-2005

[GG11]<http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=es&q=define:File> Título: Definiciones de FILE en la web; Autor: Fecha de ingreso : 18-01-2006

[GG12] <http://es.wikipedia.org/wiki/FTP>, Título: define:FTP - Búsqueda en Google, Fecha de ingreso: 6-12-2005

[GG13]<http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=en&q=define:FTPS> Título: Definiciones de FTPS en la web; Autor: Fecha de ingreso: 18-01-2006

[GG14] <http://calview.casil.ucdavis.edu/glossary.html>, Título: Glossary - via CaSIL, Fecha de ingreso: 27-12-2005

[GG15]<http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=es&q=define:Gopher> Título: Definiciones de GOPHER en la web; Autor: Fecha de ingreso : 18-01-2006

[GG16] [www.chenico.com/glosariogh.htm](http://www.chenico.com/glosariogh.htm), Título: Definiciones de GPS en la Web, Fecha de ingreso: 14-10-2005

[GG17]<http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=es&q=define:HTTP> Título: Definiciones de HTTP en la web; Autor: Fecha de ingreso : 18-01-2006

[GG18]<http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=es&q=define:HTTPS> Título: Definiciones de HTTPS en la web; Autor: Fecha de ingreso : 18-01-2006

[GG19][http://www.google.com.ec/search?hl=es&lr=lang\\_es&oi=defmore&defl=es&q=define:Internet+Information+Server](http://www.google.com.ec/search?hl=es&lr=lang_es&oi=defmore&defl=es&q=define:Internet+Information+Server) Título: Definiciones de Internet Information Server en la web; Autor : Fecha de acceso: 17-01-2005

[GG20]<http://www.google.com.ec/search?q=define:JNI&hl=es&lr=&oi=definel&defl=en>, Título: define:JNI - Búsqueda en Google, Fecha de ingreso: 12-10-2005

[GG21] <http://www.jump-project.org/project.php?PID=JTS&SID=OVER> , Título: JTS Topology Suite (JTS). Fecha de acceso: 24-01-2005

[GG22]<http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=es&q=define:LDAP> Título: Definiciones de LDAP en la web; Autor: Fecha de ingreso : 18-01-2006

[GG23]<https://www.raf.com.sv/fotosenlinea/Glosario.php>, Título: define:LZW - Búsqueda en Google, Fecha de ingreso: 19-11-2005

[GG24]<http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=es&q=define:Mac> Título: Definiciones de MAC en la web; Autor: Fecha de ingreso : 17-01-2006

[GG25][http://www.maptools.org/php\\_mapscript/](http://www.maptools.org/php_mapscript/), Título: PHPMapScript.MapTools.org, Autor: DM Solutions Group, Fecha de ingreso: 10-10-2005

[GG26] <http://ogr.maptools.org/> Título: OGR Simple Feature Library; Autor: Doxygen; Fecha de Acceso:16-01-2005

[GG27]<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/odbc/hm/dasdkodbcoverview.asp>, Título: Welcome to the MSDN Library, Autor: MSDN, Fecha de ingreso: 27-12-2005

[GG28] <http://www.php.net/>, Título: PHP: Hypertext Preprocessor, Autor: The PHP Group, Fecha de ingreso: 10-10-2005

[GG29]<http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=en&q=define:quadtree> Título: Definiciones de quadtree en la web en inglés; Autor: Fecha de acceso: 12-12-2005

[GG30][krypton.mnsu.edu/~spiral/eta/glossary/indxGlossOOxml.html](http://krypton.mnsu.edu/~spiral/eta/glossary/indxGlossOOxml.html), Título: define:SAX - Búsqueda en Google, Fecha de ingreso: 20-11-2005

[GG31] <http://www.interceptcorporation.com/definition.shtml>, Título: Definitions of Terms, Autor: ©1993-2006 Intercept Corporation, Fecha de ingreso: 19-11-2005

[GG32]<http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=es&q=define:Telnet> Título: Definiciones de TELNET en la web; Autor: Fecha de ingreso : 18-01-2006

[GG33][www.galeon.com/filoesp/glosario/glos\\_F.htm](http://www.galeon.com/filoesp/glosario/glos_F.htm), Título: Glosario del E-business, Fecha de ingreso: 27-12-2005

[GG34] Tutorial de Prácticas ArcGIS preparado por el Ing. Paúl Ochoa, Universidad del Azuay – mayo 2005

<http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=es&q=define:TIN>, Título: define:TIN - Búsqueda en Google, Fecha de ingreso: 16-01-2006

[GG35]<http://www.google.com.ec/search?hl=es&lr=&oi=defmore&defl=es&q=define:Unicode> Título: Definiciones de UNICODE en la web; Autor: Fecha de ingreso: 17-01-2006

[GG36] <http://es.wikipedia.org/wiki/URI>, Título: define:URI - Búsqueda en Google, Fecha de ingreso: 14-12-2005

[GG37][www.buzoneo.info/diccionario\\_marketing/diccionario\\_marketing\\_u.php](http://www.buzoneo.info/diccionario_marketing/diccionario_marketing_u.php), Título: define:URL - Búsqueda en Google, Fecha de ingreso: 14-10-2005

[GG38] [www.laopinion.com/glossary/m.html](http://www.laopinion.com/glossary/m.html), Título: define:virtual machine, Fecha de ingreso: 14-10-2005

[GG39]<http://www.opengeospatial.org/specs/?page=specs>, Título: OpenGIS® Specifications, Autor: © 1994 - 2006 Open Geospatial Consortium, Inc., Fecha de ingreso: 21-09-2005

[GG40] <http://es.wikipedia.org/wiki/WWW>, Título: define:WWW - Búsqueda en Google, Fecha de ingreso: 4-11-2005

[GG41] [http://www.camaraalcoy.net/Servicios\\_web/glosario/Glosario/X.htm](http://www.camaraalcoy.net/Servicios_web/glosario/Glosario/X.htm), Título: define:XML - Búsqueda en Google, Fecha de ingreso: 13-11-2005

[GG42] [es.wikipedia.org/wiki/XPath](http://es.wikipedia.org/wiki/XPath), Título: define:XPath - Búsqueda en Google, Fecha de ingreso: 6-12-2005

[GG43] [es.wikipedia.org/wiki/XPointer](http://es.wikipedia.org/wiki/XPointer), Título: define:XPointer - Búsqueda en Google, Fecha de ingreso: 6-12-2005

## Glosario De Conceptos Técnicos

### A

#### API

*Application Programming Interface* - Interfaz de Programación de Aplicaciones es un conjunto de especificaciones de comunicación entre componentes software. Corresponde a una biblioteca o bibliotecas que ofrece el sistema operativo para que los programas puedan comunicarse con él e invocar sus servicios. <sup>[GG01]</sup>

#### Archivos World

Los archivos World guardan *toda* la configuración, con la excepción del terreno mismo. Aún cuando se pueden guardar los mapas de superficie y la configuración atmosférica por separado estas configuraciones se incluyen al guardar el archivo "world". <sup>[GG02]</sup>

#### ArcSDE

ArcSDE es un producto de software servidor para acceder masivamente a una gran cantidad de bases de datos geográficas multiusuario almacenadas en sistemas gestores de bases de datos relacionales (SGBDRs). Forma parte integrada de ArcGIS y un elemento central de cualquier solución GIS empresarial. Su rol principal es actuar como la puerta de acceso GIS a los datos espaciales guardados en un SGBDR. <sup>[GG03]</sup>

#### ASCII

Acrónimo de American Standard Code for Information Interchange (Código Estadounidense Estándar para el Intercambio de Información). Código utilizado por casi todos los ordenadores y sistemas para representar las letras, los números y los caracteres especiales. Este código asigna un valor alfanumérico a 128 números, utilizando 7 bits ( $2^7=128$ ) para cada uno de ellos. El código ASCII ampliado utiliza 8 bits, y puede representar 256 caracteres distintos ( $2^8=256$ ). <sup>[GG04]</sup>

## C

### **CERN**

La sigla CERN viene de su antiguo nombre Consejo Europeo para la Investigación Nuclear. Sitio donde se celebró la primera conferencia sobre World Wide Web y considerado el lugar de nacimiento de la tecnología de WWW. [GG05]

### **Codificación Filtrada**

Una expresión filtro es una estructura usada para restringir los valores de propiedad de un objeto con el propósito de identificar un subconjunto de instancias que determinen la forma de operar dicho objeto.

La codificación filtrada es un componente común que puede ser usado por varios servicios web de OGC. Algún servicio que solicite la habilidad para consultar objetos de un repositorio web puede hacerlo usando la codificación filtrada XML. Por ejemplo, en el proceso GetFeature se utiliza la codificación filtrada para definir las restricciones de la consulta. [GG06]

## D

### **Data stream**

Toda información (comandos de datos y controles) enviada sobre un dato enlazado usualmente es una operación simple de lectura o escritura. Es un flujo continuo de elementos de datos siendo transferidos, o intentando ser transferidos, en caracteres o en forma binaria (forma digital), usando un formato definido. [GG07]

### **DICT**

DICT es un protocolo de diccionario en la red creado por el Grupo de Desarrollo DICT. Su meta es superar el protocolo Webster y permitir a los clientes acceder a más diccionarios al mismo tiempo. [GG08]

## **DtD**

Document Type Definitions, es un conjunto de reglas de sintaxis para las etiquetas. Dice qué etiquetas pueden ser usadas en un documento, el orden en que ellas deben aparecer, qué etiquetas pueden aparecer dentro de otras, qué etiquetas tienen atributos, y así sucesivamente. XML no es en sí un lenguaje sino un sistema para definir lenguajes, no tiene un DTD universal como lo tiene HTML. Es por eso que cada industria u organización que desee usar XML para intercambio de datos puede definir su propio DTDs. <sup>[GG09]</sup>

## **DOM**

Document Object Model, Modelo de Objetos de Documento es al mismo tiempo una plataforma y un lenguaje neutral que permite a programas y scripts acceder y actualizar dinámicamente los contenidos, la estructura y el estilo de los documentos HTML y XML. Es independiente de cualquier lenguaje orientado a objetos. <sup>[GG10]</sup>

## **F**

### **FILE**

Define el acceso a un fichero FTP. <sup>[GG11]</sup>

### **FTP**

File Transfer Protocol o Protocolo de Transferencia de Archivos. Es el ideal para transferir datos por la red. Es un protocolo estandarizado de Internet. <sup>[GG12]</sup>

### **FTPS**

FTPS (File Transfer Protocol Security) es un estandar para software FTP en el cual los protocolos SSL/TLS se utilizan para asegurar el control y conexión de datos. No se lo debería confundir con el protocolo SFTP el cual es totalmente diferente. <sup>[GG13]</sup>

## **G**

### **GeoTiff**

Formato Tiff enriquecido, que tiene información georeferenciada. <sup>[GG14]</sup>

### **GOPHER**

Fue desarrollado por la Universidad de Minnesota, es una herramienta de búsqueda que presenta información en un sistema de menús jerárquicos parecidos a un índice. Se trata de un método de hacer menús de material disponible a través de Internet. El Gopher es un programa de estilo Cliente -Servidor, que requiere que el usuario tenga un programa cliente Gopher. Aunque se extendió rápidamente por todo el mundo, ha sido sustituido en los últimos años por el Hipertexto., también conocido como WWW (World Wide Web). <sup>[GG15]</sup>

## **GPS**

El Global Positioning System o Sistema de Posicionamiento Global, es un sistema de navegación basado en la recepción de señales de 24 satélites de la constelación Navstar puesta en órbita por el Ministerio de Defensa de los EE.UU. que giran alrededor de la tierra dos veces al día. Un receptor en tierra calcula su posición geográfica determinando su posición con respecto a un conjunto de al menos tres satélites. El receptor puede calcular la localización exacta, habitualmente con un error de un centímetro, de un objeto en la superficie de la tierra. <sup>[GG16]</sup>

## **H**

### **HTTP**

HTTP es el protocolo de la Web (WWW), usado en cada transacción. Las letras significan Hyper Text Transfer Protocol, es decir, protocolo de transferencia de hipertexto. El hipertexto es el contenido de las páginas web, y el protocolo de transferencia es el sistema mediante el cual se envían las peticiones de acceder a una

página web, y la respuesta de esa web, remitiendo la información que se verá en pantalla. Se basa en una arquitectura cliente / servidor. <sup>[GG17]</sup>

## **HTTPS**

HTTPS (Hyper Text Transfer Protocol Security) es la versión segura del protocolo HTTP. El sistema HTTPS utiliza un cifrado basado en las Secure Socket Layers (SSL) para crear un canal cifrado (cuyo nivel de cifrado depende del servidor remoto y del navegador utilizado por el cliente) más apropiado para el tráfico de información sensible que el protocolo HTTP. <sup>[GG18]</sup>

## **I**

### **IIS**

Internet Information Services (o Server), IIS, es una serie de servicios para los ordenadores que funcionan con Windows. Originalmente era parte del Option Pack para Windows NT. Luego fue integrado en otros sistemas operativos de Microsoft destinados a ofrecer servicios, como Windows 2000 o Windows Server 2003. Windows XP Profesional incluye una versión limitada de IIS. Los servicios que ofrece son: FTP, SMTP, NNTP y HTTP/HTTPS. <sup>[GG19]</sup>

## **J**

### **JNI**

Java Native Interface, Interfaz Nativa de Java. Es un marco de programación que permite al código de Java correr en la máquina virtual de Java (VM, Virtual Machine), para llamar y ser llamado por aplicaciones nativas (programas específicos para las plataformas de hardware y sistemas operativos) y librerías escritas en otros lenguajes como C, C++ y Ensamblador. <sup>[GG20]</sup>

## **JTS**

*Java Topology Suite* (JTS) es un API que provee un modelo de objeto especial y funciones geométricas Fundamentals; implementa el modelo geométrico definido en el *Simple Features Specification* para SQL del OpenGis. <sup>[GG21]</sup>

## **L**

### **LDAP**

LDAP (Lighweight Directory Access Protocol) en sí es un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red. LDAP puede considerarse una base de datos (aunque su sistema de almacenamiento puede ser otro diferente) al que pueden realizarse consultas. <sup>[GG22]</sup>

### **LZW**

Método de compresión de archivos desarrollado por Lempel, Ziv y Welch, que se basa en aprovechar la repetición de secuencias de código y codificarlas de una manera más simple, sin pérdidas para reducir el tamaño de archivos de imagen. <sup>[GG23]</sup>

## **M**

### **MAC**

Apple Macintosh (abreviado Mac) es el nombre de una serie de ordenadores fabricados y comercializados por Apple Computer desde 1984. Apple autorizó a otras compañías, como Motorola, Umax o PowerComputing para la fabricación de clones Macintosh en los 90, aunque en la actualidad sólo Apple comercializa ordenadores Macintosh. <sup>[GG24]</sup>

### **MapScript**

Es una interfaz que permite el acceso a las principales funcionalidades de MapServer desde una variedad de ambientes de programación. <sup>[GG25]</sup>

## **O**

### **OGR**

OGR es una librería de código abierto de C++ (y herramientas de línea de comandos) facilita la lectura (y en algunas ocasiones la escritura), el acceso a una variedad de formatos de archivos vector incluyendo archivos shape de la ESRI, PostGIS, Oracle Spatial, mid/mif de Mapinfo y otros. <sup>[GG26]</sup>

### **ODBC**

Open Database Connectivity, Conectividad abierta de bases de datos. Permite una máxima interoperabilidad, haciendo posible que una aplicación pueda acceder a datos en una diversidad de database management systems (DBMS) a través de una simple aplicación. Además, la aplicación será independiente de cualquier DBMS desde la que esté accediendo los datos. <sup>[GG27]</sup>

## **P**

### **PHP**

Es un lenguaje de scripting de uso general extensamente usado que está especialmente preparado para el desarrollo de Web y puede ser embebido en HTML. <sup>[GG28]</sup>

## **Q**

### **QUADTREE**

Es un índice espacial el cual descompone recursivamente un conjunto de datos (imágenes por ejemplo) en celdas de diferentes tamaños hasta que cada celda tenga un

valor homogéneo. Los Quadrees se utilizan frecuentemente para guardar datos raster.<sup>[GG29]</sup>

## S

### SAX

“Simple API for XML” es un compilador API de acceso serial para XML. Un compilador SAX maneja la información XML en forma unidireccional, esto es, no puede renegociar un nodo sin primero establecer un nuevo enlace al documento y recompilarlo. Es más rápido que el compilador DOM. Provee una interfaz estandarizada para la interacción de aplicaciones con muchas herramientas XML.<sup>[GG30]</sup>

## SCV

Un archivo de datos *Comma Separated Value* es un archivo físico estructurado ASCII que contiene registros cuyos valores están separados o delimitados por comas. Son usados para una representación portable de una base de datos o una hoja de cálculo.<sup>[GG31]</sup>

## T

### TELNET

Tele Network. Tele Red. Se trata de un protocolo de Internet que sirve para conectarse a otros sistemas de ordenadores remotos en red, incluso los más potentes que el propio, permitiendo usar sus programas o recibir su información. Telnet permite acceder mediante una red a otra máquina y manejarla, siempre en modo terminal (no hay gráficos).<sup>[GG32]</sup>

## **TIFF**

Tagged Image File Format, formato de etiquetado de archivos de imágenes. Es un formato de archivos gráficos comprimido desarrollado por Aldus como formato estándar internacional. Lamentablemente, existen diferentes versiones de TIFF. Las diferencias más importantes se encuentran en la manera en que los formatos para MAC y PC comprimen los datos. <sup>[GG33]</sup>

## **Tin**

Triangular Irregular Network. Estructura espacial de datos generada por la partición del espacio en triángulos ajenos.

Modelos TIN: Son representaciones que modelizan digitalmente el terreno en tres dimensiones, están constituidos por una serie de puntos irregularmente distribuidos en un determinado espacio territorial, estos contienen coordenadas x, y, z, con estos puntos, el TIN traza líneas que delimitan a su vez triángulos, los cuales representan la superficie del territorio analizado. <sup>[GG34]</sup>

## **U**

### **UNICODE**

Unicode es una norma de codificación de caracteres. Su objetivo es asignar a cada posible carácter de cada posible lenguaje un número y nombre único, a diferencia de la mayor parte de los juegos ISO, que sólo definen los necesarios para un idioma o zona geográfica. <sup>[GG35]</sup>

### **URI**

Uniform Resource Identifier, identificador unificado de recursos. Se utiliza también el término URL para este concepto. <sup>[GG36]</sup>

**URL** Uniform Resource Locator, Localizador Uniforme de Recursos.

Es la dirección de Internet, incluye: "http" que indica el nombre del protocolo usado, "www" que es el nombre del servidor, "dir" es un directorio, "subdir" un subdirectorio y "file" el nombre de un archivo. Es la manera estándar de asignar direcciones de cualquier recurso en Internet que forma parte del WWW. <sup>[GG37]</sup>

## **V**

### **Virtual Machine**

Máquina virtual. Las capacidades de procesamiento de un sistema de computación creado por medio de software y en ocasiones mediante hardware en una computadora distinta. <sup>[GG38]</sup>

## **W**

### **WMC**

Web Map Context, Contexto del Mapa.

Describe como un grupo de especificaciones de uno o más mapas de uno o más servidores de mapas pueden ser descritos en una plataforma portable e independiente y con un formato de almacenamiento en un repositorio o para la transmisión entre clientes. Esta descripción es conocida como un "Web Map Context Document" o simplemente un "Contexto". Un documento de contexto incluye información acerca de los servidores proveyendo los layers dispuestos en el mapa global, el bounding box, la proyección del mapa, suficientes metadatos para que el software del cliente pueda reproducir el mapa y adicionalmente metadatos usados para anotar o describir el mapa y su proveniencia. Un documento de contexto está estructurado usando eXtensible Markup Language (XML). <sup>[GG39]</sup>

## **WWW**

La World Wide Web (telaraña o malla mundial), conocida como la Web, WWW o W3, creada a principios de la década de los 90. Sistema de arquitectura cliente/servidor creada en un principio por el CERN y permite la distribución y obtención de información en Internet basado en hipertexto e hipermedia (combinación de textos con gráficos, imágenes, animaciones e incluso música). Para ver la información se utiliza una aplicación llamada navegador web para extraer elementos de información llamados "documentos" o "páginas web", de los servidores web o "sitios" y mostrarlos en la pantalla del usuario. Actualmente su desarrollo está a cargo de la Organización World Wide Web (W3O).<sup>[GG40]</sup>

## **X**

### **XML** Extensible Markup Language

Lenguaje universal de marcado para documentos estructurados y datos en la web, desarrollado por el W3 Consortium para permitir la descripción de información contenida en el WWW a través de estándares y formatos comunes. Diseñado con la intención de reemplazar al estándar actual HTML. Más amplio, más rico y más dinámico que HTML. No solo es un lenguaje de marcado, sino también un metalenguaje que permite describir otros lenguajes de marcado. Permite el uso ilimitado de los tipos de datos que pueden utilizarse en Internet, lo cual resuelve los problemas que surgen entre las organizaciones que deben intercambiar datos procedentes de estándares distintos.<sup>[GG41]</sup>

### **XPath**

XPath es un lenguaje basado en XML que permite seleccionar subconjuntos de un documento XML. La idea es parecida a las expresiones regulares para seleccionar partes de un texto sin atributos (plain text).<sup>[GG42]</sup>

## **XPointer**

Lenguaje de punteros XML es un estándar del World Wide Web Consortium (W3C) que proporciona una forma única de identificar fragmentos de un documento XML para realizar vínculos. <sup>[GG43]</sup>

## Diccionario de Siglas

### A

AGI *Association for Geographic Information*, Asociación para Información Geográfica

### C

CIMI *Consortium for the Interchange of Museum Information*

CGI *Common Gateway Interface*, Interfaz de Pasarela Común

### F

FGDC *Federal Geographic Data Committee*, Comité Federal de Datos Geográficos

### G

GPL *General Public License*, Licencia Pública General

GPS *Global Position System*, Sistema de Posicionamiento Global

### I

IERSE Instituto de Estudios de Régimen Seccional del Ecuador

### N

N.C.G.I.A. *The National Center for Geographic Information and Analysis*, Centro Nacional para Información Geográfica y Análisis

### O

OGC CAT Catálogo del OGC

OGC *Open GIS Consortium*, Consorcio de GIS Libre

## **P**

PSAD56 o SAM56 *Provisional South American 1956*, Provisional de 1956 para América del Sur

## **S**

SIG Sistemas de Información Geográfica

## **U**

URL *Uniform Resource Locator*, Localizador Uniforme de Recursos

USMARC U. S. *Machine Readable Catalog*, Catálogo Legible de Máquina

UTM *Universal Transversal Mercator*

## **W**

WCS *Web Coverage Service*

WFS *Web Feature Service*

WGS84 *World Geodetic System* de 1984, Sistema Geodésico Mundial

WMS *Web Map Service*

WWW *World Wide Web*

## **X**

XML *Extensible Markup Lenguaje*