



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

Una solución Web-SIG basada en tecnología Open Source para la divulgación de información cartográfica de la Coordinación General de Adaptación al Cambio Climático (CGACC), del Instituto Nacional de Ecología y Cambio Climático (INECC)

INFORME DE TRABAJO

**QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN COMPUTACIÓN**

PRESENTA

JESUS YAÑEZ VILLA

ASESOR: M. C. ALEJANDRO VELÁZQUEZ MENA



CIUDAD UNIVERSITARIA, FEBRERO DE 2015

DEDICATORIA

A mis padres, como un testimonio de cariño y eterno agradecimiento por mi existencia, valores morales y formación profesional. Porque sin escatimar esfuerzo alguno, han sacrificado gran parte de su vida para formarme y porque nunca podré pagar todos sus desvelos, ni aún con las riquezas más grandes del mundo.
Por lo que soy y por todo el tiempo que les robé pensando en mí...

A mis hijas Ivana Sofia y Dana Valeria, que son el motivo y la razón que me ha llevado a seguir superándome día a día, para alcanzar mis más apreciados ideales de superación, ellas fueron quienes en los momentos más difíciles me dieron su amor y comprensión para poder superarlos, quiero también dejar a cada una de ellas una enseñanza que cuando se quiere alcanzar algo en la vida, no hay obstáculo que lo impida para poder lograrlo...

A todos los que directa e indirectamente me ayudaron a la realización de este proyecto...

“Lo importante en la vida no es el triunfo sino la lucha. Lo esencial no es haber vencido, sino haber luchado bien.”
(Barón Pierre de Coubertin)

AGRADECIMIENTOS

A la UNAM por haberme dado cobijo y por las lecciones que aprendí en ella, asimismo, por haberme dado su voto de confianza y por todo el apoyo otorgado a mi persona.

Al INECC por el apoyo brindado para la realización de este trabajo.

Quiero agradecer muy especialmente M. C. Nora Elizabeth Esquivel Esquivel por su invaluable apoyo y confianza que me han brindado.

A mi asesor de tesis, el M. C. Alejandro Velázquez Mena por todo el apoyo que ha brindado.

A mis hermanas: Adriana y Laura Patricia por haberme apoyado todo este tiempo. Gracias por sus consejos y ánimos constantes.

A mis amigos, porque sin ellos no hubiera sido lo mismo.

Gracias, eternamente gracias, a todos.

ÍNDICE

Dedicatoria.....	i
Agradecimientos.....	ii
Índice.....	iii
Índice de ilustraciones.....	v
Resumen.....	viii
Introducción.....	ix
Antecedentes.....	xi
Objetivos.....	xiv
• Objetivo General.....	xiv
• Objetivos Específicos.....	xiv
Capítulo 1. Marco Teórico.....	1
1.1 Sistemas de Información Geográfica (SIG).....	1
1.2 Open Geospatial Consortium (OGC).....	2
1.2.1 Especificaciones OGC.....	3
1.3 Infraestructura de datos espaciales (IDE).....	4
1.4 European Petroleum Survey Group (EPSG).....	6
1.5 The Open Source Geospatial Foundation (OSGeo).....	7
1.6 Software SIG – Free and Open Source Software (FLOSS).....	8
1.6.1 Servidores de Mapas.....	9
1.6.1.1 MapServer.....	11
1.6.2 Base de Datos Geográfica.....	14
1.6.2.1 PostgreSQL.....	16
1.6.2.2 PostGIS.....	17
1.6.3 Frameworks de desarrollo.....	18
1.6.3.1 OpenLayers.....	18
1.6.3.2 Heron Mapping Client (Heron –MC).....	19
1.6.4 SIG de escritorio.....	22
1.6.4.1 gvSIG.....	22

Capítulo 2. Metodología.....	23
2.1 Definición de contenido sobre información geográfica.....	24
2.2 Entorno tecnológico y de comunicaciones.....	29
2.3 Diseño de la base de datos geográfica.....	36
2.4 Desarrollo de la interfaz del visor de mapas.....	44
2.5 Implementación.....	59
2.5.1 Instalación de Apache.....	62
2.5.2 Instalación de PostgreSQL.....	64
2.5.3 Instalación de PostGIS.....	67
2.5.4 Instalación de PHP.....	74
2.5.5 Instalación de MapServer.....	77
Capítulo 3. Resultados.....	83
Capítulo 4. Conclusiones y trabajo futuro.....	88
Referencias.....	90
Glosario.....	94

ÍNDICE DE ILUSTRACIONES

Figura 1.-	Componentes de un Sistema de Información geográfica.....	1
Figura 2.-	Marco conceptual de Software Libre y de Código Abierto.....	9
Figura 3.-	Esquema típico de un Servidor de Mapas.....	9
Figura 4.-	Descripción gráfica de los componentes en un entorno MapServer.....	12
Figura 5.-	Jerarquía de los objetos, Mapfile.....	13
Figura 6.-	Arquitectura Heron Mapping Client.....	21
Figura 7.-	Diagrama de flujo de las tecnologías Open Source utilizadas en este proyecto.....	24
Figura 8.-	Clasificación del acervo cartográfico.....	29
Figura 9.-	Arquitectura de la aplicación.....	30
Figura 10.-	Tendencia del uso de Servidores Web.....	32
Figura 11.-	Porcentaje de sitios Web usando diferentes Servidores Web.....	35
Figura 12.-	Definición de un objeto espacial en el estándar OGC.....	35
Figura 13.-	Definición de un objeto espacial en el estándar OGC.....	37
Figura 14.-	Modelo conceptual para el Sistema de consulta de mapas digitales sobre planificación territorial y biodiversidad, siguiendo el estándar OGC.....	38
Figura 15.-	Modelo lógico para el Sistema de consulta de mapas digitales sobre planificación territorial y biodiversidad, siguiendo el estándar OGC.....	40
Figura 16.-	Diagrama de flujo de la implementación de la base de datos.....	43
Figura 17.-	Creación de vistas, gvSIG.....	45
Figura 18.-	Vista; cargando y visualizando información almacenada en la base de datos espacial, gvSIG.....	46
Figura 19.-	Conexión base de datos espacial, gvSIG.....	46
Figura 20.-	Servidor local y servicio OGC, gvSIG.....	48
Figura 21.-	Definiendo el directorio y nombre del archivo MapFile, gvSIG.....	49
Figura 22.-	Parámetros del Servicio, gvSIG.....	50
Figura 23.-	Publicación del Recurso, gvSIG.....	51
Figura 24.-	Publicación de vista, MapFile asociado, gvSIG.....	54
Figura 25.-	Descripción específica de la estructura del visor de mapas.....	55
Figura 26.-	Lenguajes de programación.....	57

Figura 27.-	Funciones desempeñadas por cada script.....	58
Figura 28.-	Interfaz del visor de mapas. Categoría de depresiones, tormentas y huracanes, Océano Pacífico.....	58
Figura 29.-	Interfaz del visor de mapas. Categoría de depresiones, tormentas y huracanes, Océano Atlántico.....	59
Figura 30.-	Raid 10.....	61
Figura 31.-	Escaneo de puerto para Apache, nmap.....	63
Figura 32.-	Prueba de funcionamiento, Apache.....	64
Figura 33.-	Propietario de las tablas, postgis_template.....	72
Figura 34.-	Tabla geometry_columns.....	73
Figura 35.-	Tabla spatial_ref_sys.....	73
Figura 36.-	Prueba de funcionamiento de PHP.....	77
Figura 37.-	Prueba de funcionamiento MapServer.....	80
Figura 38.-	Prueba de funcionamiento PHP/MapScript.....	81
Figura 39.-	Sito Web, Cartografía en Línea.....	83
Figura 40.-	Herramientas de navegación.....	84
Figura 41.-	Herramientas para consulta de atributos.....	84
Figura 42.-	Paneles, visor.....	85
Figura 43.-	Coordenadas.....	85
Figura 44.-	Herramientas de dibujo.....	85
Figura 45.-	Acceso al acervo de información geográfica desde el SIG de escritorio gvSIG y navegador de Internet.....	86
Figura 46.-	Estadísticas de acceso al sistema (Febrero a Diciembre, 2013).....	87
Figura 47.-	Estadísticas de acceso al sistema (Enero a Septiembre, 2014).....	87

Tabla 1.-	Códigos EPSG.....	7
Tabla 2.-	Proyectos respaldados por la OSGeo.....	8
Tabla 3.-	Ficha técnica, MapServer.....	14
Tabla 4.-	Ficha técnica, PostgreSQL.....	16
Tabla 5.-	Ficha técnica, PostGIS.....	17
Tabla 6.-	Ficha técnica, framework OpenLayers.....	19
Tabla 7.-	Ficha técnica, framework Heron Mapping Client.....	21
Tabla 8.-	Ficha técnica, gvSIG.....	22
Tabla 9.-	Descripción de la cartografía temática.....	28
Tabla 10.-	Ficha técnica, CentOS.....	32
Tabla 11.-	Ficha técnica, Servidor Apache.....	35
Tabla 12.-	Hardware, configuración.....	59
Tabla 13.-	Lista del Software Libre y de Código Abierto usado en este proyecto.....	60
Script 1.-	Contenido de compila_apache.sh para Apache.....	62
Script 2.-	Contenido de compila_postgresql.sh para PostgreSQL.....	65
Script 3.-	Contenido de compila_postgis.sh para PostGIS.....	68
Script 4.-	Parte del contenido del archivo profile.....	69
Script 5.-	Contenido del archivo ld.so.conf.....	69
Script 6.-	Contenido de compila_php.sh para PHP.....	74
Script 7.-	Contenido de index.php.....	76
Script 8.-	Contenido de compila_gdal.sh para GDAL/OGR.....	78
Script 9.-	Contenido de compila_mapserver.sh para MapServer.....	79

RESUMEN

Hoy en día el uso de las Tecnologías de la Información y Comunicación (TIC) se incrementa cada vez más, introduciéndose en la mayoría de los procesos cotidianos de las organizaciones. Las mismas son utilizadas en varios sistemas informáticos que se utilizan como apoyo en la toma de decisiones en diferentes sectores como: medio ambiente, salud, entre otros. Entre estas herramientas se encuentran los Sistemas de Información Geográfica (SIG), los cuales son utilizados a todos los niveles pues garantizan la disponibilidad de información espacial. En este documento se presenta el resultado del proyecto “Sistema de Consulta de Mapas Digitales sobre Planificación Territorial y Biodiversidad”, el cual contiene, un Servidor de Mapas cartográfico interactivo de interés público, que permite la visualización de datos espaciales distribuidos a través de Internet, siguiendo estándares internacionales recomendados por la Open Geospatial Consortium (OGC) como es el Web Map Service (WMS).

Este sistema funciona a través de un navegador Web donde el usuario accede de una manera interactiva a cartografía dinámica, multiescala, personalizable según sus preferencias, realizar consultas temáticas específicas, calcular distancias, entre otras. De esta manera un usuario no técnico y sin conocimientos avanzados en SIG podrá consultar la información sin dificultad.

El proceso se inició con una identificación de las necesidades y el posterior desarrollo de una serie de pasos en orden secuencial para disponer de una estructura combinada de elementos de almacenamiento, acceso y análisis de información; que pudiera asegurar la satisfacción de estas necesidades. Debido a que los requerimientos de los usuarios así lo demandan, se ha tenido en cuenta tanto a usuarios que trabajan en entornos Linux como a otros que lo hacen en entornos Windows. Es por esta razón, que se optó por un sistema basado en software libre y de código abierto utilizando Apache como Servidor Web, MapServer como Servidor de Mapas, PostGIS sobre “PostgreSQL” para la administración, conversión y análisis de datos; y gvSIG como interfaz gráfica de usuario. Otros programas utilizados para propósitos específicos han sido GDAL/OGR y Heron Mapping Client.

INTRODUCCIÓN

Debido a las necesidades para una toma de decisiones más rápida y eficiente, entidades gubernamentales, académicas y privadas han ido generando, empleando y demandando mayor cantidad de información geográfica. Los Sistemas de Información Geográfica (SIG) han permitido el tratamiento digital de esa información y han facilitado la consulta, el análisis y su aplicación en las distintas problemáticas que se presentan en la gestión de nuestro territorio.

Esta acumulación de información proveniente de distintas fuentes, y generada con diferente software SIG, plantea una serie de dificultades de integración y de acceso, por mencionar algunas: la falta de interoperabilidad y compatibilidad debido al uso de formato de datos propietarios no estandarizados, entre otras.

Hoy en día las soluciones tecnológicas en el campo de los SIG, deben superar la fase actual, caracterizada por soluciones propietarias y difícilmente escalables, evolucionando hacia arquitecturas compuestas por componentes abiertos, reutilizables y basados en estándares, que permitan la construcción de sistemas distribuidos e interoperables.

Bajo este contexto, existe una urgente necesidad de utilizar procedimientos más sofisticados para integrar datos de una forma sistemática y escalable, con la finalidad de que cualquier organización disponga de sistemas informáticos para proporcionar información de una forma abierta y estandarizada.

A partir del año 2012, el Instituto Nacional de Ecología y Cambio Climático (INECC), a través de la Coordinación General de Adaptación al Cambio Climático (CGACC) inició un proceso de evaluación, planeación y desarrollo de servicios y herramientas informáticas, con el fin de poner a disposición del público en general los resultados de los proyectos de las distintas líneas de investigación que se realizan en la CGACC, destacando temas como: planeación territorial, conservación de ecosistemas y cambio climático que son de utilidad para apoyar la toma de decisiones y la política pública ambiental.

El objetivo del presente trabajo es describir los elementos de dicho proceso, así como las experiencias obtenidas, especialmente con relación a la adopción de estándares para el intercambio de información geográfica, el uso de software libre y de código abierto.

ANTECEDENTES

El Instituto Nacional de Ecología y Cambio Climático (INECC) tiene como antecedente el Instituto Nacional de Ecología (INE), creado en 1992 como un órgano desconcentrado de la Secretaría de Desarrollo Social, con atribuciones técnicas y normativas en materia de ecología.

El INE mantuvo la misma naturaleza jurídica con la creación de la Secretaría de Medio Ambiente, Recursos Naturales y Pesca en 1994; y con la creación de la Secretaría de Medio Ambiente y Recursos Naturales (SEMARNAT) en 2001; sin embargo, en este año se reforma, de un órgano operativo y de gestión, en un instituto de investigación científica en materia de medio ambiente y recursos naturales.

En 2012, el INECC se crea a partir del INE, según lo dispuesto en el artículo 13 de la nueva Ley General de Cambio Climático, publicada en el Diario Oficial de la Federación con fecha de 6 de junio de 2012. Ahora es un organismo público descentralizado de la Administración Pública Federal, con personalidad jurídica, patrimonio propio y autonomía de gestión, sectorizado en la SEMARNAT, y tiene entre otros objetivos: coordinar y realizar estudios y proyectos de investigación científica o tecnológica con instituciones académicas, de investigación, públicas o privadas, nacionales o extranjeras en materia de cambio climático, protección al ambiente y preservación y restauración del equilibrio ecológico.

Desde el año 2000 y a la fecha, la Dirección General de Investigación de Ordenamiento Ecológico y Conservación de los Ecosistemas (DGIOECE), actualmente Coordinación General de Adaptación al Cambio Climático (CGACC) ha trabajado en la implementación de mecanismos de difusión e intercambio de información geográfica. Esfuerzo que se ha intensificado desde el 2011, momento en que se replanteo el proyecto *“Sistema de consulta de mapas digitales sobre planificación territorial y biodiversidad”* con una visión muy clara: fortalecer el sistema, mediante el desarrollo de un visor cartográfico basado en software libre y herramientas de código abierto, para la integración de servicios web geográficos en el portal del INECC.

Para tal efecto, mediante el uso de Internet realice una búsqueda de casos de éxito en el desarrollo de visores de mapas implementados con tecnologías libres y estándares abiertos de datos.

Con esta búsqueda, primer acercamiento al ecosistema del software libre para SIG, me percate que la mayoría de los desarrollos Web Mapping son implementados con las siguientes tecnologías; Servidores de mapas: MapServer, GeoServer; Sistemas Manejadores de Bases de Datos (SMBD): PostGIS; SIG de escritorio: gvSIG, QGIS, GRASS, GDAL/OGR; Frameworks: MapFish, Geomajas, Mapbender, OpenLayers, Heron Mapping Client; Lenguajes de desarrollo: JavaScript, PHP, JAVA.

Identificadas las tecnologías libres procedí a explorar sus capacidades técnicas, haciendo énfasis en la interoperabilidad con otros sistemas de software, además de verificar su calidad, nivel de estabilidad, facilidad de uso, documentación, cumplimiento de estándares, si su comunidad de desarrolladores es activa y aporta mejoras y si funcionan en los sistemas operativos más utilizados en la actualidad.

Con esto, pude concluir que las tecnologías que más se ajustan a las necesidades de este proyecto fueron gvSIG como software de escritorio, MapServer como Servidor de Mapas, PostGIS como sistema manejador de bases de datos, Heron Mapping Client como interfaz de usuario y como lenguaje de desarrollo JavaScript.

Por lo anterior y para fortalecer las capacidades técnicas para el desarrollo de este proyecto, en Julio de 2011 asistí a las primeras Jornadas de gvSIG bajo el lema “La información geográfica al alcance de todos”, celebradas en Montevideo, Uruguay. Evento que representó un espacio de difusión, intercambio de experiencias e información entre usuarios y desarrolladores en temas vinculados a los SIG, las IDE y la Geomática. Derivado de la participación en dicho evento, entregué un informe de trabajo a la CGACC en el cual describí como el avance tecnológico, el software libre y de código abierto han llegado a los SIG, que son utilizados para estudiar la cobertura vegetal, usos de suelo, ordenamiento territorial, modelos hidrológicos y catastro, y de cómo se pueden generar mapas.

A partir de estos argumentos, se aceptó la propuesta del uso de software libre y de código abierto para liberar al INECC de la compra de software propietario y disminuir costos de licenciamiento, medida que favoreció esta visión integradora.

OBJETIVOS

OBJETIVO GENERAL

Mi trabajo ha consistido en hacer realidad el objetivo de la CGACC, el cual consiste en contar con un sistema informático que centralice la información geográfica sobre ecosistemas y planificación territorial de una manera eficaz, para potencializar su administración, análisis y consulta desde diferentes tipos de interfaces. Garantizando que dicho sistema permita publicar en Internet el acervo cartográfico de acuerdo a los criterios establecidos por la CGACC.

OBJETIVO ESPECÍFICOS

1. Proporcionar los medios de acceso y consulta a la información geográfica para usuarios internos y externos, así como la capacidad de presentar amplios volúmenes de información temática permitiendo a los usuarios interactuar de una forma amigable.
2. Diseñar, desarrollar e implementar una interfaz (visor de información cartográfica), que permita la publicación en Internet de mapas digitales que se consideren de acceso público con base en el inventario cartográfico existente para el territorio nacional.
3. Crear un sistema informático, que sienta precedente, en la infraestructura de los SIG, en el INECC, con interés particular en brindar la información con carácter público, a cualquier ciudadano interesado en consultar y descargar cualquier capa de información o metadato.

CAPÍTULO 1.- MARCO TEÓRICO

1.1 Sistemas de Información Geográfica (SIG)

Un SIG es un sistema de hardware, software y procedimientos almacenados para facilitar la obtención, gestión, administración, análisis, modelado y salida de datos espacialmente georreferenciados, para resolver problemas complejos de planificación y gestión [1].

Un SIG consiste en información de naturaleza diversa sobre una determinada área, almacenada en un conjunto de bases de datos, tanto gráficas como alfanuméricas, cuya relación con el territorio se realiza a través de un sistema de referencia geográfico y se gestiona a través de uno o varias aplicaciones informáticas, lo anterior soportado por un equipo de cómputo y por personal especializado [2].

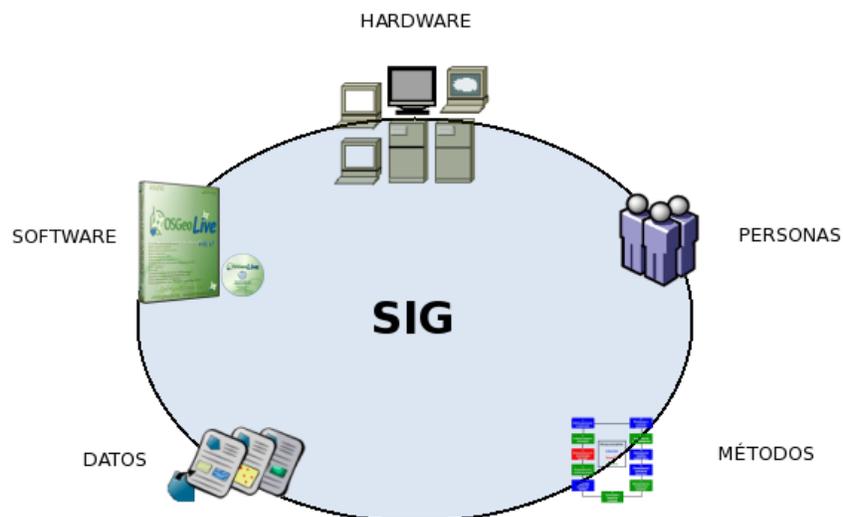


Figura 1. Componentes de un Sistema de Información Geográfica.

Un SIG, ante todo, es una herramienta para gestionar datos, analizarlos y obtener resultados de ese análisis, es decir para resolver problemas concretos. No se debe perder de vista este carácter, poniendo el énfasis no en la herramienta en sí, sino el uso que se hace de ella y en el beneficio que pueda obtenerse de ese uso.

En un SIG se suelen distinguir los siguientes componentes (**Figura 1**):

- **Hardware:** Se refiere a los equipos de cómputo, donde se ejecutan los programas.
- **Software:** Se refiere a los programas que proveen funciones y herramientas necesarias para almacenar, analizar y visualizar información geográfica. La elección del software depende de las aplicaciones concretas para las que se vaya a emplear y del tipo de análisis que se requiera de él.
- **Datos:** Considerada como la parte más importante de un SIG. La información es la base de todo el sistema y el elemento más costoso. La información geográfica resulta difícil de generar y de actualizar.
- **Metodología:** Los procedimientos que se establezcan para el desarrollo de un proyecto mediante SIG, estarán determinados por el plan de trabajo que se diseñe en función de los objetivos del proyecto.
- **Personas:** Todos los que vayan a trabajar con el SIG, tanto técnicos especializados como público en general. Estos últimos tienen que tener acceso a la información, lo que supondrá que determinadas funciones del programa deben ser fáciles de emplear pero también requerirá de ellos una cierta formación.

Muchos autores de textos sobre SIG hacen referencia al carácter multidisciplinario que es característico de estos sistemas. Su desarrollo y su evolución solo ha sido posible gracias a las aportaciones de distintas ciencias y tecnologías, y en sus aplicaciones cabe la participación de expertos de muy diversos campos, muchos de ellos sin relación directa con la cartografía.

1.2 Open Geospatial Consortium (OGC)

El Open Geospatial Consortium, conocido habitualmente como OGC, es una organización internacional fundada en 1994 sin fines de lucro, conformada por más de 400 empresas privadas, administraciones públicas, universidades y centros de investigación; su misión, promover el uso de estándares y tecnologías abiertas en el área de los sistemas y tecnologías de la información geográfica y áreas a fines [3].

1.2.1 Especificaciones OGC

Para facilitar la reutilización de la información geográfica por diferentes aplicaciones y compartir información entre organismos sin necesidad de adaptaciones, se implementan normas de interoperabilidad. Las normas de interoperabilidad permiten crear un ambiente donde el intercambio de información entre organismos sea consistente, potencializando la integración de la información geográfica de un territorio o área en particular. Las normas ISO (ISO/TC 211) y las especificaciones del OGC son ejemplos de normas que facilitan el intercambio de información geográfica.

Las especificaciones más importantes surgidas del OGC son las OpenGIS Web Services [3]. Entre ellos cabe destacar los siguientes:

- Servicio de mapas Web Map Service (WMS). Produce mapas de manera dinámica a partir de información geográfica vectorial o raster presentando la información como imágenes digitales (JPEG, PNG) susceptibles de ser visualizadas en pantalla.
- Servicio de publicación de objetos Web Feature Service (WFS). Permite descargar y modificar datos espaciales en formato vectorial codificados en Geography Markup Language (GML).
- Servicio de coberturas Web Coverage Service (WCS). Permite la obtención de datos espaciales en forma de “coberturas” que representan fenómenos de variación espacial. A diferencia de los servicios de mapas, estos proporcionan los datos con su semántica original, permitiendo además de su representación estática, su interpretación y extrapolación.
- Servicio de catálogo de metadatos Catalogue Service for Web (CSW). Estandariza la forma en que se publican los metadatos de la información geográfica. Permite la publicación, búsqueda y consulta de metadatos, servicios y otros recursos relacionados.

Estos estándares facilitan el intercambio de información geográfica. En cuanto a la definición de estándares para el procesamiento distribuido, el OGC ha establecido el Web Processing Service (WPS). Un WPS proporciona acceso a operaciones o cálculos de datos espaciales, de distinta complejidad, mediante Servicios Web como interfaces. Puede abarcar operaciones para tratar tanto datos vectoriales como matriciales, provenientes de la red o del propio servidor.

1.3 Infraestructura de Datos Espaciales (IDE)

En la actualidad, existen diferentes definiciones acerca de lo que es una IDE, según distintas instancias internacionales:

Una IDE según, la Global Spatial Data Infrastructure Association (GSDI), es una colección importante de tecnologías, políticas y disposiciones institucionales que facilitan la disponibilidad y el acceso a datos espaciales [4].

Una IDE incluye datos geográficos y sus atributos, documentación (metadatos), un medio para buscar, visualizar y evaluar datos (catálogo y mapeo por la red), y algún método para proporcionar acceso a los datos geográficos.

Para que una IDE sea funcional, también debe incluir los acuerdos organizativos necesarios para coordinarla y administrarla a una escala local, regional, nacional o internacional [5].

Mientras que el Comité Federal de datos Geográficos (FGDC) define a una IDE como un conjunto de tecnologías, políticas, normas y recursos humanos necesarios para adquirir, procesar, almacenar, distribuir y mejorar el uso de datos geoespaciales [5].

Para la Infrastructure for Spatial Information in the European Community (INSPIRE), una IDE es conjunto de tecnologías, políticas, normas que faciliten la disponibilidad de la información geográfica relevante, concertada y de calidad de forma que se permita la formulación, implementación, monitorización y evaluación de las políticas de impacto o de dimensión territorial de la Unión Europea [6].

Por lo antes mencionado, se puede considerar a una IDE como un conjunto de datos, servicios, metodologías, normas, estándares y acuerdos, que permiten visualizar, superponer, consultar y analizar la información geográfica publicada en Internet, según estándares bien definidos, por instituciones generadoras de datos y servicios geográficos. Se logra así compartir recursos y, al mismo tiempo, la interoperabilidad entre los diversos sistemas implementados por los agentes implicados.

La IDE es una realidad gracias a dos acontecimientos fundamentales:

- La Conferencia de las Naciones Unidas sobre Medio Ambiente y Desarrollo en Río de Janeiro, en 1992. A partir de ahí se crea, en 1994, la National Spatial Data Infrastructure (NSDI) y el Open GIS Consortium, actualmente Open Geospatial Consortium.
- La iniciativa INSPIRE (Infrastructure for Spatial Information in the European Community) de la Comisión Europea, que se ampara en la Directiva 2007/3/CE del Parlamento Europeo y del Consejo y tiene como objetivo la creación de una Infraestructura de Datos Espaciales en Europa.

Los objetivos de una IDE [7] son:

- Facilitar el acceso y la integración de la información espacial, tanto a nivel institucional y comercial como de los propios ciudadanos, lo que permitirá extender el conocimiento y el uso de la información geográfica y la optimización de la toma de decisiones.

-
- Promover los metadatos estandarizados como método para documentar la información espacial, lo que permitirá la reducción de costos y evitará la duplicidad de esfuerzos.
 - Posibilitar la reutilización de la información geográfica generada en un proyecto para otras finalidades diferentes, dado el alto costo de su producción.
 - Incentivar la cooperación entre los agentes, favoreciendo un clima de confianza para el intercambio de información.

Las principales organizaciones que han contribuido a establecer las bases de lo que actualmente son y significan las IDE han sido, cada una en su ámbito de repercusión:

- A nivel global, sin duda, la Global Spatial Data Infrastructure Association (GSDI), que es una organización dedicada a la cooperación y colaboración internacional en el desarrollo de IDE, orientada a permitir a gobiernos, organismos públicos y privados y a particulares un mejor planteamiento en aspectos sociales, económicos y ambientales de mayor importancia [4].
- A nivel europeo, la iniciativa INSPIRE. Se trata de una iniciativa legal que establece estándares y protocolos de tipo técnico, aspectos organizativos y de coordinación y políticas sobre la información que incluyen el acceso a los datos y la generación y mantenimiento de información espacial [6].

1.4 European Petroleum Survey Group (EPSG)

El antiguo consorcio europeo del petróleo (EPSG), actualmente absorbido en 2005 por la International Association of Oil & Gas Producers (OGP), realiza desde 1986 varios estudios sobre Geodesia por su importancia en la localización y medida de los yacimientos petrolíferos. Fruto de estos trabajos es una abundante bibliografía disponible sobre geodesia siendo destacable un dataset público, que van actualizando periódicamente, el cual contienen una colección, especialmente pensada para desarrolladores informáticos, de parámetros de sistemas de coordenadas, proyecciones, métodos para transformar la geometría de objetos, entre otros; lo que dio lugar a la base de datos con códigos EPSG que se ha convertido en un estándar y que puede ser descargada desde su página web [8].

Los códigos EPSG hacen referencia a una gran cantidad de Sistemas de Referencia Espacial (SRS) y a otros objetos cartográficos como datums y elipsoides.

En las IDE, los códigos EPSG son utilizados para el SRS en que se sirven los mapas en un servicio WMS. Algunos códigos EPSG de uso común se muestran en la (**Tabla 1**).

Código	Datum, Sistema de coordenadas	Observaciones
4326	WGS84 Lat Lon	Sistema Geodésico Mundial 1984. Sistema de referencia espacial en coordenadas geográficas más utilizado en la actualidad, diseñado por el Departamento de la Defensa de Estados Unidos. Los sistemas GPS se basan en este sistema de coordenadas [9].
3857	WGS84/Pseudo Mercator	Usado por los principales servicios de mapas en Internet (Google Maps, OpenStreetMap, Bing Maps, entre otros). Puede encontrarse también con el código alterno 900913, desarrollado por OpenLayers [10].

Tabla 1. Códigos EPSG.

1.5 The Open Source Geospatial Foundation (OSGeo)

Es una fundación sin fines de lucro, cuyo objetivo es fomentar, promover y colaborar con el desarrollo de software geoespacial de código abierto. La fundación proporciona recursos financieros, organizacionales y apoyo jurídico a la comunidad de software libre.

También sirve como una entidad jurídica independiente a la que los miembros de la comunidad pueden ayudar, colaborar o participar en la generación de código, la financiación y otros recursos, la seguridad de que sus contribuciones se mantendrán para beneficio público.

OSGeo, además funge como una organización de promoción y divulgación de conocimiento para la comunidad geoespacial de código abierto, y ofrece un foro común y compartido de infraestructura para mejorar la colaboración entre proyectos [11]. En la (**Tabla 2**) se describen brevemente algunos de los proyectos que se desarrollan bajo la tutela de OSGeo.

Categoría	Proyecto	
Mapas Web	degree Geomajas GeoMoose GeoServer Mapbender	MapBuilder MapFish MapGuide Open Source MapServer OpenLayers
Aplicaciones de escritorio	GRASS GIS Marble QGIS	
Librerías geoespaciales	FDO GDAL/OGR GEOS	GeoTools OSSIM PostGIS
Catálogo de metadatos	GeoNetwork	

Tabla 2. Proyectos respaldados por la OSGeo [11].

1.6 Software SIG – Free and Open Source Software (FLOSS)

Los proyectos y productos de software FLOSS (**Figura 2**) están experimentando una evolución y actualización vertiginosa, considerando el lado cliente y servidor se cuenta con una gama muy amplia de aplicaciones. Si a esto se agregan los distintos servicios, Framework de desarrollo, y estándares implementados, se podría decir que el mundo del software libre y de código abierto se han posicionado como una muy buena alternativa si se desea realizar un proyecto de Geomática.

Dentro de los productos del lado del servidor, se encuentran los servidores de mapas, bases de datos geográficas, Frameworks de desarrollo Web y editores de metadatos. Este conjunto es necesario para implementar un SIG orientado a la Web.



Figura 2. Marco conceptual de Software Libre y de Código Abierto.

Fuente: <http://es.wikipedia.org/wiki/Software_libre_y_de_c%C3%B3digo_abierto>

1.6.1 Servidores de Mapas

Un Servidor de Mapas (Internet Map Server) comprende software y hardware que nos permite publicar datos geográficos en Internet (**Figura 3**), facilitando al cliente interactuar con ellos en forma dinámica por medio de un navegador Web.

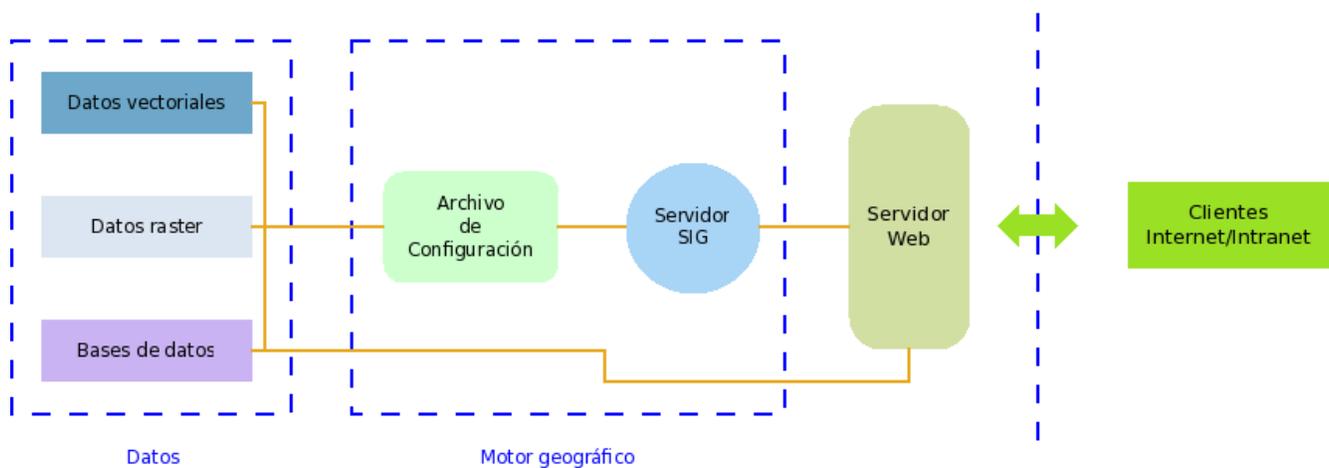


Figura 3. Esquema típico de un Servidor de Mapas.

El proceso de diseñar, implementar, generar y difundir mapas en Internet es conocido como Web Mapping [12] y la aplicación para realizar este proceso es el Servidor de Mapas, haciendo uso del modelo Cliente-Servidor.

Las características principales de un Servidor de Mapas son los OGC Web Services [12], servicios que permiten utilizar información georreferenciada (mapas) vía Web, siendo esta la principal pero no la única funcionalidad del Servidor de Mapas. Otras características [30] son:

- Capacidad de geoprocesamiento. Herramientas disponibles en forma local (buffer, clip, entre otras). Pueden estar disponibles en un mapa publicado mediante desarrollo de software Web.
- Desarrollo de herramientas, al tener la capacidad de desarrollar una interfaz para un Servidor de Mapas, se tiene la posibilidad de realizar cambios en forma dinámica a cada capa, pudiendo agregar o eliminar elementos (puntos, líneas, polígonos, etc.).
- Consulta y búsqueda de datos, se puede realizar una búsqueda mediante criterios definidos por el usuario (puntos de muestreo en cierta región, información histórica con parámetros de tiempo definidos, entre otros.).
- Carga y descarga de información georreferenciada, la flexibilidad ofrecida por el Servidor de Mapas permite realizar la carga de capas que el usuario tenga en forma local, para poder visualizarlas en forma automática dentro del Servidor de Mapas. También se puede realizar la descarga de la información georreferenciada disponible, de forma local.
- Generación dinámica de capas, la interacción con el usuario puede dar como resultado nueva información georreferenciada, esta nueva información se puede representar directamente en el mapa publicado.

1.6.1.1 MapServer

MapServer, según la página oficial del proyecto, es una plataforma para publicar información espacial y generar mapas dinámicos en la Web mediante la tecnología Internet Map Server (IMS) (**Figura 4**). MapServer fue desarrollado en la década de los 90 por la Universidad de Minnesota (UMN) en cooperación con la NASA y el Departamento de Recursos Naturales de la Universidad de Minnesota (MNDNR) [13]. MapServer es uno de los proyectos FLOSS de mayor impacto e importancia, actualmente es desarrollado por la OSGeo.

MapServer puede funcionar de dos modos diferentes: CGI y MapScript [14]:

- CGI; los programas CGI son un rasgo de Internet que amplía enormemente la capacidad de documentos HTML sencillos al permitir la interacción con datos o aplicaciones informáticas que residen en un Servidor Web. Este modo es una forma tradicional de generar contenido dinámico en la Web y permite la generación de la información espacial de una manera sencilla para aplicaciones que no requieren mucha complejidad. A pesar de ser la primera tecnología que se creó para generar contenido dinámico, en la actualidad sigue siendo utilizada.
- MapScript, permite acceder a las funcionalidades de MapServer a través de lenguajes de programación como Perl, Python, PHP y Java, y logran las mismas funcionalidades que en el modo CGI pero con la posibilidad de crear aplicaciones más sofisticadas.

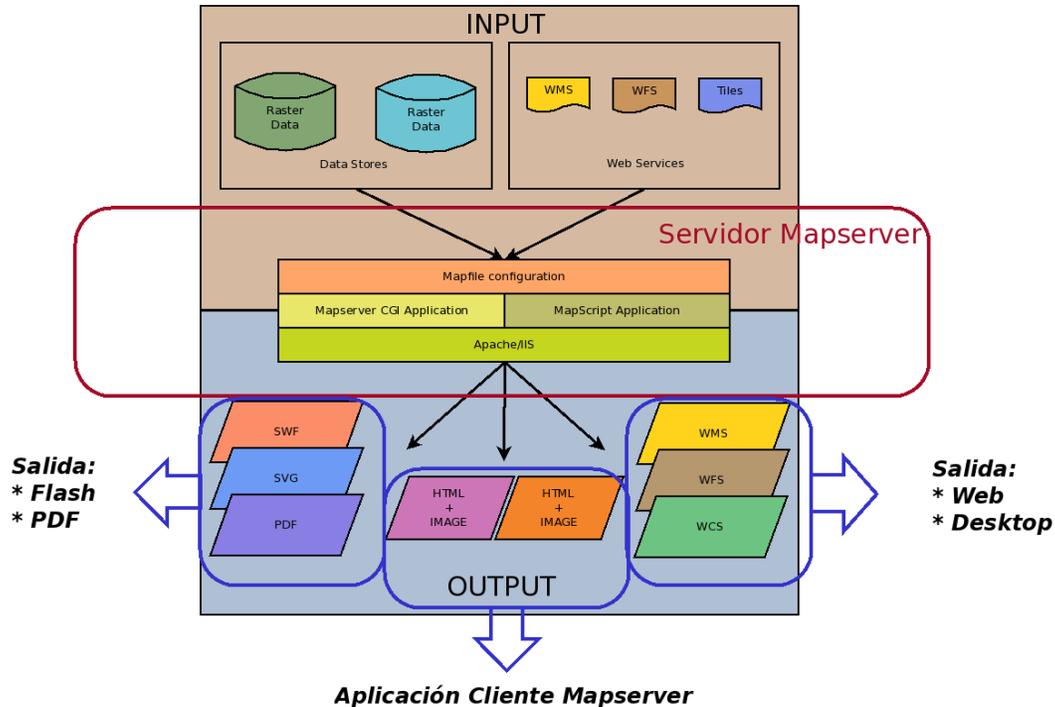


Figura 4. Descripción gráfica de los componentes en un entorno MapServer [13].

MapServer funciona directamente con archivos shapefiles de ESRI, un “formato de almacenamiento de datos”, de manera que permite utilizar en la Web cualquier tipo de datos SIG basado en formatos digitales de ESRI.

MapServer es totalmente autosuficiente; no necesita de otro programa para procesar datos y crear informes.

Una aplicación MapServer tiene en promedio cinco elementos básicos [14]:

- Un archivo de mapa con extensión “.map”, que controla los datos a publicar y la manera de cómo hacerlo. Este archivo de configuración comúnmente conocido como Mapfile, tiene una estructura jerárquica (**Figura 5**) donde se definen los datos a ser usados por una aplicación como: las capas, sus tipos y configuración, la fuente de datos y la forma de servir los mismos; sus leyendas y proyecciones; y muchas otras características que se deseen mostrar. Este archivo es definido por el usuario como un archivo de texto que se puede manipular con cualquier editor de texto.

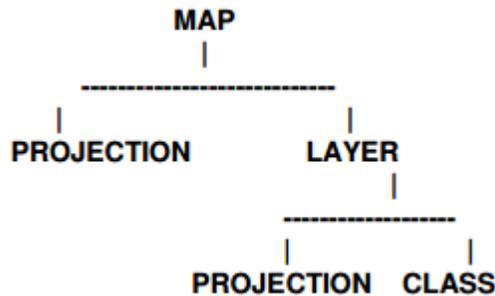


Figura 5. Jerarquía de los objetos, Mapfile.

- Un archivo de tipo plantilla con extensión “html”. Este archivo es un elemento de comunicación (interfaz de usuario) entre el usuario y MapServer.
- Cartografía (datos espaciales: vectoriales, raster).
- Un servidor HTTP (llamado también Servidor Web) que sirve las páginas Web cuando son solicitadas por el usuario.
- El programa CGI de MapServer, el cual se encargará de recibir las peticiones y devolver datos, imágenes, etc. Este CGI reside en el directorio de scripts del Servidor Web.

Para utilizar MapServer en un caso en particular, es necesario crear dos tipos de archivos, uno define la interfaz de usuario en el navegador Web, mientras que el otro determina cómo se debe crear el mapa a partir de los datos espaciales (shapefiles).

Entre sus principales características se pueden mencionar:

- Fácil de configurar y administrar.
- Multiplataforma.
- Velocidad de acceso a datos.
- Soporta una gran variedad de formatos vectoriales, raster, además de una gran cantidad de estándares OGC.

Para aumentar la funcionalidad de MapServer, se recomienda combinar su instalación con un Sistema Manejador de Bases de Datos (SMBD) como PostgreSQL y con algún lenguaje de programación como PHP, Java, entre otros. En la (**Tabla 3**) se desglosan los datos técnicos relacionados de esta plataforma.

Datos	
Nombre	MapServer
Lenguaje de desarrollo	C
Versión estable	6.4.1
Plataformas soportadas	Linux, Windows, Mac OS X, Solaris, entre otros.
Estándares OGC	WMS, WFS, WMC, WCS, Filter Encoding, SLD, GML, SOS, OM
Formatos	TIFF7GeoTIFF, EPPL7 y otros vía GDAL ESRI ShapeFile, PostGIS, ESRI ArcSDE, Oracle Spatial, MySQL y otros vía OGR
Control de versiones	SVN
Licencia	MIT-style license
Sitio Web	http://mapserver.org
Soporte	Documentación de usuario completa Documentación para desarrolladores completa Lista de correo activa

Tabla 3. Ficha técnica, MapServer [13].

1.6.2 Base de Datos Geográfica

Es un conjunto de datos espaciales que cuentan con un modelo de datos que abstrae la complejidad del mundo real y permite almacenarlos con una representación simplificada (puntos, líneas y polígonos) en un Sistema Manejador de Bases de Datos (SMBD), el cual provee un lenguaje de consulta, soporta tipos de datos espaciales y proporciona mecanismos para realizar operaciones de análisis espacial [15]. Por su parte un SMBD consiste en una colección de archivos interrelacionados y un conjunto de programas que permiten a los usuarios acceder y modificar archivos; su propósito principal es proporcionar a esos usuarios una visión abstracta de los datos [16].

La arquitectura de un SDBD se compone de:

- Base de datos. La conforma una colección de datos almacenados y utilizados por los sistemas de aplicación de una empresa [16].
- Hardware. Es un compuesto por volúmenes de almacenamiento secundario donde reside la base de datos junto con dispositivos asociados a su control.
- Software. Proporciona la interfaz entre los datos de bajo nivel almacenados en la base de datos y los programas de aplicación y consultas hechas al SDBD.
- Usuarios. Dado que el objetivo principal de un sistema de base de datos es proporcionar un entorno para recuperar información y almacenar nueva, existen tres tipos de usuarios, dependiendo de la forma en la que interactúan con el sistema.

Tipos de Usuarios

- Programador de Aplicaciones: Encargados de escribir los programas que utilizan la base de datos en todas las formas usuales como recuperación de información, creación o introducción de nuevos datos, eliminar o cambiar información existente.
- Usuario Final: Tienen la función de acceder a la base de datos por medio de una terminal, utiliza un lenguaje de consulta o recurre a un programa de aplicación que acepte instrucciones y formule solicitudes al SDBD de recuperación, creación o modificación de información.
- Administrador de Base de datos: Coordina las funciones de recopilación de información acerca de los datos; servicio a los usuarios; protección del recurso llamado datos; control de estructuras de datos y acceso a los mismos.

1.6.2.1 PostgreSQL

PostgreSQL es un potente sistema de bases de datos objeto relacional de código abierto, ya que incluye características de la orientación a objetos, como la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional, liberado bajo licencia BSD (Berkeley Software Distribution). Es multiplataforma, tiene soporte para los principales sistemas operativos, Linux, UNIX y Windows. Tiene interfaces nativas de programación C/C++, Java, .NET, Perl, Python, Ruby, TCL, ODBC, JDBC, entre otros [17].

Como muchos otros proyectos Open Source, el desarrollo de PostgreSQL no es manejado por una sola compañía sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo, dicha comunidad es denominada PGDG (PostgreSQL Global Development Group).

PostgreSQL en la actualidad, es considerado como el Sistema Manejador de Bases de Datos de código abierto más avanzado del mundo, debido a que posee características similares de los sistemas comerciales más potentes como Oracle o SQL Server. En la (**Tabla 4**) se listan algunos datos técnicos de este Sistema Manejador de Bases de Datos.

Datos	
Nombre	PostgreSQL
Lenguaje de desarrollo	C
Versión estable	9.3.5
Plataformas soportadas	Linux, UNIX, Windows
Control de versiones	SVN
Licencia	BSD
Sitio Web	http://www.postgresql.org
Soporte	Documentación de usuario completa Documentación para desarrolladores completa Lista de correo activa

Tabla 4. Ficha técnica, PostgreSQL [17].

1.6.2.2 PostGIS

PostGIS es una extensión del sistema de bases de datos PostgreSQL, esta extensión no solo permite almacenar información geográfica (puntos, líneas o polígonos) y cumplir con el estándar Simple Features Specifications for SQL (SFSS), sino además permite realizar operaciones con dicha información [15]. PostGIS almacena la información geográfica en una columna del tipo GEOMETRY (diferente del homónimo utilizado por PostgreSQL).

La OGC define dos formas estándares de expresar los objetos espaciales; estos son los formatos en modo texto (Well-Known Text) y en modo binario (Well-Known Binary) [15]. Ambos incluyen la información sobre el tipo de objetos y coordenadas que conforman el objeto.

En la actualidad, PostGIS es la mejor alternativa disponible para el almacenamiento de información geográfica en el mundo del Open Source. Como consecuencia, una capa de PostGIS se puede analizar con el software GRASS, se puede publicar en la Web mediante el Servidor de Mapas MapServer, además de ser visualizada con software de escritorio: gvSIG, uDig, QGIS o ArcGIS, y ser exportada a formatos propietarios vía OGR y Feature data Objects (FDO). En la (**Tabla 5**) se describen algunos datos técnicos de esta extensión.

Datos	
Nombre	PostGIS
Lenguaje de desarrollo	C
Versión estable	2.1.3
Plataformas soportadas	Linux, Windows, Mac OS X
Control de versiones	SVN
Licencia	GNU GPL
Sitio Web	http://www.refrations.net
Soporte	Documentación de usuario completa Documentación para desarrolladores completa Lista de correo activa

Tabla 5. Ficha técnica, PostGIS [18].

1.6.3 Frameworks de desarrollo

Un framework de desarrollo para aplicaciones Web orientadas a los SIG, es un entorno de trabajo que permite generar rápidamente una aplicación funcional [20]. Esto sucede debido a que la mayoría de los frameworks ya disponen de plantillas prediseñadas que se pueden adaptar a necesidades propias, sin embargo cada framework posee una API de desarrollo que trabaja de una manera distinta a los demás, así que la complejidad está en aprender a utilizar estas herramientas que dependerá del manejo que se tenga del lenguaje en que se desarrolló dicha herramienta. Al momento de elegir un framework se tiene que basar nuestra elección en algunos criterios mínimos como: lenguaje de desarrollo, plataformas en las que puede operar, funcionalidades que ofrece y curva de aprendizaje.

1.6.3.1 OpenLayers

Los desarrolladores de OpenLayers diseñaron la herramienta como una alternativa a la solución comercial de Google Maps; la versión 3 ya está disponible. OpenLayers ofrece un método sencillo para incluir mapas dinámicos en páginas Web [21]. La motivación principal de los desarrolladores era que fuese fácil de usar, lo cual lograron mediante la adopción del método de mosaiqueado. A pesar de que OpenLayers utiliza una gran variedad de formatos y soporta diversos estándares OGC, los servicios de tiles tienen ciertas ventajas: utilizan los servicios de directorio, simples y muy probados, de un Servidor Web para proporcionar mapas, adquiriendo así un alto grado de escalabilidad.

OpenLayers es una librería en JavaScript que integra mapas en aplicaciones Web. Su función básica, es parecida a la de Google Maps, y en realidad es superior en muchos aspectos: entre sus funcionalidades, permite sobreponer distintas capas sobre una capa base, añadir objetos geométricos (puntos, líneas polígonos) sobre el mapa con leyendas de una manera sencilla, gracias a su API; además, incorpora un conjunto de controles básicos y avanzados, soporte para WMS, WFS, además de las mencionadas tiles y cache [22].

Para su configuración se usa una sintaxis relativamente sencilla que se debe incluir en una página HTML.

OpenLayers es actualmente el estándar para los desarrolladores Web que buscan integrar mapas en sus aplicaciones con el mínimo costo. En la (**Tabla 6**) se desglosan algunos datos técnicos relacionados a este Framework.

Datos	
Nombre	OpenLayers
Lenguaje de desarrollo	JavaScript
Versión estable	2.13.1
Plataformas soportadas	Linux, Windows, Mac OS X, UNIX
Control de versiones	SVN
Licencia	2-clause BSD
Sitio Web	http://openlayers.org
Soporte	Documentación de usuario completa Documentación para desarrolladores completa Lista de correo activa

Tabla 6. Ficha técnica, framework OpenLayers [22].

1.6.3.2 Heron Mapping Client (Heron MC)

El proyecto Heron Mapping Client es uno de los más recientes en este campo, y ofrece al desarrollador de software un framework con el que se puede construir aplicaciones SIG enfocadas a Internet.

Heron MC, construido a partir de componentes JavaScript, ofrece una interfaz de usuario altamente configurable e independiente de cualquier Servidor de Mapas. La principal motivación de los desarrolladores era que la creación de aplicaciones SIG en Internet fuese fácil; para tal efecto se basaron en el uso de servicios OGC (WMS, WFS, OpenLS, WPS, CSW), además de incorporar estándares de datos (GML, GeoJSON, CSV, Shape, etc.) y la reutilización de código de librerías existentes (GeoExt, OpenLayers, ExtJS) [23].

Los objetivos con los cuales fue diseñado este framework son:

- Facilidad en el diseño, implementación, generación y publicación de mapas en Internet.
- Proporcionar una aplicación Web robusta.
- Aprovechar las capacidades de librerías existentes, utilizadas en la publicación de mapas en Internet.
- Reutilizar widgets y componentes.
- Desarrollos basados en estándares abiertos: OGC, TMS, etc.

Heron MC, está formado por varios componentes GeoExt, ExtJS y OpenLayers (**Figura 6**), OpenLayers se encarga del componente mapa, GeoExt y ExtJS para el diseño de la interfaz de usuario.

ExtJS¹ es una librería JavaScript, que proporciona un conjunto de componentes (widgets) que pueden ser incorporados dentro de una aplicación Web, como árboles de datos, menús, paneles, gráficos, etc. Es distribuida bajo licencia Open Source compatible con GNU GPL v3 y una licencia comercial.

GeoExt² es una potente librería en JavaScript, que proporciona la base para crear aplicaciones cartográficas en la Web; mediante la integración de componentes de OpenLayers con ExtJS. Esta librería proporciona un conjunto de widgets personalizables, además de soporte para el manejo de datos, que facilitan la creación de aplicaciones SIG enfocadas a la visualización, edición y elaboración de mapas a partir de datos geoespaciales.

1 <http://www.sencha.com>

2 <http://geoext.org>

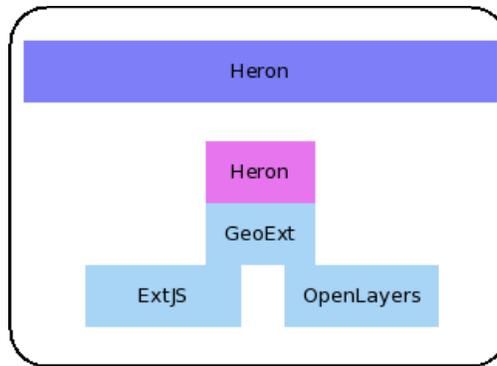


Figura 6. Arquitectura Heron Mapping Client.

Para su configuración se usan archivos JavaScript, con una sintaxis relativamente sencilla, en los cuales se definen:

- Objetos JSON/JavaScript.
- Se definen componentes (widgets).
- Se determina la estructura general de la interfaz de usuario.
- Se definen los elementos del mapa con sus respectivas propiedades.

Por último para su funcionamiento, se requiere la creación de un archivo HTML, en el cual se establezca el soporte para las librerías GeoExt, ExtJS y OpenLayers; además de incluir los archivos que contienen la estructura general de la interfaz de usuario y los elementos del mapa. . En la (**Tabla 7**) se listan algunos los datos técnicos relacionados de este framework.

Datos	
Nombre	Heron Mapping Client (Heron MC)
Lenguaje de desarrollo	JavaScript
Versión estable	1.0.3
Plataformas soportadas	Linux, Windows, Mac OS X, UNIX
Control de versiones	No disponible
Licencia	GNU GPLv3
Sitio Web	http://heron-mc.org
Soporte	Documentación de usuario completa Documentación para desarrolladores completa Lista de correo activa

Tabla 7. Ficha técnica, framework Heron Mapping Client [23].

1.6.4 SIG de escritorio

1.6.4.1 gvSIG

gvSIG es un potente SIG desarrollado en Java, muy completo y capaz de manejar información geográfica en sus formatos más comunes e integrar datos de distintas fuentes, tanto locales como remotas [24].

El desarrollo de gvSIG ha coincidió en el tiempo con las IDE, un nuevo modelo descentralizado que utiliza Internet para procesar, almacenar publicar y distribuir información geográfica. gvSIG ha sido diseñado desde esta nueva visión y es un SIG multiplataforma, extensible, bajo licencia GPL (General Public License, que protege la libre distribución, modificación y uso de software), siguiendo estándares internacionales y gratuito.

En la actualidad es considerado como uno de los pilares de los SIG libres y gratuitos, cada vez más importante en América latina. En la (**Tabla 8**) se desglosan algunos datos técnicos relacionados de este SIG.

Datos	
Nombre	gvSIG (Generalitat Valenciana Sistema de Información Geográfica)
Lenguaje de desarrollo	Java
Versión estable	2.0
Plataformas soportadas	Linux, Windows, Mac OS X
Control de versiones	No disponible
Licencia	GNU GPL v2
Sitio Web	http://www.gvsig.org
Soporte	Documentación de usuario completa Documentación para desarrolladores completa Lista de correo activa

Tabla 8. Ficha técnica, gvSIG [24].

CAPÍTULO 2.- METODOLOGÍA

Para el desarrollado de este sistema, seguí un conjunto de pasos en orden secuencial para disponer de una estructura combinada de elementos de almacenamiento, acceso y análisis de la información (**Figura 7**).

Para conseguir lo antes mencionado, el proyecto se enfocó en tres líneas de acción muy concretas:

- Definición de contenido de la información geográfica, de acuerdo a la información existente para el territorio mexicano.
- Sistematización y normalización de la información territorial existente
- Creación de mecanismos Web que faciliten el uso y consulta de dicha información.

Cabe hacer mención que esta metodología podría servir como documento de consulta para la creación y puesta en marcha de otros tipos de visores de mapas cartográficos.

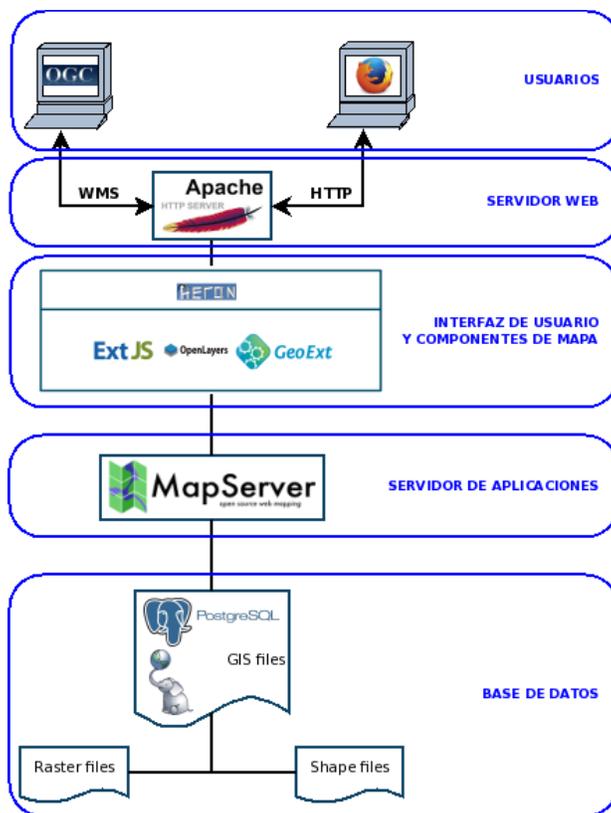


Figura 7. Diagrama de flujo de las tecnologías Open Source utilizadas en este proyecto.

2.1 Definición del contenido de información geográfica

Como punto de partida para la conceptualización y creación del visor de mapas cartográfico, se definieron los insumos de acuerdo a la información geográfica existente para el territorio mexicano.

Esto se hizo a partir de la consulta, recopilación y valoración de los acervos de información cartográfica de fuentes oficiales como la Secretaría de Medio Ambiente y Recursos Naturales (SEMARNAT), Instituto Nacional de Estadística y Geografía (INEGI) e información resultante de proyectos de la CGACC.

Posteriormente se planteó un conjunto de información básica para el desarrollo del sistema, en formato vectorial “ShapeFile” a escalas 1:250,000 y 1:50,000 (**Tabla 9**).

Subtema 1	Subtema 2	Título de Mapa	Nivel de despliegue
MEDIO BIÓTICO	Flora y Fauna	Especies registradas de Flora y Fauna por Cuencas Hidrológicas	Nacional
		Especies registradas de Flora y Fauna por tipos de vegetación, protegidas por la NOM-059-SEMARNAT-2001	Nacional
	Flora	Especies registradas de Flora por Cuenca hidrográfica protegidas por la NOM-059-SEMARNAT-2001	Nacional
		Número de registros de Flora por regiones ecológicas	Nacional
		Número de registros de Flora por tipos de vegetación	Nacional
	Fauna	Especies registradas de Fauna por Cuenca Hidrográfica protegidas por la NOM-059-SEMARNAT-2001	Nacional
		Número de registros de Fauna por regiones ecológicas	Nacional
		Número de registros de Fauna por tipos de vegetación	Nacional
		Distribución de la vaquita (<i>Phocoena sinus</i>) en el Alto Golfo de California, a partir de avistamientos (1972-1997) y encuentros acústicos (2000-2005)	Nacional
		Distribución de la Ballena Jorobada (<i>megaptera novaeangliae</i>) en Isla Socorro (Archipiélago de la Revillagigedo), Colima del 2004 al 2005	Nacional
		Distribución de la Ballena Jorobada (<i>megaptera novaeangliae</i>) en Bahía de Banderas, Jalisco-Nayarit, del 2000 al 2006	Nacional
	Vegetación	Comparativa de uso de suelo y vegetación 1976, 1993 y 2000	Nacional
		Índice de presión de uso circundante de la vegetación 2000	Nacional
		Índice de fragmentación de la vegetación 2000	Nacional
		Grado de modificación de la cobertura vegetal 2000	Nacional
		Grado de naturalidad de la vegetación 2000	Nacional
	MEDIO SOCIAL	Indicadores 1990	Categoría de atracción migratoria por municipio, hasta 1990
Densidad de población por municipio, 1990			Nacional
Densidad de población por localidad, 1995			Nacional
Dependencia económica de la población por municipio, 1990			Nacional
Desarrollo socioeconómico de la población por municipio, 1990			Nacional
Especialización ocupacional agropecuaria por municipio, 1990			Nacional
Especialización de la población ocupada por actividad económica y por municipio, 1990			Nacional
Especialización de la población ocupada por sector económico y por municipio, 1993			Nacional

		Grado de marginación de la población por municipio, 1995	Nacional
		Categoría de atracción migratoria por municipio, hasta 1990	Nacional
		Categoría de atracción migratoria por municipio, 1985-1990	Nacional
		Participación económica de la población por municipio, 1990	Nacional
		Distribución porcentual de la población ocupada en la actividad extractiva por municipio, 1990	Nacional
		Distribución porcentual de la población ocupada en la actividad manufacturera por municipio, 1990	Nacional
		Principales flujos migratorios por estado según municipio de recepción, 1985-1990	Nacional
		Crecimiento de la población por municipio, 1950-1990	Nacional
		Variación del crecimiento demográfico por municipio y por decenio, 1950-1990	Nacional
	Indicadores 2000	Categoría de atracción migratoria acumulada por municipio, 2000	Nacional
		Categoría de atracción migratoria reciente por municipio, 2000	Nacional
		Distribución de la población según categoría rural, mixta o urbana por municipio, 2000	Nacional
		Crecimiento demográfico por municipio, 1950-1970	Nacional
		Crecimiento demográfico por municipio, 1970-1990	Nacional
		Crecimiento demográfico por municipio, 1990-2000	Nacional
		Densidad de población por municipio, 2000 (m/gm)	Nacional
		Densidad de población por municipio, 2000 (Oficial)	Nacional
		Índice de dependencia económica de la población por municipio, 2000	Nacional
		Índice de dispersión de la población por municipio, 2000	Nacional
		Distribución de la población hablante de lengua indígena según tamaño de localidad por municipio, 2000	Nacional
		Especialización económica por sector de actividad según población ocupada por municipio, 2000	Nacional
		Especialización económica según valor agregado censal bruto por municipio, 1999	Nacional
		Grado de escolaridad de la población de 15 años y más por municipio, 2000	Nacional
		Índice de marginación de la población por municipio, 2000	Nacional
		Índice de urbanización por municipio, 2000	Nacional

		Distribución de la población inmigrante reciente según categoría de localidad por municipio, 2000	Nacional	
		Participación porcentual de la población hablante de lengua indígena por municipio, 2000	Nacional	
		Rezago educativo de la población de 15 años y más por municipio, 2000	Nacional	
		Tipología según ritmo de crecimiento demográfico por municipio, 1950-1990	Nacional	
MEDIO ECONÓMICO	Indicadores Sector Primario	Existencia de cabezas de ganado bovino y porcino por municipio, 1991	Nacional	
		Índice de tecnicidad agrícola por municipio, 1991	Nacional	
		Índice de tecnicidad en ganado menor (avícola y porcino) por municipio, 1991	Nacional	
		Producción agrícola por municipio, 1991	Nacional	
		Tipificación de la actividad pecuaria por municipio, 1991	Nacional	
		Tipificación de la actividad agrícola en superficie agrícola, 1991	Nacional	
	Indicadores Sector Secundario	Tipificación de la actividad extractiva por municipio, 1993	Nacional	
		Especialización económica y contribución porcentual del valor agregado censal bruto generado por la actividad extractiva al total por municipio, 1993	Nacional	
		Tipificación de la actividad manufacturera por municipio, 1993	Nacional	
		Especialización económica y contribución porcentual del valor agregado censal bruto generado por la actividad manufacturera total por municipio, 1993	Nacional	
	Indicadores Sector Terciario	Especialización económica por sector y por municipio, 1993	Nacional	
		Tipificación de la actividad comercial por municipio, 1993	Nacional	
		Especialización económica y contribución porcentual del valor agregado censal bruto generado por la actividad comercial al total por municipio, 1993	Nacional	
		Tipificación de la actividad turística por municipio, 1993	Nacional	
		Especialización económica y contribución porcentual del valor agregado censal bruto generado por servicios diversos al total por municipio, 1993	Nacional	
		Especialización económica y contribución porcentual del valor agregado censal bruto generado por servicios turísticos al total por municipio, 1993	Nacional	
	MEDIO FÍSICO	Hidrología	Cuencas Hidrográficas de México - INEGI/INE/CONAGUA 2007	Nacional
			Regionalización de las Cuencas Hidrográficas de México	Nacional
Zonas Funcionales de las Cuencas de México			Nacional	

		Red de drenaje	Nacional
		Tipología de la red de drenaje	Nacional
		Cuencas Hidrográficas - INE 2003	Nacional
	Relieve	Altimetría cada 200 metros	Nacional
		Disección vertical	Nacional
		Ángulo de la pendiente	Nacional
	Mares y Costas	Propuesta para la Regionalización de los Mares Mexicanos	Nacional
REGIONALES	Ordenamiento Ecológico Regional y Marino del Golfo de México y Mar Caribe	Aptitud sectorial	Nacional
		Unidades de Gestión Ambiental	Nacional
	Regionalización ecosistémica con base en indicadores ambientales de tres áreas marinas en el sur de México	Bahía Banderas - Ambiental	Nacional
		Bahía Banderas - Biológico	Nacional
		Bahía Banderas - Socioeconómico	Nacional
		Sistema Lagunar Huave - Ambiental	Nacional
		Sistema Lagunar Huave - Socioeconómico	Nacional
		Laguna de Términos - Ambiental	Nacional
	Laguna de Términos - Socioeconómico	Nacional	

Tabla 9. Descripción de la cartografía temática. Fuente: Dirección de Ordenamiento Ecológico (INE, 2003).

La información se enfocó principalmente en cartografía temática sobre el Ordenamiento Ecológico, Conservación de los Ecosistemas y el Manejo Integral de Cuencas Hídricas; la cual fue distribuida en 6 temas: medio físico, biótico, social, económico así como estudios regionales (**Figura 8**).

Cabe mencionar que este proceso de recopilación y valoración de información cartográfica, así como la clasificación de la misma fue realizado por personal de la Subdirección de Sistemas de Información Geográfica.

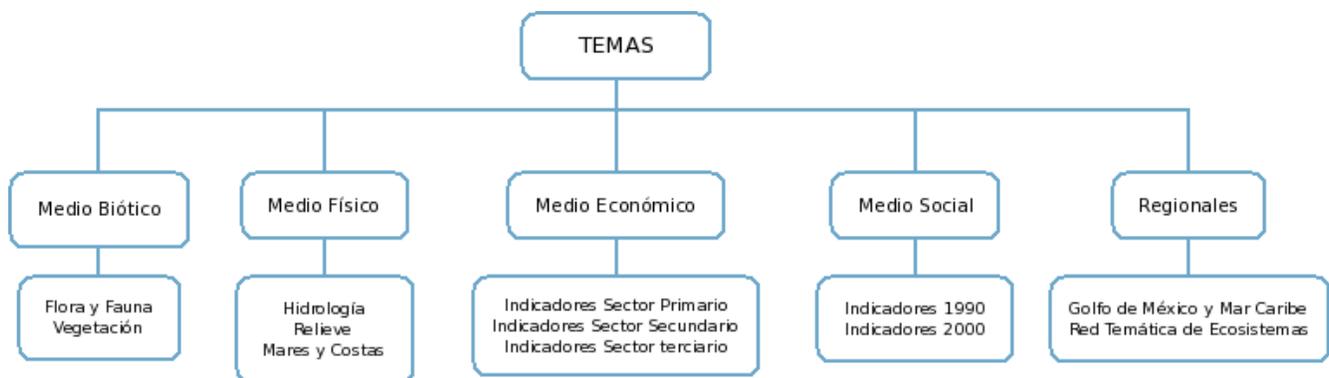


Figura 8. Clasificación del acervo cartográfico.

Aunque, debido a las posibilidades de actualizar y agregar más información al sistema, próximamente se realizarán las adecuaciones correspondientes para poder incorporar más información relacionada con el Cambio Climático y el Programa de Ordenamiento Ecológico General del Territorio.

2.2 Entorno tecnológico y de comunicaciones

El visor de mapas para el “Sistema de Consulta de Mapas Digitales sobre Planificación Territorial y Biodiversidad” fue un desarrollo basado íntegramente en software libre y de código abierto lo que garantiza bajo costo de mantenimiento e independencia tecnológica, ya que la aplicación puede correr en cualquier plataforma.

La arquitectura de la aplicación (**Figura 9**), es modular y orientada a servicios WMS definidos por la OGC debido a que el visor se planteó como un sistema accesible a través de diferentes clientes, ya sean clientes web o de escritorio. Esta característica hace que el sistema sea interoperable.

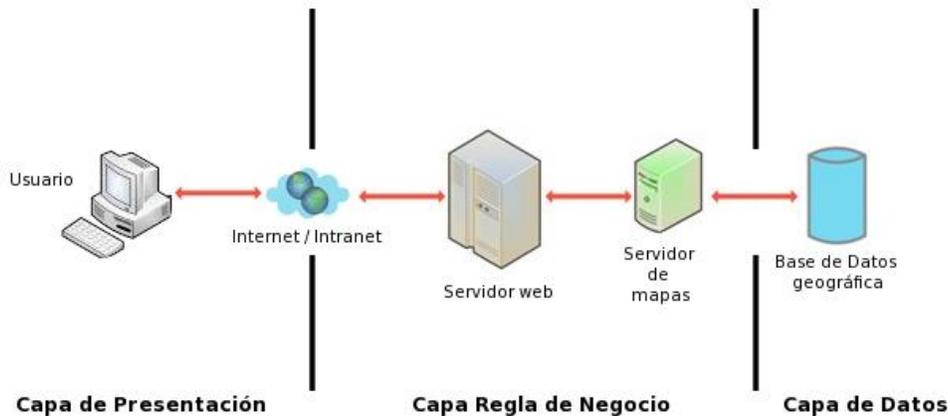


Figura 9. Arquitectura de la aplicación [25].

Por otra parte, el uso exclusivo de tecnologías JavaScript/HTML en la parte del cliente, mejora notablemente la accesibilidad de la aplicación, ya que no es necesaria la instalación de software adicional por parte del cliente para su correcto funcionamiento.

Por otro lado, el proceso de elección de software es un aspecto fundamental para una buena implementación y mantenimiento de cualquier aplicación informática, así pues, es recomendable hacer algunos cuestionamiento sobre el software libre con las siguientes preguntas: ¿El proyecto está bien documentado?, ¿Es conocido el equipo de desarrollo?, ¿Es un software modular?, ¿Qué tan grande es la comunidad de desarrollo?, ¿Qué tan grande es la comunidad de usuarios? [26]; cuantas más preguntas sean contestadas de forma positiva, se podrá confirmar la solidez de un proyecto.

Atendiendo estos cuestionamientos, la implementación práctica del proyecto denominado “Sistema de Consulta de Mapas Digitales sobre Planificación Territorial y Biodiversidad”; se estableció siguiendo un esquema basado en cuatro niveles.

I. Sistema operativo

o Sistema Operativo CentOS release 5.10

Las razones para seleccionar la distribución CentOS:

- ✓ CentOS es un producto derivado de Red Hat Enterprise Linux, Sistema Operativo comercial. CentOS ofrece a sus usuarios estabilidad debido a las similitudes en diseño con el sistema lanzado comercialmente. CentOS solo ejecuta versiones básicas y estables de programas, reduciendo sí el riesgo de fallas en el sistema.
- ✓ Sistema diseñado para correr SELinux (Security-Enhanced Linux), aplicación que permite a los administradores de sistemas controlar las normas de seguridad de acceso al kernel de Linux con módulos muy sofisticados. Originalmente se basó en proyectos iniciados por la Agencia de Seguridad de los Estados Unidos y se considera una aplicación indispensable para servidores Web, de bases de datos y de correo que son accesibles vía Internet.
- ✓ Este Sistema Operativo puede operar por mucho tiempo sin la necesidad de realizar actualizaciones adicionales del sistema. Las actualizaciones de hardware se han desarrollado para ser concurrentes con las actualizaciones de Red Hat Enterprise Linux. El ciclo de soporte de actualizaciones es de aproximadamente (10 años).
- ✓ Se utiliza más a nivel de servidores Web como lo demuestra una reciente encuesta realizada por Web Technology Surveys (**Figura 10**), en septiembre del presente año, donde CentOS es utilizado por el 7.6% de todos los sitios Web.

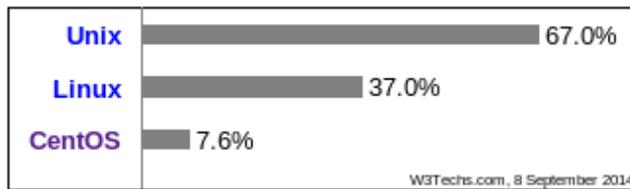


Figura 10. Porcentaje de sitios Web utilizando CentOS. Fuente: <<http://w3techs.com>>

En la (**Tabla 10**) se listan algunos datos técnicos de este Sistema Operativo.

Datos	
Nombre	CentOS (Community Enterprise Operating System)
Desarrollador	El proyecto CentOS, afiliado con Red Hat Enterprise Linux
Modelo de desarrollo	Software Libre
Versión estable	7.0-1406
Núcleo	Linux
Tipo de núcleo	Monolítico
Licencia	GPL
Idiomas	Multilingüe
Método de actualización	Paquete Yum
Gestor de paquetes	RPM
Interfaces de usuario	GNOME y KDE
Sitio Web	http://www.centos.org
Soporte	Documentación de usuario completa Documentación para desarrolladores completa Lista de correo activa

Tabla 10. Ficha técnica, CentOS. Fuente: <<http://www.centos.org/>>

II. Servidor de base de datos

- PostgreSQL v. 9.1.3
- PostGIS v. 1.5.3

Las razones para seleccionar PostgreSQL y PostGIS:

- ✓ PostGIS es una extensión que convierte a PostgreSQL en una base de datos espacial. La combinación de ambos es una solución perfecta para el almacenamiento, administración y mantenimiento de bases de datos espaciales.
- ✓ Compatible con estándares OGC

-
- ✓ Permite exportar datos a través de los comandos (shp2pgsql, pgsq2shp, ogr2ogr, dxf2postgis)
 - ✓ Fiabilidad, estabilidad y excelente rendimiento.
 - ✓ Interoperabilidad, existe un gran número de software SIG de escritorio para visualizar datos PostGIS (gvSIG, uDig, QGIS, OpenJUMP, GRASS, ArcGIS) y servidores de mapas (MapServer, GeoServer, MapGuide, ArcGIS Server).

III. Servidor de mapas

- MapServer v. 6.0.2

Las razones para seleccionar MapServer:

- ✓ MapServer se considera el producto de software libre más maduro y popular para publicar información espacial, así como la creación de mapas interactivos en Internet.
- ✓ Ahorro en desarrollo, hardware y licenciamiento.
- ✓ Mejor rendimiento. Un trabajo de nombre “Benchmarking³” presentado en el congreso anual de FOSS4G, celebrado en septiembre de 2010, en Barcelona, España; demostró que en sistemas Linux, MapServer obtiene un mejor rendimiento al publicar servicios WMS.

- Librerías

- ✓ GEOS v. 3.3.2
- ✓ GDAL v. 1.9.04
- ✓ Proj.4 v. 4.8.0

Estas librerías de código abierto permiten desarrollar aplicaciones adaptadas a los estándares. Proporcionan una implementación de las especificaciones del OGC según van apareciendo. Su diseño y concepción modular hace que numerosas implementaciones de software libre en el ámbito de los SIG hagan uso de ellas [19].

3 <<http://2010.foss4g.org/presentations/benchmarking2010.pdf>> [Consulta: Septiembre de 2014].

IV. Servidor Web

- Apache v. 2.2.22

Las razones para seleccionar Apache:

- ✓ La arquitectura del servidor de Apache es muy modular. El servidor consta de una sección core y diversos módulos que aportan mucha de la funcionalidad que podría considerarse básica para un Servidor Web. El servidor de base puede ser extendido con la inclusión de módulos externos entre los cuales se encuentra el de páginas dinámicas en PHP, Python, Ruby, entre otros.
- ✓ Apache ha incorporado en su soporte a una amplia gama de lenguajes de programación Web, como Perl, PHP y Python, y debido a esto cada programador difiere de las funciones que utiliza (ya que pocas funciones fueron construidas internamente en el servidor).
- ✓ Apache también incluye soporte "SSL" y "TLS". Estos son los protocolos para enviar datos cifrados a través de Internet, y son muy importantes en el desarrollo de aplicaciones que requieren privacidad.
- ✓ Apache cuenta con una gran comunidad de usuarios de soporte técnico, el soporte de Apache se extiende a lo largo de múltiples países, empresas, foros, etc.
- ✓ Apache es multiplataforma. Esto significa que se puede instalar en una amplia variedad de servidores y sistemas operativos. Apache es capaz de ejecutarse en todas las versiones del Sistema Operativo UNIX, Linux, Windows y Mac OS.
- ✓ En general, Apache es uno de los servidores web más utilizado en la actualidad. como lo demuestran recientes encuestas realizadas por Netcraft (**Figura 11**) en agosto de 2014, y por Web Technology Surveys (**Figura 12**) en el mes de septiembre del mismo año; en ambas encuestas Apache figura como el líder en servidores Web.

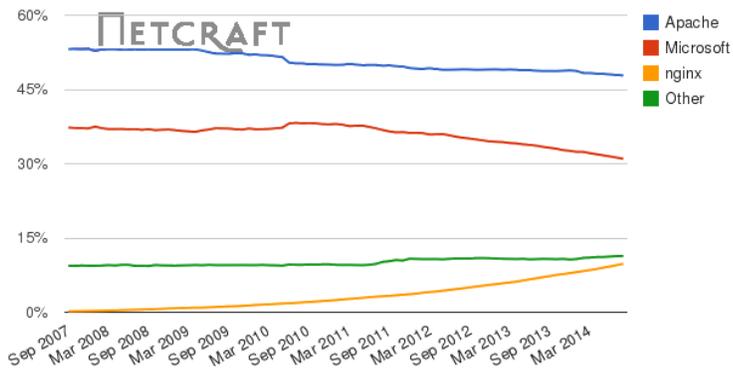


Figura 11. Tendencia del uso de servidores Web. Fuente: <http://www.centos.org/>

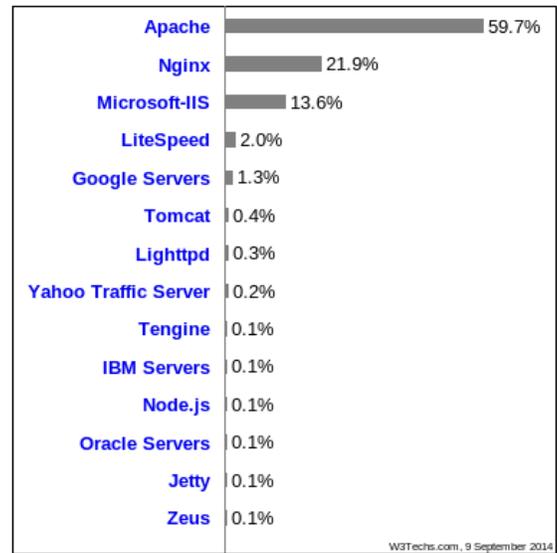


Figura 12. Porcentaje de sitios Web usando diferentes servidores Web. Fuente: <http://w3techs.com>

En la (**Tabla 11**) se describen algunos datos técnicos de este Servidor Web.

Datos	
Nombre	Servidor HTTP Apache
Desarrollador	Apache Software Foundation
Lenguaje de desarrollo	C
Modelo de desarrollo	Software Libre
Lanzamiento inicial	1995
Versión estable	2.4.10
Plataformas soportadas	Unix, Linux, Windows, Mac OS X
Licencia	Apache 2.0, compatible con GPL v3.
Sitio Web	https://httpd.apache.org
Soporte	Documentación de usuario completa Documentación para desarrolladores completa Lista de correo activa

Tabla 11. Ficha técnica, Servidor Apache. Fuente: <http://www.apache.org/>

- Desarrollo Web

- ✓ HTML, JavaScript
- ✓ Visor de mapas: Heron Mapping Client 1.0.0

Las razones para seleccionar Heron Mapping Client:

- ✓ Por las facilidades que ofrece en el diseño de interfaces Web. Heron MC, ofrece una interfaz de usuario de fácil configuración independientemente de cualquier Servidor de Mapas. Permite interactuar con servicios WMS, WFS, Web Map Tile Service (WMTS)⁴.

2.3 Diseño de la base de datos cartográfica

Con el contenido cartográfico definido en el capítulo 2.1, se generó la propuesta del modelo conceptual en cuanto a la forma y estructura de la base de datos geográfica.

El modelo conceptual para una base de datos espacial se refiere a la forma de cómo están caracterizados los elementos del mundo real cuando se almacenan en una base de datos. El modelo entidad-relación es el modelo conceptual más usado en el diseño de una base de datos.

Este modelo representa las entidades, las relaciones entre ellas y los atributos que describen. En este modelo conceptual se usaron elementos del modelo entidad-relación.

El modelo conceptual de los objetos espaciales está formado por tres tablas (**Figura 13**): una que define el objeto en sí (TABLA_NAME), y otras dos tablas que definen el sistema de referencia espacial, llamada SPATIAL_REF_SYS, y el atributo geométrico, llamado GEOMETRY_COLUMNS [27], [28].

⁴ <http://www.opengeospatial.org/standards/wmts>

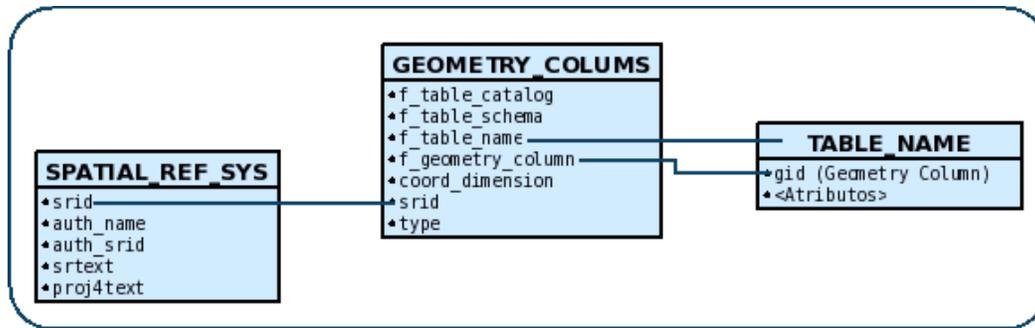


Figura 13. Definición de un objeto espacial en el estándar OGC [29].

La tabla TABLE_NAME almacena el conjunto de fenómenos espaciales. Las columnas de esta tabla representan los atributos de interés; esto significa que cada tupla hace referencia a una característica geográfica.

Cada Geometry Column en la base de datos tiene un registro en la tabla GEOMETRY_COLUMNS. Los datos almacenados para cada Geometry Colum, se describen a continuación:

- Nombre de la tabla TABLE_NAME a la cual la Geometry Column pertenece.
- Nombre de la Geometry Column.
- Identificador del sistema de referencia espacial de la Geometry Column.
- Dimensión de coordenadas para la Geometry Column.

Cada Geometry Column está asociado a un sistema de referencia espacial. El sistema de referencia espacial identifica el sistema de coordenadas para todos los objetos geométricos almacenados en la columna, y dan sentido a los valores numéricos de las coordenadas para cualquier objeto geométrico almacenado en la columna.

La tabla SPATIAL_REF_SYS almacena la información de cada sistema de referencia espacial en la base de datos. La columna SRID constituye una llave entera única para los sistemas de referencia espacial dentro de la base de datos, cuya codificación se basa en la base de datos de la EPSG).

Basándose en las consideraciones anteriores, se elaboró el siguiente modelo conceptual (Figura 14).

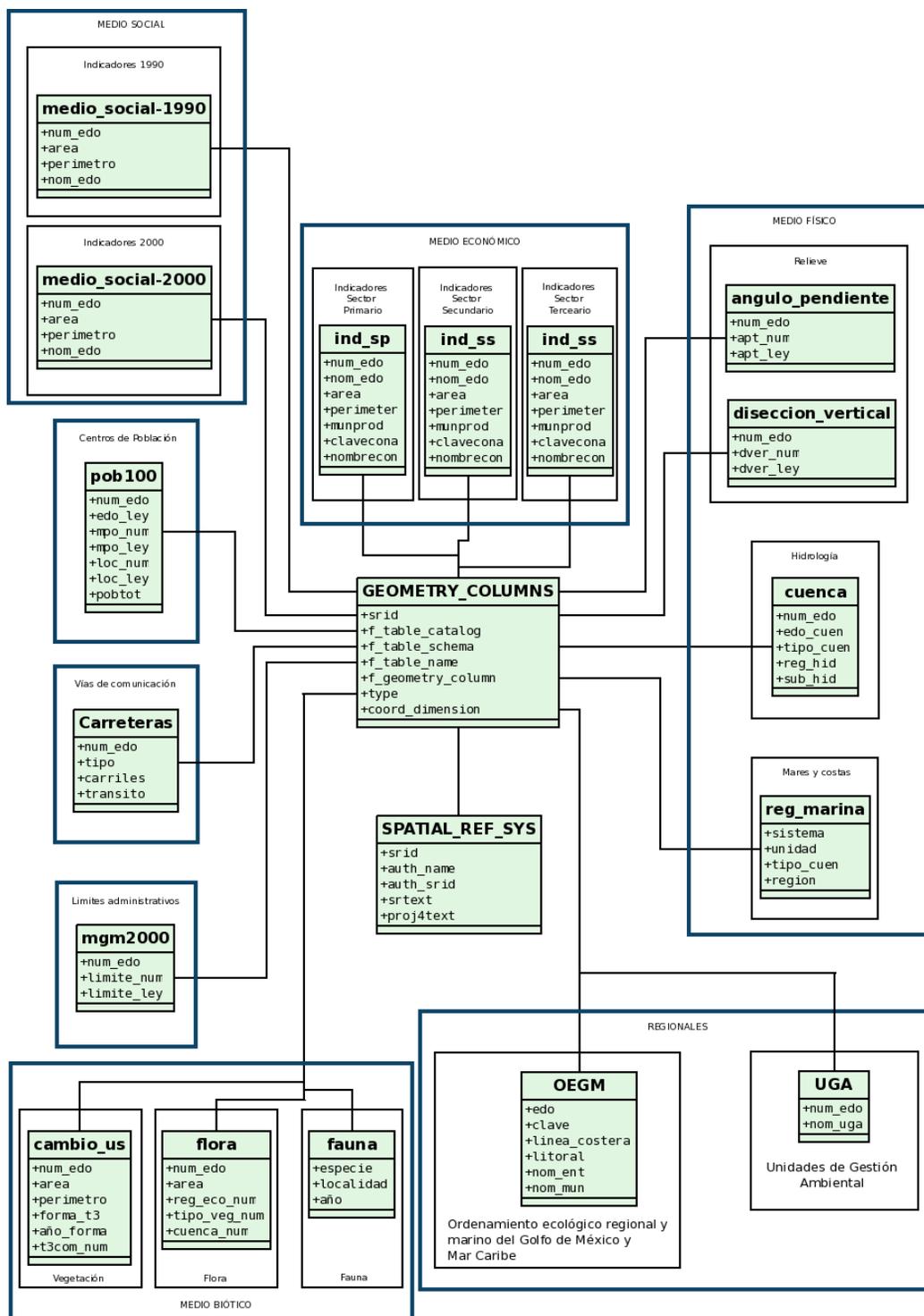


Figura 14. Modelo conceptual para el Sistema de consulta de mapas digitales sobre planificación territorial y biodiversidad, siguiendo el estándar OGC.

Una vez definido el modelo conceptual, se procedió a generar el modelo lógico. El modelo lógico es el desarrollo completo y detallado del modelo conceptual en el que se encuentra una descripción detallada de cada una de las entidades, el diseño de las tablas y los niveles de la información geográfica, con sus atributos, identificadores, relaciones, tipo de dato, longitud del dato y la geometría (punto, línea o polígono) que constituyen la base de datos espacial.

Con base en lo anterior, se planteó el siguiente modelo lógico (**Figura 15**).

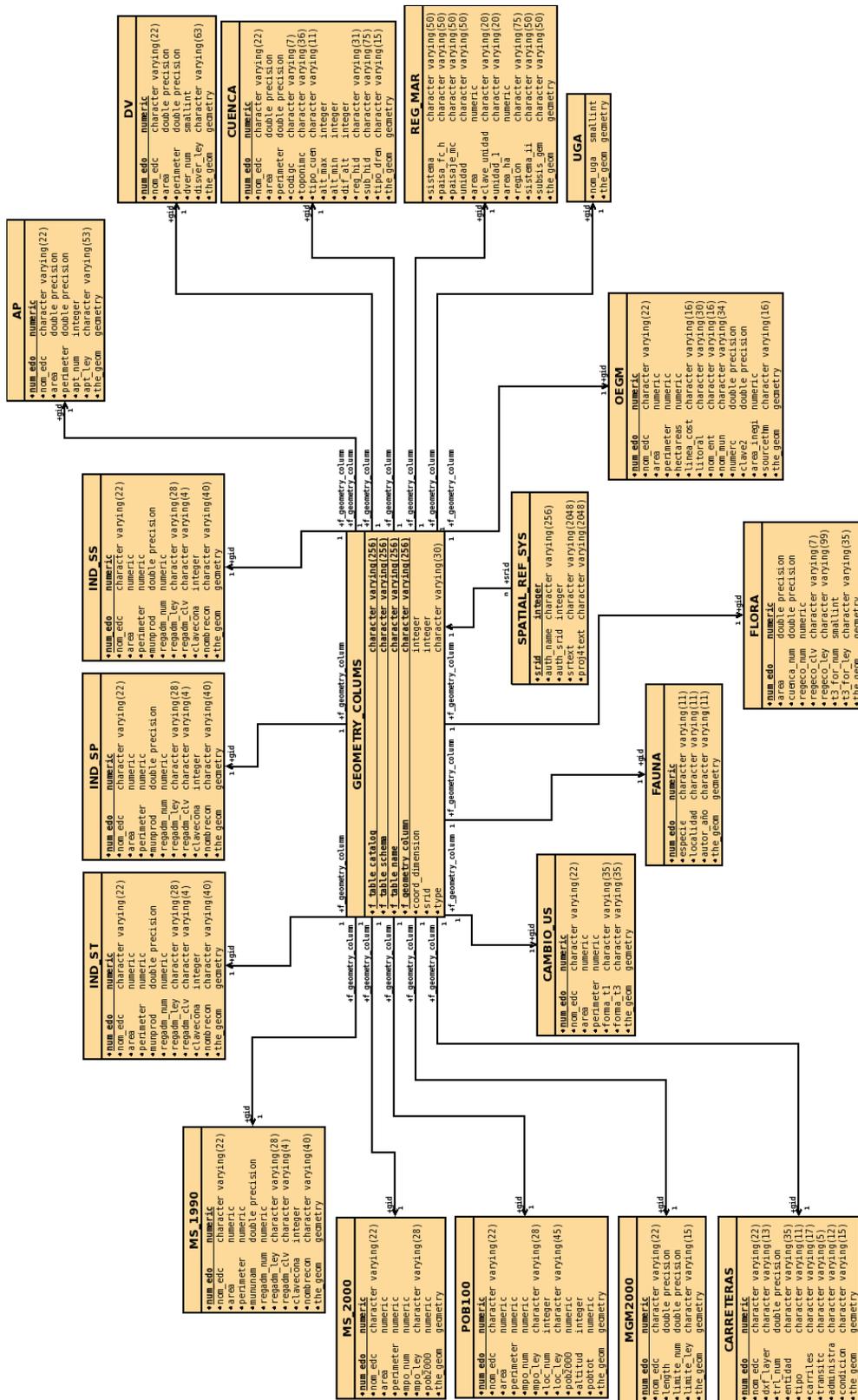


Figura 15. Modelo lógico para el Sistema de consulta de mapas digitales sobre planificación territorial y biodiversidad, siguiendo el estándar OGC.

Consideraciones del modelo lógico.

- En una base de datos espacial, el proceso de normalización encontrará que todas las tablas se encuentran siempre en la primera forma normal [16], [29]:
 - Todos sus atributos son atómicos: Un atributo es atómico si los elementos del dominio son indivisibles, mínimos; *todos los atributos de las tablas cumplen esta condición.*
 - Cada tabla contiene una clave primaria: *La clave primaria de la tabla SPATIAL_REF_SYS es srid. La clave primaria de la tabla GEOMETRY_COLUMNS está formada por los atributos f_table_catalog, f_table_schema, f_table_name y f_geometry_column. La clave primaria del resto de las tablas geográficas está basada en el atributo que se almacena en el campo f_geometry_column de la tabla GEOMETRY_COLUMNS, que dará origen a la clave primaria GID en cada tabla.*
 - La tabla no contiene atributos nulos: *Cada tabla contiene atributos con al menos un valor.*
 - Una columna no puede tener múltiples valores. Los datos son atómicos (si a cada valor X le pertenece un valor Y, entonces a cada valor Y le pertenece un valor X). *Todas las columnas de las tablas cumplen esta condición.*
 - Existe una relación uno a muchos entre las tablas GEOMETRY_COLUMNS y SPATIAL_REF_SYS, ya que en la tabla SPATIAL_REF_SYS, en la columna srid se definen los diferentes sistemas de referencia espaciales que usarán otras tablas, y la tabla GEOMETRY_COLUMNS almacenará los sistemas de referencia que usen esas tablas, con base en los sistemas definidos en SPATIAL_REF_SYS. Es decir, existe una relación de uno a muchos, ya que existen varios sistemas de referencia definidos en SPATIAL_REF_SYS, y cada valor único definido para el sistema de referencia con el que sea insertada cada tabla en la base de datos, se almacena en la tabla GEOMETRY_COLUMNS.

-
- Existe una relación uno a uno entre la tabla GEOMETRY_COLUMNS y cada una de las demás tablas de la base de datos, en el sentido de que en el campo f_geometry_column de la tabla GEOMETRY_COLUMNS se almacenan los valores del campo the_geom de cada una de las tablas. De esta forma, se genera un campo en dichas tablas llamado GID, que será la clave primaria de cada una de las tablas geográficas.
 - Existe otra relación uno a uno entre la tabla GEOMETRY_COLUMNS con las demás tablas de la base de datos, pero no es necesario tenerlo en cuenta en el diagrama: se da en el mismo sentido de que cada tabla como tal se almacena en el campo f_table_name de la tabla GEOMETRY_COLUMNS.

A partir del modelo lógico, el siguiente paso fue la construcción de la base de datos espacial dentro de un lenguaje estándar de consultas con extensión espacial, el cual es el encargado de gestionar la información de forma relacional, a partir de consultas específicas. Para lograr esto, fue necesario definir el SMBD, lo que se hizo en base en el estándar OGC, el cual proporciona el soporte de datos relacionales y dispone de la capacidad de implementar álgebra espacial para su tratamiento; así mismo, se consideraron algunos criterios de funcionamiento, entre ellos, que funcionara en los sistemas operativos más utilizados, interoperabilidad con programas SIG (gvSIG, uDig, ArcGIS Desktop, entre otros); es decir que la información espacial almacenada en la base de datos pudiera ser utilizada por los programas antes mencionados; a partir de esto el SMBD seleccionado para el trabajo fue PostgreSQL con la extensión espacial PostGIS.

El primer paso para trabajar con la información espacial dentro del sistema, fue la adecuación geográfica de los datos (objetos), se re proyectaron los datos vectoriales (shapefiles) al sistema de coordenadas World Geodetic System (WGS84) utilizando su código EPSG: 4326, esto fue posible haciendo uso del comando ogr2ogr de la librería GDAL/OGR.

```
ogr2ogr -f "ESRI shapefile" -t_srs EPSG: 4326 archivo_destino.shp archivo_origen.shp
```

Posteriormente mediante el empleo del comando `shp2pgsql` de PostgreSQL, se exportaron los archivos de formato ESRI (`shp`) a nuestra base de datos espacial.

```
shp2pgsql -s 4326 -c -D -I -W latin1 archivo.shp public.nombre_tabla | pgsq -d nombre_db -h localhost -U nombre_usuario
```

Con lo anterior, se cargaron las tablas con extensión espacial al SMBD, lo que permitirá realizar consultas espaciales que localizarán objetos, a partir de condiciones particulares establecidas por el usuario.

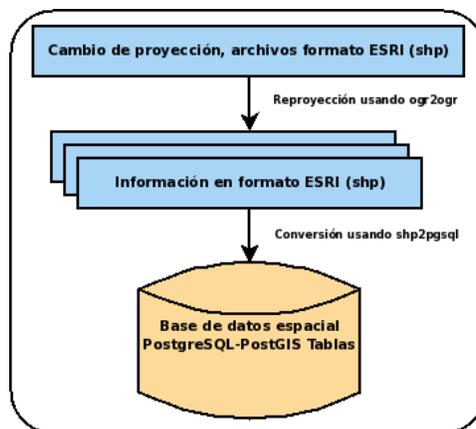


Figura 16. Diagrama de flujo de la implementación de la base de datos.

Con este último procedimiento se logró el almacenamiento de la información espacial estandarizada y homologada; es decir, se obtuvo la base de datos *gisdb*, con contenidos de información tabular y gráfica (**Figura 16**).

2.4 Desarrollo de la interfaz del visor

Para desarrollar la interfaz, se utilizaron servicios Web Map Service (WMS), bajo el estándar publicado por la OGC. Estos servicios son los encargados de procesar y desplegar la información vectorial y alfanumérica a través de Internet, mediante el Servidor de Mapas MapServer, a partir del siguiente procedimiento:

Definición de proyecciones. Uno de los principales problemas a los que nos enfrentamos al momento de mostrar información cartográfica es la multiproyección de cada una de las capas que se pretenden publicar. El problema se acentúa cuando se utilizan proyecciones tan específicas como la de Google Mercator, a veces denominada como EPSG: 900913.

Google Maps es frecuente utilizado para la interpretación de territorio, es por ello que utilizaremos la capa base principal de su servicio de mapas.

Para poder incluir los mapas proporcionados por Google Maps y poder hacer uso de sus servicios, se tuvo la necesidad de establecer su proyección definiendo el código (EPSG: 900913) creado por OpenLayers en el archivo epsg ubicado en (/usr/local/share/proj).

```
# Proyección Mercator
<900913> +proj=merc +a=6378137 +b=6378137 +lat_ts=0.0 +lon_0=0.0 +x_0=0.0
+y_0=0.0 +k=1.0 +units=m +nadgrids=@null +no_defs
```

De esta manera, se dotó a nuestro sistema de la capacidad de reconocer y publicar mapas de Google Maps y OpenStreetMap a través del Servidor de mapas MapServer.

Creación de áreas de trabajo de información cartográfica (Vistas geográficas). Como se mencionó en el apartado 1.6.4.1, el software gvSIG es una herramienta orientada al manejo de información geográfica, por ende la creación de las vistas se generaron bajo esta herramienta SIG (**Figura 17**), con los parámetros de configuración que se describen a continuación:

- Unidades de mapas: en grados;
- Unidades de medida: en metros;
- Sistema de proyección: EPSG: 4326

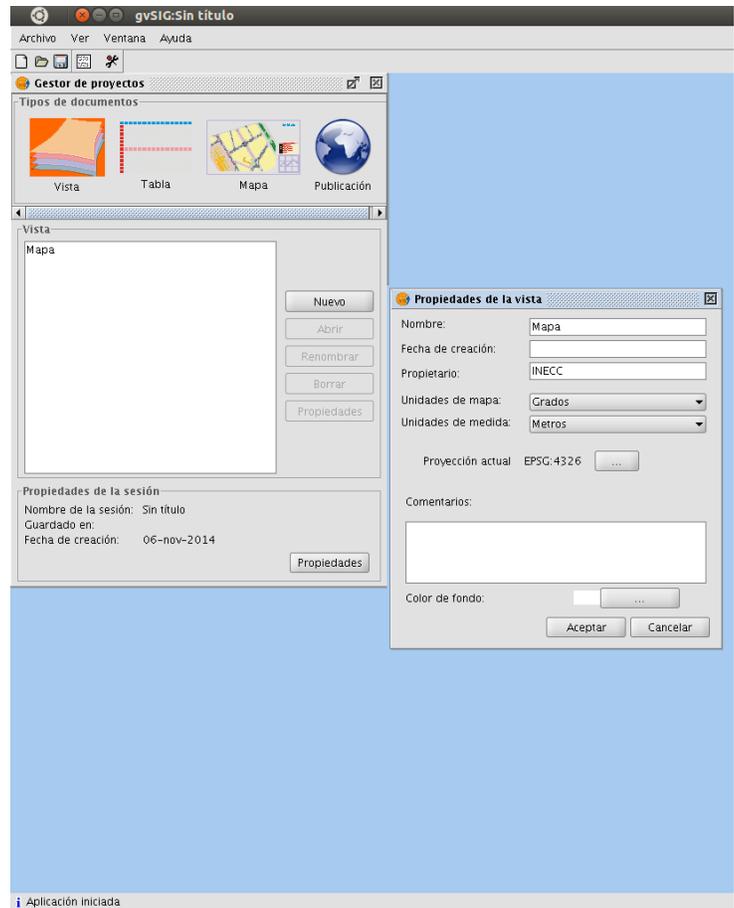


Figura 17. Creación de vistas, gvSIG.

Una vez establecidos los parámetros de cada una de las vistas, se procedió a cargar las capas de información asociadas a cada servicio (**Figura 18**), esto se logró mediante la creación de una conexión a la base de datos espacial (**Figura 19**) a través del controlador JDBC (Java Database Connectivity) especificando los siguientes parámetros.

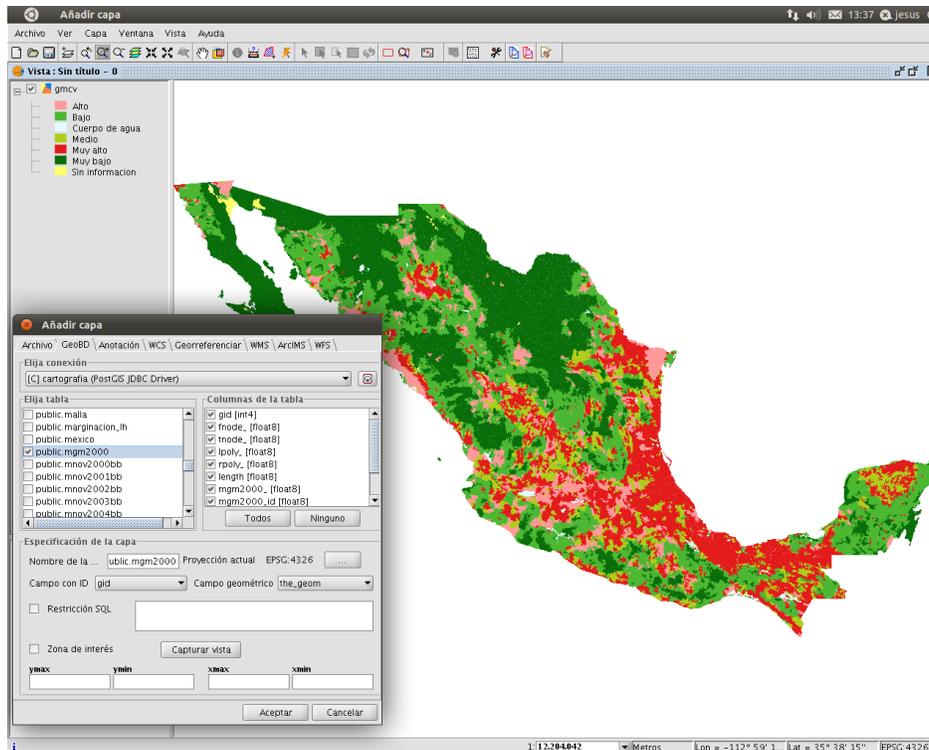


Figura 18. Vista; cargando y visualizando información almacena en la base de datos espacial, gvSIG.

Parámetros de conexión

- Nombre de la conexión: Cartografía;
- Driver: PostGIS JDBC Driver;
- URL del Servidor: localhost;
- Puerto: 5432;
- Nombre de BD: gisdb;
- Usuario: gis
- Contraseña:

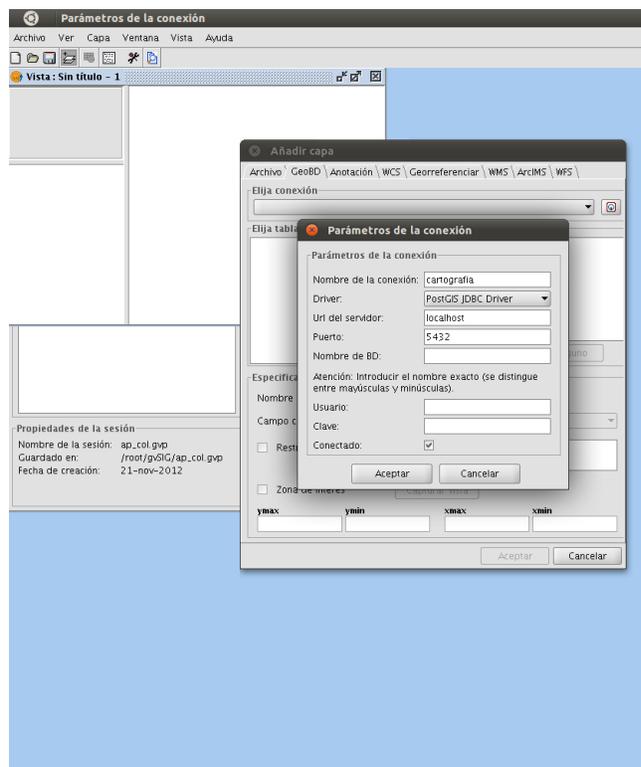


Figura 19. Conexión base de datos espacial, gvSIG.

Publicación de las vistas. Tras configurar cada una de las vistas, se procedió a generar los servicios OGC. Para ello se empleó la **Extensión de Publicación** de gvSIG versión 1.1. 2. El objetivo de esta extensión es dotar a gvSIG de la capacidad de publicar información que se está manipulando en sus vistas en un Servidor que implementa servicios OGC como WMS, WFS o WCS con gran facilidad. Básicamente esta extensión añade la funcionalidad a gvSIG para permitir a un usuario generar configuraciones de los servicios siguientes [24]:

- MapServer 5.x WMS con fuente de datos shapefiles, raster y PostGIS.
- MapServer 5.x WS con fuente de datos shapefiles y PostGIS.
- MapServer 5.x WCS con fuente de datos raster.
- GeoServer 1.5.4 WFS con fuentes de datos PostGIS y shapefiles.

Por lo anterior, el empleo de esta extensión nos permitió convertir los recursos locales visibles en este SIG de escritorio en recursos remotos como “Layers WMS” (**Figura 20**); para tal efecto se realizó lo siguiente:

- A través de gvSIG, abrimos la vista a ser publicada.
- Posteriormente, se creó un documento de Publicación por medio del Gestor de proyectos.
- Para el documento de publicación se eligió a MapServer 5.x como Servidor Local, además de establecer como servicio OGC un WMS 1. 1. 1; y se definió la URL del servidor Local “http://localhost/cgi-bin/mapserv”.

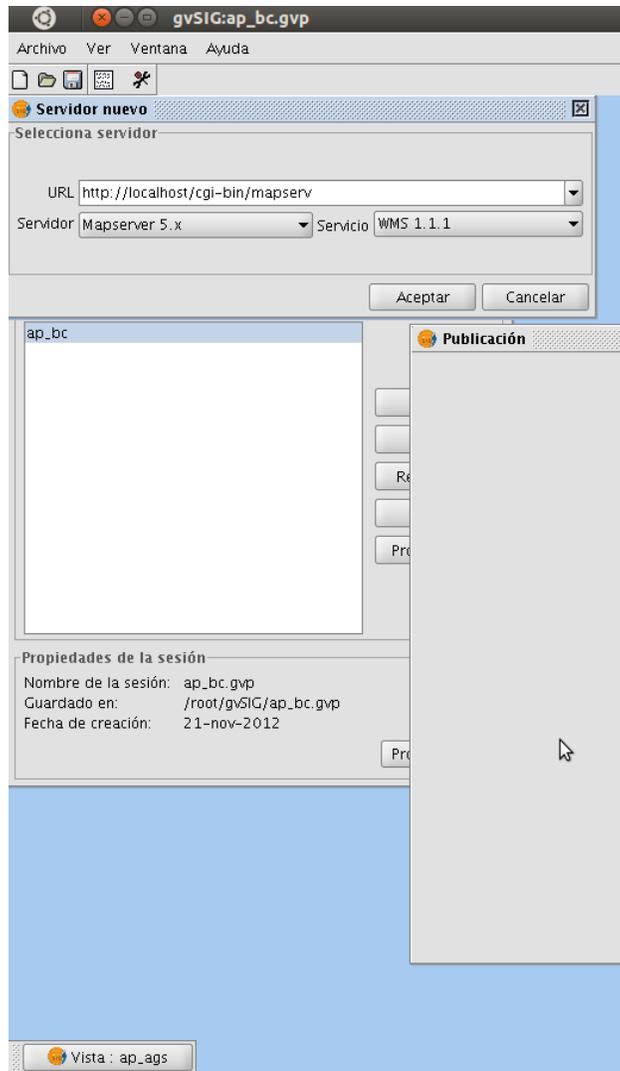


Figura 20. Servidor local y servicio OGC, gvSIG.

- Se personalizaron los parámetros para cada uno de los componentes de la publicación: el Servidor, el servicio y los recursos remotos accesibles a través del servicio OGC. Por recursos se entiende cualquier tipo de entidad que podemos manipular con gvSIG y que es susceptible a convertirse en un recurso remoto accesible a través de un servicio Web como capas WMS, WFS o WCS.

A continuación se describen los parámetros básicos para configurar el Servidor de Mapas MapServer:

Servidor

- Archivo de configuración. Se trata de un archivo .map (MapFile) donde se genera la configuración del servicio (**Figura 21**). Este archivo contiene una serie de parámetros que definen al mapa, es decir, el sistema de referencia, conexión a la base de datos PostGIS, la jerarquía de las capas disponibles, el estilo con el que se representarán, su simbología y su escala; es decir, todo lo relacionado con un servicio en particular está definido aquí.

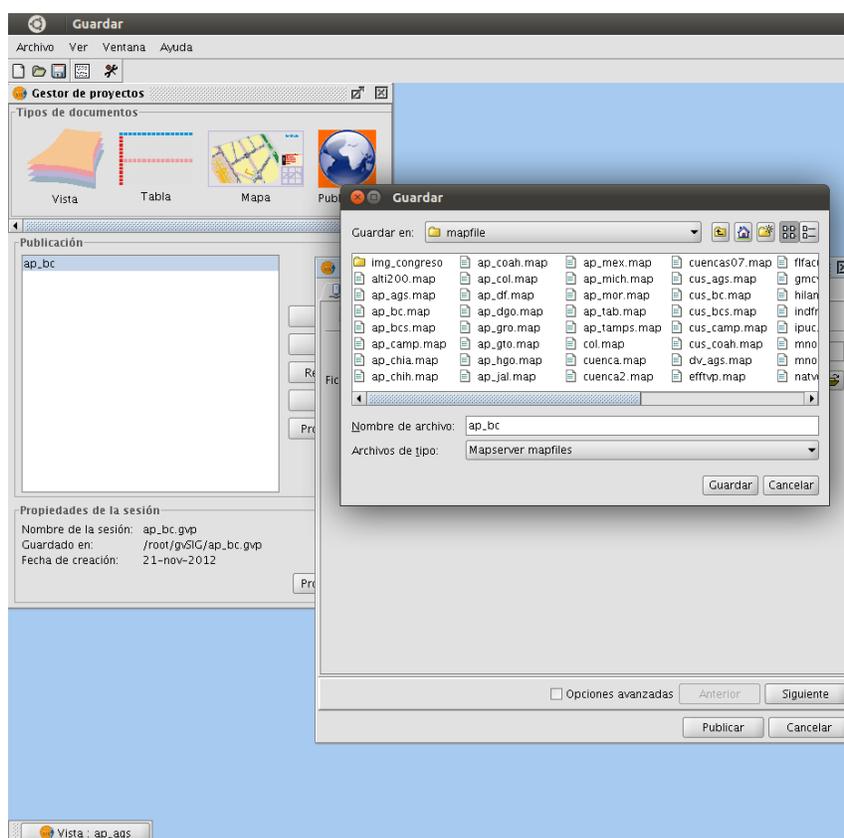


Figura 21. Definiendo el directorio y nombre del archivo MapFile, gvSIG.

Servicio

- Título. Nombre asociado a un servicio.
- Online resource (URL del servicio). Se estableció la ruta del Servidor local y la ruta del archivo MapFile (**Figura 22**). Es importante aclarar que en este caso, la ruta del archivo MapFile coincidió con la ruta donde se generó dicho archivo.

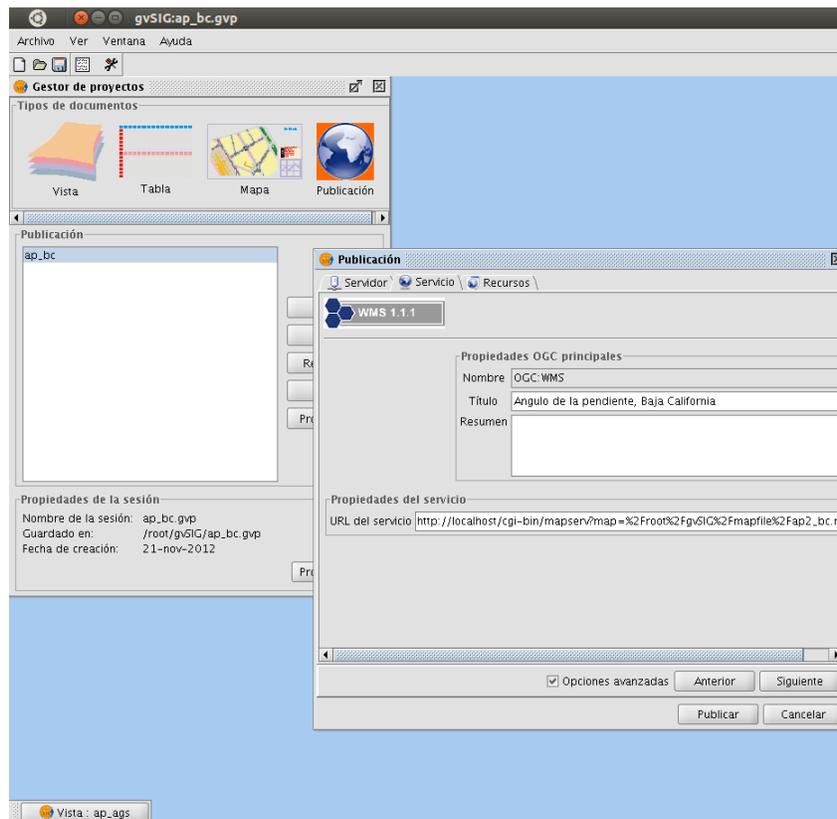


Figura 22. Parámetros del Servicio, gvSIG.

Recursos

- Nombre. Seudónimo que identifica el recurso remoto. Este pseudónimo fue generado automáticamente por gvSIG y es utilizado por el cliente del servicio de una forma transparente para el usuario. Cabe destacar que en MapServer, el nombre del recurso es el nombre de la capa definida en la vista de gvSIG, por consiguiente, este parámetro fue modificado.
- Título. Nombre que permite al usuario identificar el recurso.
- Resumen. Descripción del recurso.
- Consultable. Se activó esta opción para habilitar la operación GetFeatureInfo, con la finalidad de consultar los valores alfanuméricos de la capa.
- Por último, para finalizar la configuración se presionó el botón Publicar (**Figura 23**); y como resultado se obtuvo un mensaje que indicaba que la publicación se había generado correctamente; obteniendo así un servicio WMS.

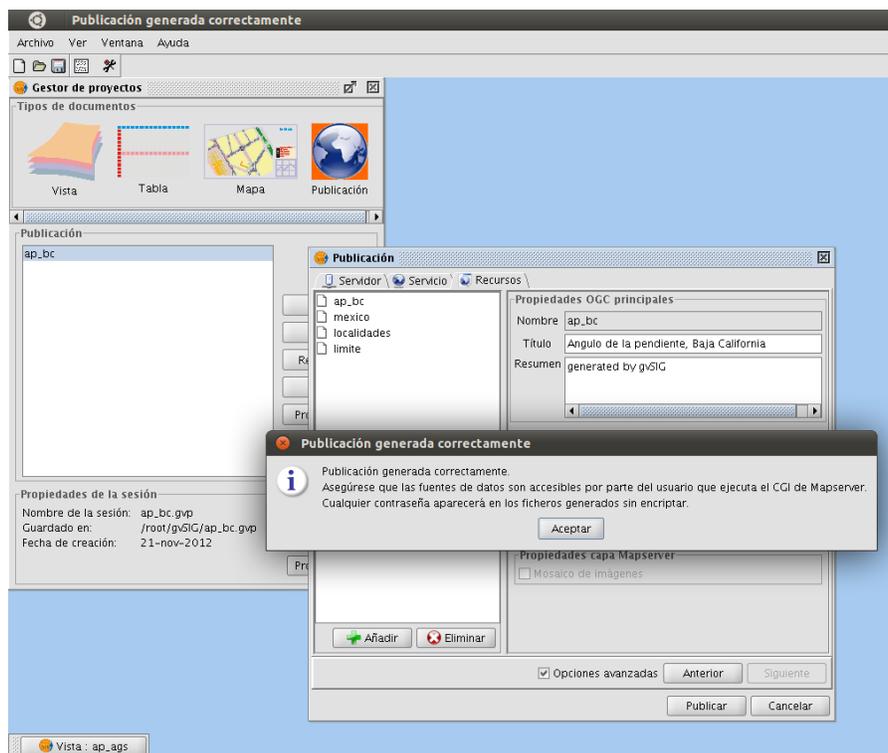


Figura 23. Publicación del Recurso, gvSIG.

Una vez completado el proceso de publicación de las vitas, se procedió a realizar los cambios necesarios en los archivos MapFile, para que el Servidor de Mapas pudiera entregar los datos requeridos a través de dichos archivos (**Figura 24**). Cabe hacer mención que la creación de los archivos MapFile generó automáticamente otros archivos para el correcto funcionamiento. Estos archivos fueron el de símbolos (symbols.sys), el de fuentes (fonts.txt) y una fuente truetype (Vera.ttf) que se encuentran en el mismo directorio del MapFile. Por consiguiente el primer paso fue editar el archivo “.sys” para que MapServer pudiera interpretar los símbolos; para ello fue necesario añadir la palabra reservada “SYMBOLSET” al inicio y “END” al final del archivo, además de eliminar todos los símbolos del tipo CARTOLINE.

Después, se especificó el nivel jerárquico de la información topológica, es decir, la forma de representar la información. Por ejemplo polígonos para los estados, líneas para las vías de comunicación y puntos para localidades. El orden en que se colocó la información en los archivos MapFile fue polígono como capa base, línea como información subsecuente y punto como información final.

También se agregó el parámetro CONFIG MS_ERRORFILE, para establecer la ubicación del archivo que guardará el reporte de incidencias de MapServer.

```
CONFIG "MS_ERRORFILE" "../../../tmp/mapserver/log.txt"
```

Asimismo se añadió el parámetro CONFIG PROJ_LIB, para definir la ubicación de las proyecciones establecidas en la librería Proj.4.

```
CONFIG "PROJ_LIB" "/usr/local/share/proj"
```

Como en nuestro caso se tienen diferentes sistemas de proyección (EPSG: 900913, capas base de OpenStreetMap y Google Maps para el territorio nacional y EPSG: 4326 para todas las demás) fue necesario definir estas proyecciones de tal manera que MapServer pudiera re proyectar los datos de las capas al sistema de referencia espacial del mapa al momento de dibujar las capas. Para ello se modificó el parámetro “PROJECTION”.

```
PROJECTION
  "init=epsg: 900913"
END
```

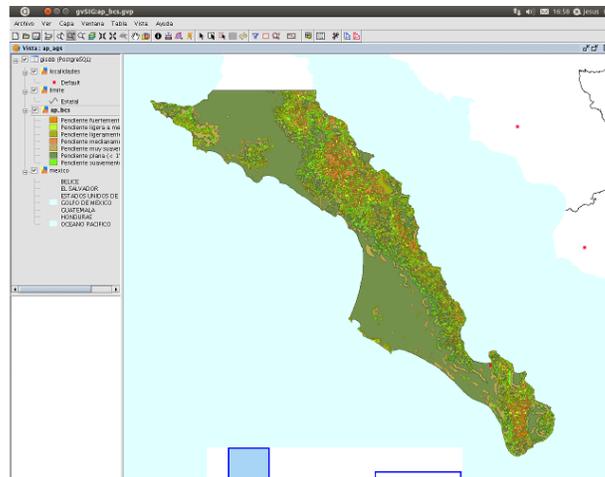
De igual forma, fue imprescindible redefinir la sección WEB a nivel global y METADATA a nivel de capa, para establecer los parámetros que serán usados por los servicios OGC (WMS y GML).

A nivel Global:

```
WEB
  METADATA
    "wms_encoding" "UTF-8"
    "wms_title" "nombre mapa"
    "wms_abstract" "descripción básica del servicio"
    "wms_srs" "EPSG: 4326 EPSG: 900913"
    "wms_onlineresource" "el URL del servicio de mapa"
    "wms_format" "image/png"
    "wms_enable_request" "*"
    "wms_feature_info_mime_type" "text/html"
    "wms_server_version" "1.1.1"
  END
END
```

A nivel de capa:

```
METADATA
  "wms_title" "algún nombre"
  "wms_abstract" "descripción básica del servicio"
  "wms_extent" "[x_min] [y_min] [x_max] [y_max]"
  "gml_include_items" "all"
  "gml_nombre_campo_alias" "algún_nombre"
  "gml_exlude_items" "campos_a_excluir"
END
```



MapFile

```

MAP
NAME ap_bc_gvsg
EXTENT -117.57051153665 12.8070830739669
8.4312757004108 33.5554905577201
SIZE 400 400
SYMBOLSET "J:\\J.J.J\\symbols\\symbols.sym"
FONTSET "J:\\J.J.J\\fonts\\fonts.txt"
UNITS DD
SCALEBAR
UNITS KILOMETERS
INTERVALS 3
TRANSPARENT TRUE
_OUTLINECOLOR 0 0 0
END
IMAGETYPE PNG
OUTPUTFORMAT
NAME png
DRIVER AGG/PNG
MMETYPETYPE "image/png"
IMAGECODE RGB
EXTENSION "png"
END
LEGEND
IMAGECOLOR -1 -1 -1
LABEL
FONT "Vera"
ANGLE FOLLOW
COLOR 128 128 128
TYPE justtype
SIZE 8
STATUS ON
TRANSPARENT ON
END
CONFIG "MS_ERRORFILE"
"J:\\J.J.J\\mapserver\\log.txt"
WEB
METADATA
"xml:encoding" "UTF-8"
"xml:id" "Angulo de la pendiente, Baja
California"
"xml:abstract" ""
"xml:ats" "EPSG:4326"
"xml:onlineonly" ""
"http://mexas.inecg.gob.mx/cgi-bin/ap_bc"
"xml:format" "image/png"
"xml:enable_request" ""
"xml:feature_info_mime_type" "text/html"
"xml:server_version" "1.1.1"
END
PROJECTION
"proj=epsg:4326"
END
LAYER
NAME "ap_bc"
STATUS ON
TYPE POLYGON
DATA "the_geom from public.ap_bc using
"the_geom and using srid=4326"
CONNECTIONTYPE POSTGIS
CONNECTION "user='*****' password='*****'
host='localhost' port='5432' dbname='*****'"
MINSCALE -1.0
MINSCALE -1.0
TRANSPARENCY 70
DUMP TRUE
TEMPLATE ""
CLASSITEM "just_ley"
PROJECTION
"proj=epsg:4326"
END
CLASS
STYLE
COLOR 255 251 195
_OUTLINECOLOR 255 251 195
WIDTH 1
END
EXPRESSION "Pendiente plana (< 1°)"
NAME "Pendientes planas menores a 1°"
END
CLASS
STYLE
COLOR 255 255 0
_OUTLINECOLOR 255 255 0
WIDTH 1
END
EXPRESSION "Pendiente muy suavemente
inclinada (1° - 3°)"
NAME "Pendientes muy suavemente
inclinadas de 1° a 3°"
END
CLASS
STYLE
COLOR 166 255 165
_OUTLINECOLOR 166 255 165
WIDTH 1
END
EXPRESSION "Pendiente suavemente
inclinada (3° - 5°)"
NAME "Pendientes suavemente inclinadas de
3° a 5°"
END
CLASS
STYLE
COLOR 2 220 0
_OUTLINECOLOR 2 220 0
WIDTH 1
END
EXPRESSION "Pendiente ligeramente
inclinada (5° - 10°)"
NAME "Pendientes ligeramente e inclinadas de
5° a 10°"
END
CLASS
STYLE
COLOR 146 174 47
_OUTLINECOLOR 146 174 47
WIDTH 1
END
EXPRESSION "Pendiente ligera a
medianamente inclinada (10° - 15°)"
NAME "Pendientes ligera a medianamente
inclinadas de 10° a 15°"
END
CLASS
STYLE
COLOR 255 173 0
_OUTLINECOLOR 255 173 0
WIDTH 1
END
EXPRESSION "Pendiente medianamente
inclinada (15° - 20°)"
NAME "Pendientes medianamente inclinadas
de 15° a 20°"
END
CLASS
STYLE
COLOR 255 0 0
_OUTLINECOLOR 255 0 0
WIDTH 1
END
EXPRESSION "Pendiente fuertemente
inclinada (20° - 30°)"
NAME "Pendientes fuertemente inclinadas de
20° a 30°"
END
CLASS
STYLE
COLOR 210 143 84
_OUTLINECOLOR 210 143 84
WIDTH 1
END
EXPRESSION "Pendiente muy fuertemente
inclinada (30° - 45°)"
NAME "Pendientes muy fuertemente
inclinadas 30° a 45°"
END
CLASS
STYLE
COLOR 161 102 50
_OUTLINECOLOR 161 102 50
WIDTH 1
END
EXPRESSION "Pendiente abruptas mayores
de 45°"
NAME "Pendientes abruptas mayores de 45°"
END
METADATA
"xml:id" "ap_bc_gvsg"
"xml:abstract" "Generated by gvSIG"
"xml:extent" "-117.12496449825
27.9993368487541 -112.764308973938
32.7189729752263"
"xml:layer_group" "igido (PostgreSQL)"
"xml:include_items" "just_ley"
"xml:app_ley_alias" "Pendiente"
END
END # Layer
END # Map File

```

Figura 24. Publicación de vista, MapFile asociado, gvSIG.

Como proceso final para integrar la interfaz, se optó por encapsular diferentes funcionalidades en secciones separadas; por lo anterior el diseño gráfico de la interfaz respondió al siguiente esquema (Figura 25).

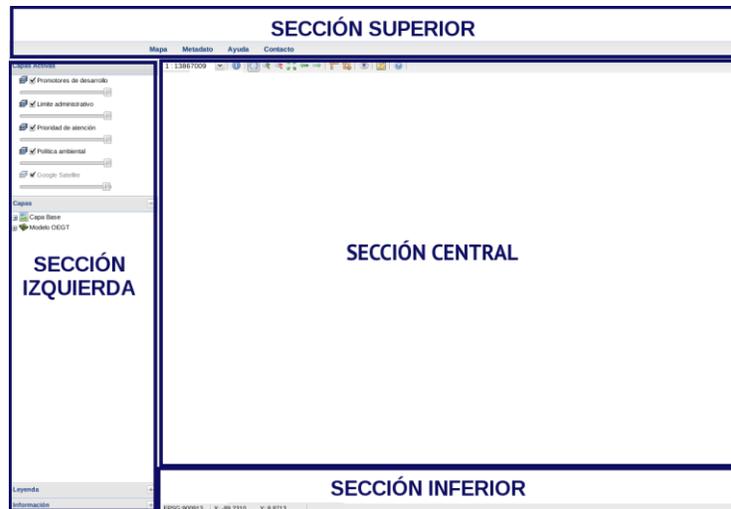


Figura 25. Descripción específica de la estructura del visor de mapas.

Sección superior

Esta sección está constituida por el logotipo de la institución, el nombre del servicio y una barra de menú con las siguientes pestañas: mapa, metadato, ayuda y contacto.

- Mapa: Al iniciar la aplicación, esta pestaña siempre está activada. Dentro de esta pestaña se encuentra un área especial para el despliegue del mapa.
- Metadato: Esta pestaña muestra un documento que describe las características principales del recurso según la norma FGDC (Content Standards for Digital Geospatial Metadata).
- Ayuda: Muestra información básica relacionada con los componentes y estructura del visor.
- Contacto: Contiene información para que los usuarios tengan la posibilidad de comunicarse con el responsable del sistema.

Sección central

Esta sección corresponde a la zona de despliegue del mapa, a través del cual el usuario puede visualizar e interactuar con las capas geográficas de interés desplegadas, asociadas con una simbología y etiquetado, así como comportamientos a través de las escalas, etc. También se encuentra una barra de herramientas con funcionalidades explícitas; básicamente muestra las herramientas útiles al usuario que permiten la navegación, visualización tanto espacial como de sus atributos, además de conocer la escala aproximada del mapa, etc.

Sección izquierda

En esta sección se encuentran los paneles de:

- Capas Activas
- Capas
- Leyenda
- Información

Sección inferior.

Esta sección contiene los indicadores del mapa, que permiten al usuario conocer el código EPSG del sistema de referencia, así como las coordenadas geográficas del mapa.

Es importante destacar, que estas secciones de trabajo son independientes entre sí, a pesar de que todas ellas utilizan la misma interfaz; además de que cada sección tiene asociado uno o más módulos en donde se pueden cargar diferentes funcionalidades.

Metodológicamente, la interfaz utilizó diferentes librerías y lenguajes de programación (**Figura 26**):

- HTML. Definición de contenido de página Web;
- CSS. Estilo y formato de los objetos de la página;
- JavaScript. Funciones e interacción con el usuario a través del sistema;

Librerías desarrolladas en JavaScript

- OpenLayers. Cliente Web de mapas;
- GeoExt. Librería de funcionalidades geográficas avanzadas;
- ExtJS. Librería de funcionalidades avanzadas para Web.

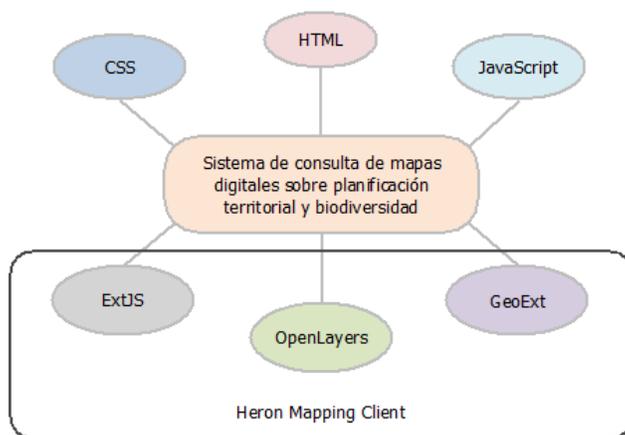


Figura 26. Lenguajes de programación.

A nivel físico, la interfaz se estructuró mediante la personalización del cliente Heron Mapping Client en cuatro archivos o scripts escritos en lenguajes HTML (index.html), JavaScript (layout.js y options.js) y un archivo de texto (logo.txt). En la (**Figura 27**) se resumen las funciones de cada uno de ellos.

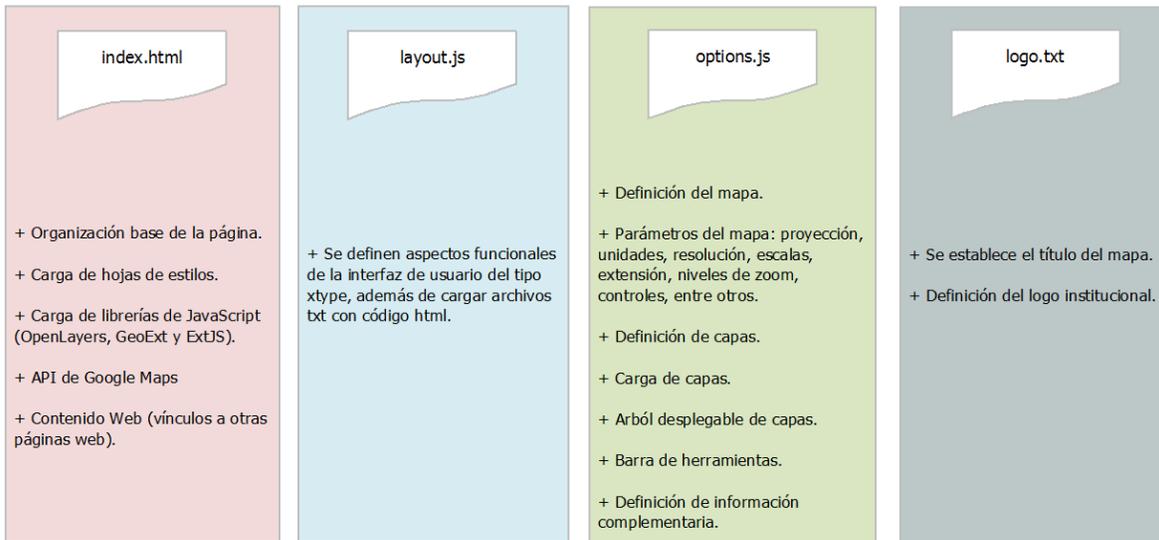


Figura 27. Funciones desempeñadas por cada script.

Logrando así la presentación de contenidos en cuanto a las herramientas de navegación (acercamiento, desplazamiento), funciones de consulta, de medición, de dibujo, etc. (Figura 28, 29).

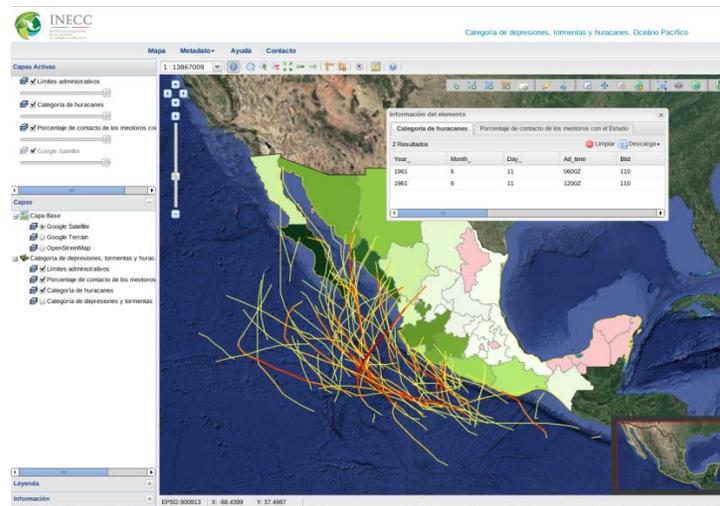


Figura 28. Interfaz del visor de mapas. Categoría de depresiones, tormentas y huracanes, Océano Pacífico.

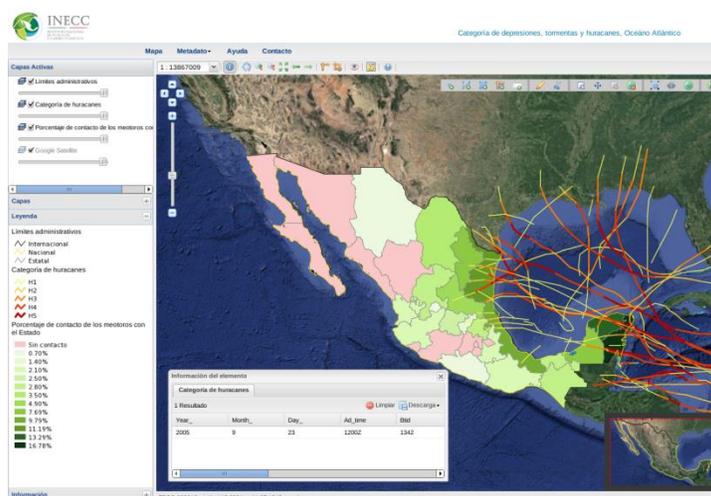


Figura 29. Interfaz del visor de mapas. Categoría de depresiones, tormentas y huracanes, Océano Atlántico.

2.5 Implementación

De acuerdo a la tecnología necesaria para la implementación y puesta en funcionamiento del “Sistema de Consulta de Mapas Digitales sobre Planificación Territorial y Biodiversidad”, la evaluación de los recursos tecnológicos, se llevó a cabo bajo dos enfoques: hardware y software.

En cuanto al hardware, el servidor donde se implementó el sistema tiene los siguientes aspectos técnicos (**Tabla 12**).

Servidor PowerEdge 4600	
CPU	Intel® Xeon™ 2.8Ghz
Memoria	24GB
Disco duro	8x36 GB 10K RPM, SCSI
Configuración de disco duro	RAID 10
Disco óptico interno	CD-ROM
Conectividad TCP/IP	2 tarjetas 10/100/1000

Tabla 12. Hardware, configuración.

Evaluando el hardware existente y tomando en cuenta que la infraestructura de red física del instituto cumple las necesidades del sistema, no fue necesario realizar inversión alguna, ya que los mismos satisfacen los requerimientos establecidos tanto para el desarrollo y puesta en funcionamiento del sistema propuesto.

En cuanto al software, el INECC cuenta con todas las herramientas que se emplearon para el desarrollo del proyecto y funcionamiento del sistema (**Tabla 13**), debido a que el proyecto se basó en software libre y de código abierto.

Categoría	Herramientas OpenSource
Sistema Operativo	Linux/CentOS
Creación de datos	gvSIG
Conversión de datos	ogr2ogr (GDAL/OGR), shp2pgsql (PostGIS)
Edición de datos	PostGIS
Análisis de datos	PostGIS
Visualización de datos (Internet)	Apache, MapServer, Heron Mapping Client

Tabla 13. Lista del software libre y de código abierto usado en este proyecto.

En lo que respecta a la seguridad de la información, que es fundamental para cualquier aplicativo que maneje datos sensibles o en su defecto muy solicitados; se utilizó una tecnología de almacenamiento en múltiples discos llamada (RAID), para formar una única unidad lógica en la que se almacenaron los datos de una forma redundante, garantizando así la disponibilidad del sistema y la protección frente a pérdidas de datos ante daños potenciales en un disco duro. [31].

Tomando en cuenta los aspectos técnicos del Servidor y la naturaleza del sistema, se optó por implementar un RAID 10 (**Figura 30**). La razón de utilizar esta configuración fue para obtener los beneficios de la redundancia (espejo) que ofrece el RAID 1 (primer nivel) y la unión de espacio que ofrece el RAID 0 (segundo nivel), para así conseguir un alto rendimiento, debido a que los datos se distribuyen en bandas y no hay sobrecarga de paridad [31].

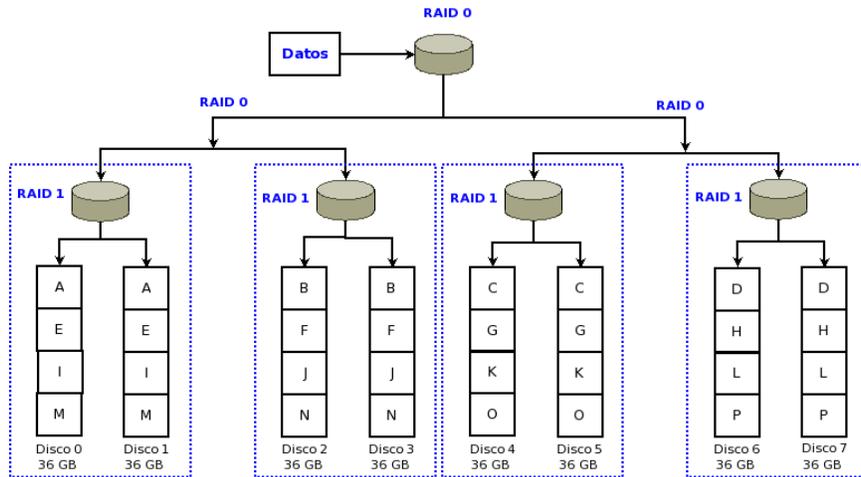


Figura 30. Raid 10 [31].

Una vez creado el RAID, se procedió a la instalación del Sistema Operativo Linux/CentOS, el cual es un clon Open Source de las soluciones Linux empresariales de RedHat; 100% compatibles con estas, con una gran ventaja, no tiene costo. La instalación del sistema escapa a los objetivos de este documento; no obstante es importante mencionar, que para la instalación del sistema fue necesario crear un diseño personalizado de particiones, con lo cual se garantizó una mejor organización del sistema, además de evitar la pérdida total de datos en caso de presentarse una falla grave en el sistema.

Una vez definido el sistema de particiones, se procedió a la instalación del sistema base en el servidor de una forma básica, es decir, en modo texto y sin ningún servicio.

Una vez concluido este proceso, se inició la instalación y configuración de los paquetes, librerías y dependencias requeridas para la puesta en funcionamiento del visor. Cabe mencionar que fue necesario crear un directorio de trabajo, para almacenar los archivos fuentes. Por lo anterior todos los archivos que se descargaron de sitios Web, se almacenaron en “/opt/fuentes”.

```
[root@mapas ~]# mkdir -p /opt/fuentes
```

Instalación de Apache

El primer paso, fue instalar los compiladores de C y C++, así como sus respectivas dependencias. Para ello desde una consola, invocamos el siguiente comando.

```
[root@mapas ~]# yum install gcc gcc-c++
```

Del sitio <http://www.apache.org/> se descargó el archivo `httpd-2.2.22.tar.gz` y mediante el comando `tar`, se procedió a su descompresión.

```
[root@mapas fuentes]# tar -zxvf httpd-2.2.22.tar.gz
```

Dentro del directorio `httpd-2.2.22`, se creó un archivo de nombre `compila_apache.sh` (**Script 1**) con el editor `vi`, donde se establecieron los parámetros de instalación.

```
./configure \  
--prefix=/usr/local/apache2 \  
--enable-deflate \  
--enable-info \  
--enable-mime-magic \  
--enable-rewrite=shared \  
--enable-so \  
--enable-speling \  
--enable-ssl \  
--enable-unique_id \  
--enable-usertrack \  
--with-mpm=prefork \  
--with-apxs2
```

Script 1. Contenido de `compila_apache.sh` para Apache.

Se cambiaron sus atributos para hacerlo ejecutable:

```
[root@mapas httpd-2.2.22]# chmod a+x compila_apache.sh
```

Para iniciar el proceso de instalación, ejecutamos el script “compila_apache.sh”.

```
[root@mapas httpd-2.2.22]# ./compila_apache.sh
```

Posteriormente, se utilizó el comando “make” para generar los archivos ejecutables de Apache y “make install” para su instalación.

```
[root@mapas httpd-2.2.22]# make & make install
```

Finalmente, se inició el servicio de Apache con la instrucción siguiente.

```
[root@mapas httpd-2.2.22]# /usr/local/apache2/bin/apachectl start
```

Para realizar pruebas de funcionamiento, realizamos un escaneo al puerto 80 con el comando nmap (**Figura 31**), lo anterior debido a que este puerto es utilizado cuando se inicia Apache.

```
[root@mapas ~]# nmap -p 80 IP
```

```
[root@mapas ~]# nmap -p 200.33.112.237
Starting Nmap 6.40 ( http://nmap.org ) at 2014-11-24 11:10 CST
Nmap scan report for mapas.inecc.gob.mx (200.33.112.237)
Host is up (0.00020s latency).
PORT      STATE SERVICE
80/tcp    open  http
Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
[root@mapas ~]# █
```

Figura 31. Escaneo de puerto para Apache, nmap.

Finalmente, para comprobar su funcionamiento se escribió en un navegador Web la dirección http://localhost (**Figura 32**).

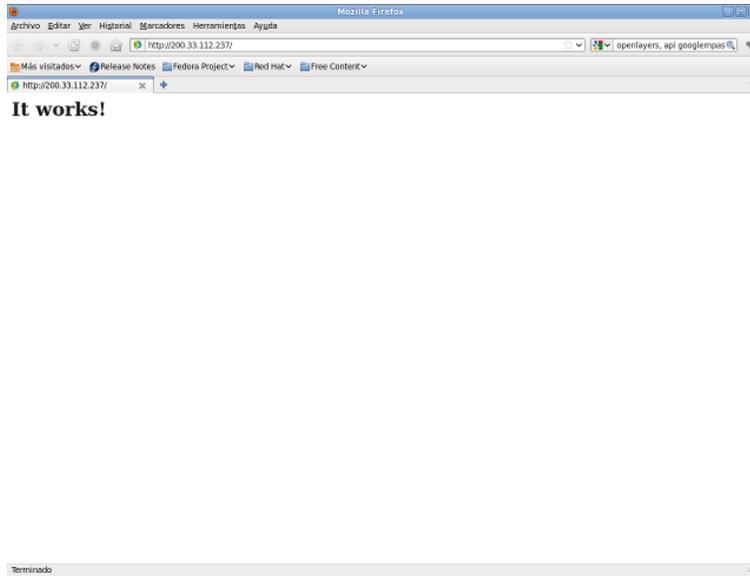


Figura 32. Prueba de funcionamiento, Apache.

Una vez instalado y configurado el Servidor Apache, se realizó la instalación y configuración del Sistema Manejador de Bases de Datos PostgreSQL.

Instalación de PostgreSQL

Para compilar PostgreSQL es necesario instalar algunos programas y librerías; con sus respectivas herramientas de desarrollo. Estas dependencias están en función de la distribución y versión de Linux, además de la forma de instalación del Sistema Operativo. Es por ello, que para la instalación de PostgreSQL en nuestro sistema, fue necesario instalar los siguientes programas y librerías.

1. libxml2 y libxml2-devel
2. readline y readline-devel
3. zlib y zlib-devel
4. python y python-devel
5. ncurses y ncurses-devel

La instalación de estas dependencias, se realizó a través de los paquetes disponibles de Linux/CentOS.

```
[root@mapas ~]# yum install libxml2 libxml2-devel
[root@mapas ~]# yum install readline readline-devel
[root@mapas ~]# yum install zlib zlib-devel
[root@mapas ~]# yum install python python-devel
```

Se descargó del sitio <http://www.apache.org/> el archivo `httpd-2.2.22.tar.gz` y mediante el comando `tar`, se procedió a su descompresión.

```
[root@mapas fuentes]# tar -zxvf postgresql-9.1.3.tar.gz
```

Dentro del directorio `postgresql-9.1.3`, creamos un archivo de nombre `compila_postgresql.sh` (**Script 2**) con el editor `vi`, donde se definieron parámetros de instalación.

```
LDFLAGS=-lstdc++ ./configure \
--prefix=/usr/local/pgsql/data \
--with-perl \
--with-python \
--with-krb5 \
--with-openssl
```

Script 2. Contenido de `compila_postgresql.sh` para PostgreSQL.

Se cambiaron sus atributos para hacerlo ejecutable:

```
[root@mapas postgresql-9.1.3]# chmod a+x compila_postgresql.sh
```

Para iniciar el proceso de instalación, ejecutamos el script “`compila_postgresql.sh`”.

```
[root@mapas postgresql-9.1.3]# ./compila_postgresql.sh
```

Posteriormente, se utilizó el comando “`gmake`” para generar los archivos ejecutables de PostgreSQL y “`gmake install`” para su instalación.

```
[root@mapas postgresql-9.1.3]# gmake & gmake install
```

Finalizada la instalación, se creó un usuario administrador de las bases de datos, asignándole el control sobre el directorio donde residirán las bases. Para ello se realizó lo siguiente:

1. Creación del usuario postgres

```
[root@mapas postgresql-9.1.3]# adduser postgres
```

2. Posteriormente se creó el directorio data, el cual almacenará las bases.

```
[root@mapas postgresql-9.1.3]# mkdir /usr/local/pgsql/data
```

3. Se asignó como propietario del directorio data al usuario postgres.

```
[root@mapas postgresql-9.1.3]# chown postgres /usr/local/pgsql/data/
```

4. Como usuario postgres, iniciamos el cluster de base de datos. Esto inicializa el directorio previamente creado, copiando los archivos y directorios requeridos por el Servidor para alojar las bases de datos.

```
[postgres@mapas ~]# /usr/local/pgsql/bin/initdb -D /usr/local/pgsql/data
```

5. Iniciamos el PostgreSQL en background y redireccionamos la salida estándar y de errores al archivo logfile.

```
[postgres@mapas ~]# /usr/local/pgsql/bin/postgres -D /usr/local/pgsql/data >logfile 2>&1 &
```

6. Finalmente, se creó la base de datos.

```
[postgres@mapas ~]# /usr/local/pgsql/bin/createdb pruebaGIS
```

Con lo anterior, quedó instalado el PostgreSQL, el espacio de almacenamiento inicializado y la primera base de datos “pruebaGIS” creada. Después de la instalación de PostgreSQL, se procedió a la instalación de PostGIS.

Instalación PostGIS

Para tener una perfecta instalación de PostGIS primero fu necesario instalar unos programas y librerías que resuelven problemas de dependencia. Para ello fue necesario instalar las siguientes librerías.

- Proj4. Esta librería de proyecciones cartográficas es empleada para poder realizar reproyecciones, cambios de Datum o Sistema de Coordenada dentro del motor de datos. Por lo anterior se descargó del sitio <http://trac.osgeo.org/proj/> el archivo `proj-4.8.0.tar.gz` y mediante el comando `tar`, se procedió a su descompresión.

```
[root@mapas fuentes]# tar -zxvf proj-4.8.0.tar.gz
```

Dentro del directorio `proj-4.8.0`, invocamos el script de configuración “`configure`”.

```
[root@mapas proj-4.8.0]# ./configure
```

Posteriormente, se utilizó el comando “`make`” para generar los archivos ejecutables de esta librería y “`make install`” para su instalación.

```
[root@mapas proj-4.8.0]# make & make Install
```

Al igual que la librería Proj4 también existe la librería GEOS, la cual es indispensable para la instalación de PostGIS, es por ello que procedimos a su instalación.

- GEOS. Esta librería es utilizada para realizar pruebas entre geometrías tales como: `touches()`, `contains()`, `intersects()` y operaciones de geoprocésamiento tales como: `buffer()`, `geomunion()`, `difference()`, etc., dentro del motor de base de datos. Por lo anterior, se descargó del sitio <http://trac.osgeo.org/geos/> el archivo `geos-3.3.2.tar.gz`, y mediante el comando `tar`, se procedió a su descompresión.

```
[root@mapas fuentes]# tar -zxvf geos-3.3.2.tar.gz
```

Como en el caso anterior, dentro del directorio geos-3.3.2, invocamos el script de configuración “configure”.

```
[root@mapas geos-3.3.2]# ./configure
```

De la misma manera, se utilizó el comando “make” para crear los archivos ejecutables de esta librería y “make install” para su instalación.

```
[root@mapas geos-3.3.2]# make & make Install
```

Concluidos los pasos antes descritos, se procedió a la instalación de PostGIS, para ello se descargó del sitio <http://www.refractions.net> el archivo postgis-1.5.3.tar.gz, y mediante el comando tar, se procedió a su descompresión.

```
[root@mapas fuentes]# tar -zxvf postgis-1.5.3.tar.gz
```

Como en el casos anterior, dentro del directorio postgis-1.5.3, se creó un archivo de nombre compila_postgis.sh (**Script 3**) con el editor vi, donde se establecieron los parámetros de instalación.

```
./configure \  
--with-pgconfig=/usr/local/pgsql/bin/pg_config \  
--with-geosconfig=/usr/local/bin/geos-config \  
--with-gui
```

Script 3. Contenido de compila_postgis.sh para PostGIS.

Se cambiaron sus atributos para hacerlo ejecutable:

```
[root@mapas postgis-1.5.3]# chmod a+x compila_postgis.sh
```

Para iniciar el proceso de instalación, ejecutamos el script “compila_postgis.sh”.

```
[root@mapas postgis-1.5.3]# ./compila_postgis.sh
```

Posteriormente, se utilizó el comando “make” para generar los archivos ejecutables de PostGIS y “make install” para su instalación.

```
[root@mapas postgis-1.5.3]# make & make install
```

Finalizado este proceso, observamos que se crearon dos archivos llamados DUMPERS⁵ (shp2pgsql y pgsq2shp), PostGIS utiliza estos archivos para cargar y descargar archivos de tipo ShapeFile de ESRI hacia y desde una base de datos [15], los cuales se encuentran en el directorio de “/usr/local/pgsql/bin”.

Por lo anterior, fue indispensable agregar este directorio al PATH del Sistema Operativo, para poder así ejecutar estos archivos desde cualquier directorio. Para ello fue necesario editar el archivo “/etc/profile” (**Script 4**), añadiendo las siguientes líneas.

```
[root@mapas ~]# vi /etc/profile
```

```
export PGSQL_HOME=/usr/local/pgsql
export PATH=$PATH:$PGSQL_HOME/bin
export LD_LIBRARY_PATH=/lib:/usr/lib:$PGSQL_HOME/lib:/usr/local/lib
```

Script 4. Parte del contenido del archivo profile.

En seguida, se dio de alta en el Sistema Operativo las librerías generadas por Proj4, GEOS y PostGIS (**Script 5**). Esto se realizó ejecutando la siguiente instrucción:

```
[root@mapas ~]# echo /usr/local/lib >> /etc/ld.so.conf
[root@mapas ~]# echo /usr/local/pgsql/lib >> /etc/ld.so.conf
```

```
Include ld.so.conf.d/*.conf
/lib
/usr/lib
/usr/local/lib
/usr/local/pgsql/lib
```

Script 5. Contenido del archivo ld.so.conf.

5 Convertidores en batch.

Por último se actualizó las librerías del sistema a través del comando “ldconfig”.

```
[root@mapas ~]# ldconfig
```

Una vez concluidos los pasos anteriores, fue necesario crear una base de datos habilitada espacialmente “postgis_template”. Para ello, como usuario postgres ejecutamos lo siguiente:

```
[postgres@mapas ~]# createdb postgis_template
```

A dicha base, se le cargó el lenguaje de procedimientos almacenados “plpgsql”, a través de la siguiente instrucción.

```
[postgres@mapas ~]# createlang plpgsql postgis_template
```

En seguida se localizaron dos archivos “postgis.sql y spatial_ref_sys.sql”, los cuales se ejecutaron para cargar el objeto PostGIS, las definiciones de función y los identificadores de los distintos sistemas de coordenadas. Lo anterior, se realizó a través de las siguientes instrucciones.

```
[postgres@mapas ~]# psql -d postgis_template -f /usr/local/pgsql/share/contrib/postgis-1.5/postgis.sql
```

```
[postgres@mapas ~]# psql -d postgis_template -f /usr/local/pgsql/share/contrib/postgis-1.5/spatial_ref_sys.sql
```

Por último, se comprobó la instalación de PostGIS, para ello ejecutamos la instrucción siguiente.

```
[postgres@mapas ~]# psql -d postgis_template -c “SELECT posn(,)”
```

Obteniendo como resultado el siguiente mensaje.

```
postgis_full_version
```

```
-----  
POSTGIS="1.5.3" GEOS="3.3.2-CAPI-1.7.2" PROJ="Rel. 4.8.0, 6 March 2012" LIBXML="2.6.26" USE_STATS
```

El cual indica, que el sistema base tiene el soporte para habilitar la funcionalidad espacial a PostgreSQL.

Para garantizar una mejor administración y generar un entorno más seguro, fue necesario la creación de roles⁶ de inicio de sesión para proyectos SIG. Para tal efecto, a través del usuario administrador de bases de datos accedimos a la base de datos postgres e ingresamos lo siguiente.

```
[postgres@mapas ~]# psql postgres
Psql (9.1.3)
Type "help" for help
postgres=# CREATE ROLE gisgroup NOSUPERUSER NOINHERIT CREATEDB NOCREATE ROLE;
```

El objetivo de la instrucción anterior, fue crear un rol “gisgroup”; mediante el cual se pudieran crear objetos en la base de datos, sin ser administrador, sin heredar permisos de los grupos y restringir la creación roles en el cluster de bases de datos.

Después, se creó un rol “gis” con contraseña sin heredar permisos, para ello fue necesario ejecutar lo siguiente.

```
postgres=# CREATE ROLE gis LOGIN PASSWORD '*****' NOINHERIT;
```

Al rol “gis”, se le asignaron privilegios del rol “gisgroup” ejecutando la siguiente instrucción.

```
postgres=# GRANT gisgroup TO gis;
```

A continuación, se otorgaron permisos al rol “gis” para las tablas (geometry_columns y spatial_ref_sys” de la base “postgis_template” (**Figura 33**), mediante los siguientes pasos.

⁶ Los usuarios en bases de datos individuales se llaman roles de inicio de sesión.

```
[postgres@mapas ~]# psql -d postgis_template
Psql (9.1.3)
Type "help" for help
postgis_template=# ALTER TABLE geometry_columns OWNER TO gis;
postgis_template=# ALTER TABLE spatial_ref_sys OWNER TO gis;
postgis_template=# CREATE SCHEMA gis_schema AUTHORIZATION gis;
```

```
postgis_template=# \dt
List of relations
Schema | Name | Type | Owner
-----+-----+-----+-----
public | geometry_columns | table | gis
public | spatial_ref_sys | table | gis
(2 rows)
```

Figura 33. Propietario de las tablas, postgis_template.

Finalmente, concluidos los pasos anteriores se procedió a crear la base de datos “gisdb” haciendo uso de postgis_template.

```
[postgres@mapas ~]# createdb -E UNICODE -T postgis_template -O gis gisdb
```

Una vez realizada la instalación, se procedió a verificar el funcionamiento del sistema, se realizó el procedimiento de visualización de la información a partir de consultas realizadas desde el sistema manejador de bases de datos (PostgreSQL) y su extensión espacial (PostGIS), de la siguiente manera:

- Se verificó la existencia de la base de datos con características espaciales llamada “gisdb”;
- Se corroboró que la base de datos generada tuviera por omisión dos tablas “geometry_columns” (**Figura 34**) y “spatial_ref_sys” (**Figura 35**), de los aspectos geométricos y de sistema de referencia espacial, respectivamente;

```

postgres@mapas:~
Archivo Editar Ver Buscar Terminal Ayuda
gisdb=#
gisdb=#
gisdb=#
gisdb=#
gisdb=# \d geometry_columns
          Table "public.geometry_columns"
  Column          |          Type          | Modifiers
-----+-----+-----
 f_table_catalog | character varying(256) | not null
 f_table_schema  | character varying(256) | not null
 f_table_name    | character varying(256) | not null
 f_geometry_column | character varying(256) | not null
 coord_dimension | integer                 | not null
 srid             | integer                 | not null
 type            | character varying(30)  | not null
Indexes:
 "geometry_columns_pk" PRIMARY KEY, btree (f_table_catalog, f_table_schema, f_table_name, f_geometry_column)
gisdb=#

```

Figura 34. Tabla geometry_columns.

```

postgres@mapas:~
Archivo Editar Ver Buscar Terminal Ayuda
gisdb=# \d spatial_ref_sys
          Table "public.spatial_ref_sys"
  Column          |          Type          | Modifiers
-----+-----+-----
 srid             | integer                 | not null
 auth_name        | character varying(256) |
 auth_srid        | integer                 |
 srtext           | character varying(2048) |
 proj4text        | character varying(2048) |
Indexes:
 "spatial_ref_sys_pkey" PRIMARY KEY, btree (srid)
gisdb=#

```

Figura 35. Tabla spatial_ref_sys.

- Se cargó la información generada a partir de los archivos *.shp;
- Por último, se comprobó por medio de consultas SQL que la información almacenada y la base de datos cumplieran con todos los requisitos y funciones para su utilización en el visor de mapas cartográfico.

El siguiente paso fue, la instalación del lenguaje de programación PHP para desarrollar sitios Web con contenido dinámico, además de dotar a este lenguaje de la capacidad de acceder, administrar y representar información cartográfica a través de la API de MapServer (PHP/MapScript).

Instalación de PHP

Como en casos anteriores, el primer paso fue resolver las dependencias que requiere PHP para su instalación. Es por ello, que se instalaron los siguientes programas y librerías:

1. curl y curl-devel
2. freetype y freetype-devel
3. libtiff y libtiff-devel
4. libpng y libpng-devel

De igual forma, la instalación de estas dependencias, se realizó a través de los paquetes disponibles de Linux/CentOS.

```
[root@mapas ~]# yum install curl curl-devel
[root@mapas ~]# yum install freetype freetype-devel
[root@mapas ~]# yum install libtiff libtiff-devel
[root@mapas ~]# yum install libpng libpng-devel
```

Resueltas las dependencias, se descargó del sitio <http://www.php.net> el archivo `php-5.3.21.tar.gz`, y mediante el comando `tar`, se procedió a su descompresión.

```
[root@mapas fuentes]# tar -zxvf php-5.3.21.tar.gz
```

Dentro del directorio `php-5.3.21` se creó un archivo de nombre `compila_php.sh` (**Script 6**) con el editor `vi`, donde se establecieron los parámetros de instalación.

```
./configure \
--prefix=/usr/local/apache2/php \
--with-apxs2=/usr/local/apache2/bin/apxs \
--with-config-file-path=/usr/local/apache2/php \
--disable-cgi \
--with-openssl \
--with-kerberos \
--with-zlib \
--with-curl \
--with-freetype-dir=/usr \
--with-gd \
--with-pgsql \
--with-gettext \
--with-regex=system
```

Script 6. Contenido de `compila_php.sh` para PHP.

Una vez creado el archivo, se cambiaron sus atributos para hacerlo ejecutable.

```
[root@mapas php-5.3.21]# chmod a+x compila.sh
```

Para iniciar el proceso de instalación, ejecutamos el script “compila_php.sh”.

```
[root@mapas php-5.3.21]# ./compila_php.sh
```

Posteriormente, se utilizó el comando “make” para generar los archivos ejecutables de PHP y “make install” para su instalación.

```
[root@mapas php-5.3.21]# make & make install
```

Cabe mencionar que este proceso de instalación no generó el archivo “php.ini”, que es indispensable para el funcionamiento de PHP. Básicamente este archivo contiene los parámetros de configuración las cuales determinan el comportamiento del interprete PHP. Por lo anterior, fue necesario copiar y renombrar el archivo de nombre “php.ini-development” de la distribución al directorio “/usr/local/apache2/php”.

```
[root@mapas php-5.3.21]# cp php.ini-development /usr/local/apache2/php/php.ini
```

Para aumentar el tiempo de procesamiento como los recursos de memoria, fue necesario definir en el archivo “php.ini” los parámetros siguientes.

```
max_execution_time = 150  
max_input_time = 300  
memory_limit = 256M
```

Al mismo tiempo, para poder desplegar mensajes de error, se estableció el siguiente parámetro.

```
error_reporting = E_ALL & ~E_NOTICE
```

Además, de definir el directorio para las extensiones, para ello se estableció la siguiente ruta.

```
extension_dir = "/usr/local/apache2/php/ext"
```

Para comprobar la correcta instalación de PHP, se corroboró que en el archivo de configuración "httpd.conf" de Apache se encontrara cargado el módulo de PHP que se generó al ejecutar el comando "make install":

```
LoadModule php5_module modules/libphp5.so
```

Por otra parte, para que el Servidor Apache pudiera interpretar automáticamente código PHP, fue necesario añadir la siguiente línea con extensión ".php", tal como se muestra a continuación:

```
<IfModule dir_module>
  DirectoryIndex index.html index.php index.htm
</IfModule>
```

Además de definir las extensiones de php, añadiendo las siguientes líneas:

```
AddType application/x-httpd-php .php .phtml
AddType application/x-httpd-php-source .phps
```

Para hacer efectivos los cambios en los archivos "php.ini" y "httpd.conf", se reinició el servicio de Apache.

Para verificar el funcionamiento de PHP se creó un archivo de nombre "info.php" (**Script 7**) dentro del directorio de Apache "/usr/local/apache2/htdocs".

```
<?php
  Date_default_timezone("America/Mexico_City");
  phpinfo();
?>
```

Script 7. Contenido de info.php.

Finalmente, para comprobar su correcto funcionamiento, nuevamente haciendo uso de un navegador Web, se accedió al siguiente URL (<http://localhost/info.php>); teniendo como resultado información detallada del Servidor Apache y de PHP (**Figura 36**).

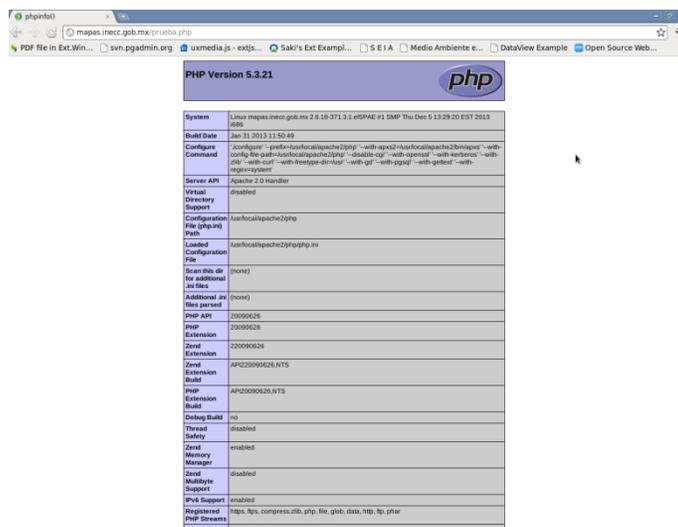


Figura 36. Prueba de funcionamiento de PHP.

Instalación de MapServer

Para instalar MapServer, es necesario compilar el archivo fuente del programa, así como las librerías necesarias para su funcionamiento. Por lo anterior, fue necesario instalar:

1. Librería de gráficos GD.
2. Librería para el manejo de datos vectoriales, raster (GDAL/OGR).
3. Librería para la transferencia de información a través de la Web (CURL).
4. Librería para el manejo de imágenes GIF (giflib).
5. PostGIS para tener un servidor con este formato espacial.

Cabe mencionar que tanto la librería CURL como PostGIS, ya se encuentran instalados en el sistema, por lo que solo se abordara la instalación de las librerías GD, GDAL/OGR y giflib.

De igual forma que en casos anteriores, la instalación de GD y giflib, se realizó a través de los paquetes disponibles de Linux/CentOS.

```
[root@mapas ~]# yum install gd gd-devel
[root@mapas ~]# yum install giflib giflib-devel
```

Para la instalación de GDAL/OGR, fue necesario descargar el archivo gdal-1.9.0.tar.gz del sitio <http://download.osgeo.org/gdal/>, y mediante el comando tar, se procedió a su descompresión.

```
[root@mapas fuentes]# tar -zxvf gdal-1.9.0.tar.gz
```

Dentro del directorio creado gdal-1.9.0, se creó un archivo de nombre compila_dgal.sh (**Script 8**) con el editor vi, donde establecieron los parámetros de instalación.

```
./configure \
--with-png \
--with-libtiff \
--with-jpeg \
--with-gif \
--with-pg=/usr/local/pgsql/bin/pg_config \
--with-geos=/usr/local/bin/geos-config \
--with-static-proj4=/usr/local/lib
```

Script 8. Contenido de compila_dgal.sh para GDAL/OGR.

Una vez creado el archivo, se cambiaron sus atributos para hacerlo ejecutable.

```
[root@mapas gdal-1.9.0]# chmod a+x compila_dgal.sh
```

Para iniciar el proceso de instalación, ejecutamos el script “compila_dgal.sh”.

```
[root@mapas gdal-1.9.0]# ./compila_dgal.sh
```

Posteriormente, se utilizó el comando “make” para generar los archivos ejecutables de esta librería y “make install” para su instalación.

```
[root@mapas gdal-1.9.0]# make & make install
```

Concluido este proceso, se tuvo la necesidad de actualizar las librerías del sistema, para ello se empleó el comando “ldconfig”.

```
[root@mapas ~]# ldconfig
```

Una vez instalados los requisitos previos “dependencias”, se procedió a la instalación de MapServer, para ello fue necesario descargar el archivo mapserver-6.0.2.tar.gz del sitio <http://mapserver.org/download.html>, y mediante el comando tar, se procedió a su descompresión.

```
[root@mapas fuentes]# tar -zxvf mapserver-6.0.2.tar.gz
```

Dentro del directorio php-5.3.21 se creó un archivo de nombre compila_mapserver.sh (**Script 9**) con el editor vi, donde definieron los parámetros de instalación.

```
./configure \  
--with-gdal=/usr/local/bin/gdal-config \  
--with-ogr=/usr/local/bin/gdal-config \  
--with-geos=/usr/local/bin/geos-config \  
--with-httpd=/usr/local/apache2/bin/httpd \  
--with-php=/usr/local/apache2/php \  
--with-curl-config=/usr/bin/curl-config \  
--with-proj=/usr/local \  
--with-postgis=/usr/local/postgis/bin/pg_config \  
--with-xml2-config=/usr/bin/xml2-config \  
--with-libiconv \  
--with-tiff \  
--with-gd \  
--with-jpeg \  
--with-zlib \  
--with-png=/usr \  
--with-freetype \  
--with-threads \  
--with-sos \  
--with-kml \  
--with-wcs \  
--with-wfs \  
--with-wmsclient \  
--with-wfsclient \  
--enable-debug
```

Script 9. Contenido de compila_mapserver.sh para MapServer.

Se cambiaron sus atributos para hacerlo ejecutable.

```
[root@mapas mapserver-6.0.2]# chmod a+x compila_mapserver.sh
```

Para iniciar el proceso de instalación, ejecutamos el script “compila_mapserver.sh”.

```
[root@mapas mapserver-6.0.2]#./compila_mapserver.sh
```

Posteriormente, se utilizó el comando “make” para generar los archivos ejecutables de MapServer.

```
[root@mapas php-5.3.21]# make
```

Para finalizar la instalación, no fue necesario hacer uso del comando make install, lo anterior debido a que los archivos ejecutables y librerías de MapServer, se copiaron manualmente en el directorio cgi-gin del Servidor Apache.

```
[root@mapas mapserver-6.0.2]# cp legend /usr/local/apache2/cgi-bin/  
[root@mapas mapserver-6.0.2]# cp mapserv /usr/local/apache2/cgi-bin/  
[root@mapas mapserver-6.0.2]# cp scalebar /usr/local/apache2/cgi-bin/  
[root@mapas mapserver-6.0.2]# cp shp2img /usr/local/apache2/cgi-bin/  
[root@mapas mapserver-6.0.2]# cp shptree /usr/local/apache2/cgi-bin/  
[root@mapas mapserver-6.0.2]# cp shptreetst /usr/local/apache2/cgi-bin/  
[root@mapas mapserver-6.0.2]# cp shptreevis /usr/local/apache2/cgi-bin/  
[root@mapas mapserver-6.0.2]# cp sortshp /usr/local/apache2/cgi-bin/  
[root@mapas mapserver-6.0.2]# cp tile4ms /usr/local/apache2/cgi-bin/
```

Con lo anterior quedó instalado y configurado MapServer con los parámetros de instalación que se definieron en el script 9.

Para comprobar que la instalación de MapServer (**Figura 37**), se ejecutó lo siguiente.

```
[root@mapas cgi-bin]# ./mapserv -v
```

```
[root@mapas cgi-bin]# ./mapserv -v  
MapServer version 6.0.2 OUTPUT=GIF OUTPUT=PNG OUTPUT=JPEG OUTPUT=KML SUPPORTS=PROJ SUPPORTS=AGG SUPPORTS=FREETYPE SUPPORTS=ICONV SUPPORTS=WMS  
SERVER SUPPORTS=WMS_CLIENT SUPPORTS=WFS_SERVER SUPPORTS=WFS_CLIENT SUPPORTS=WCS_SERVER SUPPORTS=50S_SERVER SUPPORTS=THREADS SUPPORTS=GEOS IN  
PUT=POSTGIS INPUT=0GR INPUT=GDAL INPUT=SHAPEFILE
```

Figura 37. Prueba de funcionamiento MapServer.

Por último, un aspecto importante a destacar, es que el sistema soporta MapScript para PHP. Esta extensión es una librería que extiende el lenguaje de programación PHP con las funciones necesarias para ingresar a la interfaz de programación de MapServer dando a PHP la capacidad de acceder, administrar y representar información cartográfica [13]. La incorporación de esta extensión, se realizó a través del parámetro “--with-php=/usr/local/apache2/php” definido en el archivo “compila_mapserver.sh”. Este parámetro le indicó a MapServer que PHP se encuentra en el sistema; y por ende el proceso de compilación generó el archivo “php_mapscript.so”, el cual se copió al directorio “/usr/local/apache2/php/ext”.

```
[root@mapas mapserver-6.0.2]# cp mapscript/php/php_mapscript.so /usr/local/apache2/php/ext/
```

Finalmente, para activar esta extensión, fue necesario definir el parámetro **extension="php_mapscript.so"**, dentro del archivo php.ini.

Finalmente, para comprobar su correcto funcionamiento, se reinició el Servidor Apache y mediante un navegador de Web se accedió al siguiente URL (<http://localhost/info.php>), obteniendo como resultado información detallada de la configuración de MapServer, así como la versión del PHP/MapScript (**Figura 38**).

MapScript

MapServer Version	MapServer version 6.0.2 OUTPUT=GIF OUTPUT=PNG OUTPUT=JPEG OUTPUT=KML SUPPORTS=PROJ SUPPORTS=AGG SUPPORTS=FREETYPE SUPPORTS=ICONV SUPPORTS=WMS_SERVER SUPPORTS=WMS_CLIENT SUPPORTS=WFS_SERVER SUPPORTS=WFS_CLIENT SUPPORTS=WCS_SERVER SUPPORTS=SOS_SERVER SUPPORTS=THREADS SUPPORTS=GEOS INPUT=POSTGIS INPUT=OGR INPUT=GDAL INPUT=SHAPEFILE
PHP MapScript Version	(\$Revision: 12740 \$ \$Date: 2011-11-02 10:47:24 -0700 (Wed, 02 Nov 2011) \$)

Figura 38. Prueba de funcionamiento PHP/MapScript.

Quedando así, conformada la plataforma SIG para el “Sistema de Consulta de Mapas Digitales sobre Planificación Territorial y Biodiversidad”.

Finalmente, cabe destacar que la selección de todas estas herramientas, permiten conservar de manera íntegra la información inherente a la cartografía, de manera que las imágenes cartográficas no pierden sus propiedades de origen como la escala, semiología, entre otras.

CAPÍTULO 3.- RESULTADOS

Como resultado se obtuvo un sitio Web (**Figura 39**) de acceso público que se puede consultar a través del URL (<http://mapas.inecc.gob.mx>), dirección donde se aloja el sistema a través del portal del INECC. En este espacio se encuentra el visor de mapas de forma gráfica.

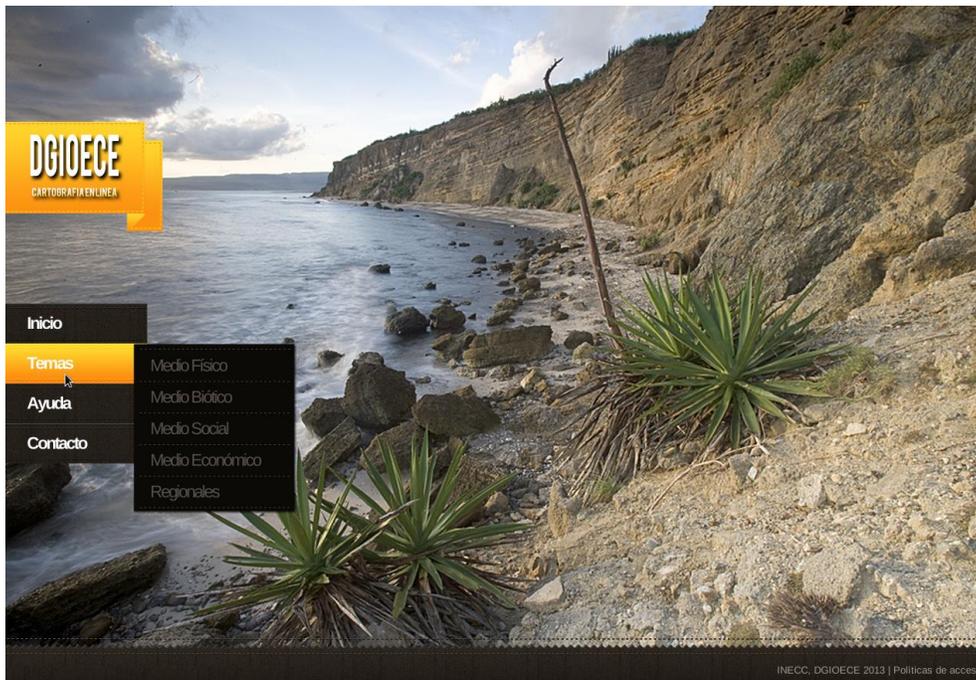


Figura 39. Sitio Web, Cartografía en Línea.

El diseño final de la interfaz gráfica está compuesto por una vista general que se compone a su vez en varias secciones o paneles.

En la sección superior se encuentra una barra de menú. Este panel muestra la información que puede ser consultada.

- Mapa: Permite examinar información geográfica.
- Metadato: Permite examinar la documentación del tema.
- Ayuda: Muestra información básica del visor de mapas.
- Contacto: Muestra información del administrador del sitio Web.

La sección central es donde se dibujan y visualizan las capas de información activas con sus coordenadas extremas, es decir, permite explorar el tema desde una perspectiva geográfica. En este panel se encuentran disponibles:

- Herramientas de navegación (**Figura 40**) en el mapa (ir a la extensión de una escala determinada, desplazar, acercar, alejar, medir distancias, medir áreas, mapa de referencia, herramientas de dibujo, ayuda).

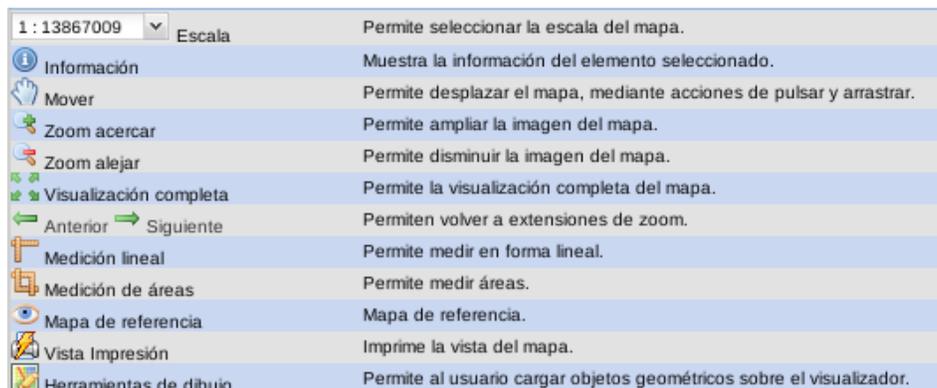


Figura 40. Herramientas de navegación.

- Herramientas que funcionan en el contexto de la capa activa (consulta de atributos) (**Figura 41**). El resultado de la consulta puede ser descargado en formato XLS como CSV.



Figura 41. Herramientas para consulta de atributos.

En la sección izquierda se encuentran las capas de información las cuales el usuario puede activar o desactivar mediante checkboxes, la leyenda, además de información complementaria. En este panel (**Figura 42**) se encuentran:

- **Capas Activas** (panel): Permite examinar las capas del tema. Las capas tienen tres propiedades comunes: visible, activo y superposición.
- **Leyenda** (panel): Permite visualizar el significado de los símbolos empleados en cada capa.
- **Información** (panel): Muestra información básica del tema.

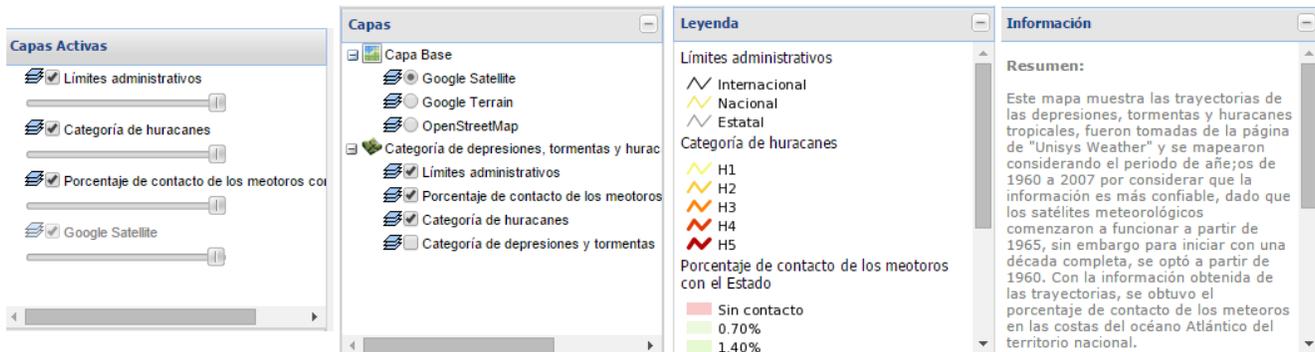


Figura 42. Paneles, visor.

En la sección inferior (**Figura 43**) se muestra información contextual sobre las acciones que se realizan en el mapa: código EPSG y, coordenadas de la posición del ratón sobre el mapa.



Figura 43. Coordenadas.

Entre las funcionalidades más sobresalientes del visor se encuentra la herramienta de dibujo (**Figura 44**), que permite al usuario entre otras cosas dibujar objetos sobre el mapa; y estos a su vez pueden ser descargados en formatos (GML, KML, ShapeFile).



Figura 44. Herramientas de dibujo.

Para finalizar, resaltar el resultado altamente satisfactorio obtenido en el proyecto “Sistema de Consulta de Mapas Digitales sobre Planificación Territorial y Biodiversidad”, utilizando herramientas libres y de código abierto, además de la reutilización de componentes para visores de mapas Web, ponen en evidencia la alta calidad tecnológica de las tecnologías libres y de código abierto.

A la calidad tecnológica habría que sumar otros factores de gran interés como la inexistencia de costos de licenciamiento, la no dependencia tecnológica, así como la intervención directa en el diseño del sistema.

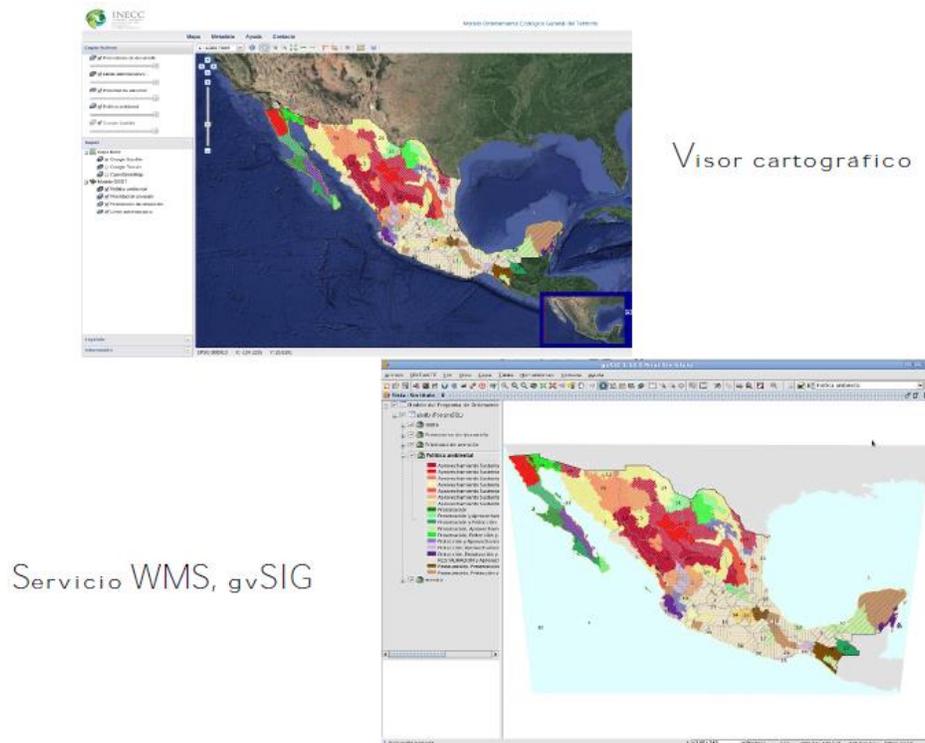


Figura 45. Acceso al acervo de información geográfica desde el SIG de escritorio gvSIG y navegador de Web.

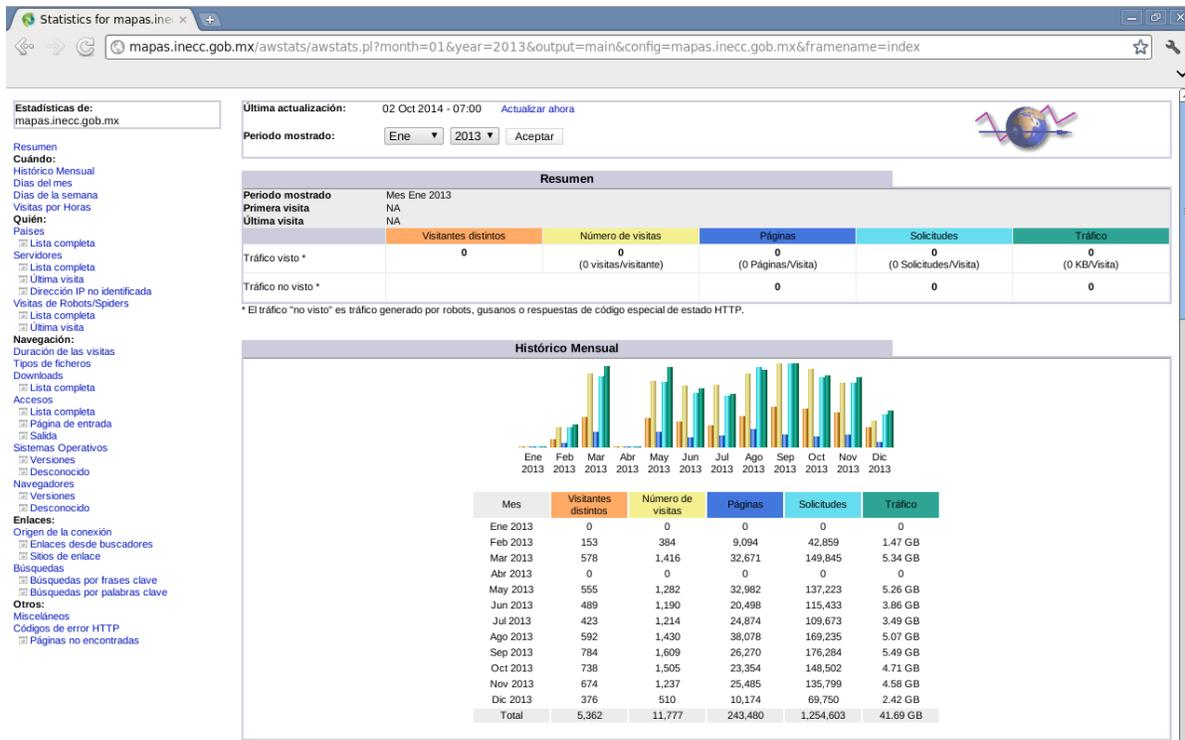


Figura 46. Estadísticas de acceso al sistema (Febrero a Diciembre, 2013). Fuente: Departamento de Sistemas Interactivos para el Análisis de la Vulnerabilidad, Riesgo y Adaptación (INECC, 2014).

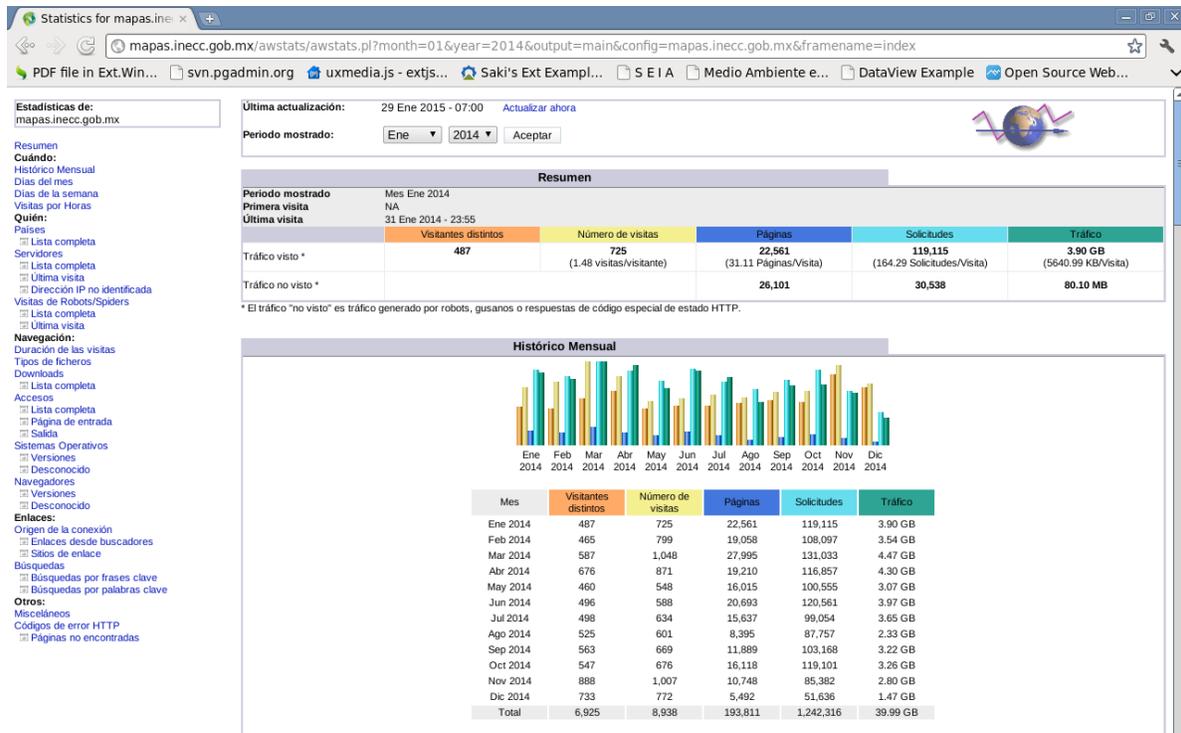


Figura 47. Estadísticas de acceso al sistema (Enero a Diciembre, 2014). Fuente: Departamento de Sistemas Interactivos para el Análisis de la Vulnerabilidad, Riesgo y Adaptación (INECC, 2014).

CAPÍTULO 4.- CONCLUSIONES Y TRABAJO FUTURO

Los SIG han demostrado ser una herramienta muy valiosa para la administración, consulta, visualización y análisis de datos geográficos, y por ende para su aplicación en temas relacionados con el territorio. Además el avance en las tecnologías de la información y de la comunicación han logrado que estos sistemas mejoren en cuanto a su capacidad de distribución o publicación masiva de datos a través de Internet, lo que ha impulsado en las últimas décadas el desarrollo de proyectos en temas relacionados con la Geomática.

Para aprovechar las ventajas que ofrece Internet es necesario contar con herramientas que faciliten la publicación de información geográfica en Internet de una forma sencilla. El producto presentado es este informe de trabajo “Sistema de Consulta de Mapas Digitales sobre Planificación Territorial y Biodiversidad”, es un Servidor de Mapas para Internet, capaz no sólo de publicar mapas, sino de operar en conjunto con otros servidores que sigan el estándar de OGC de forma que se puede convertir en una herramienta muy útil para la integración de sistemas de información geográfica en el contexto de una administración pública con departamentos y oficinas que trabajan con sistemas heterogéneos, o que simplemente están interesados en dar acceso público a parte de su acervo cartográfico a través de Internet sin necesidad de realizar un gran esfuerzo para adaptar sus sistemas a la red.

La información geográfica ofrecida de forma interoperable y estandarizada a través de los servicios Web que siguen las especificaciones OGC, es un aspecto muy importante en las administraciones públicas, resultado clave la implementación de servicios Web que puedan satisfacer los deseos y necesidades de los ciudadanos que garanticen una óptima gestión por parte de las Instituciones.

Este sistema sigue la filosofía de la libre distribución de la información de una manera gratuita, se crea como el entorno natural de acceso e intercambio de información entre administraciones públicas e introduce la necesidad de promover el desarrollo de servicios de valor agregado que aprovechen las ventajas de disponer a través de Internet información geográfica de calidad, actualizada y conforme a normas y estándares internacionales.

La interfaz de usuario ha sido diseñada cuidando al máximo la interacción del usuario con el visor, facilitando de este modo la presentación de los datos, la cartografía y manteniendo una sencilla metodología de acceso a los mismos. Ya se piensan en nuevas funcionalidades que contribuirán al fortalecimiento de este sistema, tarea que resultará fácil al emplear software con acceso libre a su código.

La finalidad de este sistema, es que se convierta en una herramienta de ayuda para el entendimiento y generación de políticas públicas orientadas hacia un desarrollo sustentable en nuestro país, además de que este sistema sea considerado como un proyecto impulsor en el INECC que avance en el futuro, hacia la implementación de una IDE, en cuanto a las ventajas que representa la utilización de las tecnologías de la información y comunicación para integrar información geográfica por entidades gubernamentales y otras organizaciones, bajo especificaciones OGC que garanticen la interoperabilidad de la información.

El presente trabajo, en esta ocasión pretende ser un aporte más para futuros profesionales e interesados en desarrollar aplicaciones Web Mapping utilizando software libre y de código abierto.

Como líneas futuras se proponen las siguientes mejoras en el sistema:

1. Incorporar servicios OGC como WFS y CWS.
2. Permitir al usuario cargar capas de información en formatos (GML, GeoJSON, GPX, KML, CSV (con X,Y), ShapeFile).
3. Creación de un módulo de impresión.
4. Cabe mencionar que el visor se encuentra en una fase de actualización, se incorporará más información particularmente en el tema de adaptación al cambio climático.

REFERENCIAS

- [1] Goodchild, M. F. (2014). **What is Geographic Information Science?** [en línea]. Disponible: <<http://www.ncgia.ucsb.edu/giscc/units/u002/u002.html>> [Consulta: Julio de 2014].
- [2] Arellano, Javier, Betts Sandra Yazmín, Ramírez Oliva (Editores) (2011). **Gestión de Datos Lidar a través de SIG y Base de Datos Geoespacial** [en línea]. Disponible: <http://www.inegi.org.mx/eventos/2011/conf_iber0/doc/ET1_25_ARELLANO.pdf> [Consulta: Agosto de 2014].
- [3] Reed, Carl: **OGC standards. Enabling the geospatial web** en Li, Songnian, Dragičević, Suzana y Veenendaal, Bert (Editores): **Advances in Web-Bases GIS, Mapping Services and Applications**, Taylor & Francis Group, London, 2011, pages 328-329, 331, 334, 336-337.
- [4] Global Spatial Data Infrastructure Association (GSDI) (2014). **Spatial Data Infrastructure Cookbook – Curren & Active (Wiki)** [en línea]. Disponible: <http://www.gsdi docs.org/GSDIWiki/index.php/Main_Page> [Consulta: Julio de 2014].
- [5] Federal Geographic Data Committee (FGDC) (2014). **National Spatial Data Infrastructure** [en línea]. Disponible: <<https://www.fgdc.gov/nsdi/nsdi.html>> [Consulta: Julio de 2014].
- [6] Infrastructure for Spatial Information in the European Community (INSPIRE) (2014). **Directive 2007/2/EC of the European Parliament and of the Council of 14 March 2007 establishing an Infrastructure for Spatial Information in the European Community (INSPIRE)** [en línea]. Disponible: <<http://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX:32007L0002>> [Consulta: Julio de 2014].

-
- [7] Capdevila, Joan. (2004, Agosto 1). **Infraestructura de Datos Espaciales (IDE). Definición y Desarrollo Actual de España**. Scripta Nova Revista Electrónica de Geografía y Ciencias Sociales [en línea], vol. VIII, número 170 (61). Disponible: <<http://www.ub.edu/geocrit/sn/sn-170-61.htm>> [Consulta: Julio de 2014].
- [8] International Association of Oil & Gas Producers (OGP) (2014). **EPSG home**. Disponible: <<http://www.epsg.org>> [Consulta: Julio de 2014].
- [9] United Nations Office for Outer Space Affairs (UNOOSA) (2014). **World Geodetic System 1984** [en línea]. Disponible: <http://www.oosa.unvienna.org/pdf/icg/2012/template/WGS_84.pdf> [Consulta: Agosto de 2014].
- [10] Spatial Reference (2014). **sr-org projection 7483 – epsg: 3857 – wgs84 web Mercator (auxiliary sphere)**. Disponible: <<http://spatialreference.org/ref/sr-org/7483/>> [Consulta: Agosto de 2014].
- [11] OSGeo Foundation (2014). **The Open Source Geospatial Foundation**. Disponible: <<http://www.osgeo.org>> [Consulta: Agosto de 2014].
- [12] Tyler Mitchell. 2005. **Web Mapping Illustrated. Using Open Source GIS Toolkits**. Ed. O'Reilly Media, Inc., pages 5-11, 211-215.
- [13] MapServer (2014). **Welcome to MapServer**. Disponible: <<http://mapserver.org>> [Consulta: Agosto de 2014].
- [14] Kropla, Bill. 2005. **Beginning MapServer. Open Source GIS Development**. Ed. Apress, pages XXV-XXVII, 31-35.
- [15] Obe Regina O., Hsu, Leo S. (2011). **PostGIS in Action**. Ed. Manning Publications Co., pages 5, 13-14, 18, 85.
-

-
- [16] Date, C. J. (2001). **Introducción a los Sistemas de Bases de Datos**. 7.^a ed., Ed. Prentice Hall, páginas 5-15, 43-54, 348-363.
- [17] The PostgreSQL Global Development Group (2014). **PostgreSQL**. Disponible: <<http://www.postgresql.org/>> [Consulta: Agosto de 2014].
- [18] PostGIS (2014). **Spatial and Geographic objects for PostgreSQL**. Disponible: <<http://postgis.net/>> [Consulta: Agosto de 2014].
- [19] Davis, Scott (2007). **Gis for Web Developers. Adding Where to Your Web Applications**. 1.^a ed., Ed. Pragmatic Bookshelf, pages 243-248.
- [20] DocForge (2014). **Web application framework** Disponible: <http://docforge.com/wiki/Web_application_framework> [Consulta: Agosto de 2014].
- [21] OpenLayers Documentation (2014). **What is OpenLayers?**. Disponible: <<http://docs.openlayers.org>> [Consulta: Agosto de 2014].
- [22] Santiago, Antonio (2013-2014). **The book of OpenLayers 3. Theory & Practice**. Ed. Leanpub, pages iii – vi.
- [23] Heron (2014). **Web Mapping Made Easy**. Disponible: <<http://heron-mc.org/>> [Consulta: Agosto de 2014].
- [24] gvSIG (2014). Portal **gvSIG**. Disponible: <<http://www.gvsig.org/web>> [Consulta: Agosto 2014].
- [25] Fu, Pinde, Sun, Jiulin (2011). **Web Gis. Principles and Applications**. Ed. ESRI Press, Redlands, California, pages 33-48, 70-74.

-
- [26] Ramsey, P. (2007, September 15). **The State of Open Source Gis** [en línea]. Disponible:
<<http://www.refractions.net/expertise/whitepapers/opensourcesurvey/survey-open-source-2007-12.pdf>> [Consulta: Septiembre de 2014].
- [27] OGC (2014). **Open Geospatial Consortium**. Disponible:
<<http://www.opengeospatial.org/>> [Consulta: Septiembre de 2014].
- [28] International Organization for Standardization (ISO) (2014). **ISO 19125-2:2004 Geographic information – Simple feature access – Part 2: SQL option** [en línea]. Disponible: <<https://www.iso.org/obp/ui/#iso:std:iso:19125:-2:ed-1:v1:en>> [Consulta: Septiembre de 2014].
- [29] OGC (2014), **Simple Features Specification For SQL** [en línea]. Disponible: <https://portal.opengeospatial.org/files/?artifact_id=829> [Consulta: Septiembre de 2014].
- [30] Monge de la Cruz, L., Torres Herrera, J., López Chico, L., Navarro Cota, C. (2010). **Análisis comparativo de servidores de mapa**. GeoFocus Revista Internacional de Ciencia y Tecnología de la Información Geográfica [en línea], Informes y comentarios, nº 10. [Consulta: Septiembre de 2014].
- [31] Vadala, Derek (2003). **Managing RAID on LINUX**. Ed. O' REILLY, pages 3-7, 11-15, 17-28.
- [32] Kabir, Mohammed J. (2002). Apache Server 2 Bible. Ed. Hungry Minds, Inc., pages 3-13.
- [33] Open Source Initiative (2014). **Open Source Licenses**. Disponible: <<http://opensource.org/>> [Consulta: Agosto 2014].
-

-
- [34] ArcGIS Resources (2014). **Georreferenciación y sistemas de coordenadas** [en línea]. Disponible: <<http://resources.arcgis.com/es/help/getting-started/articles/026n0000000s000000.htm>> [Consulta: Julio de 2014].
- [35] Spatial Information Clearinghouse (2014). **Spatial Information Clearinghouse – Data Topics**. Disponible: <<http://www.jmu.edu/cisr/research/sic/topics.htm>> [Consulta: Julio de 2014].
- [36] FGDC (2014). **Content Standard for Digital geospatial Metadata. FGDC-STD-001-1998** [en línea]. Disponible: <http://www.fgdc.gov/standards/projects/FGDC-standards-projects/metadata/base-metadata/v2_0698.pdf> [Consulta: Septiembre de 2014].
- [37] GeoJSON (2014). **The GeoJSON Format Specification** [en línea]. Disponible: <<http://geojson.org/geojson-spec.html>> [Consulta: Noviembre 2014].
- [38] Nieves, Eugenia. (2007, Julio – Septiembre). **Geomática: ciencia geográfica y tecnología aplicada al territorio**. @-scholarum Revista Académica Electrónica [en línea], Año 1, Número 1. Disponible: <<http://genesis.uag.mx/escholarum/vol1/eschol1.pdf>> [Consulta: Agosto de 2014].
- [39] Turner, Tania E. (2012). Software Libre y abierto: comunidades y redes de producción digital de bienes comunes. Tesis, Facultad de Ciencias Políticas y Sociales, UNAM, México. P. 39-45. [en línea] <<http://flosshub.org/sites/flosshub.org/files/Tesis.pdf>> [Consulta: Noviembre 2014].
- [40] Instituto Nacional de Estadística y Geografía (INEGI) 2014. **Sistema de Posicionamiento Global (GPS)** [en línea]. Disponible: <<http://www.inegi.org.mx/geo/contenidos/geodesia/gps.aspx?dv=c1>> [Consulta: Noviembre 2014].
-

-
- [41] Internet FAQ Archives (2014). **RFC 1945 - Hypertext Transfer Protocol - HTTP/1.0** [en línea]. Disponible: <<http://www.faqs.org/rfcs/rfc1945.html>> [Consulta: Octubre de 2014].
- [42] ESRI (2003, February). **System Desing Strategies. An ESRI White Paper** [en línea]. Disponible: <http://downloads.esri.com/support/whitepapers/ao_/435sysdesig.pdf> [Consulta: Agosto de 2014].
- [43] Sohdheim, M., Gardels, K., Buehler, K. (1999): **GIS interoperability** en P. Longley, M. F. Goodchild, D. J. Maguire, D. W. Rhind (Editores): **Geographical Information Systems. Principles and Technical Issues**. New York, J. Wiley and Sons, 2° ed., pages 347-358.
- [44] Instituto Panamericano de Geografía e Historia (2014). **Guía de Normas - Edición en Español. Comité ISO/TC 211 Información Geográfica/Geomática** [en línea]. Disponible: <http://www.isotc211.org/Outreach/ISO_TC_211_Standards_Guide_Spanish.pdf> [Consulta: Noviembre de 2014].
- [45] Eckel, Bruce (2006). **Thinking in Java**. 4.ª ed., Ed. Prentice Hall, pages 15-18.
- [46] GDAL - Geospatial Data Abstraction Library (2014). **KML - Keyhole Markup Language** [en línea]. [Disponible: <http://www.gdal.org/drv_kml.html> [Consulta: Noviembre de 2014].

-
- [47] Manso, M. (2009). **El uso de los metadatos para el desarrollo de un modelo de interoperabilidad para las Infraestructuras de Datos Espaciales**. Tesis, Departamento de Ingeniería Topográfica y Cartográfica, Escuela Técnica Superior de Ingenieros en Topografía, Geodesia y Cartografía. Madrid, España. p. 9-12. [en línea] <[http://latingeo.upm.es/intranet/CCD/Lists/DI_Publicaciones/Attachments/275/El uso de los metadatos para el desarrollo de un modelo de interoperabilidad para las IDE.pdf](http://latingeo.upm.es/intranet/CCD/Lists/DI_Publicaciones/Attachments/275/El_uso_de_los_metadatos_para_el_desarrollo_de_un_modelo_de_interoperabilidad_para_las_IDE.pdf)> [Consulta: Julio de 2014].
- [48] ISO (2014). **Standards** [en línea]. Disponible: <<http://www.iso.org/iso/home/standards.htm>> [Consulta: Julio de 2014].
- [49] Ramm, Frederik, Topf, Jochen, Chilton, Steve (2011). **OpenStreetMap. Using and Enhancing the Free Map of the World**. Ed. Cambridge, UK: UIT Cambridge, pages 3-7, 163-171.
- [50] ArcGIS Resources (2014). **Propiedades de una referencia espacial** [en línea]. Disponible: <<http://resources.arcgis.com/es/help/main/10.1/index.html#//003n0000001v000000>> [Consulta: Julio de 2014].
- [51] Path, (2014). **Path Definition** [en línea]. Disponible: <http://www.linfo.org/path_env_var.html> [Consulta: Septiembre de 2014].
- [52] ESRI (1998). **Shapefile Technical Description. An ESRI White Paper** [en línea]. Disponible: <<http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>> [Consulta: Agosto de 2014].
- [53] Masó J., Zabala A., Julià N. (2010b): **Fast Satellite Data Dissemination of Large JPEG2000 Compressed Images with OGC Standards**. Geomatics for Crisis Management 2010 (Gi4DM20710). Torino.
-

[54] Instituto Nacional de Estadística y Geografía (INEGI) 2015. **Imágenes Digitales Orto-rectificadas Fotográficamente (Ortofotos Digitales)** [en línea]. Disponible: <http://www.inegi.org.mx/inegi/SPC/doc/INTERNET/15-%20imagenes_digitales_ortorectificadas_fotogrametricamente.pdf> [Consulta: Enero de 2015].

GLOSARIO

Apache. Es un Servidor Web de código abierto, multiplataforma que surgió en 1995; su desarrollo se basó en el código de NCSA HTTPd (Servidor Web desarrollado por Robert McCool). Se distribuye bajo licencia Apache 2.0 [32].

BSD. Licencia de software para sistemas BSD (Berkeley Software Distribution). Licencia permisiva, sin copyleft, con cláusula de advertencia [33].

BDS 2-clause. Es una licencia simple, libre y abierta [33].

CentOS. Sistema Operativo, distribución basada en RedHat Enterprise Linux, desarrollada por una comunidad de colaboradores que crece constantemente. En la actualidad, va por la versión 7.0, e incluye versiones para arquitecturas basadas en microprocesadores Intel y AMD. Contiene todos los servidores populares de código abierto, instalación gráfica y un escritorio GNOME. Se encuentra disponible bajo la licencia GNU/GPL. <<http://www.centos.org/>> [Consulta: Septiembre de 2014].

Cliente-Servidor. Es la tecnología que proporciona al usuario final el acceso transparente a las aplicaciones, datos, servicios de cómputo o cualquier otro recurso, a través de la organización, en múltiples plataformas. El modelo soporta un medio distribuido en el cual los requerimientos de servicio hechos por computadoras (clientes), resultan en un trabajo realizado por otras computadoras llamadas servidores [16].

CSV (Comma Separated Values). Formato de archivo de texto donde cada dato, columna o valor se presenta separado por comas y cada fila o registro por saltos de línea. <<http://www.ietf.org/rfc/rfc4180.txt>> [Consulta: Noviembre 2014].

Dato georreferenciado. Se refiere a entidades geográficas que tienen una localización expresada en coordenadas geográficas, mediante la cual se pueden situar en la superficie terrestre. La georreferenciación es el proceso que permiten asignar coordenadas geográficas a objetos que no tienen un sistema de coordenadas. Un sistema de coordenadas de latitud-longitud es un ejemplo de un sistema de referencia [34].

Datum. De acuerdo con [35], un datum es un modelo matemático en el cual se incluyen un conjunto de parámetros, mediante los cuales se puede calcular la posición de objetos geográficos en la superficie terrestre. Los parámetros del datum son los siguientes: Las dimensiones del elipsoide de referencia (eje mayor, menor y la relación achatamiento de los polos); y las coordenadas del punto de origen. La mayoría de los datums son creados para su uso de manera regional, pero el WGS84 puede ser usado globalmente.

Elipsoide. Debido a que la superficie terrestre no es una esfera perfecta (que es más ancha en el ecuador que en los polos), un elipsoide se utiliza a menudo para modelar su forma. El elipsoide de referencia es definido por sus dimensiones del semieje mayor (radio ecuatorial de la tierra) y su semieje menor (radio polar de la tierra) o el semieje mayor y su relación de achatamiento [35].

Environmental Systems Research Institute (ESRI). Compañía líder a nivel mundial en el desarrollo y comercialización de software para Sistemas de Información Geográfica. <<http://www.esri.com>> [Consulta: Agosto de 2014].

EPSG (European Petroleum Survey Group). Organización relacionada con la industria petrolera en Europa. Este organismo estuvo formado por especialistas en geodesia, topografía y cartografía aplicadas al área de exploración y desarrolló un repositorio de parámetros geodésicos que contiene información sobre sistemas “marcos de referencia” antiguos y modernos “geocéntricos”, proyecciones cartográficas y elipsoides para todo el mundo [8].

Feature data Objects (FDO). API que se utiliza para la administración, definición y análisis de información geoespacial. <<http://fdo.osgeo.org>> [Consulta: Agosto de 2014].

FGDC (Federal Geographic Data Committee). Organismo del gobierno de los Estados Unidos de América encargado de desarrollar la National Spatial Data Infrastructure en aquellos aspectos que afecten la distribución de los datos geográficos. Este organismo desarrolló la norma de contenido para metadatos geoespaciales digitales (FGDC-STD-001-1998) [36].

FLOSS (Free and Open Source Software). Debido a la cercanía entre las licencias de código abierto y las de software libre, y dado que a efectos prácticos ambos tipos de software evolucionan conjuntamente dentro de un marco común de desarrollo de proyectos, es usual referirse a ambos conceptos de manera conjunta FLOSS [39].

FOSS4G. Congreso anual de software libre y de código abierto para el tratamiento de datos geoespaciales, organizado por OSGeo. <<http://foss4g.org/>> [Consulta: Septiembre de 2014].

GDAL/OGR. Librería de lectura y escritura para formatos geoespaciales, tanto raster con GDAL como vectoriales con OGR [19].

General Public Licence (GNU). Licencia libre, abierta con copyleft [33].

GeoJSON. Formato basado en JSON para codificar una variedad de estructuras de datos geográficos. Bajo este formato se pueden representar objetos geométricos, una cualidad o una colección de éstas. Los tipos geométricos soportados son puntos, líneas, polígonos, multipunto, entre otros [37].

Geomática. La Geomática es una ciencia emergente, cuyo dominio de conocimiento está basado en la convergencia de disciplinas tradicionales (Geografía: Cartografía, Geodesia Fotogrametría, Estadística espacial) y tecnológicas (Ciencias de la computación e Informática: Sistemas de bases de datos). La Geomática agrupa a las disciplinas que

comprenden el desarrollo científico necesario para la adquisición, procesamiento, almacenamiento, manejo, administración, representación, comunicación y diseminación de la información geográfica [38].

GeoServer. Es un Servidor Web de código abierto desarrollado en Java, multiplataforma, que permite servir mapas y datos geoespaciales. Diseñado para la interoperabilidad y publicación de datos de cualquier fuente de datos espaciales con estándares abiertos. <<http://geoserver.org/>> [Consulta: Agosto de 2014].

GML. Lenguaje basado en gramática XML para expresar características geográficas, se utiliza como lenguaje de modelado para sistemas geográficos y como formato abierto para el intercambio de datos vectoriales [3].

GNU GPLv3. Licencia libre, abierta con copyleft [33].

GPS (Global Position System). Sistema de navegación desarrollado por el Departamento de la defensa de los Estados Unidos de América. Originalmente fue desarrollado sólo con fines militares y con el tiempo se extendió su uso hacia aplicaciones de navegación, posicionamiento de puntos en tierra, mar y aire [40].

GPX (GPS Exchange Format). Formato estándar mantenido por TopoGrafix para guardar los resultados de un receptor de GPS. Este estándar basado en XML admite puntos de referencia y recorrido del GPS para guardarlo y compartirlo. Los puntos se recopilan en la proyección WGS84 y se guardan en pares de latitud-longitud, usualmente con información de tiempo y elevación. <<http://www.topografix.com/gpx.asp>> [Consulta: Noviembre de 2014].

GSDI (Global Spatial Data Infrastructure). Organismo internacional, en el que colaboran más de 100 países, que se ha planteado como labor fundamental la homogeneización de la información geográfica [4].

HTTP (Hypertext Transfer Protocol). Protocolo de Transferencia de Hipertexto cliente-servidor que articula los intercambios de información entre los clientes Web y los servidores HTTP. La especificación completa del protocolo HTTP 1/0 está recogida en el RFC 1945. Fue propuesto por Tim Berners-Lee, atendiendo a las necesidades de un sistema global de distribución de información como el World Wide Web [41].

INSPIRE (Infrastructure for Spatial Information in Europe). Infraestructura para la Información Espacial en Europa. Directiva 2007/2/CE del Parlamento Europeo y del Consejo, de 14 de marzo de 2007, por la que se establece una infraestructura de información espacial en la Comunidad Europea [6].

Interfaz de programación de aplicaciones (API). Conjunto de comandos, funciones y protocolos que definen como invocar desde un programa un servicio que éstos prestan. Una API especifica cómo deben interactuar los componentes de software y se utiliza en la programación de la interfaz gráfica de usuario (GUI). <<http://www.techterms.com/definition/api>> [Consulta: Agosto de 2014].

Internet Map Server (IMS). Tecnología para la publicación de datos SIG a través de Internet [42].

Interoperabilidad. Este término se refiere a la creciente necesidad de que, no solo los datos geográficos, también los diversos elementos de software que constituyen las tecnologías geográficas, puedan ser intercambiables y disponibles de un modo sistemático por los distintos usuarios [43].

ISO/TC 211. Norma que tiene el objetivo de establecer una normativa de referencia en el campo de la información geográfica digital, pensada tanto para la transferencia de datos y el mundo de los SIG alistados, como para los servicios y las Infraestructuras de Datos Espaciales (IDE) [44].

Java. Lenguaje de programación orientado a objetos. Esta tipo de programación permite crear programas de forma más parecida al pensamiento humano, simplifica el problema dividiéndolo en objetos y permitiendo centrarse en cada objeto, para que de esa forma se simplifique la complejidad. Cada objeto se programa de forma autónoma y esa es su la principal virtud [45].

JDBC (Java DataBase Connectivity). JDBC proporciona a las aplicaciones Java un mecanismo uniforme para el acceso a datos. La tecnología JDBC consiste en la utilización de un conjunto de clases (API JDBC), que disponen de una serie de métodos para operar con la base de datos. Utilizando estos métodos, la aplicación dirige todas las peticiones hacia un software intermediario, conocido como driver JDBC, cuya misión es traducir las llamadas a los métodos a órdenes nativas del gestor de base de datos. <<http://profesores.fi-b.unam.mx/carlos/java/JDBC.html>> [Consulta: Octubre de 2014].

KML (Keyhole Markup Language). Formato basado en XML para almacenar datos geográficos, es un estándar oficial del Consorcio Geoespacial abierto (OGC) [3]. KML es un formato habitual para compartir datos geográficos con personas que no utilizan SIG, ya que se puede enviar fácilmente por Internet y se puede ver en muchas aplicaciones gratuitas, como Google Earth, entre otras. Los archivos KML tienen una extensión .kml o .kmz (para archivos KML comprimidos). Cabe mencionar que KML por especificación sólo utiliza una proyección, EPSG: 4326 [46].

Metadato. Conjunto estructurado de información que describe datos, su estructura interna y a sus servicios, cuyo propósito es incrementar el conocimiento y contestar preguntas del tipo 'qué', 'quien', 'dónde', 'cuánto' y 'cómo'. También pueden considerarse productos autónomos asociados a los datos que permiten mantener un inventario de los mismos, facilitar su publicación y consulta a través de catálogos en las Infraestructuras de Datos Espaciales (IDE) y facilitar la reutilización de los datos [47].

MIT-style license. Licencia libre, permisiva, copyleft limitado [33]. Software con copyleft es software libre cuyos términos de distribución no permiten a los redistribuidores agregar ninguna restricción adicional cuando lo redistribuyen o modifican, o sea, la versión modificada también deberá ser libre.

Mosaiqueado. Se refiere a los archivos unitarios de imágenes de un mapa según su ubicación y nivel de acercamiento (zoom), Este método es bastante eficiente cuando la información del mapa es relativamente constante, debido a que las aplicaciones que utilizan teselas, acceden al vector de datos de manera anticipada para devolver las celdas más tarde con un procesamiento mínimo al recibir la petición; pero si los contenidos de dichos mapas son actualizados con cierta frecuencia, este método no es recomendable [49].

Netcraft. Compañía inglesa de servicios de Internet, que entre otros servicios se encarga de estar analizando constantemente el estado de los sitios Web de Internet; publica mensualmente un reporte sobre cómo se distribuye el uso de los diferentes servidores web. <<http://news.netcraft.com>> [Consulta: Septiembre de 2014].

Normas ISO. Las normas ISO son documentos que describen los criterios a seguir; definen las características que deben poseer un producto y los servicios que han de tener una compatibilidad para ser usados a nivel internacional. La finalidad principal de las normas ISO es orientar, coordinar, simplificar y unificar los usos para conseguir menores costos y aumentar la efectividad, garantizando que los productos y servicios sean seguros, confiables y de buena calidad [48].

NSDI (National Spatial Data Infrastructure). Infraestructura de Datos Espaciales de los Estados Unidos de Norteamérica. El proyecto está liderado por el FGDC (Federal Geographic Data Committee) [36].

OGC (Open Geospatial Consortium). El Open Geospatial Consortium fue creado en 1994 y agrupa a más de 400 organizaciones públicas, privadas, universidades, centros de investigación y administraciones públicas. Su Objetivo es la definición de estándares abiertos e interoperables dentro de los Sistemas de Información Geográfica. Persigue acuerdos entre las diferentes empresas del sector que posibiliten la interoperabilidad de sus sistemas de geoprocesamiento, y de esta manera facilitar el intercambio de información geográfica en beneficio de los usuarios [3].

OGP (International Association of Oil & Gas Producers). Organismo que absorbió en 2005 a la European Petroleum Survey Group (EPSG) [8].

OpenLS (Location Services). Especificación OGC que estandariza la prestación de servicios basados en la localización, incluyendo servicios de cálculo de rutas y navegación en tiempo real de vehículos (sector que está teniendo un gran auge en el campo SIG a raíz de la extensión del uso de navegadores GPS en vehículos de uso privado) [27].

OpenStreetMap (OSM). Es una iniciativa cuyo objetivo es crear y proporcionar datos geográficos libres y editables, tales como calles, mapas de carreteras, entre otros, de todo el mundo [49].

OSGeo (Open Source Geospatial Foundation). Fundación creada para promover y construir software geoespacial de código abierto de la más alta calidad [11].

PATH. Variable del sistema que utiliza el Sistema Operativo para buscar los ejecutables necesarios desde la línea de comandos o una ventana “terminal” [51].

Proj.4. Librería que se puede utilizar para transformar coordenadas entre diferentes sistemas de referencia [19].

RAID (Redundant Array of Independent Disks). Hace referencia a una arquitectura para el almacenamiento de datos en discos duros, que basada en niveles define el tipo de tolerancia del sistema y la forma en la que los datos se distribuyen entre dos o más discos que conforman el arreglo. RAID utiliza una técnica llamada “striping” para dividir la información antes de distribuirla en bloques que son almacenados de forma organizada en los diferentes discos del arreglo. Cabe mencionar que un sistema RAID puede ser interno o externo y su implementación puede llevarse a cabo por hardware o software. Por último incidir en el hecho de que RAID hace referencia a la arquitectura que dota de redundancia o tolerancia a fallos al sistema de almacenamiento, pero en ningún caso al arreglo de discos en sí mismo [31].

ShapeFile. Es un formato de representación vectorial desarrollado por ESRI. Consta de varios archivos, en los que se almacena digitalmente la localización de los elementos geográficos (archivo shape *.shp) junto con sus atributos o características tabulares o de atributos (tabla dBASE *.dbf) [52].

Sistemas de Referencia Espacial (SRS). Los sistemas de referencia espacial son utilizados para poder localizar objetos en un mapa [50]. Existen varios SRS, de los cuales destacan los siguientes: World Geodesic System (WGS), National Geodetic Survey (NSRS), European Petroleum Survey Group (EPSG).

Tesela o Píxel. Elemento bidimensional con forma cuadrada o rectangular que corresponde al elemento mínimo de una imagen, al que se le asocia un valor temático y una posición espacial [54].

TMS (Tile Map Service). No es un estándar oficial pero si un proyecto oficial de OSGeo, donde se describe una posible solución para la comunicación entre una arquitectura cliente/servidor utilizando un servicio de teselas. Define como el servidor debe proporcionar la información relacionada con los mapas que contiene: las capas disponibles, conjunto de escalas que soporta, conjunto de teselas para cada escala, entre otras; así como la forma de acceder a dichas teselas. Esta especificación tiene la ventaja de poder ofrecer capas con distintas configuraciones de SRS, tamaño de teselas, escalas, entre otras; pero en su contra,

requiere tener todas las teselas pregeneradas porque el acceso vía HTTP es directo al archivo siguiendo la interfaz REST y la nomenclatura definida en la especificación. <http://wiki.osgeo.org/wiki/Tile_Map_Service_Specification> [Consulta: Agosto 2014].

URL (Uniform Resource Locator). Es un Localizador Uniforme de Recurso. Es el sistema de direcciones en Internet. El modo estándar de escribir la dirección de un sitio específico o parte de una información en el Web [41].

Vistas geográficas. Es el área de trabajo donde se carga cartografía como capas. También es donde se realiza el maquetado y análisis de la información geográfica [24].

Web Technology Surveys. Empresa que proporciona información sobre el uso de diversas tecnologías empleadas en la Web. Las tecnologías incluyen sistemas de gestión de contenidos (CMS), lenguajes de programación del lado del cliente y servidor, lenguajes de marcado, codificación de caracteres, servidores Web, sistemas operativos y herramientas de análisis de tráfico. <<http://w3techs.com>> [Consulta: Septiembre de 2014].

Widget. Es un es un elemento de una interfaz gráfica de usuario (GUI), que muestra información con la cual el usuario puede interactuar. Por ejemplo: ventanas, cajas de texto, checkboxes, listbox, entre otros. Un widget provee un punto de interacción con el usuario para la manipulación directa de un tipo de dato dado. En otras palabras, los widgets son bloques básicos y visuales de construcción que, combinados en una aplicación, permiten controlar los datos y la interacción con los mismos. <<http://whatis.techtarget.com/definition/widget>> [Consulta: Septiembre de 2014].

WMTS (Web Map Tile Service). El WMTS proporciona un enfoque complementario al WMS; mientras el WMS se centra en la renderización de mapas personalizados, el cual es una solución ideal para datos dinámicos, el WMTS renuncia a la personalización de los mapas para obtener una mayor escalabilidad, sirviendo datos prerenderizados donde la envolvente y las escalas han sido restringidas a un conjunto discreto de mosaicos que siguen una geometría de malla regular. Ortofotos o mapas con poca actualización temporal son ejemplos de capas ideales para elegir WMTS como servicio de difusión de mapas. El conjunto de

mosaicos permite la ejecución de un servicio WMTS utilizando un Servidor Web que simplemente devuelve los archivos existentes y el uso de mecanismos de red estándar de escalabilidad, tales como sistemas de caché distribuidos [53].