

Capítulo 2

Análisis de Herramientas y Tecnología

En el presente capítulo se explicará de manera detallada las principales herramientas y tecnologías que se ocuparon a lo largo de este proyecto. Es importante el estudio de cada una de ellas, pues de esa manera uno puede aprovechar las bondades que ofrecen o bien, eliminar su uso debido a las desventajas que presentan. El capítulo 2 es de gran importancia, pues brinda un panorama claro y general de la tecnología usada para este proyecto.

2.1 Javascript

Javascript es muy fácil de aplicar para el desarrollo de un sitio Web, pues las funciones básicas de este lenguaje son soportadas por la mayoría de los navegadores que se utilizan diariamente. Una desventaja, es que su aparente simplicidad se basa en una disponibilidad de objetos limitada, por lo que algunos procedimientos, aparentemente muy sencillos, requieren *scripts* bastante complejos.

La característica principal de Javascript es la de ser el lenguaje de “scripting” por excelencia y, sin lugar a dudas, el más usado. Esta particularidad conlleva una notable serie de ventajas y desventajas según el uso que se le deba dar, teniendo en cuenta la relación que

se establece entre el mecanismo cliente-servidor [html point.com 2001]. Para explicar con pocas palabras dicha relación, podemos decir que el servidor envía los datos al cliente y estos datos pueden llegar en dos formatos: en formato texto (o ASCII) o en formato binario (o código máquina).

El concepto de “script” queda bien expresado con una comparación que podemos leer en el texto de Michael Moncur sobre Javascript. La explicación es muy eficaz y merece citarse: “script” en inglés significa "guión". De hecho, el uso es exactamente éste: el navegador lee una línea, la interpreta y la ejecuta, después pasa a la sucesiva y hace lo mismo, y así hasta el cierre del “script”.

2.1.1 Ventajas y Desventajas de Javascript

El lenguaje de scripting es seguro y fiable: porque está en claro que debe ser interpretado, por lo que puede ser filtrado para el mismo Javascript. La seguridad es casi total y sólo en su primera versión el CIAC (Computer Incident Advisory Committee) señaló problemas de leve entidad, entre ellos la lectura de la caché y de los sitios visitados, de la dirección e-mail y de los archivos presentes en el disco. Sin embargo, estos fallos se corrigieron ya en las versiones de Netscape sucesivas a la 2.0

Los script tienen capacidades limitadas, por razones de seguridad, por lo cual no es posible hacer todo con Javascript, sino que es necesario usarlo conjuntamente con otros

lenguajes evolucionados, posiblemente más seguros, como Java. Dicha limitación es aún más evidente si queremos operar en el hardware de la computadora, como, por ejemplo, la fijación en automático de la resolución video o la impresión de un documento.

El código es visible y puede ser leído por cualquiera, incluso si está protegido con las leyes del copyright.

El código Javascript se ejecuta en el cliente por lo que el servidor no es solicitado más de lo debido.

El código del script debe descargarse completamente antes de poderse ejecutar. Si los datos que un script utiliza son muchos (por ejemplo, una recopilación de citas que se mostrará de manera casual), el tiempo que tardará en descargarse será muy largo, mientras que la interrogación de la misma base de datos en el servidor sería más rápida.

2.2 Servlets

Los Servlets son una mezcla de código Java y código HTML. Estos programas se ejecutan en un servidor Web y construyen páginas Web dinámicas. Construir páginas Web dinámicas es sumamente útil, ya que gracias a esto se puede interactuar con una página y no solo leer como pasa con páginas desarrolladas con simple código HTML. [Hall, 2001]

2.2.1 Ventajas de Servlets

Los Servlets Java son eficientes, fáciles de usar, poderosos, portables y fáciles de implementar para cualquier persona que conozca el lenguaje de programación Java.

Conveniencia. Los Servlets tienen una gran infraestructura para análisis automático y decodificación de datos de formularios HTML, leer y seleccionar cabeceras HTTP, manejar cookies, seguimiento de sesiones, y muchas otras utilidades.

Potencia. Los servlets se pueden Comunicar directamente con el servidor Web, lo cual simplifica las operaciones necesarias para buscar imágenes y otros datos almacenados en situaciones estándares. Los Servlets también pueden compartir los datos entre ellos, haciendo las cosas útiles como almacenes de conexiones a bases de datos fáciles de implementar. También pueden mantener información simplificando cosas como seguimiento de sesión y el caché de cálculos anteriores.

Portable. Los Servlets están escritos en Java y siguen un API bien estandarizado; consecuentemente, los servlets escritos, digamos en el servidor I-Planet Enterprise, se pueden ejecutar sin modificarse en Apache, Microsoft IIS, o WebStar. Los Servlets están soportados directamente o mediante plug-in en la mayoría de los servidores Web.

2.3 JSP

Java Server Pages (JSP) es una tecnología que nos permite introducir código Java en una página desarrollada con HTML estático. También existe la posibilidad de ejecutar comandos de Java y antes de terminar, por ejemplo un ciclo, ejecutar comandos HTML, con lo cual se facilita la programación de páginas dinámicas. Los JSP's son muy similares a los servlets, de hecho al compilarse un JSP, este es convertido a servlet. [Hanna, 2002].

2.3.1 Ventajas de JSP

Contra los Servlets. JSP no nos da nada que no pudiéramos hacer con un servlet, pero es mucho más conveniente escribir (y modificar) HTML normal que tener un gran número de sentencias “print” que generen HTML. Además, es posible agregar código Java una página que fue anteriormente diseñada con puro HTML.

Contra JavaScript. JavaScript puede generar HTML dinámicamente en el cliente; esta es una capacidad útil, pero sólo maneja situaciones donde la información dinámica está basada en el entorno del cliente. Con la excepción de las cookies, el HTTP y el envío de formularios no están disponibles con JavaScript. Debido a que se ejecuta en el cliente, JavaScript no puede acceder a los recursos en el lado del servidor, como bases de datos y catálogos. Por ejemplo: si se desea guardar los datos que se introducen en un formulario

dentro de una base de datos, no es posible hacerlo con JavaScript, ya que la base de datos se encuentra en el servidor, y los formularios se ejecutan en el cliente.

Sin embargo, es posible aprovechar las ventajas de JSP y Javascript si se utilizan conjuntamente. Una de las principales ventajas de Javascript es el manejo de entornos visuales; y la mayor ventaja de JSP sobre Javascript es la capacidad de comunicación dentro de un servidor; por lo cual se pueden mezclar para validar formularios que han de ser enviados al servidor. Con esto, se logra una rápida y fácil validación de los datos que el usuario del sistema introduce.

2.4 UML

Lenguaje de Modelamiento Unificado (UML - Unified Modeling Language) es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. UML provee una forma de modelar aspectos conceptuales como lo son procesos y funciones de sistema, además de aspectos concretos como lo son escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables. [OMG, 2003]

UML se puede usar para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real. UML ofrece nueve diagramas en los cuales modelar sistemas.

- Diagramas de Casos de Uso para modelar los procesos 'business'.
- Diagramas de Secuencia para modelar el paso de mensajes entre objetos.
- Diagramas de Colaboración para modelar interacciones entre objetos.
- Diagramas de Estado para modelar el comportamiento de los objetos en el sistema.
- Diagramas de Actividad para modelar el comportamiento de los Casos de Uso, objetos u operaciones.
- Diagramas de Clases para modelar la estructura estática de las clases en el sistema.
- Diagramas de Objetos para modelar la estructura estática de los objetos en el sistema.
- Diagramas de Componentes para modelar componentes.
- Diagramas de Implementación para modelar la distribución del sistema.

Para la descripción de este proyecto, se utilizaron los diagramas de uso. Sin duda alguna, los diagramas de casos de uso facilitan la comprensión del problema resuelto, así como la visualización del sistema en general.

2.5 MVC (Model View Controller)

El MVC es una arquitectura que se aplica al concepto de programación orientada a objetos. En sus inicios fue desarrollado para el lenguaje de programación “Smalltalk”. El

MVC propone una división lógica de una aplicación en tres partes (Modelo, Vista, Controlador.) y cada una de ellas debe ser independiente de cualquier otra. [Hanna, 2002].

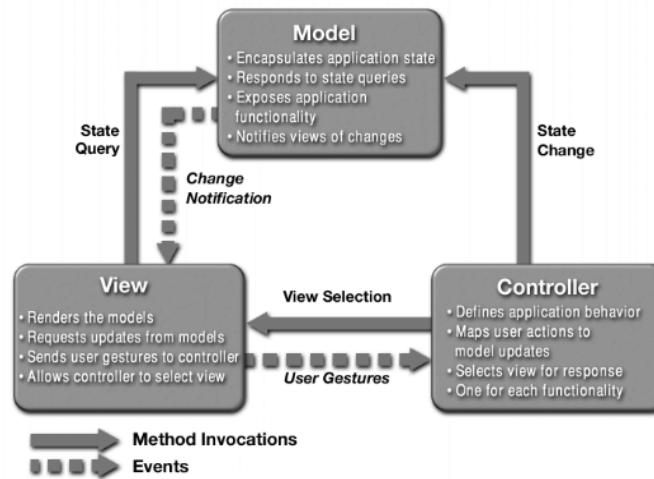


Figura 2.1: Estructura MVC

Fuente: Hanna (2002)

Modelo: Es la parte lógica del sistema; en la cual se ejecutan las “opciones” generales del sistema.

Vista: Es la representación del modelo; no requiere programación compleja, prácticamente es la interfaz gráfica con el usuario.

Controlador: Es quien toma las entradas que proporciona el usuario y da las instrucciones al modelo.

[Hanna, 2002]

A continuación se muestra un ejemplo del funcionamiento del Model View Controller (MVC) en el cual se envía una solicitud al Controlador y después se envía al visor o al modelador (según sea necesario) y finalmente el visor muestra los resultados. (Figura 2.2)

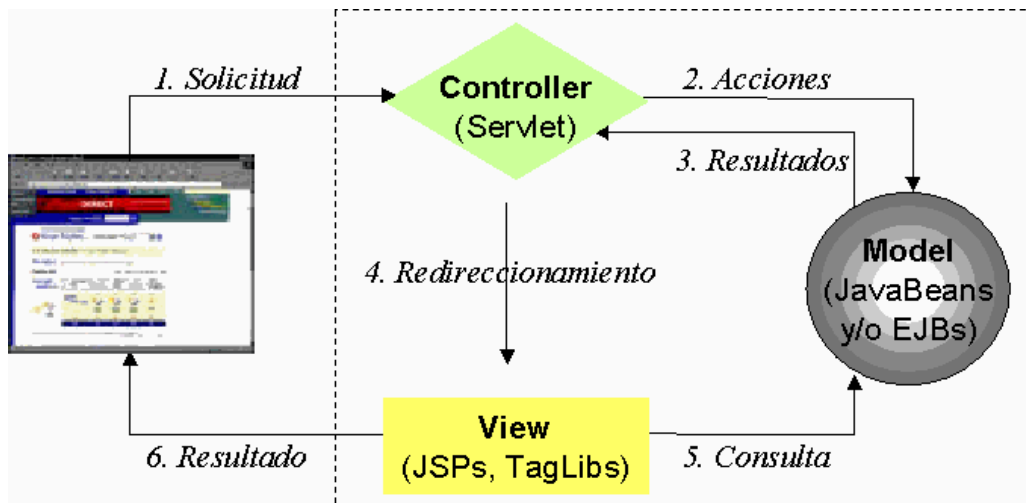


Figura 2.2: Ejemplo MVC

Fuente: Hanna (2002)

2.6 ArcIMS

En lo que se refiere a la tecnología de GIS (Geographical Information System), existe una compañía muy poderosa (ESRI), la cual ha desarrollado un conjunto de aplicaciones con el fin de facilitar el manejo de datos necesarios para el desarrollo y despliegue de datos GIS.

Una de estas herramientas es ArcIMS, la cual tiene entre otras aplicaciones, funcionar como servidor de mapas espaciales, situación que resulta importante para el despliegue de un mapa espacial. Además ArcIMS cuenta con varias herramientas capaces de interactuar con el mapa espacial; una de ellas permite realizar queries espaciales. [ESRI, 2003]

ArcIMS está basado en una arquitectura “multi-nivel”, altamente distribuida y escalable. Por otro lado, este sistema se compone de 3 niveles: clientes, servicios y administración de datos (Figura 2.3). [ESRI, 2003]

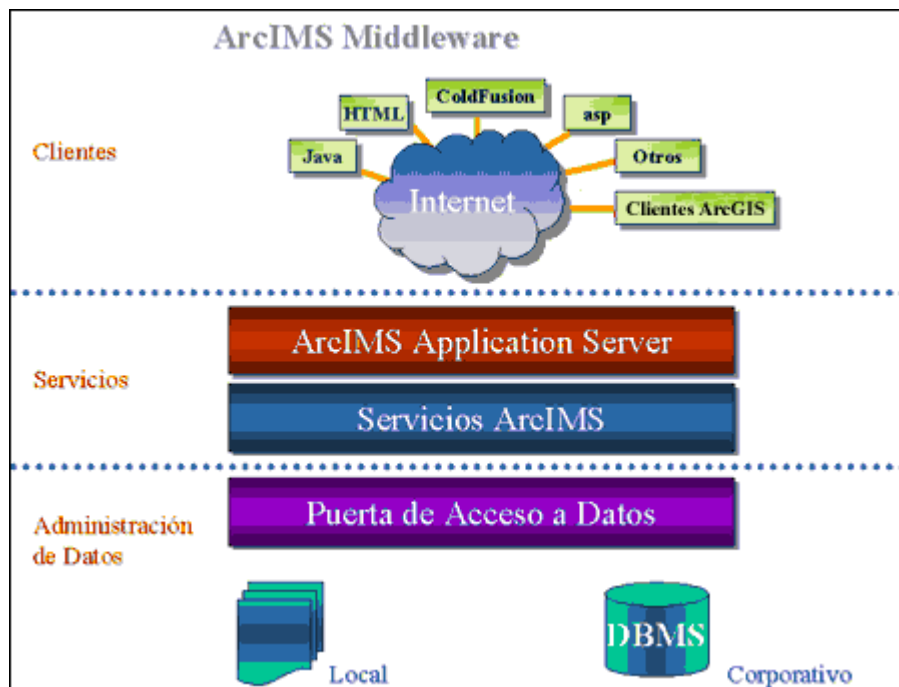


Figura 2.3: Estructura de ArcIMS

Fuente: ESRI (2003)

ArcIMS por parte del servidor está conformado por cuatro partes que realizan un trabajo específico, las cuales son:

- **Servidor Espacial:** Es el encargado de procesar las peticiones de mapas e información alfanumérica asociada. En general, el servidor espacial se encarga de realizar las siguientes tareas: [ESRI, 2003]
 1. Creación de mapas cartográficos en formato de imagen, (jpg, gif o png).
 2. Creación de mapas cartográficos en formato vector.
 3. Búsqueda de información en la base de datos.
 4. Geocodificación de direcciones.
 5. Acceso y recorte de los datos para su posterior envío en formato shapefile.

- **Servidor de Aplicaciones:** Maneja el balance de los procesos y demandas entrantes, y mantiene un registro de los servicios de mapas ejecutados en los servidores espaciales. [ESRI, 2003].

- **Conectores del Servidor de Aplicaciones:** Conecta el servidor de Web al Servidor de Aplicaciones. ArcIMS provee tres tipos de conectores - Servlet (Java), ColdFusion y ActiveX (VB,ASP). [ESRI, 2003]. Lo anterior puede observarse gráficamente en la figura 2.4

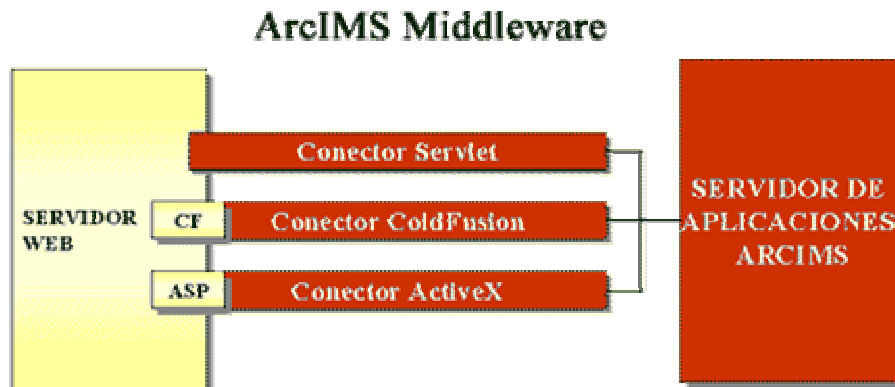


Figura 2.4: Conector ArcIMS

Fuente: ArcIMS (2003)

- **Administrador:** Conjunto de asistentes de fácil uso para el manejo de todas las funciones y tareas relacionadas con el servidor. [ESRI, 2003]

ArcIMS incluye dos tipos de clientes, un visualizador HTML y un visualizador Java, que difieren en su apariencia gráfica y funcionalidades. El cliente Java ofrece más funciones. [ESRI, 2003]

ArcIMS ofrece dos tipos de servicios de mapas, el servidor de imágenes y el servidor de elementos geográficos [ESRI, 2003]

Servidor de imágenes (image server): Genera y transmite los mapas en formato JPEG, PNG o GIF a los clientes. Las imágenes cartográficas pueden ser creadas desde archivos shape, datos ArcSDE y diferentes tipos de datos ráster. [ESRI, 2003]

Servidor de elementos geográficos (feature server): Transmite (solo) al cliente JAVA archivos shape y capas ArcSDE en formato comprimido. El cliente recibe al mismo tiempo instrucciones adecuadas para procesar los datos transmitidos. Este tipo de servicio de mapas permite una funcionalidad más avanzada, tales como:

- Poner etiquetas a elementos geográficos en el cliente.
- Cambiar la simbología del mapa.
- Selección espacial en el cliente.
- Superposición de elementos geográficos procedentes de varios servidores de mapas o del sistema local

Por otro lado, la parte del cliente solo está conformado por los visualizadores, de los cuales solo existen 2 tipos, que se clasifican por su capacidad de procesamiento de tareas de los mapas espaciales; estos dos tipos son: Visores HTML y Visores Java. [ESRI, 2003]

Visor HTML.- Es un visor ligero, el cual puede ser integrado en cualquier navegador. Debido a que este tipo de visor es más ligero y compatible, se debe sacrificar la operabilidad de algunas tareas que pueden ser realizadas en un mapa espacial, pero se tiene

la seguridad de que el mapa podrá ser visualizado en cualquier navegador (Fig 3.5), además que no se requiere un ancho de banda tan grande como en el Visor Java. Unas de las tareas que se pueden realizar con el Visor HTML son: queries de atributos, queries espaciales (como medición de distancias), realizar acercamiento en el mapa etc. [ESRI, 2003]

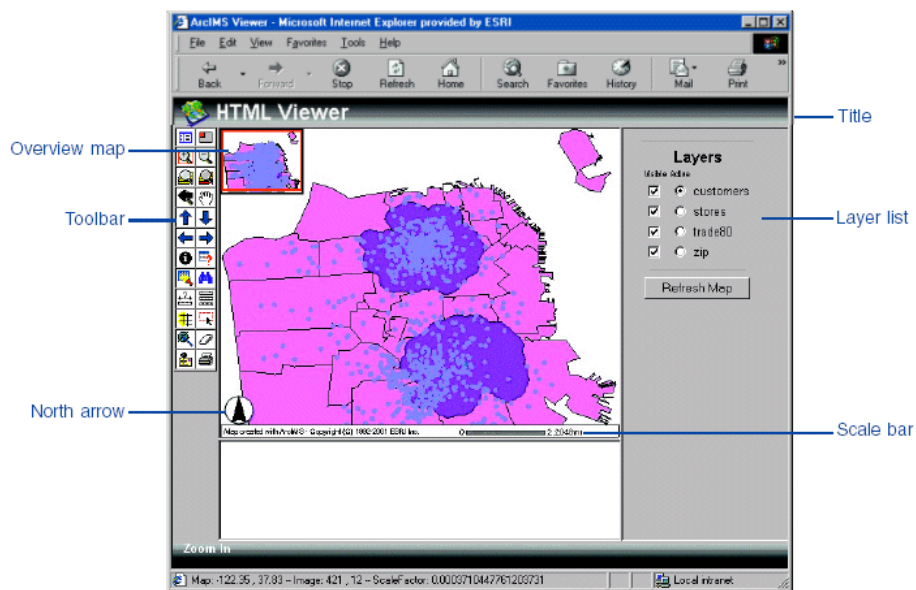


Figura 2.5: Ejemplo de un mapa implementado en Visor HTML

Fuente: ESRI (2003)

Visor Java.- Es un visor más pesado en el cual se pueden realizar más tareas que en el visor HTML, razón por la cual se requiere un navegador que tenga el plug-in de la maquina virtual de java, además de contar con una conexión bastante rápida, ya que el cliente debe descargar 3.8 mega bytes antes de poder visualizar y manipular el mapa. A continuación se muestra un ejemplo de un mapa diseñado para ser visto en el “Java Viewer” (Fig. 2.6) [ESRI, 2003]

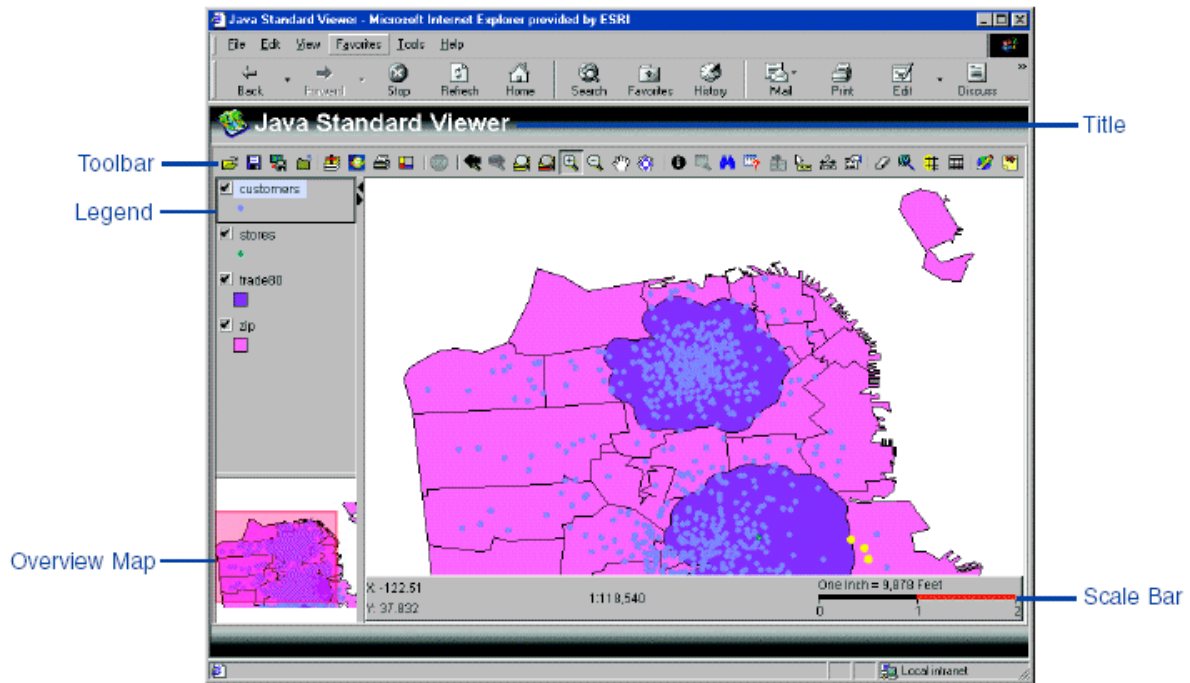


Figura 2.6: Ejemplo de un mapa implementado en Visor Java

Fuente: ESRI (2003)

2.7 SIG (Sistema de Información Geográfica)

Un SIG es un sistema computarizado que está conformado por una colección de datos y programas que están georeferenciados. Los programas tienen como objetivo almacenar, procesar y desplegar información espacial que está asociada en una base de datos, la cual contiene los valores de los atributos.

Un SIG está compuesto por 5 partes esenciales las cuales son: hardware, software, gente, métodos y datos [ESRI, 2003]

Un SIG necesariamente debe funcionar como conexión entre el usuario y la base de datos espacial. La base de datos, es una base de datos “normal” ya que depende de un manejador de bases de datos ordinario, por ejemplo Oracle o Informix. La base de datos es creada de tal manera que la información es organizada para facilitar la búsqueda de datos.

Existen herramientas comerciales, capaces de realizar los procesos de construcción, utilización y mantenimiento de un SIG de forma eficiente. Generalmente los procesos antes mencionados se manejan por separado. Un usuario no debe tener acceso a las tablas de la base de datos, en cambio, un administrador debe poder realizar los cambios pertinentes, con el fin de crear o dar mantenimiento a una base de datos.

La mayor utilidad de un SIG, es la capacidad que este tiene a responder preguntas espaciales o no espaciales y desplegarlas visualmente. Las preguntas espaciales (queries) son aquellas que requieren datos propios de las cartografías digitales, con lo cual se pueden medir distancias, buscar ubicaciones por medio de coordenadas etc. Los queries no espaciales, son aquellos que se relacionan con los atributos de los objetos que pertenecen a la cartografía digital. Tanto los queries espaciales como los no espaciales pueden ser desplegados de forma visual.

2.8 Bases de datos

A continuación se presenta una comparación entre dos manejadores (DBMS'S) de bases de datos más utilizadas en la actualidad, MySQL y Oracle. Cada uno de los manejadores antes mencionados tiene características específicas que se reflejan en ventajas y desventajas propias de cada uno. Sin duda alguna, es importante conocer los beneficios que se van a tener al momento de escoger el manejador de base de datos que se va a utilizar.

2.8.1 Ventajas de MYSQL

- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo, lo cual la hace una base de datos muy veloz.
- Soporta gran cantidad de tipos de datos para las columnas.
- Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, etc).
- Gran portabilidad entre sistemas.
- Soporta hasta 32 índices por tabla.
- Gestión de usuarios y passwords, manteniendo un muy buen nivel de seguridad en los datos.

[Pecos, 2000].

2.8.2 Desventajas de MYSQL

- Subconsultas: tal vez ésta sea una de las características que más se echan de menos, aunque gran parte de las veces que se necesitan, es posible reescribirlas de manera que no sean necesarias.
- SELECT INTO TABLE: Esta característica propia de Oracle, todavía no está implementada.
- Triggers y Procedures: Se tiene pensado incluir el uso de procedures almacenados en la base de datos. Más no el de triggers, ya que los triggers reducen de forma significativa el rendimiento de la base de datos, incluso en aquellas consultas que no los activan.
- Transacciones: a partir de las últimas versiones ya hay soporte para transacciones, aunque no por defecto (se ha de activar un modo especial).
- Integridad referencial: aunque sí que admite la declaración de claves ajenas en la creación tablas, internamente no las trata de forma diferente al resto de campos. [Pecos, 2000].

- **Mantenimiento y soporte:** Debido a que MySQL es “open-source”, no es posible contratar algún tipo de mantenimiento o soporte. Si se da el caso de que el manejador de bases de datos falla, el usuario deberá resolver el problema sin poder contar con ayuda técnica de los desarrolladores del software.

2.8.3 Oracle

Sin duda alguna. Oracle es uno de los manejadores de bases de datos más robustos que existen actualmente. Brinda todos los servicios que tiene un manejador de base de datos básico, tal como MySQL, pero sin olvidar opciones avanzadas como las subconsultas. A pesar de contar con un gran número de posibilidades para realizar consultas complejas, no es lento al momento de ejecutar las queries.

El problema máximo al que se enfrenta un usuario cuando decide utilizar Oracle, es el precio. Siendo Oracle un manejador de bases de datos muy poderoso, su precio es un punto importante a considerar; ya que sólo está al alcance de empresas medianas o grandes o para instituciones bien consolidadas.

2.9 Jakarta Tomcat

Tomcat es la implementación de referencia oficial para las especificaciones Servlet 2.2 y JSP 1.1. Puede ser usado como pequeño servidor para probar páginas JSP y servlets, o

puede integrarse en el servidor Web Apache. Tomcat, al igual que el propio Apache es gratuito. Sin embargo, también al igual que Apache (que es muy rápido, de gran rendimiento, pero un poco difícil de configurar e instalar), Tomcat requiere significativamente más esfuerzo para configurarlo que los motores de servlets comerciales. [Luc, 2002]

Tomcat es el sucesor del jserv y apoya el jsp y los servlet-APIs. Desafortunadamente Tomcat es muy difícil de construir de fuente, porque está utilizando su propio edificio-sistema llamado "hormiga". Hay también una lista muy larga de requisitos previos [Luc, 2002]

2.10 Flash

Flash es una tecnología desarrollada por Macromedia. Permite crear y reproducir contenido Web basado en vectores. Gracias a esa característica, el texto puede utilizar cualquier tipo de fuente y determinadas animaciones se reproducen en menos tiempo del habitual. Además Flash permite comprimir audio y utilizarlo en la Web. [Candela, 2002]

Para reproducir estas ventajas se necesita el reproductor Macromedia Flash Player. El reproductor es gratuito y le permite reproducir miles de páginas que ya utilizan esta tecnología. El 90% de los usuarios tienen alguna versión de Flash instalada. [Candela, 2002]

Claro que no todo son ventajas. Sus cuatro principales inconvenientes son:

- No todos los usuarios pueden ver los contenidos en Flash. Si el visitante no tiene instalado el reproductor, no verá su página o algún contenido. Además es muy probable que si no ha instalado el reproductor hasta el momento, tampoco lo instale para ver su sitio Web.
- El texto escrito en Flash no se integra correctamente en los motores de búsqueda. Los principales motores de búsqueda rastrean todas las palabras escritas de un sitio HTML. Esta característica permite encontrar su sitio a partir de cualquier palabra del texto. En cambio, los motores de búsqueda no rastrean el texto en Flash.
- El texto no cambia de tamaño con el botón del explorador para ampliar y reducir las fuentes. Los usuarios están forzados a leer el texto con el tamaño que el diseñador ha especificado o cambiar su tamaño pulsando el botón derecho del ratón. Esta opción tiene el inconveniente de no ser estándar del navegador y el usuario puede no conocerla. Además cuando una imagen de Flash se amplía no cabe en el espacio establecido por el diseñador y obliga al usuario a utilizar otra opción no estándar y muy incómoda: arrastrar la imagen.
- La opción de abrir el enlace en nueva ventana no existe. Esta carencia molesta al usuario experto que quiere explorar su sitio desde varias ventanas.

Posiblemente Flash sea una buena alternativa si está pensando en desarrollar cortos de animación o videojuegos para la red.

De lo contrario, lo aconsejable es no deslumbrarse con las posibilidades de esta tecnología.

2.11 Dreamweaver

Dreamweaver es una editor profesional para crear y mantener sitios Web utilizando la técnica WYSIWYG: What You See Is What You Get (lo que se ve es lo que se obtiene). De esta manera no es necesario conocer ni recordar código HTML comúnmente utilizado para esta tarea. Con Dreamweaver se trabaja de manera similar a la de un procesador de texto. [Edutech, 2000]

A diferencia de otros editores que utilizan el método WYSIWYG, Dreamweaver permite trabajar con DHTML (Dynamic HTML), insertar capas en las páginas, insertar componentes multimedia (active x), applets creados en Java, películas creadas con Flash y películas Shockwave.

Dreamweaver es uno de los editores de páginas Web más profesionales. A grandes rasgos, podemos decir que es una máscara que se coloca delante del código HTML para que su edición sea más intuitiva y visual.

2.11.1 La tecnología WYSWYG

HTML es un lenguaje informático basado en texto e instrucciones. Como únicamente utiliza texto, en principio con cualquier editor de textos se podría crear un documento HTML. Sin embargo, esta forma de trabajar resulta francamente desalentadora, pues durante todo el proceso debes imaginar cómo va a quedar la página una vez acabada, ya que no estamos componiendo de forma visual. Además, debes tener presente la lista de “tag” y las normas sintácticas para hacer un buen trabajo. [McFedries, 1997]

Para solucionar estos problemas, surgieron los editores de páginas Web, que utilizan un entorno visual, mucho más creativo y agradable. De estos editores, Dreamweaver es uno de los mejores, ya que intenta emular la tecnología WYSWYG (Lo que ves es lo que hay), característico de los programas de diseño, como QuarkXPress, InDesign o Freehand. De hecho, con Dreamweaver se pueden diseñar páginas Web sin tener un ápice de conocimiento de HTML.

Dreamweaver posee una doble manera de crear páginas Web: la manual (mediante código HTML) y la visual. Ambas formas de trabajo están perfectamente relacionadas, de modo que a medida que se crea un documento en la ventana de Dreamweaver, éste va generando el código HTML necesario. De igual forma, también se puede acceder al código HTML que se ha ido generando y editarlo de forma manual; automáticamente, los cambios realizados, se actualizan en la ventana visual del documento.

Dreamweaver ha ido evolucionando desde sus primeras versiones, en las que únicamente era un editor de páginas Web, hasta las últimas versiones, en las que se han integrado opciones avanzadas como funciones JavaScript predefinidas (llamadas comportamientos), herramientas de productividad como plantillas y bibliotecas, y aplicaciones para gestionar un sitio Web completo como el cliente FTP incluido en el programa.

A continuación se comenzará a escribir sobre la tecnología de GIS (Geographical Information System) ocupada en este proyecto. Con ello podrá observarse la manera en como se complementa lo anteriormente expuesto con esta tecnología nueva. Además, la principal ventaja radica en la manera en como las tecnologías se complementan para lograr una aplicación que pueda ser utilizada para el mejoramiento de los procesos administrativos de una área.