



SOLUCIONES POTENCIALES DE SIG LIBRES PARA CUBRIR LAS NECESIDADES DE GESTIÓN MUNICIPAL EN AYUNTAMIENTOS MEDIANOS Y PEQUEÑOS

José Carlos Martínez Llario

Profesor Funcionarios Doctores

Departamento de Ingeniería Cartográfica, Geodesia y Fotogrametría.
E.T.S.I. Geodésica, Cartográfica y Topográfica, Universidad Politécnica de Valencia

Eloina Coll Aliaga

Profesora Funcionarios Doctores

Departamento de Ingeniería Cartográfica, Geodesia y Fotogrametría.
E.T.S.I. Geodésica, Cartográfica y Topográfica, Universidad Politécnica de Valencia

Jesús Irigoyen Gaztelumendi

Profesor Funcionarios Doctores

Departamento de Ingeniería Cartográfica, Geodesia y Fotogrametría.
E.T.S.I. Geodésica, Cartográfica y Topográfica, Universidad Politécnica de Valencia

1. INTRODUCCIÓN

En las últimas décadas los Sistemas de Información Geográfica (SIG) han evolucionado rápidamente. Han pasado de ser unas herramientas muy específicas en manos de unos pocos expertos, a constituir uno de los fundamentos de cualquier análisis espacial riguroso.

El software SIG tiene una importancia fundamental en cualquier proceso y cálculo del análisis geográfico y conviene desarrollar estas herramientas desde diversos puntos de vista. En este sentido, además del Software Propietario, se están desarrollando Sistemas de Información Geográfica Libres. Esta clase de software juega un papel muy importante en el avance de las tecnologías SIG, puesto que favorece la aparición de aplicaciones experimentales y proporciona acceso a la tecnología SIG a usuarios que no puedan o no quieran recurrir al software propietario.

El Software Libre representa una revolución en el modo en que están concebidos los actuales programas informáticos. Esta clase de software no pone límites a la creatividad ni al ingenio del posible usuario. Además se trata de software que siempre resultará más económico que el Software Propietario. La licencia GNU General Public License ofrece plenas libertades de uso al usuario. Las únicas restricciones que impone esta licencia protegen las aportaciones de los distintos usuarios frente a su posible utilización en Software Propietario. Los usuarios de SIG necesitan desarrollar sus capacidades en función de sus necesidades. Por este motivo la idea de los SIG de código abierto resulta tan necesaria.

Estos estudios se engloban dentro de las tareas preliminares realizadas en la primera fase del proyecto de investigación SIGMUN, Proyecto BIA2003-07914 financiado por el Ministerio de Ciencia y Tecnología y los fondos FEDER, proyecto descrito con más detalle en el artículo presentado en este mismo congreso con el título “Estudio piloto de las necesidades municipales para la implantación de un sistema de información geográfica para la gestión municipal en los ayuntamientos medianos y pequeños”.

En dicho proyecto de investigación se ha realizado un estudio de las necesidades para la implantación de un SIG en ayuntamientos medianos y pequeños, como se comenta en este proyecto uno de los agentes que intervienen en el modelo de solución son empresas desarrolladoras de software encargadas de crear las aplicaciones SIG personalizadas para los ayuntamientos:

“Estas empresas están actualmente capacitadas para buscar soluciones informáticas a la gestión municipal mediante herramientas SIG de fácil manejo, a bajo coste y adaptables a las necesidades particulares de cada ayuntamiento. Son muchos los ayuntamientos receptivos a disponer de las ventajas que suponen estas herramientas. Constituyen, por tanto, muchos clientes a los que implantar los SIG y mantener su software actualizado con las nuevas versiones de sistemas operativos o los nuevos adelantos técnicos. Lógicamente, estas empresas tienen que cobrar sus servicios de creación, desarrollo y personalización del software así como el de la implantación y adaptación a cada caso particular. Es necesario que un organismo como pueden ser las diputaciones sean las encargadas de centralizar este tipo de proyectos que involucran a los ayuntamientos. Los costes del software y su adaptación a cada caso deberían correr a cargo de las propias diputaciones”.

Es en este punto donde entra nuestra propuesta de utilizar software SIG libre. De esta forma desaparecerían los costes de licencias, quedando únicamente a cargo de las diputaciones la actualización, personalización, mejora, y soporte de los programas utilizados por los ayuntamientos.

En este artículo vamos a describir varios productos SIG con licencia GPL (*General Public License*). El objetivo de este estudio es mostrar las diferentes alternativas que disponemos en el mundo de SIG libre, alternativas que aunque las vamos a plasmar mediante un software específico en realidad representan a bloques diferenciados según la metodología que se desee emplear para cubrir las necesidades del usuario final del producto.

Estos bloques vendrían definidos por las siguientes características:

- Un bloque A que representa a grandes programas consolidados en el mercado de los SIG, pero cuya utilización por parte del usuario final presenta dificultades debido entre otras causas al austero interfaz empleado por el software. En este bloque describiremos el programa GRASS originalmente desarrollado por el USA-CERL (U.S. Army Construction Engineering Research Laboratories). Página oficial de GRASS: <http://grass.itc.it/index.html>
- Un bloque B que representa un modelo de programa con un interfaz mucho más agradable y fácil de utilizar pero que no está tan consolidado en el mercado ni tampoco presenta una funcionalidad tan amplia como los programas del primer bloque. El software que describiremos en este bloque es el programa JUMP en cuya creación han intervenido entre otros organismos el grupo Vivid Solutions. Página oficial de JUMP: <http://www.jump-project.org/>
- Un bloque C que representa quizás la solución tecnológica más avanzada. En este bloque describiremos el software PostGIS que implementa una base de datos espacial sobre el Sistema Gestor de Bases de Datos (SGBD) PostgreSQL. Por contra, esta solución tiene el gran inconveniente de no presentar un entorno gráfico de introducción o edición de datos espaciales. PostGIS ha sido desarrollado por Refrations Research Inc como un proyecto de investigación en tecnologías de bases de datos en código abierto. Página oficial de PostGIS: <http://postgis.refrations.net/>

Los dos bloques siguientes no representan a programas completos de SIG como los anteriores, sino a una librería de desarrollo (OPENMAP), y a un servidor de cartografía (MAPSERVER), estos bloques han sido tratados de forma más breve en este artículo.

- Un bloque D que representa a librerías de programación mediante las que se puede implementar un SIG propio. Esta opción es la que más libertad deja al programador para cubrir las necesidades de los ayuntamientos pero también es la que más trabajo supone. En este bloque comentaremos las librerías OPENMAP, versión liberada como aplicación Java y código abierto en 1998 y en cuya creación interviene el grupo BBN Technologies. Página oficial de OPENMAP: <http://openmap.bbn.com/>
- Por último vamos a describir un bloque E que engloba a las soluciones en el ámbito de los servidores de cartografía, con el objetivo de publicar los contenidos cartográficos en Internet mediante la construcción de diferentes tipos de visualizadores que se integran dentro del navegador. Este bloque viene representado por el servidor de mapas MAPSERVER, originalmente desarrollado por la Universidad de Minnesota. Página oficial: <http://mapserver.gis.umn.edu/index.html>

2. GRASS

2.1. DESCRIPCIÓN Y CAPACIDADES

GRASS (*Geographical Resources Analysis Support System*) es un GIS con capacidad raster y vectorial. Además cuenta con sistemas integrados de visualización de datos y de procesado de imagen.

GRASS incluye más de 350 módulos para gestión, procesado, análisis y visualización de datos georeferenciados.

GRASS es GNU GPL desde 1999. Esta licencia protege a los contribuidores de GRASS frente al uso de sus contribuciones en proyectos propietarios que no permiten el libre acceso a su código fuente. La GPL asegura que todo el código publicado basado en código GPL, solo pueda ser publicado a su vez bajo licencia GPL.

GRASS está disponible a través de Internet y CD-ROM como versiones binarias precompiladas para las diferentes plataformas UNIX, Mac OS X y MS-Windows. Además de los archivos binarios, podemos obtener el código fuente completo en lenguaje C. Como la GPL proporciona a GRASS el estatus de Software Libre y protege a sus autores, pueden ofrecerse servicios comerciales relacionados con GRASS y son bienvenidos tanto por los desarrolladores como por la comunidad de usuarios.

GRASS es un SIG modular con datos organizados como coberturas raster, vectoriales y de puntos. GRASS proporciona una gran variedad de herramientas que permiten clasificarlo como un SIG de altas prestaciones. A continuación se resumen las principales funcionalidades de GRASS:

- Integración de datos espaciales
- Procesado de datos raster
- Procesado de datos vectoriales
- Procesado de datos puntuales
- Procesado de imágenes
- Visualización
- Modelado y simulaciones
- Soporte para datos temporales
- Procesado de datos 3D
- Enlaces con otras herramientas

La versión 5.0 de GRASS tiene un soporte limitado para datos temporales y 3D, sin embargo, la versión experimental 5.7 (la versión 5.7.0 ha sido liberada recientemente en Junio de 2004) está siendo diseñada como un SIG 3D completo con soporte para datos raster 3D, datos vectoriales 3D y datos puntuales 3D.

GRASS soporta la importación, exportación de los formatos más conocidos tanto raster como vectoriales.

2.2. INTERFAZ DE USUARIO

Aunque GRASS puede funcionar bajo sistemas operativos *Windows NT, 2000 o XP* (utilizando *cygwin*, que es un entorno Linux para Windows), para asegurarnos el correcto funcionamiento es aconsejable utilizar la versión para GNU/Linux.

Cada capacidad o funcionalidad de GRASS se materializa con el correspondiente comando en modo consola, lo que representa un handicap para el usuario final y más aún sino tiene conocimientos de LINUX. Aunque es cierto que se ha implementado un entorno de ventanas con TCL/TK no deja de ser una solución intermedia ya que no ofrece la potencialidad de utilizar un GUI (*Graphical User Interface*) como GNOME o KDE.

2.3. PROGRAMACIÓN Y CÓDIGO FUENTE

El código de GRASS está disponible a toda la comunidad en lenguaje C. Aunque el código está ampliamente documentado, presenta dificultades propias de organización y complejidad al no haber utilizado un lenguaje de programación orientado a objetos (POO), ya que el programador se encuentra con cientos de métodos con sus correspondientes argumentos y necesita un periodo largo de aprendizaje para iniciarse en la programación del entorno.

El código fuente se puede obtener actualizado a partir del repositorio CVS (*Concurrent Versions System*) que proporciona una red de trabajo transparente para el control del código fuente para los grupos de desarrolladores.

En la versión 5.7, aún en desarrollo, se introducen importantes mejoras como: una nueva librería de geometría vectorial, soporte de formatos externos sin necesidad de conversión, soporte de PostGIS, mejora de la indexación espacial, almacenamiento de los atributos en un SGBD, nuevo interfaz gráfico, etc.

3. JUMP

3.1. DESCRIPCIÓN Y CAPACIDADES

El proyecto JUMP (*Unified Mapping Platform*), es una aplicación basada en un interfaz gráfico de usuario (GUI) para visualizar y procesar datos espaciales. Incluye muchas de las funciones más comunes usadas en SIG. JUMP ha sido diseñado también como un entorno de trabajo extensible para el desarrollo y personalización de aplicaciones para el proceso de datos espaciales.

Es una colección de componentes Java, con las siguientes funcionalidades:

- Creación de clases para la representación espacial de los datos mediante el paquete de topología JTS:

JTS es una API que proporciona un modelo de objetos espaciales y funciones fundamentales geométricas. Esta API implementa el modelo de geometría definido por el OpenGIS Consortium en el apartado de especificaciones para elementos sencillos mediante SQL. A partir de la utilización de esta API, JUMP proporciona funciones de superposición (intersección, deferencia, unión, buffer), comprobación de validación de topología, funciones de área y distancia, etc. JTS está publicada bajo licencia GNU LGPL.
- El entorno de visualización para el desarrollo de aplicaciones JUMP, que proporciona el principal interfaz de usuario y API para crear aplicaciones extensibles con el fin de desarrollar y personalizar aplicaciones para el proceso espacial de datos.
- Clases de algoritmos para la manipulación de datos espaciales, integración y operaciones de correcciones topológicas (JCS Conflation Suite):

JCS ayuda a resolver los problemas ocasionados al combinar dos o más capas de datos espaciales que han sido obtenidas de diferentes fuentes. Un mismo elemento sobre el terreno puede tener diferente representación geométrica, atributos y errores posicionales asociados en función del origen de datos del que provenga. Mediante esta API se proporcionan técnicas para la integración de datos, aunque siguen siendo dependientes de una edición manual supervisada.

JUMP trabaja originalmente con los siguientes formatos: JCS GML, FME GML, Esri ShapeFile. Estos formatos pueden ser ampliados a partir de los correspondientes plugins.

3.2. INTERFAZ DE USUARIO

En cuanto al interfaz de usuario es mucho más amigable que el empleado por GRASS, y presenta ciertas semejanzas a productos comerciales mucho más utilizados como el software ArcView de ESRI.

Al estar programado en su totalidad en código 100% Java, es multiplataforma ejecutándose correctamente por ejemplo tanto en entornos LINUX como MS-Windows.

3.3. PROGRAMACIÓN Y CÓDIGO FUENTE

El código fuente de JUMP, así como el código de las librerías JTS y JCS en las que se apoya, están disponibles en lenguaje Java.

Todas las clases de Java están documentadas utilizando JavaDoc y CVS. Aunque no existe una documentación o ejemplos tan amplios como en GRASS, a partir del manual del desarrollador y la consulta de la documentación de las clases se puede empezar a desarrollar en un tiempo razonablemente corto.

El código fuente se encuentra accesible mediante la utilización del servicio CVS. La versión actual de JUMP es la 1.1.1, liberada en Diciembre de 2003.

4. POSTGIS

4.1. DESCRIPCIÓN Y CAPACIDADES

PostGIS añade el soporte para objetos geográficos a la base de datos objeto-relacional PostgreSQL. De esta forma podemos decir que PostGIS proporciona la capacidad espacial a un servidor PostgreSQL, permitiendo ser utilizado como un cliente GIS de la base de datos. Podemos decir que PostGIS más PostgreSQL forman una base de datos espacial donde se almacena y gestiona tanto la geometría de los elementos geográficos como los atributos temáticos de los mismos.

PostGIS permite al SGBD orientado a objetos PostgreSQL la gestión de objetos geográficos. PostGIS consigue que el servidor de bases de datos PostgreSQL pueda manejar objetos geográficos capacitándolo para funcionar como soporte de datos espaciales en un Sistema de Información Geográfica. En este sentido, PostGIS se encontraría al nivel de la base de datos SDE de ESRI o al de la versión de Oracle con su correspondiente extensión espacial.

PostGIS sigue la norma OpenGIS referente a bases de datos SQL conocida como *Simple Features Specifications for SQL*. PostGIS ha sido desarrollado por Refrations Research Inc como un proyecto de investigación en tecnologías de bases de datos en código abierto. PostGIS se distribuye bajo licencia GNU General Public License. Los desarrolladores no se comprometen al mantenimiento del producto por tiempo indefinido. La lista de proyectos futuros incluye el desarrollo de tecnologías para carga y descarga de datos, la creación de herramientas para el acceso y manipulación directa de las bases de datos, y el soporte para topologías avanzadas, como pueden ser coberturas, redes y superficies, en entornos de servidor.

Con PostGIS podemos utilizar todos los objetos que aparecen en las especificaciones OpenGIS, como por ejemplo puntos, líneas, polígonos, multilíneas, multipuntos y colecciones geométricas.

PostGIS utiliza el motor de indexación espacial GiST basado en índices espaciales R-Tree.

Por todo esto, PostGIS es la solución tecnológicamente hablando más avanzada expuesta en este artículo (velocidad, manejo de gran volumen de datos, información centralizada, etc.), aunque presenta grandes inconvenientes para el usuario, como las dificultades de instalación y sobre todo la inexistencia de un interfaz gráfico de usuario como incluyen GRASS o JUMP con más o menos éxito.

4.2. INTERFAZ DE USUARIO

Aunque PostGIS puede funcionar bajo sistemas operativos *Windows NT, 2000 o XP* (utilizando *cygwin*, que es un entorno *Linux* para *Windows*), para asegurarnos el correcto funcionamiento es aconsejable utilizar la versión para *GNU/Linux*.

Como se ha comentado, la filosofía de PostGIS se basa en la gestión de una base de datos espacial. Actualmente no existe un interfaz gráfico de usuario que proporcione un entorno amigable y fácil de utilizar como *JUMP* o en menor grado *GRASS*. Como ejemplo, para crear un nuevo elemento geográfico deberemos utilizar la correspondiente sentencia *SQL* introduciendo tanto la geometría como los atributos.

La solución pasa por utilizar otros software libres que proporcionen soporte para datos PostGIS:

GRASS soporta actualmente PostGIS como fuente de datos a través del driver *PostGRASS*. El visor *JUMP* tiene un plugin para leer datos PostGIS, y el software *QGIS* (<http://qgis.sourceforge.net/>) presenta un buen soporte para PostGIS.

4.3. PROGRAMACIÓN Y CÓDIGO FUENTE

Los datos PostGIS pueden ser exportados a otros formatos de salida utilizando la librería en *C++* del *OGR*. Por supuesto cualquier lenguaje que admita *PostgreSQL* puede trabajar con *PostGIS*, la lista incluye *Perl*, *PHP*, *Python*, *TCL*, *C*, *C++*, *Java*.

El código fuente de PostGIS esta disponible en lenguaje *C*. El código fuente se encuentra también accesible mediante la utilización del servicio *CVS*. Tanto la documentación del código fuente como los manuales para el desarrollador son bastante pobres.

La versión actual de PostGIS es la 0.8.2 con soporte para *PostgreSQL 7.5*.

5. OPENMAP

OpenMap es un kit de componentes basado en *Java Beans* para la construcción de aplicaciones y *applets* que utilicen información geográfica. *OpenMap* es pues un conjunto de componentes *Swing* de *Java* que “entienden” coordenadas geográficas. Mediante estos componentes podemos visualizar mapas y gestionar la interacción del usuario con eventos para manipular los datos espaciales.

OpenMap es un conjunto de herramientas para desarrolladores basado en librerías *JAVA* de código abierto. *OpenMap* está siendo desarrollado por *BBN Technologies* y se trata de software de

Código Abierto, no se trata de Software Libre con licencia GNU-GPL. Es un Software de Código Abierto con licencia propia. Esta licencia reúne la mayor parte de las características más importantes propias del Software Libre, sin embargo presenta algunas restricciones que no permiten clasificarlo como Software Libre.

OpenMap permite construir rápidamente aplicaciones y *applets* que son capaces de acceder a los datos contenidos en bases de datos y en otras aplicaciones. OpenMap proporciona al usuario la capacidad de ver y manipular información espacial. OpenMap está en constante desarrollo. La última versión es la 4.6, puesta a disposición del público en febrero de 2004. El código fuente, compatible con JDK 1.2, está disponible a través del servicio CVS.

Como ya se ha comentado, se puede obtener una copia de OpenMap completamente gratis a través su página web. Además de bajarlo gratuitamente, se puede utilizar sin recargo alguno e incluso realizar cambios en el código fuente, característica que lo aproxima mucho a un Software Libre con licencia propia.

Los beneficios económicos que obtiene BBN Technologies de OpenMap no llegan de vender licencias. En cambio BBN Technologies presta servicios de consulta, integración y desarrollo por los que si obtiene beneficios.

El hecho de que OpenMap sea Software de Código Abierto posibilita el que se puedan realizar mejoras en el código fuente y que estas sean enviadas a los creadores del programa para su inclusión en futuras versiones. En este contexto se encuentran disponibles para la comunidad de usuarios gran cantidad de contribuciones y mejoras a parte de la distribución oficial del producto.

6. MAPSERVER

Es una aplicación desarrollada para trabajar como servidor de cartografía, la cual corre bajo plataformas Linux/Apache, Windows/IIS. MapServer fue originalmente desarrollado por la Universidad de Minnesota (UMN).

El núcleo de MapServer es una aplicación CGI con la finalidad de crear contextos SIG dinámicos a través de Internet. El conjunto de aplicaciones que engloba MapServer permite crear aplicaciones independientes para crear mapas, escalas gráficas, leyendas, etc.

Constantemente se están ofreciendo nuevas versiones, pero a partir de la versión 3.5 soporta WMS (*Web Map Service*). Un WMS produce mapas de datos georreferenciados. Estos "mapas" son la representación visual de los geodatos, no los datos mismos, y pueden ser generados en formatos de imagen como PNG, GIF o JPEG, con el fin de ser utilizados como una fácil respuesta cuando la información sea requerida por un cliente, de esta manera se ofrecen más capacidades de utilización al servidor de mapas.

La información visualizada por el servidor son archivos shape, sólo las imágenes PNG, GIF, o JPEG se utilizan en el momento de hacer la conexión de un servidor cliente a un servidor Web, en el caso de querer establecer comunicación entre dos servidores.

Para la generación del interfaz, se puede utilizar programación HTML, y JavaScript, o PHP. MapServer utiliza un lenguaje propio que permite el despliegue de las capas geográficas, clasificación de la leyenda, simbología y componentes adicionales como colores, ancho de línea etc., cada uno de los componentes se debe definir en el archivo *.map*. Según la versión seleccionada, el

despliegue de los datos puede hacerse utilizando el protocolo OGC de servicios de mapas que permitirá la conexión fácilmente con otro servidor.

Añadiendo las librerías GDAL y OGR, MapServer es capaz de leer los formatos GIS más comunes. Cuando GDAL se compila contra *libgrass*, MapServer lee directamente datos raster desde una localización de GRASS a través de GDAL. En futuras versiones de *libgrass* los datos puntuales y vectoriales también serán soportados.

Es posible crear potentes sistemas utilizando únicamente herramientas libres como Apache, PHP, GDAL/OGR, MapServer, PROJ4, *libgrass*, GRASS o PostgreSQL.

7. CONCLUSIONES

Como menciona el título de la comunicación las soluciones propuestas en este artículo son potenciales, es decir, todos los programas descritos requieren de una personalización bastante compleja, ya que la finalidad es crear un software libre para la gestión de ayuntamientos medianos y pequeños cubriendo todas sus necesidades según se han definido en el proyecto SIGMUN.

Debido a la falta de recursos del citado proyecto no es viable construir una aplicación utilizando la solución aportada en el bloque D (la utilización de librerías como OpenMap), ya que aunque aporta mucha libertad al programador habría que desarrollar la aplicación prácticamente desde cero, es decir, crear todo el interfaz gráfico de usuario, gestores de capas, leyendas, maquetación, etc.

Por otra parte la publicación de contenidos en Internet (bloque E), si que es totalmente posible utilizando un servidor de cartografía como MapServer y algún lenguaje de personalización como PHP, aunque esta opción es meramente un complemento al programa de SIG que en realidad se está buscando.

Por lo tanto y teniendo en cuenta las ventajas e inconvenientes de los programas analizados de los bloques A, B y C, y teniendo en cuenta siempre los recursos disponibles de SIGMUN podemos llegar a los siguientes razonamientos:

Descartamos de momento el bloque A con el programa GRASS, aunque es el SIG más completo libre que existe actualmente en el mercado, hasta que se implemente un interfaz gráfico de usuario mucho más amigable y el modelo vectorial completo de forma estable (versión 6), no nos ofrecerá los requisitos necesarios.

JUMP presenta una buena solución siempre que no queramos manejar información centralizada y/o con un volumen de datos excesivo. Presenta el interfaz gráfico más cercano a otros programas comerciales ampliamente utilizados. Aunque por supuesto habría que implementar todas las personalizaciones necesarias para cubrir las necesidades de los ayuntamientos concluimos que sería una buena solución para ayuntamientos pequeños.

En cuanto a ayuntamientos medianos con un volumen de información más elevado e incluso con la cartografía y los datos temáticos centralizados, la solución pasaría por utilizar PostGIS y PostgreSQL como SGBD. El gran inconveniente recae en la construcción de un interfaz gráfico o el apoyo en otros programas para visualizar y manejar las capas de PostGIS.

Como hemos comentado anteriormente JUMP posee un *plugin* para leer datos PostGIS, con lo cual la solución idónea sería unir las características de JUMP y PostGIS con este *plugin*

(ampliando sus funcionalidades si fuera necesario), lo que brindaría a la aplicación desarrollada una interfaz gráfica de usuario atractiva. De esta manera siempre podríamos aprovechar las funcionalidades de PostGIS como los análisis espaciales utilizando sentencias directas de PostGIS. Aún así las labores a realizar para cubrir las necesidades de los ayuntamientos se presentan como un trabajo complejo.

Todos los integrantes del proyecto de investigación SIGMUN estamos muy esperanzados en poder presentar dentro de los dos próximos años una alternativa de software libre que resuelva los problemas de gestión municipal en aquellos organismos que sus recursos económicos no hacen viable el utilizar otros programas SIG comerciales.

GRASS	JUMP	PostGIS
VENTAJAS		
Amplias funcionalidades Producto altamente consolidado Código fuente bien documentado Muchos formatos reconocidos Documentación	Programación orientada a objeto Interfaz de usuario atractivo y fácil de utilizar Extensible fácilmente mediante plugins	Base de datos espacial Gran velocidad y manejo de grandes volúmenes de datos.
INCOVENIENTES		
Interfaz gráfico de usuario pobre Modo consola Programado sin POO Proceso de aprendizaje largo Versión 5.3 aún no estable, versión esperada 6.0	Producto joven, poco consolidado Dificultad en el manejo de grandes volúmenes de datos	Se necesita visores o editores gráficos externos. Documentación

AGRADECIMIENTOS

Este artículo y los estudios de propuestas que actualmente se están estudiando son consecuencia directa del trabajo realizado en el Proyecto de investigación SIGMUN, Proyecto BIA2003-07914 financiado por el Ministerio de Ciencia y Tecnología de España y los Fondos FEDER. Queremos agradecer también la colaboración de José Morell Rama alumno de la titulación de Ingeniero en Geodesia y Cartografía de la ETSIGCT de la Universidad Politécnica de Valencia.

BIBLIOGRAFÍA

- Christopher Negus. La biblia de Red Hat Linux 7. Editorial Anaya Multimedia, Barcelona, 2000.
- Daniel Gómez Castro, Jose Carlos Martinez Llarío y Maria A. Brovelli. Progetto Issola: Costruzione delle Schede di Valutazione Ambientale nel GIS GRASS. Politecnico di Milano. Facoltà di Ingegneria di Como, 2001.
- Eloína Coll Aliaga, David Peñaranda Rod y Alberto García González. Sistemas de Información Geográfica y Urbanismo (SIG III). Ed. Servicio de Publicaciones de la Universidad Politécnica de Valencia, 2000.
- Federal Geographic Data Committee (FGDC), <http://www.fgdc.gov>, accedida en febrero 2004.
- Internacional Organization for Standardization (ISO), <http://www.iso.org>, accedida en marzo 2004.
- Open Gis Consortium (OGC), <http://www.opengis.org>, accedida en febrero 2004.

- Markus Neteler, Helena Mitasova. OPEN SOURCE GIS. A GRASS GIS Approach. Kluwer Academic Publishers. Dordrecht, The Netherlands, 2002.
- Moritz Lennert. Grass Tutorial. Copyright © 2003 GRASS Development Team.
- Software GRASS, <http://grass.itc.it/index.html>, accedida en Junio de 2004
- Software JUMP, <http://www.jump-project.org/>, accedida en Junio de 2004
- Software MAPSERVER, <http://mapserver.gis.umn.edu/index.html>, accedida en Junio de 2004
- Software OPENMAP, <http://openmap.bbn.com/>, accedida en Junio de 2004
- Software PostGIS, <http://postgis.refractions.net/>, accedida en Junio de 2004
- Structured Query Language (SQL), <http://www.sql.org>, accedida en diciembre 2003.
- World Wide Web Consortium (W3C), <http://www.w3c.org>, accedida en enero 2004.