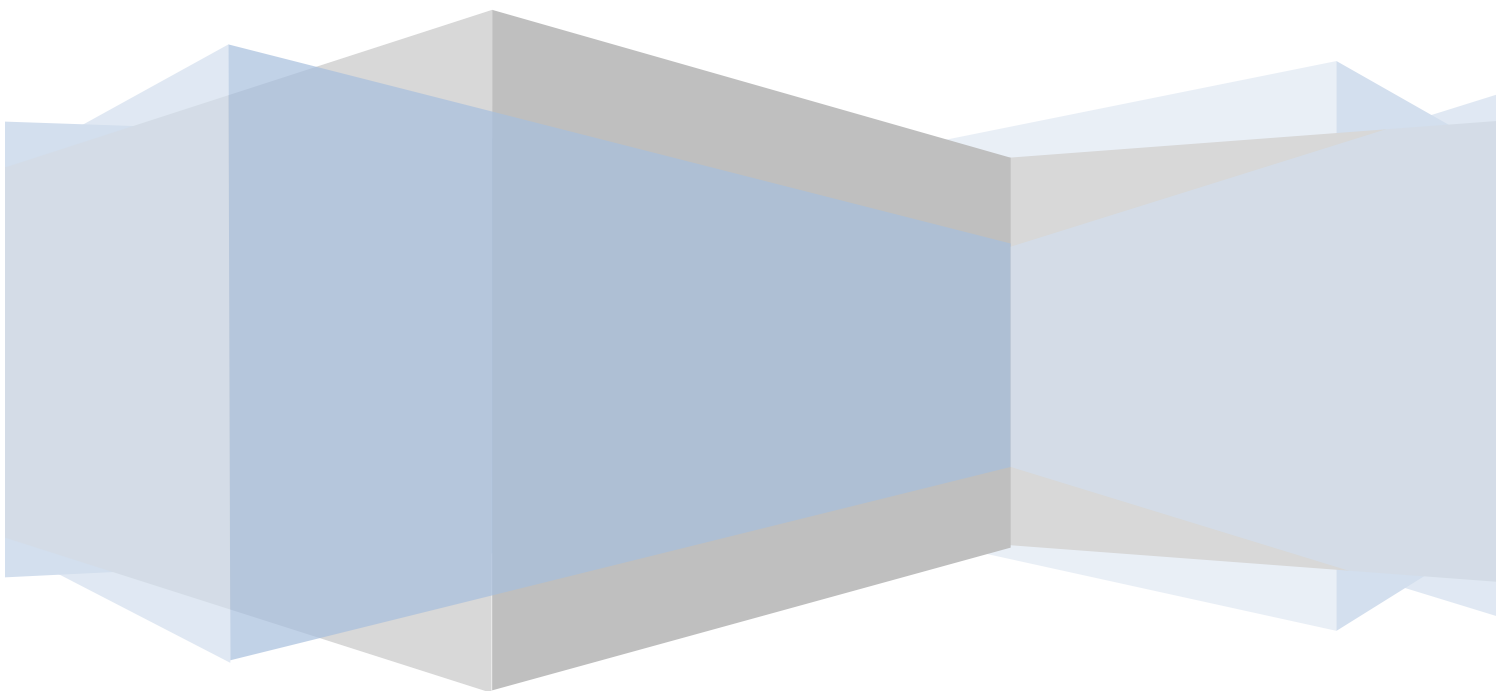


Universidad de Zaragoza

Trabajo 1:

Elección de un sistema gestor de bases de datos

Adrian Casans (590114) Sergio Pedrero (627669) Diego Sánchez (628279) Cristian Simón (611487)



Índice

1. Organización del trabajo:	3
2. Objetivo y finalidad del proyecto:	4
3. Introducción:	5
4. Bases de datos espaciales:	6
4.1 Introducción	6
4.2 Tipo de datos espaciales:	6
4.3 SIG:	7
5. Análisis de los posibles SGBD:	9
5.1 Oracle Spatial 11g	9
5.2 PostgreSQL + PostGIS	10
5.3 MS SQL server	11
5.4 MySQL	12
5.5 SpatialLite	12
5.6 GeoSpark	12
5.7 MongoDB:	13
6. Presupuesto:	15
Oracle Spatial 11g	15
PostgreSQL + PostGIS	15
MS SQL server	15
MySQL	16
SpatialLite	16
GeoSpark	16
MongoDB	17
7. Conclusiones:	17
Oracle Spatial 11G	18
Postgre + PostGIS	18
MS SQL server	18
MySQL	18
SpatialLite	18
Geospark	19
MongoDB	19
8. Problemas y soluciones:	19
9. Bibliografía:	20

1. Organización del trabajo:

Tarea	Adrián	Sergio	Diego	Cristian
Búsqueda de información	2 horas	1 hora	1 hora	2 horas
Objetivos y finalidad proyecto	1 hora			0,25 horas
Introducción (experiencias previas, recursos humanos,.....)			0,25 horas	0,25 horas
Bases de datos espaciales	0,5 horas	0,5 horas		1,5 horas
Análisis de los posibles SGBD				2,5 horas
Presupuestos			4 horas	
Conclusiones		2 horas		
Problemas y soluciones		0,5 horas		
POWER POINT	1 hora	0,5 horas		

2. Objetivo y finalidad del proyecto:

El objetivo de este proyecto es la recopilación y procesado de datos de mapas de carreteras que posteriormente, tras haber analizado también todos los posibles sistemas gestores de bases de datos y decidir cual es el más adecuado para este tipo de datos. Una vez conseguidas ambas cosas se intentará vender dicho sistema gestor de base de datos con todos los datos incluidos a las empresas que fabrican dispositivos GPS.

Para trabajar con datos vectoriales necesitaremos un sistema gestor de bases de datos que les de soporte. Además, parte del presupuesto irá destinado a una serie de consultorías con el fin de ayudar al cliente a su configuración y puesta a punto del sistema.

La finalidad de este proyecto como proponemos en el objetivo será seleccionar el sistema gestor de base de datos que mejor se adapte a nuestra finalidad de almacenar datos de mapas de carreteras. Se buscará cual es el que mejor se adapta en cuanto a características de almacenaje, como a coste económico del mismo.

3. Introducción:

ZGeoZ es una joven empresa dedicada principalmente al desarrollo software. Entre nuestros clientes se encuentran varios fabricantes de dispositivos GPS, a los que actualmente se les provee únicamente de soluciones software.

Desde ZGeoZ hemos visto la oportunidad de crear una nueva división en la empresa que se encargue de recopilar y procesar los datos de carreteras para poder proporcionárselos a nuestros clientes a un precio competitivo.

En un principio, para la consecución del proyecto, se necesitará un equipo que se encargue del tratamiento de los datos y un sistema gestor de bases de datos que de soporte a los datos geoespaciales con lo que se va a trabajar. Respecto al equipo se tiene presupuesto para contratar 1 persona que tenga conocimientos avanzados del nuevo sistema gestor, además del equipo actual compuesto por 2 personas que se encargan de la administración de las bases de datos actuales, las cuales son MySQL.

Se calcula que en un principio el SGBD necesitara 4 cores, por el volumen de clientes con los que trabajamos y de los diferentes países que son.

La dirección de la empresa ha informado que para este proyecto se dispone de un presupuesto de 20.000 €, sin contar la incorporación del experto en el nuevo SGBD.

4. Bases de datos espaciales:

4.1 Introducción

Una base de datos espacial es una base de datos que maneja datos existentes en el espacio o datos espaciales. En este tipo de bases de datos es imprescindible establecer un cuadro de referencia (SRE) para definir la localización y relación entre objetos. Estos pueden ser de dos tipos:

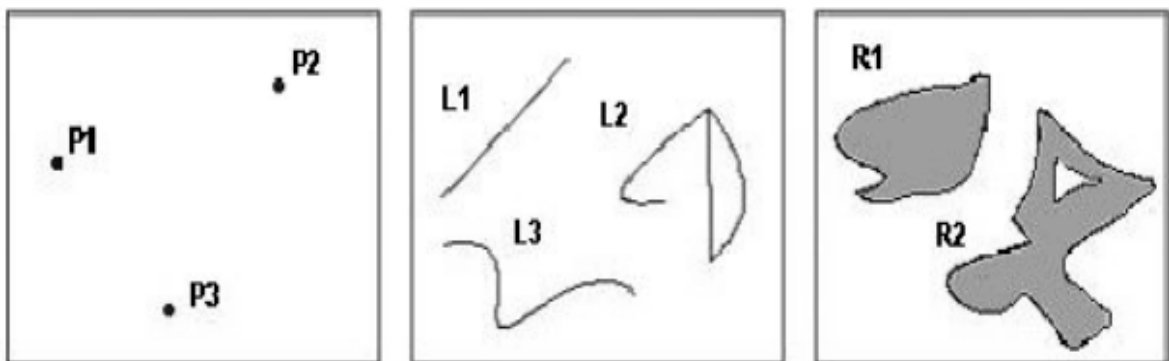
- **Georreferenciados:** Aquellos que se establecen sobre la superficie terrestre. Son los que normalmente se utilizan, ya que es un dominio manipulable, perceptible y que sirve de referencia.
- **No georreferenciados:** Son sistemas que tienen valor físico, pero que pueden ser útiles en determinadas situaciones.

4.2 Tipo de datos espaciales:

Un dato espacial es una variable asociada a una localización del espacio. Normalmente se utilizan datos vectoriales, los cuales pueden ser expresados mediante tres tipos de objetos espaciales.

Se dividen en:

- **Puntos:** Representa un objeto del cual solo interesa conocer su localización en el espacio. Se representa mediante coordenadas (x, y). Ciudades, accidentes geográficos, ...
- **Líneas:** Secuencia de puntos y por tanto, una secuencia de coordenadas (x, y), normalmente representa conexiones en el espacio. Carreteras, ríos, ...
- **Polígonos o regiones:** Figuras planas conectadas por distintas líneas u objetos cerrados que cubren un área determinada. Países, regiones, ...



4.3 SIG:

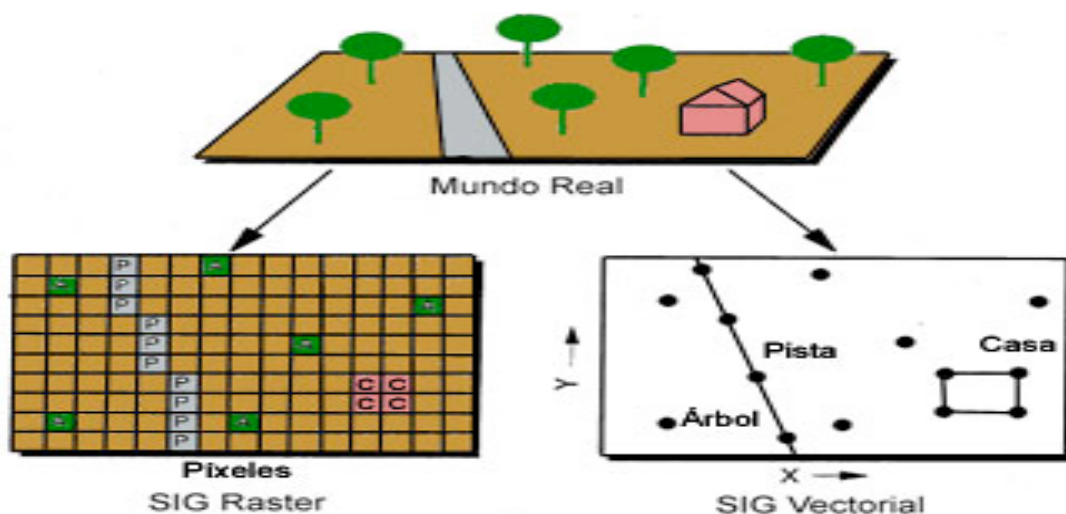
El software que trabaja con este tipo de datos es el conocido como SIG (Sistema de Información Geográfica). Este software es una herramienta que permite a los usuarios crear consultas interactivas, analizar información espacial, editar datos, mapas y presentar datos de todas estas operaciones.

Las principales cuestiones que puede resolver un sistema de información geográfica son:

- **Localización:** preguntar por las características de un lugar concreto.
- **Condición:** el cumplimiento o no de unas condiciones impuestas al sistema.
- **Tendencia:** comparación entre situaciones temporales o espaciales distintas de alguna característica.
- **Rutas:** calculo de rutas óptimas entre dos o más puntos.
- **Pautas:** detección de pautas espaciales.
- **Modelos:** generación de modelos a partir de fenómenos o actuaciones simuladas.

Los datos SIG representan objetos del mundo real (carreteras, el uso del suelo, altitudes,...). Este tipo de dato se puede dividir en dos tipos de datos fundamentales:

- **Raster:** Es, en esencia, cualquier tipo de imagen representada en mallas. Este modelo se centra en las propiedades del espacio más que en la precisión de la localización. Divide el espacio en celdas regulares donde cada una de ellas representa un único valor. Este tipo de datos es útil cuando no se quiere acudir a las propiedades topológicas del entorno.
- **Vectorial:** Se centra en la precisión de la localización de los elementos geográficos sobre el espacio y donde los fenómenos a representar tienen los límites definidos.



En las siguientes tablas se pueden ver algunas de las ventajas y los inconvenientes de cada uno de los tipos de datos:

- Ventajas:

Vectorial	Raster
Estructura de los datos compacta (Solo almacena los datos de los elementos digitalizados).	Estructura de datos simple.
Buena salida grafica.	Buen almacenamiento de imágenes digitales
Mejor mantenimiento y actualización.	Optimo para variaciones altas de datos.
Permite mayor capacidad de análisis.	Operaciones de superposición muy sencillas.

- Inconvenientes:

Vectorial	Raster
Estructura de datos compleja.	Requiere una mayor memoria de almacenamiento (Todas las celdas contienen datos).
Operaciones de superposición difíciles de implementar y representar.	Las reglas topológicas son mas difíciles de generar.
Poca eficiencia con altas variaciones de datos.	Salidas graficas menos vistosas.

Tras consultar con nuestros clientes en qué tipo de datos estarían interesados, todos han coincidido en que necesitan usar el **tipo de dato vectorial**, ya que posee un mayor grado de precisión y una mejor salida grafica.

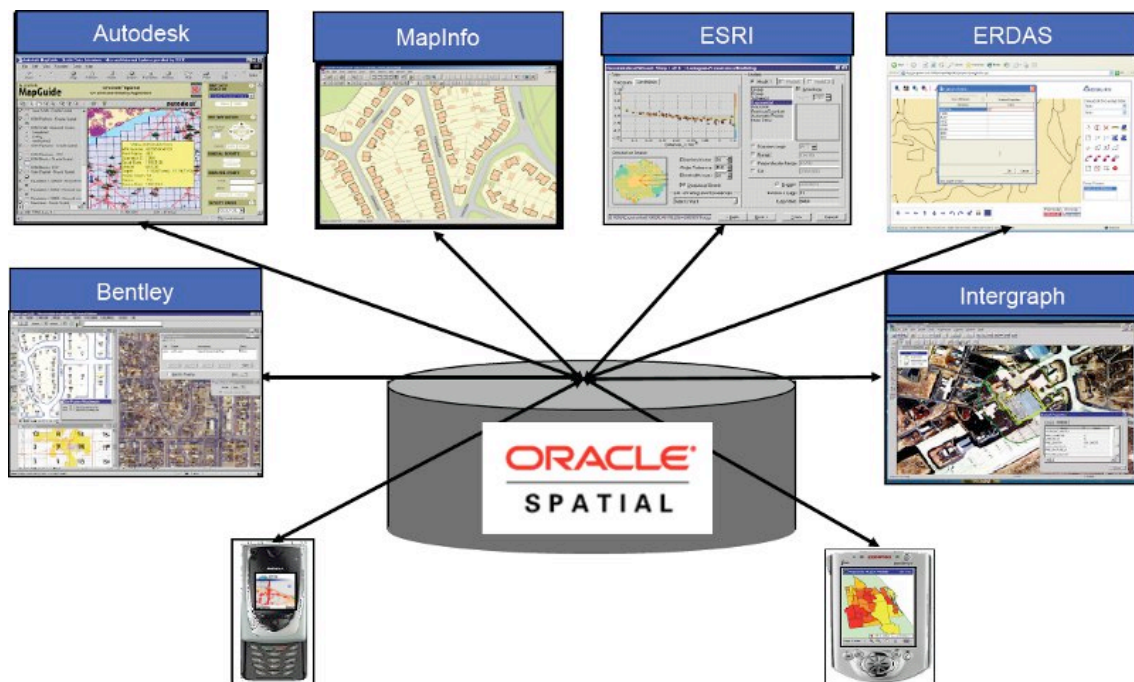
5. Análisis de los posibles SGBD

Para decidir el sistema gestor de base de datos que más se adapte a nuestros objetivos, es decir, que pueda trabajar con datos vectoriales se han analizado los siguientes SGDB:

5.1 Oracle Spatial 11g

Oracle Spatial es una opción dentro de Oracle Enterprise Edition que ofrece una gestión de bases de datos objeto-relacionales espaciales que permite el análisis detallado de los datos que se almacenan. Incluye numerosos módulos de topología, infraestructura para modelado y visualización en 3D, triangulado de datos, etc.

Soporta tanto la representación de datos de tipo raster, tipo vector, o ambos. Está diseñada para satisfacer las necesidades de los sistemas de información geográfica, siendo soportado por los distribuidores líderes en datos y herramientas espaciales, como NAVTEQ, Autodesk, Bentley, Digital Globe, entre otros.



Las licencias de este modulo van de 1 a 5 años, o con una opción de licencia permanente. El coste de las licencias es:

- **Permanente:** es de 16.500€ por el procesador y 330€ por cada usuario.
- **Anual:** es de 3.300€ por el procesador y 66€ por cada usuario.

Por otra parte, las versiones Oracle Standard Edition y Oracle Standard Edition One incluyen el modulo **Oracle Locator** sin coste adicional, el cual incluye alguna funcionalidad del modulo Spatial.

5.2 PostgreSQL + PostGIS

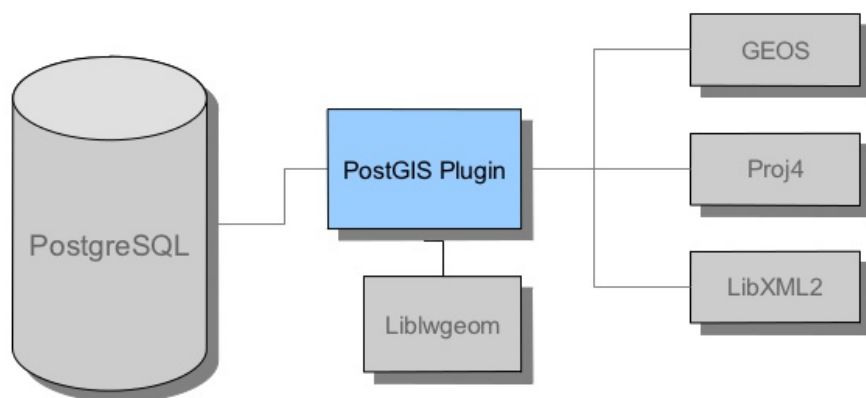
PostGIS es una extensión que convierte el sistema de base de datos PostgreSQL en una base de datos espacial, convirtiéndose, una vez juntos, en una solución perfecta para el almacenamiento, gestión y mantenimiento de datos espaciales.

Debido a que está construido sobre PostgreSQL, PostGIS hereda automáticamente sus características, así como los estándares abiertos, contando con características como:

- Se trata de un software libre.
- Es compatible con los estándares de Open Geospatial Consortium, lo que garantiza la interoperabilidad con otros sistemas también interoperables.
- Soporta tipos de datos espaciales, índices espaciales y tiene cientos de funciones espaciales.
- Permite importar y exportar datos fácilmente a través de varias herramientas.
- Compatible con un gran número de clientes SIG de escritorio y servidores de mapas web.
- Es actualmente la base de datos espacial de código abierto más utilizada.

Soporta datos de tipo raster y de tipo vector. Permite acceder a la base de datos desde varios lenguajes de programación (Java, C, PHP, Python,...). Además incluye funciones para el soporte 3D y ofrece la posibilidad de exportar datos compatibles con dispositivos GPS.

PostGIS Architecture: PostGIS



**The PostGIS spatial extension,
Provide spatial types and functions.**



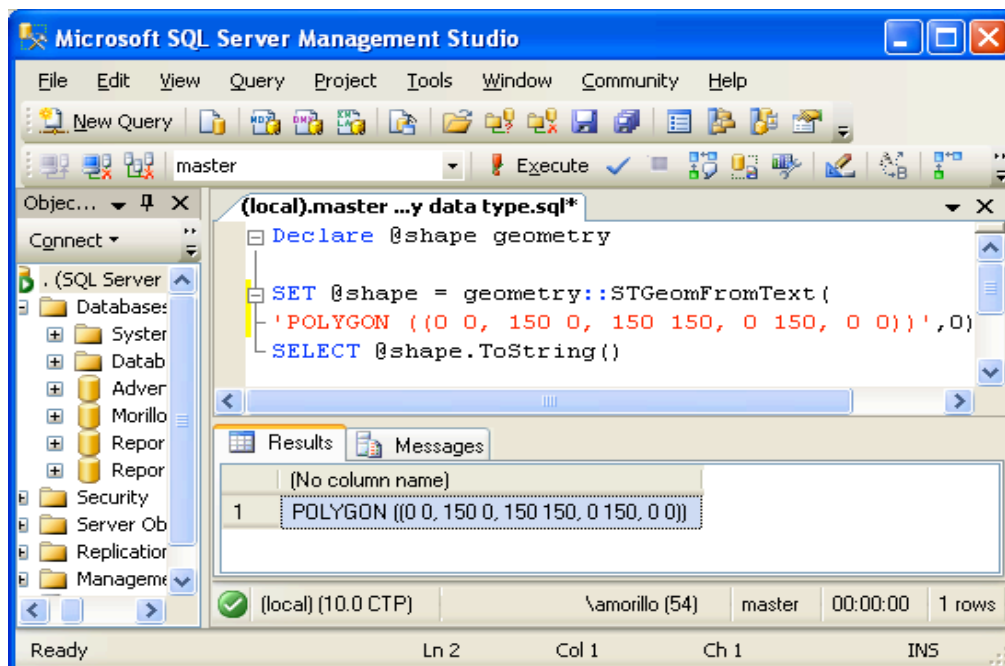
Tiene licencia pública general GNU. **Open source.**

5.3 MS SQL server

Microsoft incluye en las características de SQL Server el manejo de datos espaciales. SQL Server admite dos tipos de datos espaciales:

- El tipo **geometry**: representa los datos en un sistema de coordenadas euclidiano (plano). Suele ser suficiente para representar datos que provengan de proyecciones cartográficas o que abarquen áreas geográficas reducidas, donde las distorsiones ocasionadas al proyectar la superficie terrestre sobre un plano no son un factor crítico.
- El tipo **geography**: representa los datos en un sistema de coordenadas de tierra redonda. Ofrece mayor precisión a la hora de realizar operaciones sobre áreas de gran tamaño, ya que tiene en cuenta parámetros como la curvatura de la tierra a la hora de calcular la distancia entre dos puntos o el área en km² de una región.

Ambos tipos de datos se implementan como tipos de datos .NET Common Language Runtime (CLR) en SQL Server. Ofrece la posibilidad de importar y exportar datos compatibles con el Open Geospatial Consortium.



Las licencias de SQL Server varían de precio en función del número de cores que tengamos:

- **Web Edition:** 23€/mes con 4 cores el más barato y 92€/mes con 16 cores el más caro.
- **Standard Edition:** 320€/mes con 4 cores el más barato y 1280€/mes con 16 cores el más caro.
-

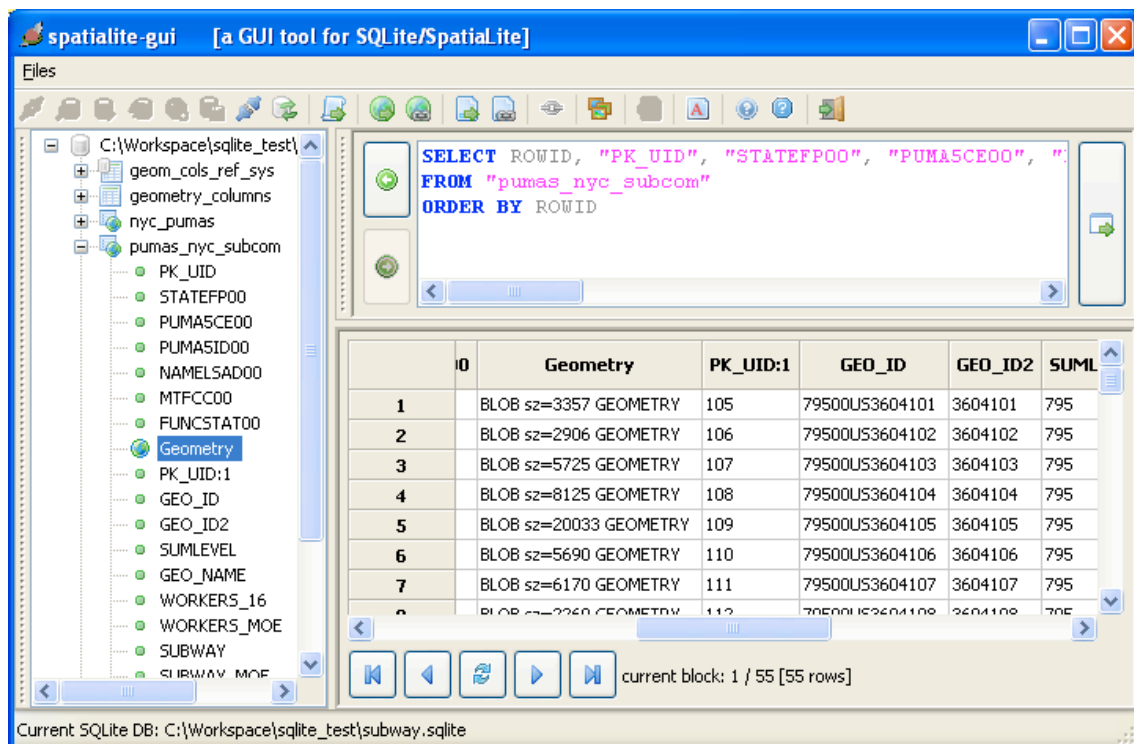
5.4 MySQL

MySQL introduce las extensiones espaciales para permitir la generación, almacenamiento y análisis de elementos geográficos. Sigue las especificaciones del Open Geospatial Consortium y es **open source**.

5.5 SpatialLite

Se trata de un motor de bases de datos SQLite al que se le han agregado funciones espaciales. Provee de una funcionalidad de base de datos espacial mediante el soporte para vectores.

Se trata de un sistema similar a PostgreSQL o MySQL que destaca sobre estos por ser un archivo reducido ~500 KB, muy flexible, portable y funcional desarrollado en C, y que no necesita previa configuración de un administrador de base de datos. El conjunto de base de datos es almacenada en un fichero estándar de extensión SQLite.

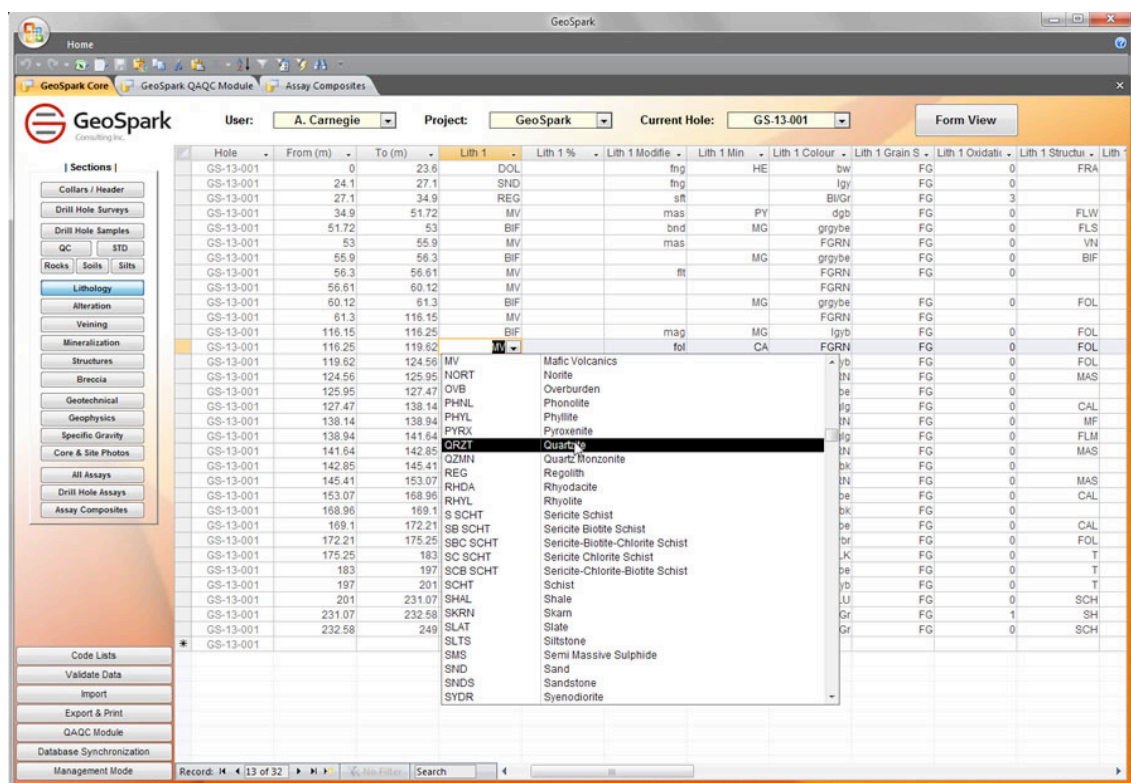


Sigue las especificaciones del Open Geospatial Consortium y es **open source**.

5.6 GeoSpark

Sistema que permite la posibilidad de trabajar con una base de datos relacional que puede estar alojada o en la nube. Garantiza la disponibilidad total de los datos y la seguridad de los mismo realizando copias de seguridad periódicamente.

Ofrece entornos de modelado en 3D y la exportación a dispositivos GPS.



Las licencias de este sistema son de pago y tenemos diferentes paquetes y módulos.

El paquete básico "Core", que incluye el software del servidor, clientes y algunos módulos cuesta 30€/mes.

Además, existen paquetes auxiliares (utilidades de seguimiento, herramientas de cálculo y gestión, análisis de datos, ...) que cuestan de 30€ a 60€ por mes y maquina.

Con todo esto, el precio anual para una maquina, incluyendo paquete básico y los módulos necesarios rondaría los 2.000€.

5.7 MongoDB:

El formato **GeoJSON** es un formato estándar abierto diseñado para representar elementos geográficos sencillos. Es un formato ampliamente utilizado en aplicaciones de cartografía en entornos web al permitir el intercambio de datos de manera rápida, ligera y sencilla.

El formato GeoJSON difiere de otros estándares SIG en que no está desarrollado y mantenido por la organización oficial, sino que es mantenido por una comunidad de desarrolladores en internet.

MongoDB se trata de una base de datos no sql que almacena objetos JSON, y que gracias a la aparición del índice 2dsphere, también objetos GeoJSON. MongoDB ofrece dos "superficies" sobre las que guardar datos espaciales (datos de tipo vectorial 2D):

- Esferico (Spherical): coordenadas esféricas (lon,lat), para puntos, líneas y polígonos.
- Plano (Flat): coordenadas planas (x,y), solo sirve para puntos.

Si los datos están sobre una superficie esférica se usa el índice 2dsphere. El formato es objetos GeoJSON con coordenadas de longitud, latitud y usando como sistema de referencia el WGS84. El 2dsphere permite almacenar y realizar consultas sobre puntos, líneas y polígonos. La diferencia con el índice 2d es que este permite hacer consultas a las geometrías sobre una esfera y no sobre un plano. El índice 2dsphere soporta tanto GeoJSON (lon, lat) como coordenadas planas (X, Y).

```
LINEA:
{
  geo: {
    type: "LineString",
    coordinates: [ [100.0, 0.0], [101.0, 1.0] ]
  }
}

POLIGONO:
{
  geo: {
    type: "Polygon",
    coordinates: [
      [ [100.0, 0.0], [101.0, 0.0],
        [101.0, 1.0], [100.0, 1.0],
        [100.0, 0.0] ]
    ]
  }
}
```

Se trata de un sistema de código abierto, basado en la arquitectura cliente-servidor y es capaz de funcionar en una gran variedad de sistemas operativos.

6. Presupuesto:

Oracle Spatial 11g

Oracle ofrece el producto de 2 formas diferentes, por usuario, que cobra 61€ al año por usuario y te pide un mínimo de 25 usuarios. O por procesador, que cobra 3039 € por procesador al año, en este caso como el requisito son procesadores, se ha realizado el presupuesto con 4 procesadores. Además el primer año piden un pago extra por la licencia y el soporte que asciende a 3342 € por procesador. Aunque aplican un pequeño descuento como se puede observar en la siguiente tabla.

Producto	Precio Unidad	Descuento	Cantidad	Precio Total
Oracle Spatial and Graph (1 Año)	3,039.00 €	2,887.05 €	4	11,548.20 €
Software Update License and Support (Primer año)	3,342.57 €	3,175.44 €		12,701.76 €

PRECIO TOTAL: 24.249,96 €

PostgreSQL + PostGIS

Es un producto Open source, pero es necesario dar formación a los trabajadores y contratar un soporte, ya que a veces pueden surgir problemas que si no se solucionan rápidamente pueden generar graves pérdidas. La formación para cada empleado es de 250 Libras más el viaje e instancia de 2 noches en Reino Unido. Por lo que la cifra asciende a unos 2500 €.

Respecto al soporte, se ha encontrado un buen soporte por 4320 Libras al año, 5.473,31 €.

PRECIO TOTAL: 7.973,31 €

Fuente del soporte:

<http://www.thinkwhere.com/files/8914/3023/3293/OpenSourceTechnicalSupportPlans.pdf>

MS SQL server

Se cree que la versión más conveniente es la Standard, la cual ofrece funcionalidad de Business Intelligence y administración de datos básica para cargas de trabajo que no son críticas, con recursos de TI mínimos.

El precio es de 3,717 € al año, además se deberán contratar 2 paquetes de 2 núcleos cada uno, que aunque no se sabe exactamente ya que se está a la espera de contestación de Microsoft, se calcula de que cada paquete oscila entre 6.000 € y 8.000€.

PRECIO TOTAL: 15.717 € - 19.717 €

MySQL

Es un producto Open Source, por lo que no hay que pagar licencias, además es el mismo SGBD que se está usando actualmente, aunque con respecto a los datos espaciales el equipo no está muy especializado y ofrece pocas alternativas para el manejo.

Respecto al soporte, se ha encontrado un buen soporte por 9.473,31 €, además se tendría que tener en cuenta la formación de los empleados que saldría por muy poco dinero ya que no hemos podido encontrar ningún curso pero teniendo en cuenta por ejemplo lo que costaría una formación interna podrían ser un total de unos 1000€-2000€ y poder exprimir las pocas funciones espaciales que tiene este gestor.

PRECIO TOTAL: 10.000 a 12.000 €

SpatialLite

También es un producto Open Source similar a PostgreSQL y MySQL, con el problema de que el equipo no ha trabajado nunca con este SGBD, por lo que conllevaría pagar algún curso al equipo.

Se ha encontrado soporte el cual incluye soporte técnico, correo electrónico de alta prioridad y soporte telefónico directamente de los desarrolladores de SQLite el cual costaría entre 8.000 € y 35.000 € anuales. Aunque hay servicios más baratos (1.500 € al año) se cree que no son suficiente dada el desconocimiento de este gestor por el equipo.

La empresa que daría el soporte es:

<http://www.hwaci.com/sw/sqlite/prosupport.html>

Respecto a cursos para el equipo no se ha encontrado ningún curso para es gestor de bases de datos.

PRECIO TOTAL = 8.000 € a 35.000€

GeoSpark

Este SGBD cuesta 39\$ al mes por cada core, el cual cuenta con los 3 módulos siguiente: QAQC Module, Strip Log Module y Drill Log Module. Además de una herramienta de pruebas, sincronización de la base de datos y el modo de administración.

Se han buscado cursos de formación para el equipo pero no se ha encontrado ninguno

PRECIO TOTAL = 1645 € al año.

MongoDB

Este sistema gestor de bases de datos también es de código abierto. Además al ser NoSQL nuestro equipo tendrá que formarse y se deberá contratar un soporte.

Cada core costaría 109,872 € al mes, por lo tanto los 4 que se estima que se van a necesitar saldrían por 5.273,856 € al año.

Además MongoDB ofrece una gran variedad de cursos online gratuitos (<https://university.mongodb.com>).

PRECIO TOTAL = 5.273,856 €

7. Conclusiones:

Analizando todos los gestores anteriores se han abstraído las siguientes conclusiones para cada uno de ellos :

- **Oracle Spatial 11G**

Es un SGBD es un SGBD muy completo ya que es muy potente y apto para grandes organizaciones. Su uso está muy extendido ya que ofrece experiencia y garantía de calidad. Su precio representa el principal problema a la hora de escogerlo ya que además de la licencia base del gestor, es necesaria una licencia adicional para el módulo "Spatial". Afortunadamente la empresa ya dispone de la licencia base, pero aún así el precio del módulo es bastante elevado.

- **Postgre + PostGIS**

Es la opción mas completa al lado de Oracle, con la gran ventaja de que es de libre distribución , es decir, sin coste de licencias. Ofrece herramientas muy potentes y múltiples utilidades para el tratamiento de datos espaciales. Es importante tener en cuenta que el soporte ofrecido puede no ser el adecuado para un entorno empresarial , por lo que la parte del presupuesto dedicada a licencias podría ser destinada a consultoría con el objetivo de cubrir esta carencia. Sin embargo, como ya se tiene experiencia previa con PostgreSQL lo que facilitaría el aprendizaje del módulo PostGIS y funciones mas avanzadas.

- **MS SQL server**

Es una opción simple y poco extendida, pero que está integrada en el propio gestor. Las funciones espaciales son muy limitadas en comparación con las otras alternativas.

- **MySQL**

Es un gestor que también es de libre distribución al igual que PostGIS. Incluye gran cantidad de controladores específicos para componentes espaciales, una gran ventaja para manejar este tipo de datos. Sin embargo, no es tan completo en funciones espaciales como PostGIS otra alternativa a tener en cuenta por el precio. El problema radica en el tipo de aplicación que se quiera desarrollar. Si se van a hacer aplicaciones donde se necesitan cosas muy básicas y puntuales MySQL puede ser una buena alternativa. Si se necesitan funciones mas especificas se utilizaría PostGIS.

- **SpatialLite**

Es una alternativa más orientada a experimentación y muy eficaz para entornos muy pequeños y con presupuestos muy ajustados. Quizás demasiado pequeños comparados con nuestro entorno.

- **Geospark**

Es un gestor muy completo que ofrece múltiples opciones y paquetes con infinidad de utilidades. Aparentemente es una elección poco conocida y extendida, y parece estar muy enfocada a un determinado tipo de sistemas. Se trata de una solución algo cara en entornos grandes con múltiples usuarios, sobre todo si se tiene en cuenta con perspectiva a varios años de desarrollo. De todos modos, no está claro el soporte que ofrece para datos vectoriales.

- **MongoDB**

En cuanto a escalabilidad es la opción prioritaria ya que es el único gestor NoSQL. Sin embargo, sus opciones espaciales son muy limitadas por lo que se desecha esta opción.

Teniendo en cuenta todos los datos anteriores nuestra empresa se decantaría por **“PostgreSQL + PostGIS”** ya que es la solución que mejor se adapta a nuestro contexto. Nuestra decisión se ha decantado de este lado por aspectos tan importantes como el planificar el coste de formación y el coste de soporte frente a problemas que puedan surgir en el desarrollo de implantación del gestor. Además como se ha comentado en el apartado anterior provee de multitud de herramientas y alternativas para bases de datos espaciales y el coste en licencias de uso es nulo.

8. Problemas y soluciones:

El principal problema ha sido el no poder probar todos y cada uno de los gestores de una forma mas exhaustiva para poder entender mejor todas las opciones que nos ofrecen.

Ya que se trata de gestores para tratar con datos espaciales y ser un campo en el que nunca habíamos investigado previamente concluimos con que el trabajo nos ha servido para aprender que tipos de gestores existen para manejar este tipo de información y como funcionan, además de ayudar a hacernos una idea general del presupuesto necesario para tener una aproximación cercana a lo que podría surgirnos en un entorno de trabajo como futuros ingenieros informáticos en una empresa.

9. Bibliografía:

https://es.wikipedia.org/wiki/Sistema_de_informaci%C3%B3n_geogr%C3%A1fica

https://um.es/geograf/sigmur/sigpdf/temario_9.pdf

<https://docencia.lbd.udc.es/six/teoria/1.-GIS-Introduccion.pdf>

https://rree.go.cr/file-db.php?id_file=3197

<https://mappinggis.com/2012/09/por-que-utilizar-postgis>

<https://postgis.net>

<https://msdn.microsoft.com/es-es/es/library/bb933790.aspx>

https://ovh.es/servidores_dedicados/precios-licencias-windows-2014.xml

<https://dev.mysql.com/doc/refman/5.7/en/spatial-extensions.html>

<https://geosparkconsulting.com/>

<https://es.wikipedia.org/wiki/GeoJSON>

<http://mappingandco.com/blog/mongodb-y-geojson>