

Universidad de las Ciencias Informáticas
Facultad 6



Título: Propuesta del diseño e implementación de una
base de datos espacial para depósitos minerales

Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Autor: Rosemary Riera Carratalá

Tutor: Ing. Eddy Dangel Quezada Rodríguez

La Habana, Junio de 2012
“Año 54 de la Revolución”

Declaración de autoría:

Declaro que soy la única autora de este trabajo y autorizo a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo. Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Firma del Autor

Rosemary Riera Carratalá

Firma del Tutor

Ing. Eddy Dangel Quezada Rodríguez

Agradecimientos:

Agradecer de forma especial a mis padres que me han apoyado de forma incondicional durante estos cinco largos años, la confianza y el amor que me profesan ha sido la fuerza y la determinación que me ha acompañado para terminar mis estudios universitarios.

A mis abuelitos y mi tía Nidia porque desde que vine al mundo han hecho de todo por mí, siempre aconsejándome y ayudándome en todo lo que he necesitado.

A mi compañero de la vida Joel por acompañarme en los buenos y los malos momentos, por ser atento, cuidarme, ayudarme y sobre todas la cosas por su confianza y amor.

A mis hermanos y mis amigos, algunos más lejos y otros más cerca, por tenerme siempre en sus corazones y hacerme saber que puedo contar con ellos.

A mi tutor y amigo Eddy por su guía y apoyo, gracias por despertar en mí el interés por conocer y aprender más, gracias por hacerme sentir que lo que hago es importante.

Dedicatoria:

Dedico este trabajo y los resultados obtenidos a todos mis abuelitos, a ellos que siempre han estado orgullosos de mí.

También dedico este trabajo a mis sobrinas y mi sobrinito porque cuando estoy con ellos soy inmensamente feliz.

Resumen

En la actualidad diversas empresas mineras del mundo entero presentan necesidades en cuanto al almacenamiento de la información generada a partir de los procesos que realizan en su trabajo diario. En muchas ocasiones esta información se encuentra registrada en papeles y es una necesidad digitalizar esos datos para poder contar con información que pueda ser consultada cuando sea necesario. Esta información puede presentar problemas de integridad por lo que es vital identificarlos y solucionarlos de forma ágil. En las actividades mineras se generan grandes volúmenes de información y por tal motivo cuando se realizan búsquedas o se consulta la misma puede demorar un tiempo prolongado siendo necesario crear diferentes mecanismos para lograr un mejor rendimiento de las aplicaciones que utilicen dicha información. El rendimiento puede ser afectado también porque se utilice un diseño para la estructura de almacenamiento de los datos al que no le han sido aplicados los recursos más óptimos. Para darle solución a esta problemática se diseñó un modelo de base de datos para el proyecto Sistema Minero Cubano con el objetivo de lograr un almacenamiento estructurado y seguro de la información. En el desarrollo de la presente investigación se estudiaron algunos de los conceptos más relevantes relacionados con el tema en cuestión así como diferentes herramientas utilizadas en las etapas de diseño, implementación y pruebas de la base de datos. Finalmente, luego de validar la presente propuesta, se obtuvo un modelo que se ajusta a las necesidades actuales de las empresas mineras.

Palabras Claves:

Bases de datos espaciales, depósito mineral, diagramas, índices espaciales, PostGIS

Contenido

Introducción	1
Capítulo 1: Elementos teóricos de la investigación	5
1.1 Bases de Datos Relacionales (BDR)	5
1.1.1 Bases de Datos Espaciales.....	6
1.2 Principales Sistemas Gestores de Bases de Datos y sus extensiones espaciales.....	8
1.2.1 PostgreSQL	9
1.2.2 Oracle	11
1.2.3 MySQL	12
1.2.4 SQLite.....	13
1.3 Datos geométricos de un depósito mineral.....	14
1.4 Análisis de soluciones existentes	15
1.5 Tecnologías y herramientas a utilizar	18
1.6 Conclusiones parciales.....	19
Capítulo 2: Descripción y análisis de la solución propuesta	20
2.1 Modelo Lógico	20
2.1.1 Diagrama de Clases Persistentes	20
2.2 Modelo Físico	21
2.2.1 Diagrama de Entidad Relación.....	21
1.2.2 Diseño por esquemas	22
1.2.3 Patrones de diseño utilizados.....	24
1.2.4 Funciones e índices relevantes	25
2.2.4 Normalización de la base de datos.....	29
2.3 Conclusiones parciales.....	30
Capítulo 3: Validación de la propuesta de diseño.....	31
3.1 Validación teórica del diseño	31
3.1.1 Integridad	31
3.1.2 Análisis de redundancia de la información.....	32
3.1.3 Seguridad y control de acceso	32
3.1.4 Trazabilidad de acciones.....	33
3.2 Validación funcional del diseño.....	34
3.2.1 Pruebas de volumen	34
3.2.2 Pruebas de desempeño	36

3.2.3 Pruebas de carga	40
3.3 Conclusiones parciales	44
Conclusiones Generales	45
Recomendaciones	47
Bibliografía Referenciada	48
Anexos.....	52

Índice de figuras

Figura 1. Jerarquía de clases geométricas.....	8
Figura 2. Modelo Entidad-Relación de la primera versión del diseño de base de datos para Geolmin.	16
Figura 3. Estructura de tablas básica de Gemcom.....	17
Figura 4. Tabla HEADER de Gemcom.....	18
Figura 5. Diagrama de Clases Persistentes del diseño de la base de datos espacial para depósitos minerales.	21
Figura 6. Diagrama de Entidad Relación del diseño de la base de datos espacial para depósitos minerales.	22
Figura 7. Funcionalidad importar datos de la herramienta Geolmin.....	35
Figura 8. Representación gráfica de las pruebas de desempeño.....	39
Figura 9. Representación gráfica de la prueba 1.....	43
Figura 10. Representación gráfica de la prueba 2.....	44

Índice de tablas

Tabla 1. Patrones de diseño utilizados.....	24
Tabla 2. Funciones de validación implementadas.....	25
Tabla 3. Funciones de conversión implementadas.....	27
Tabla 4. Índices espaciales desarrollados.....	28
Tabla 5. Total de datos introducidos por tabla.....	35
Tabla 6. Resultados de las pruebas de desempeño realizadas.....	36
Tabla 7. Resultados de la prueba 1.	42
Tabla 8. Resultados de la prueba 2.	43

Introducción

A medida que el hombre comenzó a automatizar sus procesos se hacía más urgente la necesidad de almacenar la información resultante, pero no fue hasta la década de 1960 que se desarrollaron las primeras bases de datos estructuradas jerárquicamente y seguidamente las bases de datos en red. Estos acontecimientos posibilitarían que en 1970 Edgar Frank Codd definiera el modelo relacional e hiciera público un grupo de reglas para la evaluación de administradores de sistemas de datos relacionales.

Las bases de datos surgieron para darle solución al problema del almacenamiento de grandes volúmenes de información, convirtiendo toda la información que almacenan en una serie de datos relacionados que forman una estructura lógica y a la vez reconocible desde un programa informático. Las bases de datos se enfrentan a demandas empresariales muy exigentes, elemento que ha marcado su evolución a lo largo del tiempo. Tanto han llegado a evolucionar que en la actualidad si una empresa necesita informatizar sus procesos y esta actividad implica almacenamiento de información, es muy conveniente elegir entre varios tipos existentes la base de datos que más se adecue al proceso de negocio de esa entidad.

La utilización de bases de datos se extiende a numerosas áreas de trabajo. En el área de la industria minera por ejemplo, se realizan estudios y procesos de explotaciones de los minerales naturales que son valiosos económicamente para un país y su desarrollo. Para la manipulación de la información minera se han creado sistemas que representan soluciones de software orientados a la optimización de los procesos de reconocimiento, prospección y exploración, explotación, procesamiento y comercialización de depósitos minerales. De esta forma se logra una mayor eficiencia, organización y rentabilidad en las actividades. Estos procesos generan mayormente información de tipo socioeconómica e información de tipo espacial.

Uno de los principales objetivos de la creación de soluciones de software es poder disponer de información persistente en el tiempo, que pueda ser consultada cuando se necesite. Actualmente en diversas empresas mineras mucha de la información sobre sus procesos solamente se encuentra registrada en papeles y es fundamental poder digitalizar todos estos datos. Muchas veces esta información contiene errores y problemas de integridad que deben ser identificados y solucionados de forma rápida para poder contar con información completamente lógica y fiable antes de realizar cualquier análisis.

Debido a las grandes cantidades de datos que se generan en las actividades mineras otro de los problemas que existen es que la recuperación de los datos se torna un proceso lento y engorroso. Esto puede suceder a la hora de consultar la información o realizar búsquedas en la misma, siendo necesario aplicar diferentes mecanismos para solucionar este problema. De esta manera se puede obtener un mejor rendimiento del software en general.

Una dificultad existente que también afecta el rendimiento de las aplicaciones que trabajan con información minera es que el diseño de la estructura que emplean para el almacenamiento de la información muchas veces no utiliza los recursos más óptimos desarrollados. Esto puede ocurrir ya sea por una mala investigación y análisis de las herramientas destinadas a estos fines o por problemas de presupuesto. Estas herramientas en su gran mayoría son productos que se comercializan a elevados precios, y este es precisamente otro inconveniente que sale a relucir.

En ocasiones las empresas comprenden que las herramientas que emplean para el almacenamiento de sus datos solamente responden a datos bidimensionales y sin embargo las funcionalidades que la empresa necesita desarrollar requieren del manejo tanto de datos bidimensionales como tridimensionales. De la misma forma sucede con la información socioeconómica y la información espacial, los dos tipos en que puede clasificarse la información minera. Si se utiliza un diseño de almacenamiento que solamente soporta la información socioeconómica entonces se ajusta toda la información proveniente de un depósito mineral a ese tipo en particular. Esto provoca un diseño menos eficiente que influye en el rendimiento de la aplicación que trabaje con esa información.

Cuba no está ajena a estos constantes movimientos de cambio para mejorar los resultados de sus actividades mineras, por lo que distintas entidades geológico-mineras cubanas como la ONRM, el Grupo empresarial Geominsal, el Grupo empresarial Cuba Níquel, en conjunto con la Universidad de Ciencias Informáticas comenzaron la construcción del software Geolmin. Este software debe permitir procesar la información obtenida en las actividades de prospección y exploración que se llevan a cabo por las empresas mineras cubanas. La tarea fue encomendada al centro de desarrollo de software GEySED de la UCI, específicamente al proyecto nombrado Sistema Minero Cubano que pertenece al departamento de Geoinformática.

Partiendo de esta situación problemática se determina como **problema de la investigación**: ¿Cómo contribuir a la manipulación y almacenamiento de la información espacial de un depósito mineral?

En la presente investigación se plantea como **objeto de estudio**: El Proceso de diseño de bases de datos relacionales.

Dentro de dicho objeto de estudio se enmarca el siguiente **campo de acción**: El proceso de diseño de bases de datos espaciales para depósitos minerales.

Para resolver el problema se plantea como **objetivo general** de la investigación: Proponer el diseño y la implementación de una base de datos para el proyecto Sistema Minero Cubano que soporte la manipulación y el almacenamiento de la información espacial asociada a un depósito mineral.

Para dar cumplimiento al objetivo general se trazaron las siguientes **tareas de la investigación**:

- 1- Caracterizar los conceptos asociados al dominio del problema.
- 2- Caracterizar los Sistemas Gestores de Bases de Datos actuales que soporten extensiones geométricas.
- 3- Caracterizar los datos geométricos persistentes de un depósito mineral.
- 4- Elaborar el diseño lógico y físico de la base de datos espacial.
- 5- Implementar la base de datos espacial.
- 6- Realizar pruebas a la base de datos.

Para el desarrollo de la investigación se emplearon los siguientes **métodos científicos**:

Métodos teóricos:

- Analítico-sintético: Se utilizó para realizar el análisis de los elementos relacionados con la abstracción de base de datos para luego lograr establecer una síntesis de los resultados obtenidos.
- Histórico-lógico: Se empleó para realizar el estudio de las tecnologías actuales de bases de datos y las soluciones existentes creadas para el almacenamiento de información espacial. Se utilizó realizando el análisis del desarrollo de los diferentes Sistemas Gestores de Base Datos, analizando los principales aspectos en los que fueron evolucionando para escoger el más conveniente.
- Modelación: Se utilizó para modelar los diagramas del diseño e implementación de la base de datos.

Métodos empíricos:

- Observación: Se utilizó para poder efectuar un análisis anticipado de la información para verificar y comprobar varias concepciones teóricas.

El análisis de los elementos anteriormente expuestos conduce a plantear como **idea a defender**: El diseño e implementación de una base de datos espacial para el software Geolmin permitirá contribuir a la manipulación y almacenamiento de la información espacial de un depósito mineral.

Se espera como **posible resultado** el diseño de una base de datos espacial que permita manipular y almacenar información espacial de depósitos minerales, además de un conjunto de procedimientos almacenados, vistas y consultas que permitan una recuperación de datos eficiente de la base de datos espacial.

El documento está estructurado en: resumen, introducción y desarrollo, el cual está compuesto por 3 capítulos:

- Capítulo 1 denominado “**Elementos teóricos de la investigación**”, se incluyen todos los aspectos teóricos que soportan este proyecto y se realiza un estudio del estado del arte de diferentes Sistemas Gestores de Bases de Datos y sus extensiones espaciales. Se efectúa un breve análisis de algunas soluciones existentes que se han desarrollado en el campo de la minería.
- Capítulo 2 denominado “**Descripción y análisis de la solución propuesta**” se plantean los diferentes elementos que conforman el diseño propuesto en su modelo lógico y modelo físico como son los diagramas de clases persistentes y de entidad relación, los patrones, funciones e índices desarrollados así como algunos aspectos referentes al estado de normalización del diseño propuesto de la base de datos espacial para depósitos minerales.
- Capítulo 3 denominado “**Validación de la propuesta de diseño**”, se realiza la validación teórica del diseño, teniendo en cuenta la integridad, los elementos que componen la seguridad de la Base de Datos, la trazabilidad de acciones y el análisis de redundancia de información así como una validación funcional a través de la aplicación de pruebas de volumen, desempeño y carga.

Capítulo 1: Elementos teóricos de la investigación

A continuación en el desarrollo del capítulo 1 se abordan los diferentes conceptos y fundamentaciones teóricas que componen la presente investigación. Para lograrlo se plantean los siguientes pasos: Definir las bases de datos relacionales, las base de datos espaciales, los datos espaciales, los índices espaciales, los principales Sistemas Gestores de Bases de Datos y sus extensiones espaciales. Argumentar sobre la relación existente entre los datos geométricos y un depósito mineral. Por último realizar un análisis del diseño de las soluciones existentes y definir las tecnologías y herramientas a utilizar.

1.1 Bases de Datos Relacionales (BDR)

Existen algunas definiciones asociadas al término base de datos como *“un conjunto exhaustivo de datos estructurados, fiables y homogéneos, organizados independientemente de su utilización e implementación en una computadora, accesibles en tiempo real, que pueden compartir varios usuarios con necesidades de información diferentes y no predecibles en el tiempo”*, así lo plantea (CAMALLEA 2004).

También se define como *“un conjunto de datos persistentes que es utilizado por los sistemas de aplicación de alguna empresa dada”*, es la consideración de (DATE 2001). Se podría definir también como aquel lugar donde se guardan los datos y al cual pueden acceder diferentes aplicaciones de una determinada entidad. Otra definición es *“Una base de datos es una serie de datos relacionados que forman una estructura lógica, es decir una estructura reconocible desde un programa informático. Esa estructura no sólo contiene los datos en sí, sino la forma en la que se relacionan”* de (SÁNCHEZ 2004).

En las tecnologías de bases de datos existen diferentes modelos de organización entre los cuales se encuentra el modelo jerárquico, en red, orientados a objetos y relacionales. Este último es uno de los más utilizados y ha alcanzado gran popularidad en el mundo de las Tecnologías de la Información. Una de las definiciones otorgadas a las bases de datos de tipo relacional es *“un conjunto de relaciones representadas a través de tablas que almacenan una colección específica de datos estructurados. Cada tabla puede variar con el transcurso del tiempo y se identifica de manera única por medio de un nombre”*, según el criterio de (ROZIC 2004).

Además las BDR presentan numerosas ventajas ya que permiten que los datos puedan ser normalizados y de esta forma se pueda evitar la redundancia de los mismos y garantizar su

actualización. A través de una base de datos se puede evitar la inconsistencia de datos, garantizar su integridad en todo momento, a través de permisos bien definidos se puede lograr su máxima seguridad, además una base de datos permite compartir los datos entre distintos usuarios y aplicaciones que necesiten manipularlos o modificarlos.

1.1.1 Bases de Datos Espaciales

Una base de datos espacial es un sistema de bases de datos que ofrece tipos de datos espaciales en su modelo de datos y lenguaje de consulta, soporta diversos tipos de datos espaciales en su implementación, proporcionando al menos indexación espacial y métodos de unión espacial. Los sistemas de bases de datos espaciales ofrecen la tecnología básica para los SIG¹ y otras aplicaciones. (GÜTING 1994). Una base de datos espacial permite describir los objetos espaciales que la forman a través de tres características básicas: atributos, localización y topología. (SHEKHAR 2003).

Una base de datos espacial se define también como una colección de datos referenciados en el espacio que actúa como un modelo de la realidad (ANALYSIS 1998). *“Una base de datos espacial representa una serie o aproximación de fenómenos. Esos fenómenos seleccionados son considerados suficientemente importantes para ser representados en forma digital. La representación digital puede ser para el pasado, presente o futuro”*, así lo expone (HAITHCOAT 1999).

También pueden conocerse como bases de datos geográficas, que no son más que una colección de datos organizados de tal manera que sirvan efectivamente para una o varias aplicaciones SIG. Esta base de datos comprende la asociación entre sus dos principales componentes: datos espaciales y atributos o datos no espaciales. (ESRI 1995).

Las bases de datos espaciales en la actualidad constituyen la elección de numerosas empresas que deciden emprender proyectos y necesitan un sistema capaz de abstraerse y transformar un problema complejo y real para obtener un resultado más simple que permita ser modelado por un lenguaje de programación. Se podría definir a los datos que maneja una base de datos espacial como: *“los datos acerca de los objetos y propiedades en el mundo con respecto a su ubicación”*, así lo considera (P. RAGAUX 2002).

¹ *Sistemas de Información Geográfica*

Las bases de datos espaciales tratan con objetos que tienen una posición en el espacio, así como con las relaciones espaciales entre esos objetos. Para ello posibilitan el uso de diferentes tipos de datos que manejan los resultados para que a la hora de abstraerse se mantengan los objetivos reales. Dentro de los diferentes tipos de datos espaciales que permiten manejar se encuentran el punto, línea, polilínea, polígono y multipolígono, estos proporcionan una vía de abstracción para modelar la estructura de las entidades geométricas en el espacio, así como las relaciones entre clases, propiedades y operaciones.

Los tipos específicos de datos que se utilizan varían en dependencia de las características que debe soportar la aplicación que se desea crear. Los datos espaciales refieren a entidades o fenómenos y se caracterizan por poseer una posición absoluta sobre un sistema de coordenadas, una posición relativa frente a otros elementos del paisaje, es decir una topología, además presentan una figura geométrica que las representan y atributos que los describen. (VOISARD 2002).

La OGC² es un consorcio internacional de empresas, organismos y universidades que participan en el desarrollo de soluciones conceptuales a disposición del público que puede ser útil con todo tipo de aplicaciones que manejan datos espaciales. En la especificación de geometrías la OGC propone el siguiente modelo de objetos geométricos, ver figura 1.

La clase geométrica base tiene las subclases punto, curva, superficie y colección geométrica. Cada objeto geométrico es asociado con un sistema de referencia espacial, el cual describe el espacio de coordenadas donde el objeto geométrico es definido. Las clases de colecciones de cero, uno y dos dimensiones llamadas Multipunto, Multipolilínea y Multipolígono son para modelar geometrías correspondientes a colecciones de puntos, líneas y polígonos respectivamente. (OGC 1994).

La indexación espacial es usada para apoyar la selección espacial, es decir, para buscar en un gran conjunto de objetos espaciales aquellos con una relación particular a un valor de consulta. La indexación espacial organiza el espacio y los objetos dentro de él de forma que solamente algunas partes del espacio y un subconjunto de los objetos deban tenerse en cuenta para responder a una determinada consulta.

² *Open Geospatial Consortium*

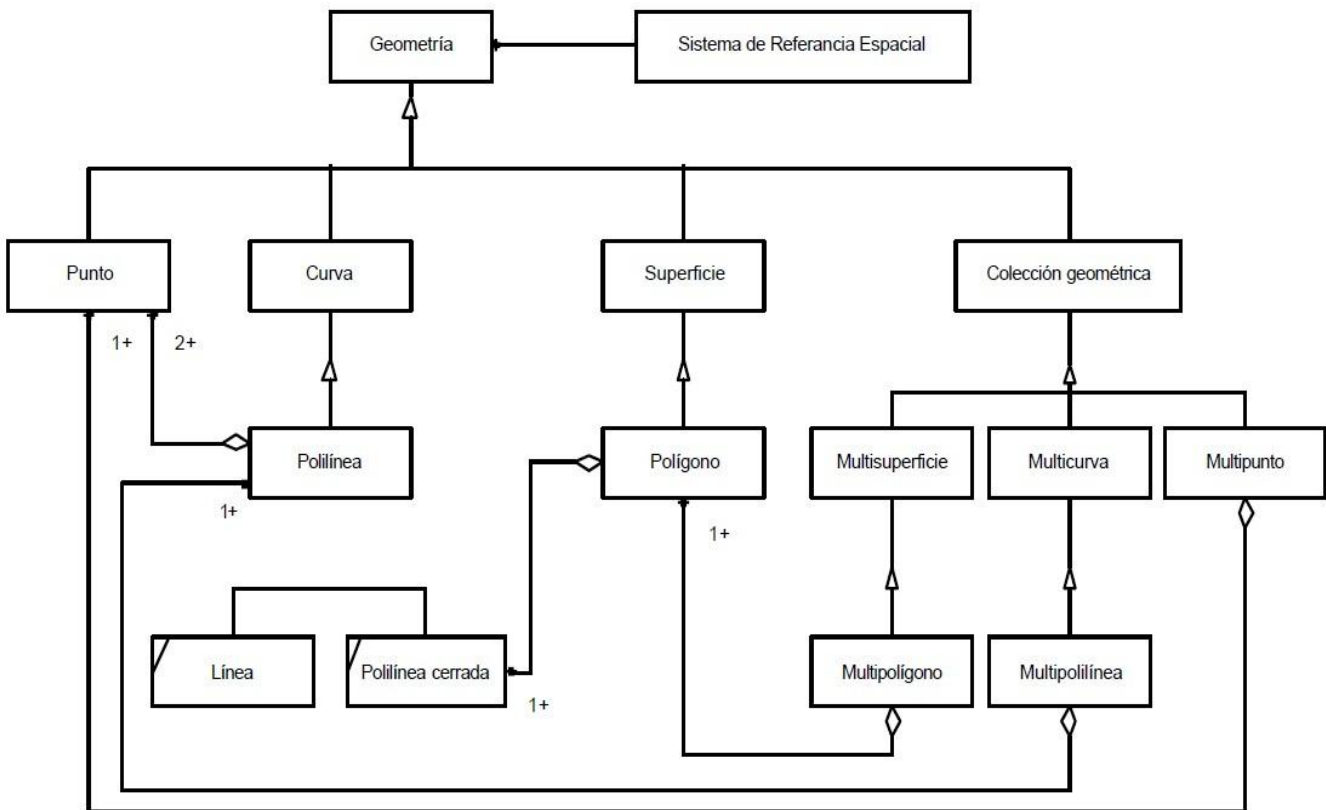


Figura 1. Jerarquía de clases geométricas. (OGC 1994)

De la misma forma que los índices simples, los índices espaciales pueden definirse como estructuras de datos que permiten optimizar las búsquedas dentro de la información contenida en la base de datos, particularmente en la información de tipo espacial. Otro concepto válido es que un índice espacial es un tipo de índice extendido que permite indexar una columna espacial. Una columna espacial es una columna de tabla que contiene datos de un tipo espacial. Con la utilización de los índices espaciales se logra una mejor recuperación de la información espacial contenida en una base de datos.

1.2 Principales Sistemas Gestores de Bases de Datos y sus extensiones espaciales

Un Sistema Gestor de Bases de Datos (SGBD) constituye *“un conjunto coordinado de programas, procedimientos, lenguajes, herramientas, que suministra, tanto a los usuarios no informáticos como a los analistas, programadores o administradores de una base de datos, los medios necesarios para describir y manipular los datos integrados en la base de datos, manteniendo su integridad, confidencialidad y disponibilidad”*, concepto que propone (DE MIGUEL CASTAÑO 1999).

Se define también a un SGBD como *“una colección de datos interrelacionados y un conjunto de programas para acceder a esos datos. El principal objetivo de un SGBD es proporcionar un entorno que sea a la vez conveniente y eficiente para ser utilizado al extraer y almacenar información de la base de datos”* (CODAZZI 2010). Un SGBD facilita un grupo de herramientas y programas que garantizan la relación y manipulación de la información almacenada en una base de datos, posibilitando que exista entre la información y las aplicaciones total independencia y sin afectar la seguridad de la información.

Un SGBD está estructurado por tres niveles fundamentales. El más bajo de ellos, el nivel físico describe cómo son almacenados realmente los datos y en él son detalladas las estructuras de datos complejas de bajo nivel. Luego el nivel lógico es quién describe a los datos que son almacenados en la base de datos y las relaciones existentes entre ellos. Por último el nivel de vistas, es el más alto de los niveles de abstracción, se definen varias vistas de una base de datos para que sean observadas por los usuarios.

Numerosas facilidades presentan los SGBD para los usuarios. *“Entre sus principales funcionalidades se encuentran: la creación y mantenimiento de Bases de Datos; garantizar el acceso, definición, actualización y recuperación de los datos; así como permitir el control centralizado de estos. También permiten garantizar la integridad, seguridad y fácil manipulación de los datos; y evitar las redundancias e inconsistencias de los mismos”*, asegura (GARCÍA 1999).

1.2.1 PostgreSQL

PostgreSQL presenta un grupo de características que lo distinguen notablemente y lo convierten en uno de los SGBD más potentes y robustos de todo el mercado mundial. (POSTGRESQL 1996). A continuación se mencionan algunas de estas características: es un servidor de bases de datos objeto relacional libre, liberado bajo la licencia BSD³. Puede ser ejecutado desde los Sistemas Operativos Linux, UNIX, Mac OS X, Solaris, Windows, mencionando solamente los más conocidos. Este SGBD puede trabajar con grandes cantidades de datos y una alta concurrencia de usuarios accediendo al

³ Berkeley Software Distribution

sistema a la misma vez, además puede soportar el uso de conexiones SSL⁴ para cifrar las comunicaciones.

Además PostgreSQL posee un soporte nativo para los lenguajes más populares como son PHP, C, C++, Perl, Python, entre otros. Soporta casi toda la sintaxis SQL e incluye además sub-consultas, transacciones, numerosos tipos de datos y funciones que además le permite definir al mismo usuario.

PostGIS

PostGIS es una extensión espacial para el SGBD PostgreSQL. PostGIS está certificado por la Open Geospatial Consortium como cumplidora de la especificación SFS⁵ 1.1, también implementa gran parte de la norma ISO “SQL Multi-media” para la funcionalidad geoespacial.

EL SGBD PostgreSQL soporta a tres tipos de índices por defecto, B-Tree que son utilizados principalmente para datos que son almacenados en solamente un eje, ejemplo números o fechas, R-Tree que divide los datos en rectángulos, a su vez en sub rectángulos y así sucesivamente, e índices GiST⁶ que separan los datos en distintas clasificaciones, y se utilizan para algunas estructuras de dato irregulares, ejemplo arrays numéricos y datos SIG. (INC 2009).

PostGIS incluye un conjunto de operaciones para realizar consultas espaciales muy bien optimizadas por sus índices R-Tree y su integración con el planificador de consultas de PostgreSQL. Utiliza las librerías Proj4 para dar soporte a la transformación dinámica de coordenadas y la librería GEOS para realizar operaciones de geometría. Utiliza bloqueo a nivel de fila, permitiendo a múltiples procesos trabajar con las tablas espaciales concurrentemente y asegurando la integridad de los datos. Los objetos soportados por PostGIS son todos los objetos y funciones especificados en la OGC y extiende el estándar con soporte de coordenadas para 3DZ, 3DM y 4D. (INC 2005).

Una vez que se haya creado la tabla espacial, se pueden subir datos a la misma mediante dos métodos: directamente mediante comandos SQL o utilizando la herramienta Shape Loader/Dumper

⁴ *Secure Sockets Layer*

⁵ *Simple Features Specifications For SQL*

⁶ *Generalized Search Tree*

que convierte ESRI Shape files⁷ en comandos SQL. Para la extracción de información espacial la forma más común es utilizar consultas SQL select, y luego pasar esta información a un archivo de transferencia. El desarrollo de PostGIS es eminente, se han agregado varias funcionalidades como soporte básico de topología, transformación de coordenadas, APIs⁸ de programación, entre otras.

1.2.2 Oracle

Oracle es una herramienta cliente-servidor utilizada para la gestión de bases de datos y altamente vendida a nivel mundial. Oracle es SGBD Orientado a Objeto y ha logrado un gran nivel de madurez debido características como: transportabilidad ya que funciona sobre decenas de plataformas, la potencia de sus instrumentos de desarrollo de aplicaciones, la gran riqueza de su diccionario de datos, una potente seguridad y confidencialidad, la realización de copias de seguridad y recuperación de los datos, proporciona conectividad con paquetes software de otros fabricantes, además de un excelente soporte técnico. (KOTHURI 2007).

Oracle Spatial

Oracle Spatial es un componente opcional, que necesita de licenciamiento, se encuentra viable únicamente para el SGBD Oracle Enterprise Edition mientras que las demás versiones presentan un componente espacial limitado llamado Oracle Locator. Oracle Spatial facilita un esquema SQL y funciones para el almacenamiento, recuperación, actualización y consulta de datos espaciales. Cumple los estándares de la SFS y OLS⁹ 1.1.

Oracle Spatial utiliza indexación R-Tree, con la utilización de este índice se aproxima cada geometría al mínimo rectángulo que encierre la geometría llamada MBR¹⁰. Soporta datos primitivos y geometrías compuestas por colecciones de esos tipos, soporta el almacenamiento e indexación de tipos de datos tridimensionales y tetradimensionales. La información geográfica se estructura a través de una ubicación la cual indica la ubicación del objeto con respecto a un sistema de coordenadas ya sea de dos, tres o cuatro coordenadas, que sirva de referencia con la posición del mundo físico.

⁷ Desarrollado por la compañía Environmental Systems Research Institute. Son un formato sencillo y no topológico que permite almacenar información geográfica y de atributos.

⁸ Applications Programming Interface

⁹ Open Geospatial Consortium Location Services

¹⁰ Minimum bounding rectangle

Oracle presenta dos subcomponentes Geometry Engine que provee funciones para analizar, comparar y manipular geometrías e Index Engine que posibilita funciones para reducir las filas devueltas, quitando las más alejadas en el procesamiento de la consulta y así acelerar el proceso de identificación de filas que satisfagan una determinada condición. Presenta operadores, funciones y procedimientos que permiten realizar operaciones de análisis espacial. Implementa funciones y procedimientos para operaciones de tuning y posee un modelo de datos topológicos para trabajar con datos de nodos, bordes y caras.

Además Oracle Spatial contiene un modelo de datos de red para representar objetos modelados como nodos y enlaces de una red. Contiene GeoRáster permitiendo almacenar, indexar, consultar y analizar datos GeoRáster, como imágenes ráster y sus metadatos asociados. Implementa 3D Geometry para brindar soporte para almacenamiento de objetos espaciales tridimensionales. (ORACLE 2010). Presenta una gama de servicios web espaciales que pueden realizar funciones espaciales de geocoding, routing, feature selection y varias opciones que pueden resultar ser de interés por los usuarios.

1.2.3 MySQL

MySQL es un sistema de gestión de base de datos relacional, multiusuario y de subprocesamiento múltiple. Es además un software libre en un esquema de licenciamiento dual ya que por un lado se ofrece bajo la licencia GNU GPL¹¹, pero en el caso de las empresas que deseen incorporarlo en productos privativos tendrían que comprar una licencia que les permita ese uso. Utiliza SQL como lenguaje de consulta estructurado, el lenguaje estándar para consultas a bases de datos utilizado a nivel mundial.

MySQL posee gran versatilidad ya que trabaja tanto con sistemas operativos basados en Unix como con el sistema operativo desarrollado por Microsoft: Windows. (MYSQL 2012). Dispone de una arquitectura que lo convierte en un sistema de gran rapidez y de fácil personalización y uso.

MySQL Spatial Extensions

EL SGBD MySQL desde la versión 4.1 da soporte a extensiones espaciales para permitir la generación, almacenamiento y análisis de tablas con información espacial. Entre sus principales características se

¹¹ GNU General Public License

encuentra el soporte a los estándares de la Open Geospatial Consortium SFS para los tipos de geometría. Solamente soporta el trabajo con objetos espaciales de dos dimensiones. Los índices espaciales son soportados únicamente para tablas MyISAM, estos solamente pueden ser de tipo R-Tree y las columnas que sean indexadas no pueden poseer datos nulos, mientras que las tablas InnoDB, NDB, BDB y ARCHIVE soportan columnas espaciales pero estas no pueden ser indexadas.

MySQL facilita un conjunto de funciones para realizar operaciones sobre los datos espaciales, entre ellas se destacan 4 grupos: Funciones para conversión del formato de la Geometría las cuales son empleadas para convertir valores de geometría entre formatos internos y entre el formato WKT¹² o WKB¹³. Las funciones de las geometrías: estas reciben como parámetro una geometría y devuelven una característica cualitativa o cuantitativa de la misma. Funciones que crean nuevas geometrías a partir de las existentes: toman como argumento un valor de geometría y devuelven un nuevo valor y por último funciones que describen relaciones entre geometrías: este tipo recibe como entrada dos geometrías y devuelve un valor cualitativo o cuantitativo de la relación entre las mismas.

1.2.4 SQLite

SQLite es proyecto de dominio público, totalmente libre, un sistema relacional programado en el lenguaje C. (SQLITE 2010). No requiere ser instalado, ni se necesita prender, reiniciar o apagar un servidor, e incluso tener que configurarlo ya que así fue concebido su diseño. Puede ser ejecutado en diferentes sistemas operativos como Windows, Linux, BSD, Mac OS X, Solaris o estar embebido en otros como QNX, VxWorks, Symbian, Palm OS, Windows CE. Entre sus principales características destacan su portabilidad debido a que la base de datos se sitúa en solamente un fichero.

Spatialite

Es una extensión de SQLite que permite cargar, almacenar y manipular datos espaciales. Spatialite implementa extensiones espaciales siguiendo los planteamientos especificados por la OGC. Establece la indexación espacial utilizando R-Tree lo que significa que un R-Tree tiene que ser definido por cada columna relacionada a la indexación espacial. Con el propósito de detener cualquier problema que pueda surgir con los datos, el R-Tree ha de ser actualizado de manera coherente para cada INSERT, UPDATE o DELETE que se realice.

¹² *Well-Known Text*

¹³ *Well-Known Binary*

Cualquier columna geométrica que se beneficia de la indexación espacial tiene que ser definida dentro de los metadatos espaciales, esto significa que se puede añadir un índice espacial solamente en las columnas geométricas creadas a través de la función definida `AddGeometryColumn()` y luego se puede crear el índice espacial llamando a la función `CreateSpatialIndex()` de SQL, esta acción además de crear el índice espacial añade los disparadores necesarios para la sincronización completa entre la tabla principal y las correspondientes del R-Tree.

Para la aplicación de un índice espacial existe una manera alternativa si se utiliza `Spatialite` y es utilizando una memoria caché MBR con el fin de acelerar una consulta espacial que se torne lenta, el resultado es un aumento de rendimiento semejante a cuando se utiliza un R-Tree. (PROJECTS 2011). La librería `Spatialite` implementa un completo motor de datos SQL, utiliza además la librería `GEOS` que implementa sofisticadas funciones de análisis espacial, además la librería `Proj4` para implementar transformaciones de coordenadas entre diferentes referencias de coordenadas en un sistema.

Soporta `EPSG`¹⁴, una extensión para bases de datos geodésicas. La librería `libiconv` que implementa soporte para `locale_charset` y además un apoyo efectivo a cualquiera de las lenguas como el chino y japonés.

1.3 Datos geométricos de un depósito mineral

Un depósito mineral es la concentración de minerales u otras sustancias inorgánicas bajo la tierra cuya explotación está connotada a su importancia económica. Una de las técnicas más utilizadas a nivel mundial para la exploración y prospección de un depósito mineral es el empleo de pozos de perforación. Cada pozo está compuesto por un collar que no es más que la boca del pozo que se encuentra sobre la superficie de la tierra, el collar se representa como un punto en el espacio de coordenadas x, y, z .

El resto del pozo que se encuentra a partir del collar bajo la superficie de la tierra y se representa como una polilínea en el espacio dividida en varios segmentos de recta unidos y consecutivos, además una polilínea está denotada por un punto de inicio y un punto final, un determinado nivel de inclinación y valor del ángulo del pozo con respecto a la superficie de la tierra. Bajo tierra y empleando un pozo de

¹⁴ *European Petroleum Survey Group*

perforación pueden ser hallados cuerpos minerales, un cuerpo mineral es representado por un polígono en el espacio que puede ser convexo o no convexo.

Las superficies de los cuerpos minerales se representan a través del empleo de mayas poligonales y pueden clasificarse como superficies cerradas o abiertas. En un pozo se establecen determinadas zonas, una zona mineral es definida por una polilínea en el espacio y a ella se asocian los distintos valores de las concentraciones minerales encontradas dentro de sus límites.

1.4 Análisis de soluciones existentes

Primera versión del diseño de la base de datos del software Geolmin

En el proyecto Sistema Minero Cubano ya se había desarrollado anteriormente un diseño para la base de datos del software Geolmin. Para el desarrollo de esa solución se propuso SQLite como Sistema Gestor de Bases de Datos, la extensión espacial SpatiaLite y el Visual Paradigm como herramienta para el diseño. Esta primera versión cuenta con una tabla básica nombrada pozo que almacena el identificador y las coordenadas x, y, z donde se encuentra ubicado el pozo en el espacio. La versión de la extensión espacial SpatiaLite que se utilizó en el momento en que se diseñó este modelo no contaba con tipos de datos espaciales con soporte para objetos geométricos tridimensionales por lo que fue necesaria la utilización de tres campos en la tabla pozo para almacenar estas coordenadas.

La base de datos resultante de esta primera versión no permite realizar consultas espaciales ni utilizar índices espaciales sobre los campos, características de mucha utilidad al tratarse de una base de datos para un software minero que genera mucha información espacial. También fue necesaria la creación de otra tabla auxiliar para cada una de las tablas básicas: litología, muestra e inclinometría. Esta tabla auxiliar presenta atributos x, y, z con valores geométricos necesarios para el trabajo de la aplicación y que se calculan a partir de los valores de los campos x, y, z almacenados en la tabla básica correspondiente. Ver figura 2.

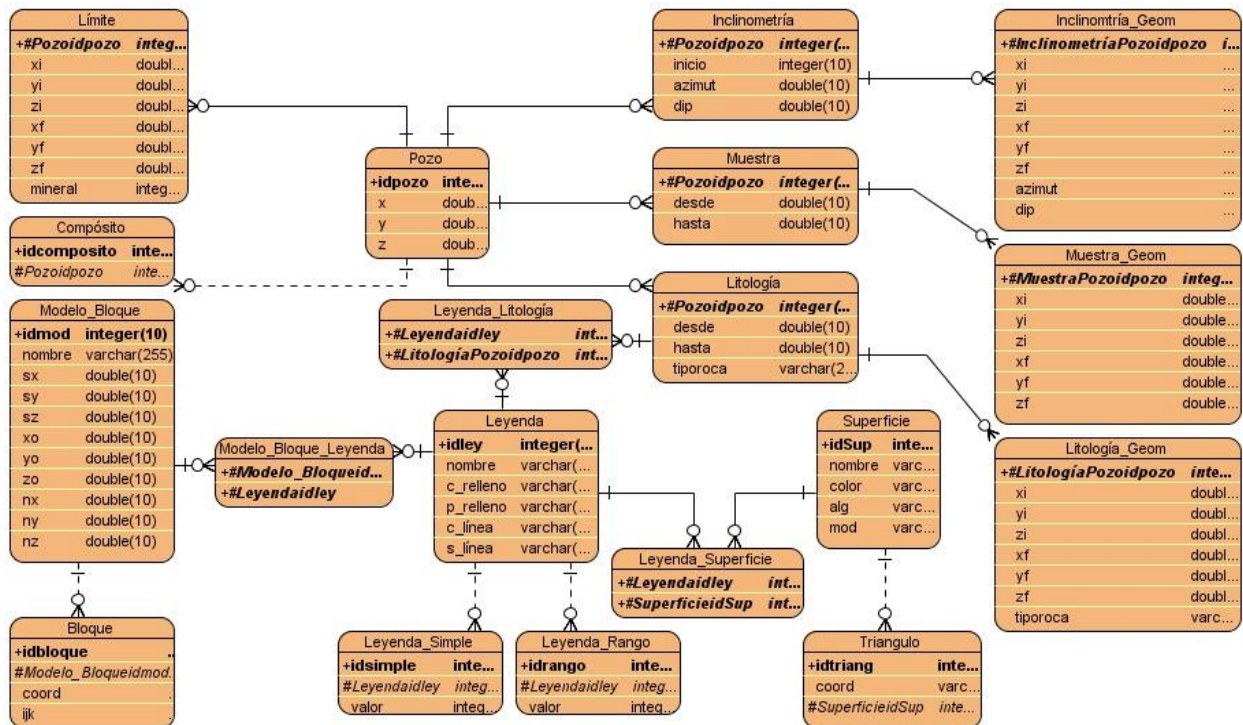


Figura 2. Modelo Entidad-Relación de la primera versión del diseño de base de datos para Geolmin. (BELLO 2011)

Diseño de la base de datos de Gemcom

Gemcom es un software de minería muy reconocido a nivel internacional, su gran prestigio se respalda en una experiencia de mercado de casi 30 años que ha aportado soluciones a algunas de las principales empresas de la industria minera del mundo. Gemcom utiliza una base de datos diseñada para Sistemas Gestores como Jet 4 con la utilización de Access 2000 y SQL server, ambos son Sistemas Gestores de Bases de Datos propietarios.

Analizando la base de datos de Gemcom se puede observar que posee un diseño genérico utilizado por numerosas aplicaciones de minería a nivel mundial. Dentro de su estructura de tablas posee cuatro tablas básicas, HEADER: donde se almacenan los datos referentes al identificador del pozo de perforación, la longitud y las coordenadas de la ubicación del pozo en el espacio; la tabla SURVEYS: donde se almacenan los datos referentes a la inclinometría del pozo; ASSAYS: en ella se almacenan

los datos referentes a las muestras extraídas del pozo y por último LITHOLOGY donde se almacena la información concerniente a la litología¹⁵ que conforma el pozo. Ver figura 3.

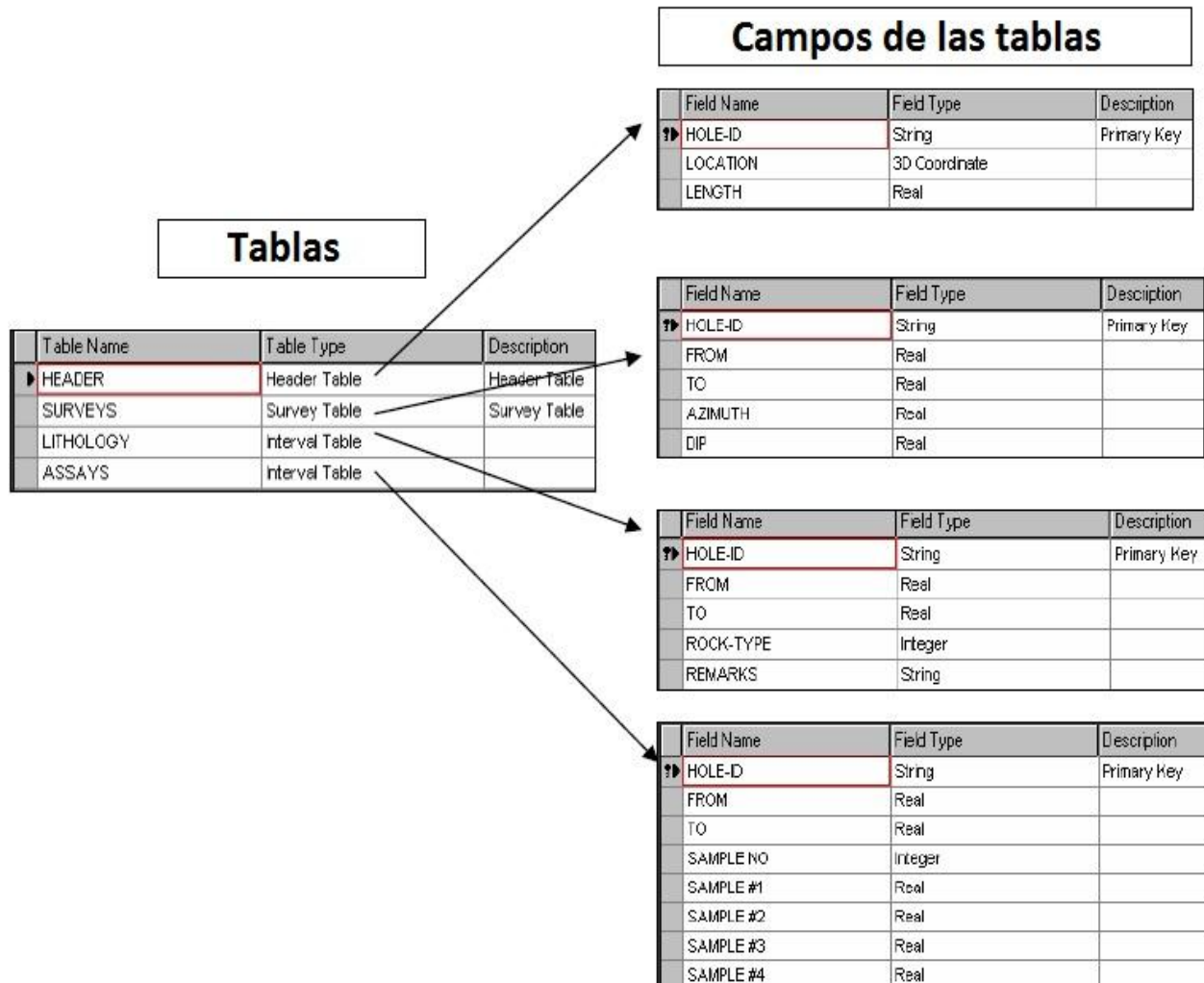


Figura 3. Estructura de tablas básica de Gemcom. (GEMCOM 1998)

En la tabla HEADER que utiliza GEMCOM existen tres campos: LOCATION[X], LOCATION[Y] y LOCATION[Z], para almacenar el valor de los puntos en el espacio de las coordenadas x, y, z respectivamente, ver figura 4. En el diseño de Gemcom no se utilizaron tipos de datos con soporte para objetos geométricos tridimensionales por lo que se necesitaron tres campos de la tabla HEADER para poder almacenar la información de las coordenadas del pozo en el espacio. No usar

¹⁵ Tipo de roca

objetos geométricos en el diseño, le impide a este modelo de base de datos la utilización de consultas espaciales y la implementación de índices espaciales.

	HOLE-ID	LOCATION[X]	LOCATION[Y]	LOCATION[Z]	LENGTH
1	DH001	10025.34	9241.56	265.73	908.32
2	DH002	10167.89	8905.84	491.77	409.54
3	DH003	9935.31	1123.27	387.09	843.07
4	DH004	11563.55	4208.89	886.65	460.67
5	DH005	10925.34	9316.09	611.78	345.65

Figura 4. Tabla HEADER de Gemcom.

1.5 Tecnologías y herramientas a utilizar

PostgreSQL

Para el diseño de la base de datos se propone utilizar PostgreSQL en su versión 9.1 como SGBD para el desarrollo de la solución propuesta. A partir del estudio realizado a los diferentes Sistemas Gestores de Bases de Datos se determinó que PostgreSQL presenta condiciones y funcionalidades que pueden ser muy bien aprovechadas para el diseño de una base de datos del tipo espacial (PostgreSQL). Con el empleo de la extensión espacial PostGIS en su versión 1.5 se garantiza el trabajo con objetos espaciales y nuevas topologías que proporcionan un mejor uso de la base de datos.

PostgreSQL posee características muy convenientes para el entorno de trabajo en la Universidad de Ciencias Informáticas ya que es un software libre, totalmente gratuito, posee una amplia documentación y puede ser ejecutado desde el Sistema Operativo Linux que es la plataforma libre a la que el país se propone emigrar totalmente desde hace algunos años. Además es una herramienta potente y capaz de ofrecer un buen funcionamiento de la base de datos espacial, ya que en conjunto con la extensión espacial PostGIS permite la utilización de índices espaciales para optimizar los tiempos de respuesta devueltos en las consultas y las búsquedas sobre la información espacial de la base de datos.

PostgreSQL posibilita además tres niveles de acceso para garantizar la seguridad a la base de datos, a partir de la utilización de los mismos se puede configurar quién y desde dónde se puede conectar a la base de datos y a que tablas específicamente. Otra facilidad que brinda la utilización de este SGBD es que archiva logs mediante los cuales se puede dar seguimiento a las acciones realizadas sobre la

información. PostgreSQL también permite la creación de esquemas, característica muy conveniente para agrupar las tablas de forma estructurada y lógica.

Visual Paradigm

Se propone Visual Paradigm versión 6.4 como herramienta a utilizar para el diseño de la base de datos, ya que es una herramienta UML¹⁶ profesional que soporta el ciclo de vida completo del desarrollo de software, desde el análisis y diseño hasta la construcción, pruebas y despliegue. (ALONSO 2009). Visual Paradigm es una herramienta con una interfaz muy amigable y fácil de utilizar, es también la herramienta que se especifica en el documento de Arquitectura: *Vista Entorno de Desarrollo Tecnológico* perteneciente al expediente del proyecto Sistema Minero Cubano.

Además con la utilización del Visual Paradigm se pueden construir aplicaciones de forma rápida, con calidad y a un menor costo. Permite dibujar todos los tipos de diagramas, entre ellos los que se necesitan diseñar en la presente investigación como son: el Diagrama de Clases Persistentes y el Diagrama de Entidad Relación. Permite generar código y documentación a partir de los diagramas. Es una herramienta colaborativa, es decir que soporta múltiples usuarios trabajando sobre el mismo proyecto. Permite realizar ingeniería inversa: del modelo físico se puede llegar al modelo lógico. Permite generar tablas con sus atributos a partir del diagrama de Entidad Relación.

1.6 Conclusiones parciales

A partir de los elementos teóricos de la investigación estudiados y analizados en el presente capítulo se puede arribar a las siguientes conclusiones:

- El uso de bases de datos espaciales posibilita la utilización del indexado espacial y el uso de consultas espaciales para acceder a la información espacial de la base de datos. El empleo de estas funcionalidades posibilitan un mejor rendimiento de la base de datos en cuanto a organización, tiempo de respuesta en las búsquedas y recuperación de la información contenida en la base de datos.
- PostgreSQL es un SGBD útil para el trabajo con información espacial ya que a través de su extensión espacial PostGIS posibilita a sus usuarios el empleo de objetos geométricos directamente en la base de datos. Existen otros SGBD que como PostgreSQL brindan las

¹⁶ *Unified Modeling Language*

mismas funcionalidades pero estos son software propietarios y con licencias altamente costosas, por lo que PostgreSQL es una excelente opción para la manipulación y almacenamiento de información espacial.

Capítulo 2: Descripción y análisis de la solución propuesta

A continuación en el desarrollo del capítulo 2 se abordan el diseño e implementación de la base de datos espacial que se propone en la presente investigación. Para lograrlo se plantean los siguientes pasos: Definir las clases persistentes del negocio, enunciar y explicar las funciones e índices creados de mayor importancia y finalmente explicar el estado de normalización en el que se encuentra la base de datos espacial.

2.1 Modelo Lógico

2.1.1 Diagrama de Clases Persistentes

En la figura 5 se muestra el Diagrama de Clases Persistentes correspondiente al diseño de la base de datos espacial para depósitos minerales que se propone en la presente investigación. Se han resaltado con color amarillo las tablas que almacenan información espacial en su composición y con un color rojo se han subrayado los atributos geométricos presentes en cada una de estas tablas. Téngase en cuenta que el Visual Paradigm como herramienta de modelación utilizada no facilita el empleo de tipos de datos GEOMETRY por lo que se ha utilizado el tipo de dato DOUBLE para caracterizarlos.

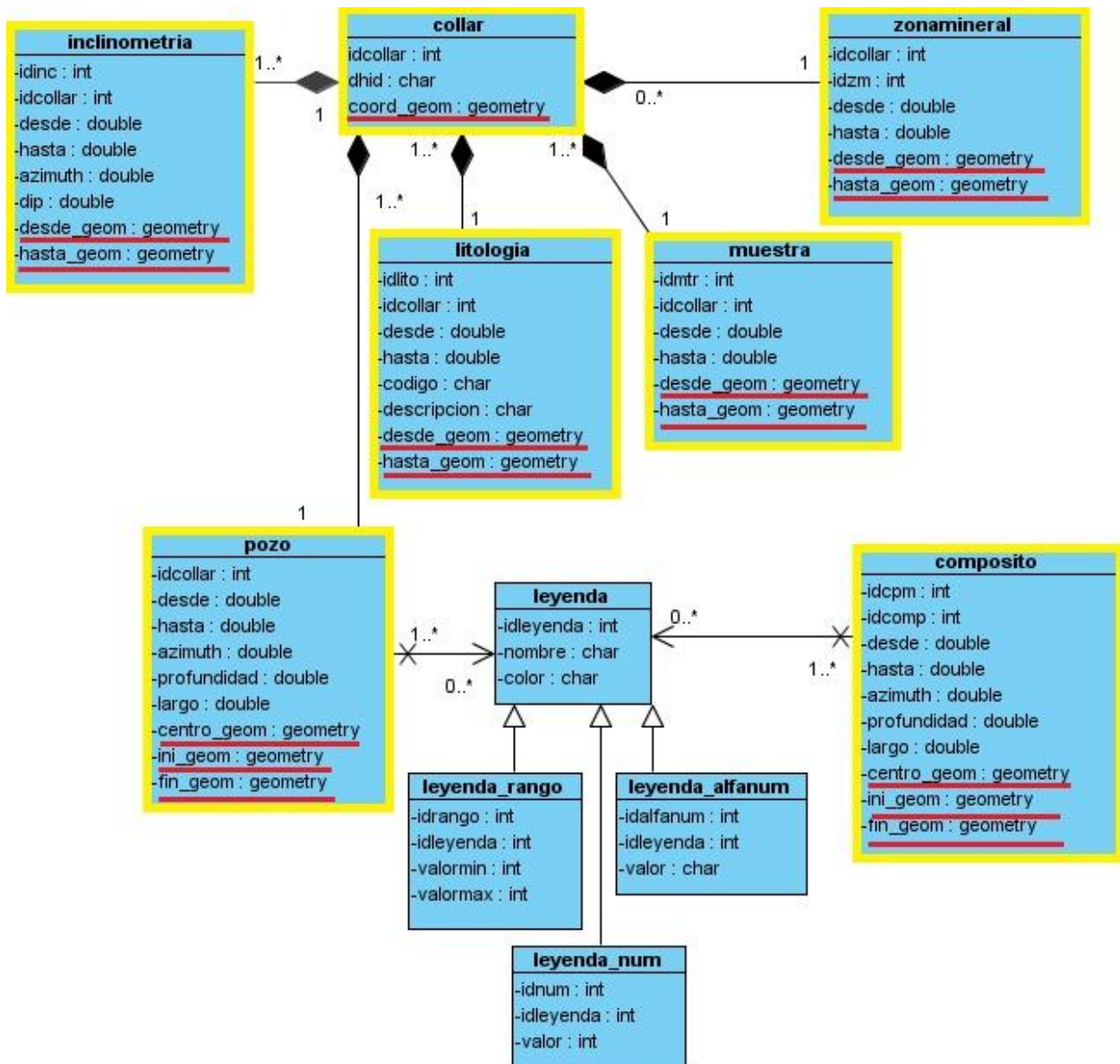


Figura 5. Diagrama de Clases Persistentes del diseño de la base de datos espacial para depósitos minerales.

2.2 Modelo Físico

2.2.1 Diagrama de Entidad Relación

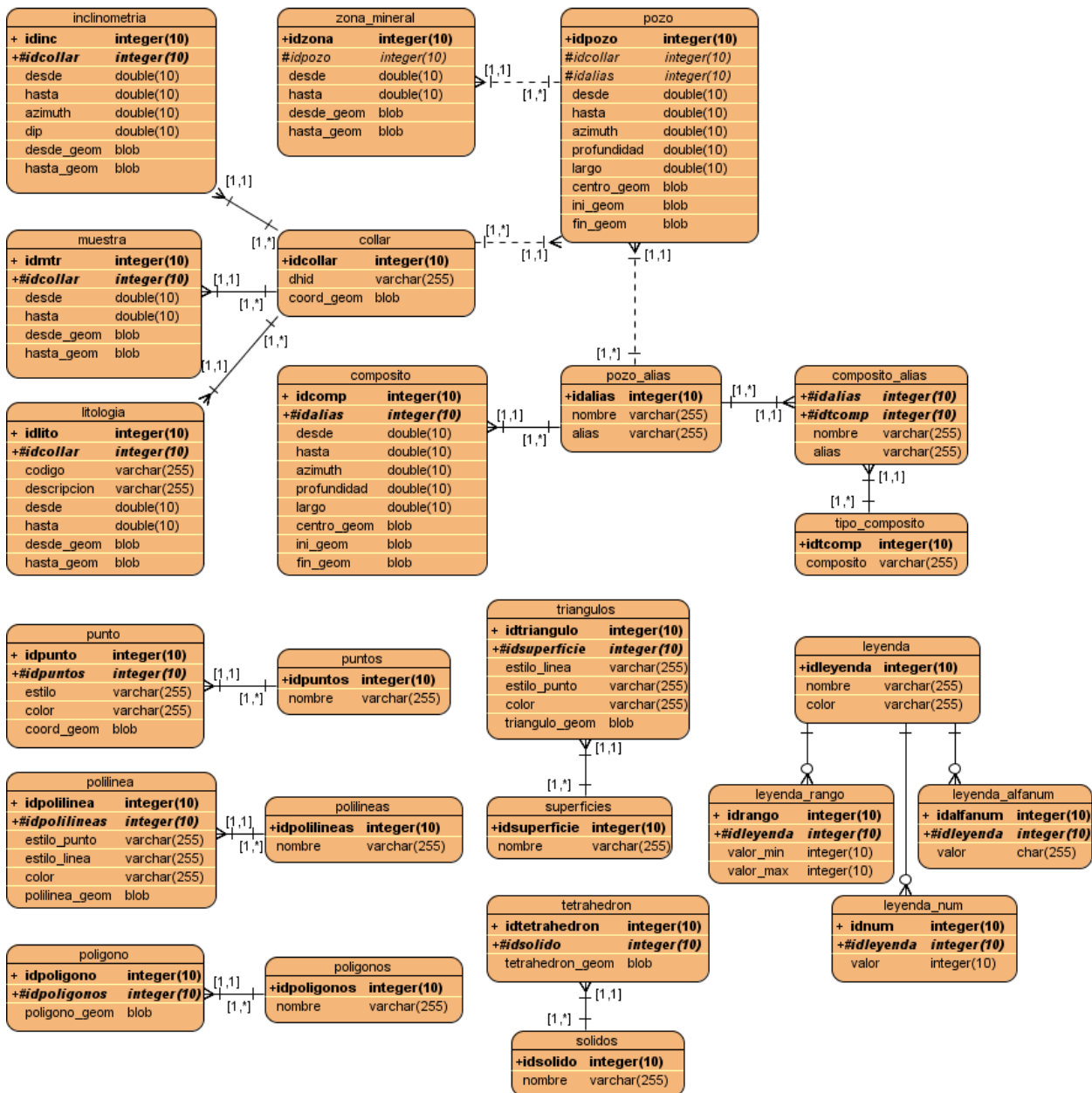


Figura 6. Diagrama de Entidad Relación del diseño de la base de datos espacial para depósitos minerales.

1.2.2 Diseño por esquemas

La base de datos está diseñada por esquemas, presenta 2 esquemas en total: gmsystem y public. La utilización de esquemas es una facilidad que brinda PostgreSQL para agrupar tablas de una forma estructurada y lógica según como el diseñador de la base de datos determine. En el esquema gmsystem se ubicaron las tablas que son del catálogo y que el sistema utiliza de forma interna como:

- tb_color
- tb_bdvalidacion
- tb_mineral
- tb_plot_mineral_color
- tb_roca_codigo
- tb_vparametro

En el esquema public se ubicaron las tablas propias del negocio como son:

- tb_collar
- tb_muestra
- tb_litologia
- tb_inclinometria
- tb_zona_mineral
- tb_pozo
- tb_pozo_alias
- tb_composito
- tb_composito_alias
- tb_tipo_composito
- tb_punto
- tb_puntos
- tb_polilinea
- tb_polilineas
- tb_poligono
- tb_polilgonos
- tb_solidos
- tb_superficies
- tb_tetrahedron
- tb_triangulos
- tb_leyenda
- tb_leyenda_rango
- tb_leyenda_alfanum
- tb_leyenda_num

1.2.3 Patrones de diseño utilizados

Para realizar el diseño de la base de datos, en la confección del Diagrama de Clases Persistentes así como el Diagrama de Entidad Relación fueron utilizados algunos patrones de diseño dentro de los cuales se encuentran los patrones Brown que propone (JIM COPLIEN 1996), los cuales son descritos a continuación:

Tabla 1. Patrones de diseño utilizados.

Nombre	Descripción
Representación de objetos como tablas	El objetivo de la utilización de este patrón es definir una tabla para cada clase de objetos persistentes. Los atributos de la clase que son tipos primitivos serán las columnas de las tablas.
Representación de relaciones como tablas	Al diseñar una base de datos se debe tener en cuenta cómo se representan las relaciones en un esquema de base de datos relacional, este patrón de diseño propone soluciones para los distintos tipos de relaciones que posee este modelo de datos. En el caso de las relaciones de uno a uno o de uno a muchos: <ul style="list-style-type: none"> - Colocar una clave ajena en la tabla de cardinalidad uno, para representar la relación de los objetos. - Crear una tabla asociativa para registrar los identificadores de cada uno de los objetos de la relación.
Identificador de objetos	Asignar un identificador independiente (OID) a cada objeto persistente.
Referencia de llaves foráneas	Asignar a cada objeto un identificador único. Luego añadir una columna por cada variable de instancia que no tenga un tipo de dato base o sea una colección. En esa columna almacenar el identificador del objeto referenciado y declarar

	la columna como llave foránea.
--	--------------------------------

1.2.4 Funciones e índices relevantes

Para conformar el modelo físico de la base de datos fueron implementadas algunas funciones entre las que se encuentran funciones de validación como:

Tabla 2. Funciones de validación implementadas.

Nombre	Objetivo y descripción
fn_intervalos_repetidos_muestra	Crear un listado de errores que contiene los identificadores de las muestras que posean intervalos repetidos dentro de un mismo pozo. Esto significa que no debe existir más de una muestra con los mismos valores en los campos “desde” y “hasta” que conforman el intervalo de un pozo.
fn_intervalos_repetidos_litologia	Crear un listado de errores que contiene los identificadores de las litologías que posean intervalos repetidos dentro de un mismo pozo. Esto significa que no debe existir más de una litología con los mismos valores en los campos “desde” y “hasta” que conforman el intervalo de un pozo.
fn_intervalos_repetidos_inclinometria	Crear un listado de errores que contiene los identificadores de las inclinometrías que posean intervalos repetidos dentro de un mismo pozo. Esto significa que no debe existir más de una inclinometría con los mismos valores en los campos “desde” y “hasta” que conforman el intervalo de un pozo.
fn_intervalos_negativos_muestra	Identificar aquellas muestras que tengan intervalos negativos, es decir que el resultado de restar el valor del campo “desde” al campo

	<p>“hasta” sea un número negativo. Desde el punto de vista de la minería es imposible que exista una muestra cuyo valor de inicio del intervalo donde fue hallada sea mayor que el valor final del intervalo.</p>
fn_intervalos_negativos_litologia	<p>Identificar aquellas litologías que tengan intervalos negativos, es decir que el resultado de restar el valor del campo “desde” al campo “hasta” sea un número negativo. Desde el punto de vista de la minería es imposible que exista una litología cuyo valor de inicio del intervalo donde fue hallada sea mayor que el valor final del intervalo.</p>
fn_intervalos_negativos_inclinometria	<p>Identificar aquellas inclinometrías que tengan intervalos negativos, es decir que el resultado de restar el valor del campo “desde” al campo “hasta” sea un número negativo. Desde el punto de vista de la minería es imposible que exista una inclinometría cuyo valor de inicio del intervalo donde fue hallada sea mayor que el valor final del intervalo.</p>
fn_intervalos_solapados_muestra	<p>Listar los intervalos solapados existentes en las muestras dentro de un mismo pozo. Los intervalos deben ser consecutivos de forma que ninguno de los valores correspondiente a un intervalo pertenezca a otros intervalos.</p>
fn_intervalos_solapados_litologia	<p>Listar los intervalos solapados existentes en las litologías dentro de un mismo pozo. Los intervalos deben ser consecutivos de forma que ninguno de los valores correspondiente a un intervalo pertenezca a otros intervalos.</p>
fn_intervalos_solapados_inclinometria	<p>Listar los intervalos solapados existentes en las inclinometrías dentro de un mismo pozo.</p>

	Los intervalos deben ser consecutivos de forma que ninguno de los valores correspondiente a un intervalo pertenezca a otros intervalos.
--	---

Las siguientes funciones fueron desarrolladas con el objetivo de convertir los valores geográficos de diferentes campos de la base de datos en valores matemáticos que utiliza la aplicación para la realización de cálculos necesarios para la visualización, posibilitando al usuario observar y analizar la información minera almacenada en la base de datos a través de gráficos tridimensionales.

Tabla 3. Funciones de conversión implementadas.

Nombre	Objetivo y descripción
fn_convertir_profundidad	Convertir el valor geográfico del campo profundidad en un valor matemático.
fn_convertir_azimuth	Convertir el valor geográfico del campo azimuth ¹⁷ en un valor matemático.
fn_calcular_coordenadas	Calcular coordenadas esféricas. Recibe por parámetros un punto inicial, los ángulos profundidad y azimuth además de un valor de radio que se obtiene de la resta de los valores de los campos “hasta” y “desde”, devuelve un punto final. Para obtener este resultado se utilizó una de las funciones que brinda PostGIS para crear una coordenada tridimensional, la función: makepoint.
fn_calcular_coordenadas_muestra	Calcular los valores geométricos de los campos “desde” y “hasta” de las muestras. Esta función utiliza el resultado devuelto de las funciones fn_calcular_coordenadas, fn_convertir_profundidad y fn_convertir_azimuth. Con su utilización se logra obtener valores matemáticos y almacenarlos en los campos desde_geom y hasta_geom de la tabla tb_muestra de la base de datos.

¹⁷ Es la distancia angular medida en sentido a las agujas del reloj desde un punto cualquiera hacia el norte terrestre. Un par de valores dados en coordenadas de azimuth y altitud indican la posición de un objeto en el espacio local en un momento dado.

fn_calcular_coordenadas_litologia	Calcular los valores geométricos de los campos “desde” y “hasta” de las litologías. Esta función utiliza el resultado devuelto de las funciones fn_calcular_coordenadas, fn_convertir_profundidad y fn_convertir_azimuth. Con su utilización se logra obtener valores matemáticos y almacenarlos en los campos desde_geom y hasta_geom de la tabla tb_litologia de la base de datos.
fn_calcular_coordenadas_inclinometria	Calcular los valores geométricos de los campos “desde” y “hasta” de las inclinometrías. Esta función utiliza el resultado devuelto de las funciones fn_calcular_coordenadas, fn_convertir_profundidad y fn_convertir_azimuth. Con su utilización se logra obtener valores matemáticos y almacenarlos en los campos desde_geom y hasta_geom de la tabla tb_inclinometria de la base de datos.

Los siguientes índices espaciales son índices del tipo GIST que brinda PostGIS y se construyeron sobre campos geométricos de la base de datos con el objetivo de optimizar el tiempo de respuesta en consultas realizadas sobre esos campos geométricos.

Tabla 4. Índices espaciales desarrollados.

Nombre	
indice_tb_muestra_hasta_geom	Índice aplicado sobre el campo desde_geom de la tabla tb_muestra.
indice_tb_muestra_desde_geom	Índice aplicado sobre el campo hasta_geom de la tabla tb_muestra.
indice_tb_punto_coord_geom	Índice aplicado sobre el campo coord_geom de la tabla tb_punto.
indice_tb_poligono_poligono_geom	Índice aplicado sobre el campo poligono_geom de la tabla tb_poligono.
indice_tb_polilinea_polilinea_geom	Índice aplicado sobre el campo polilinea_geom de la tabla tb_polilinea.
indice_tb_triangulos_triangulo_geom	Índice aplicado sobre el campo triangulo_geom de la tabla tb_triangulos.

2.2.4 Normalización de la base de datos

“La normalización es un término que deriva de la metodología que se utiliza para evitar la redundancia de datos y el fácil acceso y actualización de estos”, es una definición otorgada por (ROZIC 2004). La metodología a la que se refiere el autor antes citado consiste en definir un conjunto de normas encaminadas a eliminar redundancias e inconsistencias de dependencia en el diseño de las tablas que conforman la base de datos. En total se han definido 5 formas normales y la Forma Normal de Boyce-Codd.

El autor (KROENKE 2003) plantea que el término 1ra Forma Normal describe una relación en la cual:

- Todos los atributos claves están definidos.
- No existen grupos repetidos en la tabla.
- Todos los atributos son dependientes de la clave primaria.

La 2da Forma Normal según los autores (ROB 2003) describe una relación que:

- Está en 1ra Forma Normal.
- Todos sus atributos no primos tienen dependencias funcionales total respecto a cada una de las claves.

La 3ra Forma Normal describe una relación que:

- Está en 2da Forma Normal.
- Ningún atributo no primo depende transitivamente de ninguna clave.

Pueden existir algunos casos en que la 3ra Forma Normal no satisfaga totalmente los criterios de normalización en los casos que la relación posea más de una clave candidata y para solucionar este inconveniente se definió la Forma Normal de Boyce Codd. Una relación la cumple sí y sólo sí un determinante es una clave candidata.

La 4ta Forma Normal se aplica para dependencias multivaluadas. Una relación se encuentra en este nivel si cumple las siguientes condiciones:

- Se encuentra en la Forma Normal de Boyce Codd.

- Todas las dependencias multivaluadas de dicha relación son por defecto dependencias funcionales.

Una relación se encuentra en la 5ta Forma Normal si se satisface que toda dependencia de reunión es consecuencia de las claves candidatas de la relación. Estas son las consideraciones para la 4ta y 5ta Forma Normal planteadas por el autor (ROZIC 2004).

Al diseño propuesto en esta investigación de una base de datos espacial para depósitos minerales le han sido aplicadas las reglas de normalización hasta alcanzar el 3er nivel, es decir que el diseño de la base de datos resultante cumple con las especificaciones expuestas anteriormente para la 1ra, 2da y 3ra Forma Normal. De esta manera se obtienen varios resultados positivos ya que se logra una mejor estructuración entre las relaciones que conforman la base de datos. Además se posibilita una recuperación sencilla de los datos en respuesta a las solicitudes de consultas o reportes y se simplifica el mantenimiento de los datos al actualizarlos, insertarlos o borrarlos.

2.3 Conclusiones parciales

En el presente capítulo se puede arribar a las siguientes conclusiones:

- El diseño propuesto a través de la creación de funciones de validación garantiza que sean identificados y solucionados los errores que posee la información minera, convirtiéndola en información lógica y fiable a la hora de realizar un análisis.
- Con la creación de los índices espaciales que componen el diseño físico propuesto se garantiza una excelente recuperación de los datos espaciales de la base de datos y se optimiza el tiempo de búsqueda en la información geométrica almacenada.
- Con la utilización de la extensión espacial PostGIS se pudo diseñar una solución que almacena tanto los datos socioeconómicos como los datos espaciales de un depósito mineral además de que se puede almacenar en la base de datos información bidimensional y tridimensional.

Capítulo 3: Validación de la propuesta de diseño

A continuación en el desarrollo del capítulo 3 se abordan los elementos que componen una validación teórica del diseño de una base de datos así como los aspectos fundamentales para realizar una validación funcional del diseño que se propone en la presente investigación. Para lograrlo se plantean los siguientes pasos: Definir las características de integridad, redundancia y seguridad así como la trazabilidad de acciones respecto a la información. Definir y aplicar pruebas de volumen, desempeño y carga a la base de datos espacial para depósitos minerales.

3.1 Validación teórica del diseño

3.1.1 Integridad

Cuando se modela información real a un nivel conceptual es necesario tener en cuenta un grupo de condiciones que aseguren que el resultado de la transformación es lo más exactamente posible a la realidad. En este proceso además de identificar los datos y las relaciones entre sí, se necesita aplicar un grupo de reglas que posibiliten la integridad de la información. Lograr preservar la información completamente íntegra es el resultado de una correcta interpretación y aplicación de las reglas establecidas para estos fines.

Restricciones de Integridad en bases de datos relacionales que propone (DATE 2001):

Datos Requeridos: Establece que una columna tenga un valor no nulo. Se define efectuando la declaración de una columna como NOT NULL cuando la tabla que contiene las columnas se crea por primera vez, como parte de la sentencia CREATE TABLE.

Chequeo de Validez: Cuando se crea una tabla cada columna tiene un tipo de dato y el SGBD asegura que solamente los datos del tipo especificado sean ingresados en la tabla.

La Integridad de entidades: Exige que los atributos que componen una llave primaria tengan que estar definidos.

La Integridad referencial: Los atributos que hacen referencia a una llave primaria de otra relación tienen que tomar uno de los valores definidos para esa llave primaria o estar indefinidos (relaciones padre/hijo).

3.1.2 Análisis de redundancia de la información

Una vez constituido el diseño de una base de datos se le debe realizar una revisión exhaustiva y verificar que este no almacene datos repetidos que puedan provocar inconsistencia en la información almacenada dentro de la base de datos. Los datos repetidos contemplados dentro del diseño también pueden influir negativamente en el rendimiento y la velocidad de respuesta de las aplicaciones que trabajen con la base de datos a la que se le aplique dicho diseño. Al almacenamiento de información repetida se lo conoce como redundancia de la información y precisamente eliminar esa redundancia es el objetivo fundamental de aplicar las reglas de normalización a las bases de datos.

Aplicando las diferentes reglas de normalización a un diseño de bases de datos se pueden eliminar completamente o al menos obtener una mínima ocurrencia de redundancia de información. Lo más óptimo sería lograr una redundancia de los datos nula pero esta condición está sujeta a determinadas circunstancias. En algunos casos la complejidad de los cálculos que se necesitan implementar obliga al diseñador de la base de datos a utilizar la redundancia para obtener consultas más simples y por consecuencia ganar un mejor tiempo de respuesta de las mismas. En el presente diseño de una base de datos espacial para depósitos minerales una vez aplicada la normalización de la base de datos se logró la eliminación de la redundancia de la información almacenada.

3.1.3 Seguridad y control de acceso

En PostgreSQL existen tres niveles de acceso:

El nivel 0 es el que se encarga de las máquinas (host) y los usuarios. Es decir es el que permite configurar las máquinas y/o usuarios que se pueden conectar a la base de datos. Para la realización de estas operaciones se utilizan las opciones del fichero de configuración `pg_hba.conf`.

El nivel 1 es el que se encarga de los usuarios y las bases de datos. Es el que permite configurar a que bases de datos se puede conectar un usuario o un determinado grupo de usuarios. Se puede configurar utilizando las opciones del fichero de configuración `pg_ident.conf`.

El nivel 2 es el que se encarga de las tablas. Es el que permite configurar a que tablas puede acceder un usuario o un determinado grupo de usuarios. Para esta configuración se utilizan los comandos `GRANT` para dar permisos y `REVOKE` para eliminar los permisos.

En el presente diseño se emplea el nivel 2 ya que las tablas correspondientes al esquema de nombre `gmsystem` solamente deben ser accedidas por un usuario con permisos especiales ya que estas son las tablas del catálogo que el sistema utiliza de forma interna.

Para garantizar la protección de los datos almacenados y para que la base de datos no sufra pérdida de información, PostgreSQL posibilita la realización de salvas mediante la utilización del comando `pg_dump`. A partir de la utilización del mismo se especifican algunos parámetros para indicar el nombre de la base de datos a la que se va a realizar la salva, el usuario y contraseña para conectarse, se especifica además el nombre que toma el archivo de salva y la dirección donde se desea ubicar el mismo.

La salva no se le puede realizar a una solamente, sino a varias bases de datos a la vez, además de que se puede realizar mientras está el servicio postgres funcionando. Para restaurar una salva se emplea el comando `pg_restore` y seguidamente se indica el archivo de salva que se desea abrir. Este proceso de salvas puede presentar algunas desventajas en cuanto al tiempo que demora salvar o restaurar, todo depende de la cantidad de información almacenada en la base de datos y del comportamiento del consumo de recursos del sistema.

3.1.4 Trazabilidad de acciones

La trazabilidad se puede definir como el conjunto de acciones que posibilitan rastrear desde un origen cuando es creado el producto hasta el momento de la entrega final al cliente. Consiste en la capacidad para reconstruir la historia, recorrido o aplicación de un determinado producto. Específicamente en el caso de las bases de datos se puede observar la trazabilidad cada vez que se insertan, actualizan o eliminan datos en la base de datos ya que se registran automáticamente las tuplas afectadas, el autor de la acción y el momento preciso en que realizó dicha acción. Con toda esta información se puede

seguir y controlar las acciones realizadas sobre los datos de un registro específico y por un usuario en particular logrando así contar con un historial para cada uno de los usuarios.

Los SGBD utilizan los logs como un mecanismo para seguir las trazas en una determinada base de datos. Los logs son empleados para registrar la información sobre quién, qué, cuándo y dónde, llevando cuenta sobre cada evento que ocurre durante un período de tiempo en particular, de esta manera los administradores de bases de datos cuentan con un archivo de evidencias que pueden consultar cuando lo necesiten. En el SGBD utilizado para la presente propuesta de diseño: PostgreSQL, los logs son registrados en archivos de texto almacenados en la carpeta pg_log que crea el mismo PostgreSQL cuando es instalado.

3.2 Validación funcional del diseño

3.2.1 Pruebas de volumen

Las pruebas de volumen se emplean para determinar si el sistema puede trabajar con grandes cantidades de datos, indicando los problemas y fallas que presentaría el software cuando los límites de la capacidad de información que puede almacenar la base de datos de dicho software son alcanzados. Las pruebas de volumen además identifican las cargas continuas o el volumen que el sistema puede manejar en un tiempo determinado. La prueba resulta exitosa finalmente cuando los límites del sistema son alcanzados o excedidos sin que el software llegue a fallar.

Se tuvieron en cuenta para aplicarle pruebas de volumen 6 tablas que almacenan datos espaciales como son tb_collar, tb_muestra, tb_litologia, tb_inclinometria, tb_punto y la tabla tb_triangulos. Para la realización de estas pruebas se investigaron algunas herramientas que posibilitan el llenado voluminoso de la base de datos de forma dinámica. Tras un análisis de las herramientas se concluyó que no soportan tipos de datos para objetos geométricos tridimensionales por lo que se pobló de forma manual la base de datos utilizando las opciones para importar datos con que cuenta la herramienta Geolmin como muestra la figura 7. De esta forma fueron introducidos a la base de datos un total de 1595569 tuplas utilizando ficheros con datos geológicos reales. La cantidad de datos introducidos por tabla quedó constituida como muestra la tabla 5.

Durante el proceso de pruebas no se detectaron limitaciones de capacidad en el almacenamiento de la información, ni problemas de volumen de datos. No se produjeron desbordamientos de matrices, columnas, atributos, tipos de datos, ni peticiones excesivas de memoria que paralizaran el SGBD.

Tampoco fueron detectadas irregularidades con los tipos de datos definidos en el diseño propuesto. Todos estos resultados aseguran que el diseño propuesto de una base de datos espacial para depósitos minerales soporta los requerimientos de volumen de información resultante de los procesos de prospección y exploración minera.

Tabla 5. Total de datos introducidos por tabla.

Nombre de la tabla	Cantidad de tuplas
tb_collar	6100
tb_muestra	52000
tb_litologia	5400
tb_inclinometria	2100
tb_punto	447026
tb_triangulos	1082943

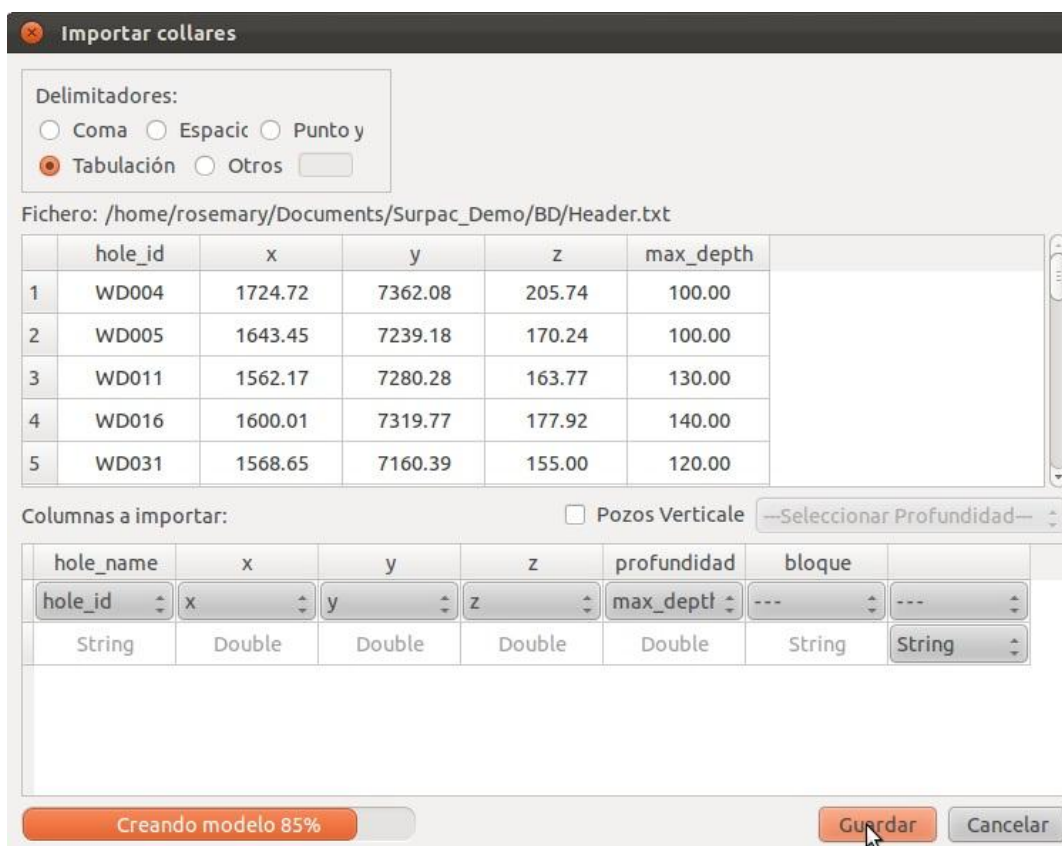


Figura 7. Funcionalidad importar datos de la herramienta Geolmin.

3.2.2 Pruebas de desempeño

Mediante las pruebas de desempeño se miden los tiempos de respuesta, las tasas de transacción y otros elementos sensibles al tiempo. La meta de las pruebas de desempeño es verificar y validar que los requerimientos de desempeño han sido alcanzados. Se considera que el cumplimiento de estas pruebas ha sido exitoso cuando no se encuentran fallas en los tiempos esperados o requeridos.

A continuación se muestra el resultado de la aplicación de pruebas de desempeño a la base de datos espacial para depósitos minerales. Con la aplicación de las mismas se midieron los resultados de los tiempos de respuesta obtenidos tras aplicar algunas de las consultas más frecuentes que se realizan a la base de datos, tanto para columnas indexadas como columnas no indexadas.

Tabla 6. Resultados de las pruebas de desempeño realizadas.

Consulta	Filas recuperadas	Tiempo de respuesta	
		Sin índices	Con índices
SELECT tb_triangulos.triangulo_geom, tb_triangulos.idtriangulo, tb_triangulos.idsuperficie, tb_superficies.nombre FROM public.tb_triangulos, public.tb_superficies WHERE tb_triangulos.idsuperficie = tb_superficies.idsuperficie AND tb_superficies.idsuperficie = 20;	81731	62159 ms	32981 ms

Consulta	Filas recuperadas	Tiempo de respuesta	
SELECT tb_puntos.nombre, tb_punto.idpunto, tb_punto.coord_geom, tb_punto.color, tb_punto.estilo FROM public.tb_punto, public.tb_puntos WHERE tb_puntos.idpuntos = tb_punto.idpuntos AND tb_puntos.idpuntos = 2;	6461	Sin índices	Con índices
		1173 ms	1146 ms
Consulta	Filas recuperadas	Tiempo de respuesta	
SELECT tb_collar.idcollar, tb_collar.coord_geom, tb_collar.dhid, tb_inclinometria.idinc, tb_inclinometria.azimuth,	35739	Sin índices	Con índices
		21097 ms	19217 ms

<pre> tb_inclinometria.profundidad, tb_inclinometria.desde_geom, tb_inclinometria.hasta_geom, tb_muestra.desde_geom, tb_muestra.hasta_geom, tb_muestra.idmtr FROM public.tb_collar, public.tb_inclinometria, public.tb_muestra WHERE tb_collar.idcollar = 613 AND tb_collar.idcollar = tb_muestra.idcollar AND tb_collar.idcollar = tb_inclinometria.idcollar; </pre>			
Consulta	Filas recuperadas	Tiempo de respuesta	
<pre> SELECT tb_collar.idcollar, tb_collar.dhid, tb_muestra.idmtr, </pre>	52000	Sin índices	Con índices
		3411 ms	3279 ms

<pre> tb_muestra.desde, tb_muestra.hasta FROM public.tb_muestra, public.tb_collar WHERE (tb_muestra.hasta- tb_muestra.desde) >= 0 AND tb_collar.idcollar = tb_muestra.idcollar; </pre>			
---	--	--	--

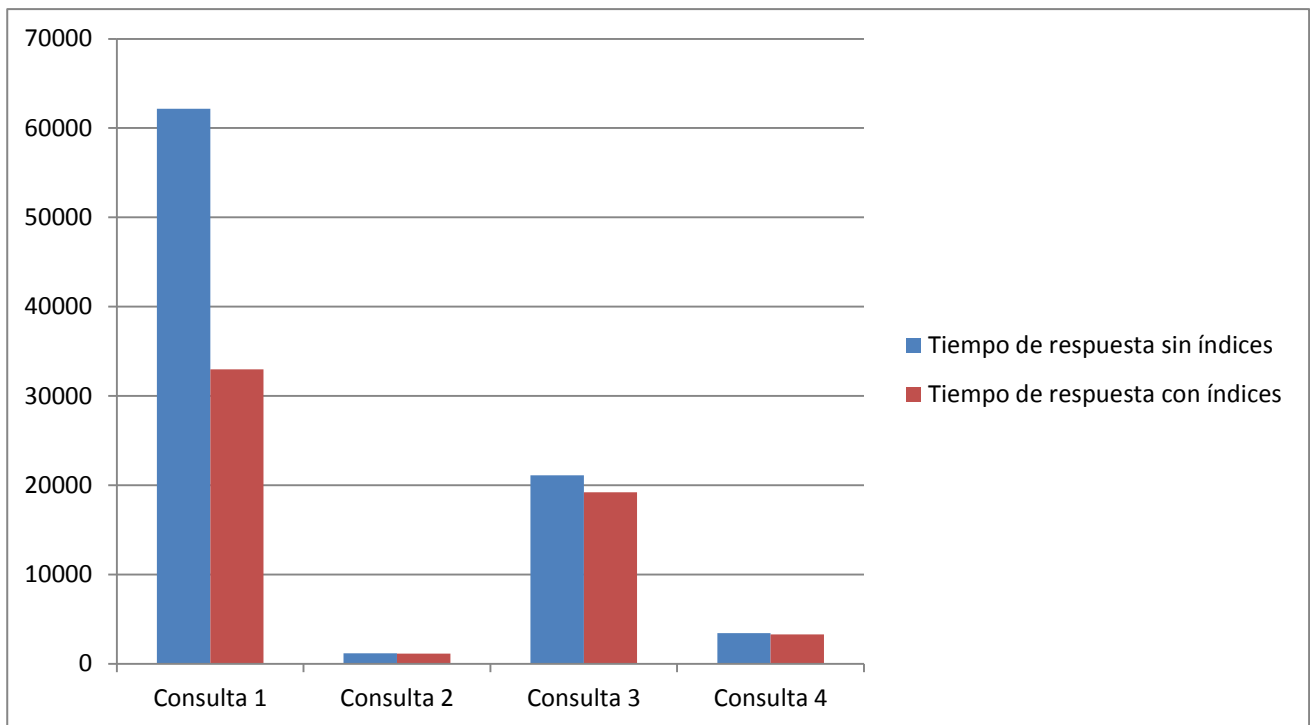


Figura 8. Representación gráfica de las pruebas de desempeño.

Al comparar los resultados que devolvieron las consultas realizadas se observa que existe una diferencia en cuanto a los tiempos de respuesta de las mismas, las consultas efectuadas sobre tablas

con índices espaciales fueron mucho más rápidas que las consultas sobre tablas sin índices espaciales. A partir de esta prueba se puede concluir que resulta ventajosa la utilización de índices espaciales en una base de datos espacial ya que se pone de manifiesto como la utilización de los mismos contribuyen a mejorar el rendimiento y la recuperación de la información.

3.2.3 Pruebas de carga

Las pruebas de carga se realizan para evaluar la habilidad que presenta el software para continuar funcionando adecuadamente y miden además las situaciones en las que el sistema se somete a variaciones en su carga de trabajo más allá de la carga de trabajo esperada. Adicionalmente, las pruebas evalúan las características de desempeño ya sean tiempos de respuestas, tasas de transacción y otros problemas sensibles al tiempo. Las pruebas de carga posibilitan probar el funcionamiento de la base de datos a través del grado de concurrencia de las consultas. Mediante la aplicación de esta prueba se puede comprobar el comportamiento de la base de datos frente a las consultas más frecuentes que se efectúan verificando que esta se comporte adecuadamente.

Para la realización de esta prueba se utilizó la herramienta Apache Jmeter en su versión 2.3.1. Apache Jmeter es una herramienta de carga desarrollada en Java que permite poner a prueba el funcionamiento y rendimiento de diversos recursos de software. Cuenta con dos componentes básicos: un plan de pruebas y los elementos que integran las diferentes etapas que se aplican en las pruebas a desarrollar. Estos elementos se organizan dentro de una estructura de árbol y de esta manera el entorno de trabajo del Apache Jmeter posibilita observarlos y editarlos para conformar el plan de pruebas que se desea.

Para poder utilizar el Apache Jmeter fue necesaria la instalación de Java Runtime Environment (JRE 6) en la estación cliente de donde se van a ejecutar las peticiones. Para establecer la conexión entre el Apache Jmeter y el SGBD PostgreSQL versión 9.1 fue necesaria la utilización del driver postgresql-9.1-901.jdbc3, del cual se explica más adelante cómo integrarlo al Apache Jmeter. El escenario creado para la aplicación de esta prueba se basa en una estación cliente donde se encuentra instalado el Apache Jmeter y una estación servidor donde se encuentra la base de datos, ambas estaciones conectadas directamente entre sí.

A continuación se describen las características del hardware utilizado para la estación cliente y la estación servidor:

Sistema Operativo: GNU Linux. Distribución Ubuntu 11.10

Memoria RAM: 1.0 GB

Microprocesador: Intel Pentium IV. Velocidad 3.0 GHz

La configuración del Apache Jmeter fue constituida de la siguiente manera:

- 1- Se adicionó al plan de pruebas un grupo de hilos donde se estableció el número de usuarios que en la simulación realizarían peticiones al servidor.
- 2- Se adicionó al plan de pruebas una configuración de conexión JDBC¹⁸, que es el tipo de conexión que se establece con una base de datos. Para configurar la conexión JDBC creada se especifica un nombre de la variable que es usada para las peticiones, se señala la dirección url mediante la cual se accede a la base de datos y la ubicación de donde se encuentra el driver, en este caso para establecer la conexión con la base de datos diseñada en PostgreSQL 9.1. Se especifican también el nombre de usuario y la contraseña que se necesitan para acceder a la base de datos.
- 3- Se adicionó al grupo de hilos una petición JDBC donde se introduce nuevamente el nombre de la variable que es usada para las peticiones, se especifica el tipo de consulta SQL que se va a realizar y se formula la consulta.
- 4- Por último se señala la petición JDBC anteriormente creada y en la barra de herramientas del Jmeter se selecciona la opción de comenzar.

El Apache Jmeter evalúa el comportamiento del plan de pruebas realizado y utiliza algunas variables para manifestar los resultados de tiempo obtenidos en la simulación. En la presente prueba solamente se va a realizar una valoración de los resultados a partir de la comprensión de 3 de esas variables. La variable Mínimo: representa el mínimo tiempo de respuesta de una petición, Máximo: representa el máximo tiempo de respuesta de una petición y Media: señala el tiempo promedio de respuesta de todas las peticiones.

Durante las pruebas la herramienta Apache Jmeter presentó algunos problemas para soportar los datos espaciales cuando se simulaban peticiones de más de 20 usuarios concurrentes. Para garantizar un resultado fiable y sin alteraciones se realizaron las pruebas simulando hasta 20 peticiones. Con la

¹⁸ *Java Database Connectivity*

realización de esta prueba se concluyó que la misma arrojó resultados satisfactorios, proporcionando un tiempo de respuesta aceptable en comparación con la gran cantidad de información almacenada.

Prueba 1:

Tabla 7. Resultados de la prueba 1.

Consulta	Total de usuarios concurrentes	Máximo(ms)	Mínimo (ms)	Media (ms)
SELECT * FROM tb_collar WHERE tb_collar.idcollar not in (select idcollar from tb_survey)	5	1476	1293	1355
	10	2875	2170	2710
	20	3960	2500	3495

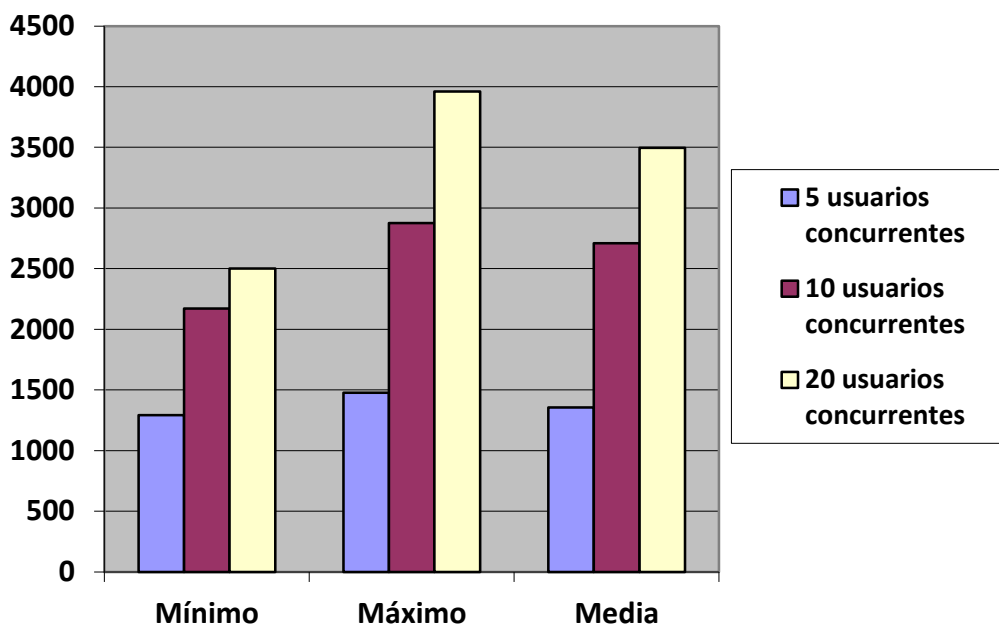


Figura 9. Representación gráfica de la prueba 1.

Prueba 2:

Tabla 8. Resultados de la prueba 2.

Consulta	Total de usuarios concurrentes	Máximo(ms)	Mínimo (ms)	Media (ms)
SELECT tb_collar.idcollar, tb_collar.coord_geom, tb_collar.dhid, tb_inclinometria.idinc, tb_inclinometria.azimuth, tb_inclinometria.profundidad, tb_inclinometria.desde_geom, tb_inclinometria.hasta_geom, tb_muestra.desde_geom, tb_muestra.hasta_geom, tb_muestra.idmtr FROM public.tb_collar, public.tb_inclinometria,	5	27557	27037	27557
	10	178326	177771	178326
	20	275238	273180	275038

public.tb_muestra				
WHERE				
tb_collar.idcollar = 613 AND				
tb_collar.idcollar =				
tb_muestra.idcollar AND				
tb_collar.idcollar =				
tb_inclinometria.idcollar;				

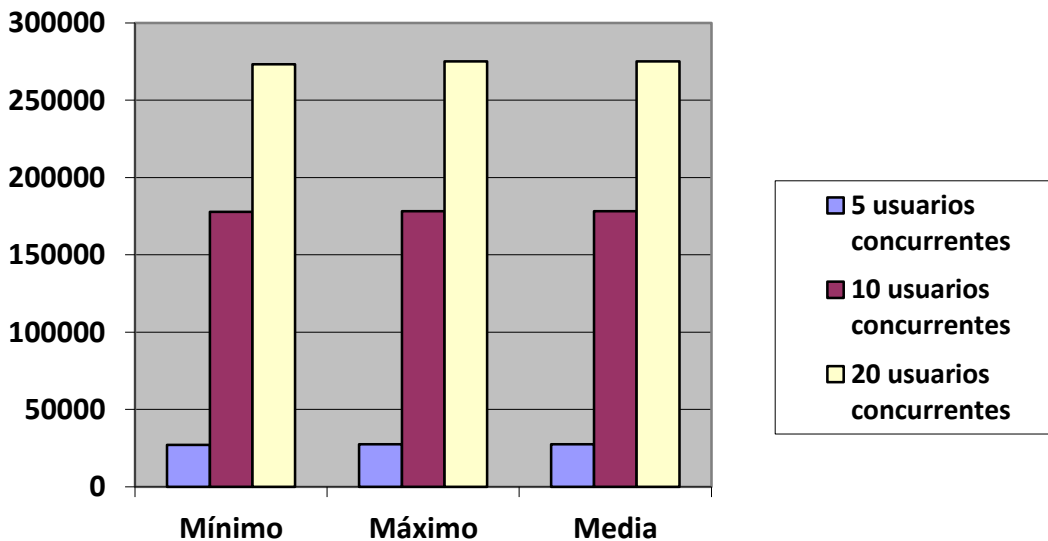


Figura 10. Representación gráfica de la prueba 2.

3.3 Conclusiones parciales

Partiendo de los resultados obtenidos mediante las pruebas realizadas en el presente capítulo se puede arribar a las siguientes conclusiones:

- El uso de índices espaciales contribuye en gran medida a optimizar la velocidad de las búsquedas que se realizan sobre la información espacial de la base de datos.

- El diseño propuesto soporta los volúmenes de información necesarios para que se realice de forma satisfactoria el almacenamiento de la información minera proveniente de un yacimiento mineral.
- El diseño propuesto soporta un número admisible de usuarios realizando peticiones de forma simultánea sobre la información almacenada sin que la estructura de almacenamiento llegue a colapsar.

Conclusiones Generales

En la presente investigación se realizó un estudio exhaustivo de los elementos que componen el diseño teórico y físico de una base de datos, con la particularidad de tratarse de una base de datos de tipo espacial. Para la confección del diseño se tuvieron en cuenta algunos elementos que posibilitaron resolver los problemas más comunes que presentan las entidades mineras en la actualidad. Se aplicó el uso de funciones de validación para detectar de forma rápida problemas de integridad en la información y el uso de índices espaciales para lograr una mejor velocidad en las búsquedas sobre la información espacial o para recuperar de manera ágil esa información cuando es consultada.

Se realizó un estudio de las herramientas más adecuadas para el manejo de la información espacial concretándose en la utilización de PostgreSQL y su extensión espacial PostGIS, herramientas libres, con una amplia gama de funcionalidades para el manejo de datos tanto bidimensionales como tridimensionales así como datos socioeconómicos y espaciales. El diseño propuesto cumple las reglas de normalización hasta la 3ra Forma Normal lo que posibilita eliminar posibles problemas de inconsistencias de los datos así como de redundancia de la información, lográndose un mejor rendimiento.

Luego de desarrollar la etapa de pruebas y analizar el comportamiento de la base de datos diseñada, poblada con datos mineros reales, se obtuvieron resultados satisfactorios por lo que se puede concluir que el diseño propuesto ha sido validado teórica y funcionalmente garantizando que el mismo presente la calidad necesaria y esperada. Se dio cumplimiento a los objetivos generales y específicos planteados, diseñándose e implementándose una base de datos espacial para depósitos minerales.

Recomendaciones

- Lograr un mayor rendimiento de la base de datos incorporando mejores prestaciones en el hardware del servidor donde esté alojada.
- Utilizar herramientas que se especialicen en datos espaciales para aplicar las pruebas y obtener resultados más efectivos.
- Aplicar además pruebas de stress a la base de datos para analizar su comportamiento ante situaciones desfavorables que puedan producirse.

Bibliografía Referenciada

1. ALONSO, E. M. *Herramientas CASE para el proceso de desarrollo de Software*. 2009. p.
2. ANALYSIS, N. C. F. G. I. A., 1998. [Disponible en: <http://ncgia.ucsb.edu>]
3. BELLO, A. A. A. *Diseño e implementación de la base de datos del producto GeolMin*. Ciudad Habana, Universidad de las Ciencias Informáticas, 2011. p.
4. CAMALLEA, N. L. N. *Gestión de Base de Datos con ADO.NET*. Ciudad de La Habana, Científico Técnica, 2004. p.
5. CODAZZI, I. G. A., 2010. [Disponible en: <http://www.igac.gov.co>]
6. DATE, C. J. *Introducción a los Sistemas de Bases de Datos*. Séptima Edición. 2001. p.
7. DE MIGUEL CASTAÑO, A., PIATTINI VELTHUIS, MARIO Y MARCOS MARTÍNEZ, ESPERANZA. *Diseño de bases de datos relacionales*. México, RAMA, 1999. p.
8. ESRI. 1995. [Disponible en: <http://www.esri.com>]
9. GARCÍA, R. M. M. *Diseño de Bases de Datos*. 1999. p.
10. GEMCOM. *GEMCOM for Windows*. 1998. p.
11. GÜTING, R. H. *An introduction to spatial database systems*. 1994. p.
12. HAITHCOAT, T. *Spatial Databases as Models of Reality*. Estados Unidos, University of Missouri Columbia, 1999. p.
13. INC, R. R., 2009. [Disponible en: <http://postgis.refractive.net>]
14. INC, R. R. *PostGIS 1.5.2 Manual*. 2005. p.
15. JIM COPLIEN, D. S., NORMAN KERTH. *Pattern Languages of Program Design*. 1996. p.
16. KOTHURI, R. *Oracle Spatial for Oracle Database*. 2007. p.
17. KROENKE, D. M. *Procesamiento de bases de datos: fundamentos, diseño e implementación*. Octava Edición. 2003. p.
18. MYSQL. 2012. [Disponible en: <http://dev.mysql.com/>]
19. OGC. 1994. [Disponible en: <http://www.opengeospatial.org>]
20. ORACLE. 2010. [Disponible en: <http://www.oracle.com>]
21. P. RAGAUX, M. S., AND A. VOISARD. *Spatial Databases: with Application in GIS*. 2002. p.
22. POSTGRESQL. 1996. [Disponible en: <http://www.postgresql.org>]
23. PROJECTS, T. G.-S. F., 2011. [Disponible en: <http://www.gaia-gis.it>]
24. ROB, P. Y. C., CARLOS. *Sistemas de bases de datos: diseño, implementación y administración*. 5ta edición. 2003. p.
25. ROZIC, S. E. *Bases de Datos y su aplicación con SQL*. Buenos Aires, MP Ediciones, 2004. p.
26. SÁNCHEZ, J. *Diseño Conceptual de Bases de Datos guía de aprendizaje*. 2004. p.

27. SHEKHAR, S. A. S. C. *Spatial Databases: A Tour*. Estados Unidos, Pearson Education Inc, 2003. p.
28. SQLITE. 2010. [Disponible en: <http://www.sqlite.org>
29. VOISARD, A. Y. D., B. *A database perspective on geospatial data modeling*. IEEE Transactions on Knowledge and Data Engineering, 2002. p.

Bibliografía Consultada

1. ALMENARES, L.S., *Cómo realizar Pruebas de Carga y Estrés en JMeter*. 2008.
2. ALONSO, E.M., *Herramientas CASE para el proceso de desarrollo de Software*. 2009.
3. ANALYSIS, N.C.F.G.I.A. 1998; Available from: <http://ncgia.ucsb.edu>.
4. BÁEZ, I.V., *Desarrollo del módulo de base de datos para un SCADA*. 2010, Universidad de las Ciencias Informáticas: Ciudad de La Habana.
5. BELLO, A.A.A., *Diseño e implementación de la base de datos del producto Geolmin*. 2011, Universidad de las Ciencias Informáticas: Ciudad Habana.
6. CAMALLEA, N.L.N., *Gestión de Base de Datos con ADO.NET*. 2004, Ciudad de La Habana: Científico Técnica.
7. CODAZZI, I.G.A. 2010; Available from: <http://www.igac.gov.co>.
8. DATE, C.J., *Introducción a los Sistemas de Bases de Datos*. Séptima Edición ed. 2001.
9. DE MIGUEL CASTAÑO, A., PIATTINI VELTHUIS, MARIO Y MARCOS MARTÍNEZ, ESPERANZA, *Diseño de bases de datos relacionales*. 1999, México: RAMA.
10. ESRI. 1995; Available from: <http://www.esri.com>.
11. GARCÍA, R.M.M., *Diseño de Bases de Datos*. 1999.
12. GARCÍA, Y.S., *Diseño de la base de datos para el Grupo de Calidad de la facultad 9*. 2010, Universidad de las Ciencias Informáticas: Ciudad Habana.
13. GEMCOM, *GEMCOM for Windows*. 1998.
14. GROUP, T.P.G.D., *PostgreSQL 8.4.11 Documentation*. 2009.
15. GUTIÉRREZ, M., *El Rol de las Bases de Datos Espaciales en una Infraestructura de Datos*. 2006, Santiago de Chile.
16. GÜTING, R.H., *An introduction to spatial database systems*. 1994.
17. HAITHCOAT, T., *Spatial Databases as Models of Reality*. 1999, University of Missouri Columbia: Estados Unidos.
18. HERNÁNDEZ LEÓN, R.A.y.C.G., SAYDA, *El proceso de Investigación Científica*. 2011, Ciudad de La Habana: Editorial Universitaria.
19. HERNÁNDEZ SAMPIERI, R., FERNÁNDEZ COLLADO, CARLOS Y BAPTISTA LUCIO, PILAR, *Metodología de la Investigación*. Segunda edición ed. 1998, México D. F.
20. INC, R.R., *PostGIS 1.5.2 Manual*. 2005.
21. INC, R.R. 2009; Available from: <http://postgis.refrations.net>.
22. JIM COPLIEN, D.S., NORMAN KERTH, *Pattern Languages of Program Design*. Vol. Vol. II. 1996.

23. KOTHURI, R., *Oracle Spatial for Oracle Database*. 2007.
24. KROENKE, D.M., *Procesamiento de bases de datos: fundamentos, diseño e implementación*. Octava Edición ed. 2003.
25. MÉNDEZ ROLDÁN, I.Y.B.R., GUILLERMO, *Las Bases de Datos Espaciales (BDE) como nueva componente en los Sistemas de Información*. 2009.
26. MySQL. 2012; Available from: <http://dev.mysql.com/>.
27. OGC. 1994; Available from: <http://www.opengeospatial.org>.
28. ORACLE. 2010; Available from: <http://www.oracle.com>.
29. P. RAGAUX, M.S., AND A. VOISARD, *Spatial Databases: with Application in GIS*. 2002.
30. POSTGRESQL. 1996; Available from: <http://www.postgresql.org>.
31. PROJECTS, T.G.-S.F. 2011; Available from: <http://www.gaia-gis.it>.
32. ROB, P.Y.C., CARLOS, *Sistemas de bases de datos: diseño, implementación y administración*. 5ta edición ed. 2003.
33. RODRÍGUEZ TORRES, A., *Bases de datos espaciales Spatial Databases*. 2009.
34. ROZIC, S.E., *Bases de Datos y su aplicación con SQL*. 2004, Buenos Aires: MP Ediciones.
35. SÁNCHEZ, J., *Diseño Conceptual de Bases de Datos guía de aprendizaje*. 2004.
36. SHEKHAR, S.A.S.C., *Spatial Databases: A Tour*. 2003, Estados Unidos: Pearson Education Inc.
37. SQLITE. 2010; Available from: <http://www.sqlite.org>.
38. VOISARD, A.Y.D., B, *A database pespective on geospatial data modeling*. 2002: IEEE Transactions on Knowledge and Data Engineering.
39. ZAYAS, C.A.D., *Metodología de la Investigación Científica*. 1995, Santiago de Cuba.

Anexos

Anexo 1: Diccionario de datos de las tablas básicas

Nombre de la tabla	tb_collar			
Alias	Collar			
Campos	Tipo de Datos		Restricciones	Descripción
	Tipo	Tamaño		
idcollar	int	8	Llave primaria	Es el campo que almacena el identificador único de los pozos.
dhid	varchar	20	único	Es el campo que almacena el nombre único de los pozos.
coord_geom	geometry			Es el campo que almacena las coordenadas espaciales del pozo.

Nombre de la tabla	tb_inclinometria			
Alias	Inclinometría			
Campos	Tipo de Datos		Restricciones	Descripción
	Tipo	Tamaño		
idinc	int	8	Llave primaria	Es el campo que almacena el identificador de la

				inclinometría.
idcollar	int	8	Llave foránea	Es el campo que almacena el identificador único de los pozos.
desde	double			Es el campo que almacena el valor desde el cual comienza el intervalo de la inclinometría.
hasta	double			Es el campo que almacena el valor en el cual termina el intervalo de la inclinometría.
azimuth	double			Es el campo que contiene el ángulo o longitud de arco.
dip	double			Es el campo que contiene el ángulo que forma el plano a medir con respecto a un plano horizontal, y debe ir acompañado por el sentido en el que el plano buza o baja.
desde_geom	geometry			Es el campo que almacena el valor espacial desde el cual comienza el intervalo de la inclinometría.
hasta_geom	gometry			Es el campo que almacena el valor espacial en el cual

				termina el intervalo de la inclinometría.
--	--	--	--	---

Nombre de la tabla	tb_litologia			
Alias	Litología			
Campos	Tipo de Datos		Restricciones	Descripción
	Tipo	Tamaño		
idlito	int	8	Llave primaria	Es el campo que almacena el identificador de la litología.
idcollar	int	8	Llave foránea	Es el campo que almacena el identificador único de los pozos.
codigo	varchar	20		Es el campo que almacena el código de la litología.
descripcion	varchar	20		Es el campo que almacena una descripción de la litología.
desde	double			Es el campo que almacena el valor desde el cual comienza el intervalo de la litología.
hasta	double			Es el campo que almacena el valor en el cual termina el intervalo de la litología.
desde_geom	geometry			Es el campo que almacena el

				valor espacial desde el cual comienza el intervalo de la litología.
hasta_geom	geometry			Es el campo que almacena el valor espacial en el cual termina el intervalo de la litología.

Nombre de la tabla	tb_muestra			
Alias	Muestra			
Campos	Tipo de Datos		Restricciones	Descripción
	Tipo	Tamaño		
idmtr	int	8	Llave primaria	Es el campo que almacena el identificador de la muestra.
idcollar	int	8	Llave foránea	Es el campo que almacena el identificador único de los pozos.
desde	double			Es el campo que almacena el valor desde el cual comienza el intervalo de la muestra.
hasta	double			Es el campo que almacena el valor en el cual termina el intervalo de la muestra.

desde_geom	geometry			Es el campo que almacena el valor espacial desde el cual comienza el intervalo de la muestra.
hasta_geom	geometry			Es el campo que almacena el valor espacial en el cual termina el intervalo de la muestra.

Glosario de términos:

Herramienta CASE: (Computer Aided Software Engineering, Ingeniería de Software Asistida por Ordenador) Las Herramientas CASE son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero.

ISO: Organización Internacional para la Estandarización, es el organismo encargado de promover el desarrollo de normas internacionales de fabricación, comercio y comunicación para todas las ramas industriales a excepción de la eléctrica y la electrónica.

Linux: Núcleo o kernel del sistema operativo libre denominado GNU/Linux. Lanzado bajo la licencia pública general (GPL - General Public License) de GNU y desarrollado gracias a contribuciones provenientes de todo el mundo.

Llave Foránea: (foreign key o clave foránea) Es aquella columna que existiendo como dependiente en una tabla, es a su vez clave primaria en otra tabla.

Llave Primaria: Llave con valores únicos, es decir, no ocurren más de una vez en el atributo.

Modelo entidad-relación: Es una técnica para el modelado de datos utilizando diagramas de entidad-relación.

Normalización: Proceso de reducción sobre una estructura de datos que procura aumentar la integridad, disminuir la redundancia y las dependencias funcionales de esa estructura.

Open source: Código abierto es el término con el que se conoce al software distribuido y desarrollado libremente.

Sistema Operativo: Es un conjunto de programas destinados a permitir la comunicación del usuario con un computador y gestionar sus recursos de una forma eficaz.

Software: Es el conjunto de los programas de cómputo, procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema de computación.

SQL: (Structured Query Language) Es un lenguaje de acceso a las bases de datos, permite especificar todas las operaciones sobre la base de datos como por ejemplo: Inserción, Borrado, Actualización. Utiliza características de álgebra y cálculo relacional permitiendo de esta forma realizar consultas a la base de datos de forma sencilla.

Trazabilidad: Aptitud de reconstruir la historia, la utilización o la localización de un producto por medio de identificaciones registradas.

Tupla: Es una hilera o fila en una tabla.