

Universidad de las Ciencias Informáticas



Facultad 5

TÍTULO: Base de Datos del Sistema para el Manejo Integral de la Perforación de Pozos de Petróleo y Gas (SIPP). Versión 2.0.

Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas.

Autor: Yordan Gallardo Avilés.

Tutores: Msc. Yaneisis Pérez Heredia.

Ing. Yunier Alexander Pimienta.

Co-tutor: Ing. David Tavares Cuevas.

Año del 52 Aniversario del Triunfo de la Revolución.



MAYO 2011

DECLARACIÓN DE AUTORÍA

Declaración de autoría:

Declaro que soy el único autor de este trabajo y autorizo a la Facultad 5 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Yordan Gallardo Avilés.
Estudiante.

Msc. Yaneisis Pérez Heredia.

Tutor

Ing. Yunier A. Pimienta Fernández

Tutor



La Universidad de las Ciencias Informáticas (UCI), específicamente el Centro de Desarrollo Industrial (CEDIN) perteneciente a la Facultad #5, conjuntamente con la dirección de CUPET, desarrolla un Sistema para el Manejo Integral de la Perforación de Pozos de Petróleo y Gas (SIPP). La investigación persigue diseñar un modelo de base de datos que elimine las deficiencias del modelo que soporta la versión 1.0, en cuanto a su organización, integridad, e independencia de los datos. Se obtiene como resultado una base de datos normalizada hasta la tercera forma normal y con una minimización de los datos redundantes. Aportando al sistema un modelo con alta estructuración de los datos y organización de los mismos. Además se propone una serie de mejoras para una futura evolución del sistema de almacenado de la información de SIPP.

Palabras Clave: base de datos, información, perforación, pozo de petróleo.



Índice de Contenido

Introducción.1

Capítulo 1: Fundamentación Teórica.9

 1.1. **Introducción.9**

 1.2. **Conceptos asociados al Sistema para el Manejo Integral de Perforación de Pozos de Petróleo y Gas (SIPP).9**

 1.3. **Descripción de procesos de negocio para el Sistema para el Manejo Integral de Perforación de Pozos de Petróleo y Gas (SIPP).14**

 1.4. **Proceso de gestión y control de la información de perforación de pozos de petróleo.17**

 1.5. **Características y arquitectura de los Sistemas de Base de Datos.18**

 1.5.1. **Características de los Sistemas de Bases de Datos (SBD).19**

 1.5.2. **Arquitectura de los Sistemas de Bases de Datos (SBD).21**

 1.6. **Modelos de Base de Datos.25**

 1.6.1. **Modelo de Base de Datos Relacional.27**

 1.6.2. **Elementos de una base de datos relacional.31**

 1.7. **Conclusiones Parciales.32**

Capítulo 2: Descripción y Análisis de la Solución Propuesta.33

 2.1. **Introducción.33**

 2.2. **Sistema Gestor de Base de Datos. PostgreSQL.33**

 2.3. **Herramienta de Diseño. Visual Paradigm.37**

 2.4. **Herramienta de Desarrollo. Pg Admin III.38**

 2.5. **Herramienta para prueba de volumen de la base de datos. EMS Data Generator 2005 para PostgreSQL.39**

 2.6. **Herramientas para replicación de base de datos.40**

 2.7. **Descripción de la Arquitectura propuesta y fundamentación.45**

 2.7.1. **Soporte Arquitectónico.46**

 2.7.2. **Estándares de Nomenclatura.47**

 2.8. **Diseño de la Base de datos SIPP versión 2.0.49**

 2.8.1. **Diagrama de Clases Persistentes.49**

 2.8.2. **Diagrama Entidad-Relación de la base de datos propuesta.50**

 2.9. **Conclusiones parciales.51**

Capítulo 3: Validación teórica del diseño.52

 3.1. **Introducción.52**

 3.2. **Integridad.52**

 3.3. **Normalización de la base de datos.57**

 3.4. **Análisis de la redundancia de la información.62**

 3.5. **Análisis de la seguridad y recuperación de la base de datos.63**

 3.6. **Análisis de la Replicación de datos.66**

 3.7. **Análisis de Prueba de volumen.67**



MAYO 2011

ÍNDICE DE CONTENIDO

3.8.	Trazabilidad de las acciones.....	69
3.9.	Análisis de los resultados obtenidos.....	70
3.10.	Conclusiones Parciales.....	71
	Conclusiones Generales.	72
	Bibliografía.....	73



Introducción.

La información ha sido siempre un recurso muy valorado. Desde las primeras civilizaciones se ha archivado, manipulado y creado grandes volúmenes de la misma en bibliotecas, monasterios o por determinadas personas. Sin embargo, nunca como hoy, ha tenido tanto valor la información, lo que ha provocado que en unos pocos años esta se haya convertido en un eslabón fundamental para el desarrollo de cualquier sociedad.

La explosión de las tecnologías de redes, la aparición de novedosos servicios, el acercamiento entre culturas y las necesidades de comunicación que estas generan han cambiado las relaciones entre los hombres. Se encuentra el hombre actual en una sociedad donde las Tecnologías de la Información y las Comunicaciones (TICs) han llegado a ser figura representativa de su cultura, a tal punto que para designar el marco de su convivencia se alude reiteradamente a la expresión “sociedad de la información”¹; y detrás de ese desarrollo tecnológico descansa la información como objeto de estudio de dicha revolución.

Hoy, la mayoría de los países buscan soluciones rentables a sus problemas, problemas que pueden ser tanto económicos, como la gestión empresarial, como sociales, en sectores educativos o de salud, apoyándose principalmente en estas tecnologías, ya que son una vía importante para llevar el control de grandes volúmenes de información. Con el desarrollo y optimización de las TICs surge la necesidad de almacenar y gestionar los datos que se capturaban o generaban mediante software o aplicaciones que se hacen cada vez más potentes debido a dichos avances. Esta necesidad dio paso al surgimiento de las bases de datos. Se podría considerar como una base de datos a cualquier recopilación organizada de información sobre la que haya habido análisis documental y que disponga de un sistema de búsqueda específica [1].

¹ *Estado de desarrollo social caracterizado por la capacidad de sus miembros para obtener y compartir cualquier información, instantáneamente, desde cualquier lugar y en la forma que se prefiera.*



Las bases de datos tuvieron sus orígenes en 1960 - 1962, cuando se empezaron a usar las máquinas que codificaban la información en tarjetas perforadas por medio de agujeros. Se crearon con el objetivo de almacenar inmensas cantidades de datos que antes se almacenaba en libros, lo que era lento, costoso y complejo cualquier actualización a realizar.

Antiguamente, los predecesores de los sistemas de bases de datos fueron los sistemas de ficheros. No hay un momento concreto en que los sistemas de ficheros hayan cesado para dar comienzo a los sistemas de bases de datos. Cuando los ordenadores evolucionan, aparecen las cintas y los discos, a la vez que las máquinas son dotadas de mucha más potencia y facilidad de manipulación, es por tanto en ese momento cuando las bases de datos comienzan a ser realmente útiles.

A medida que aumenta la informatización, los softwares de bases de datos experimentan un desarrollo vertiginoso, lo que ha traído consigo rapidez, efectividad y menos costos en los procesos de grandes flujos de información. Todo esto para satisfacer las necesidades a la hora de optimizar servicios y productos. Hoy se le presta gran importancia a la gestión de la información a través de los gestores de bases de datos. Éstos permiten manejar la información de modo sencillo ya que presentan excelentes funcionalidades para el desarrollo y manejo de bases de datos; con las que se optimizan en gran medida la fluidez de los procesos, la toma de decisiones y la calidad del trabajo en las diferentes organizaciones.

El país no está ajeno a estas demandas, hoy se trabaja de manera acelerada en busca de soluciones informáticas que agilicen el trabajo de las instituciones y empresas cubanas. La Oficina Central de Exploración y Producción de Cuba perteneciente a la empresa CUBA-PETROLEO (CUPET), se encuentra inmersa en un proceso de expansión de sus principales procesos productivos.



Las expectativas que se plantean sobre las posibilidades reales del país de la exploración, perforación y producción de petróleo en las aguas profundas del Golfo de México, han propiciado un ambiente favorable para la realización de convenios de trabajos entre la Universidad de las Ciencias Informáticas (UCI) y CUPET, debido a la imperiosa necesidad de informatizar sus principales procesos de producción, que elevaría grandemente la competitividad de esta empresa en el marco internacional.

La Dirección de Intervención y Perforación de Pozos (DIPP) radicada en la Empresa de Producción y Extracción de Petróleo del Centro (EPEPC), controla todas las operaciones y demás actividades que se realizan en los pozos de petróleo en perforación e intervención. En esta entidad los procesos de gestión para el control y flujo de información se llevan a cabo manualmente por lo que existe desactualización de la información en los partes de perforación emitidos diariamente. En ocasiones los partes contienen errores e inconsistencias, lo que conlleva a que exista redundancia y falta de organización en los elementos que lo integran. Los partes son archivados en estantes de oficina, donde no existe seguridad, esto trae consigo que sea difícil el acceso a la información, dando paso a que exista un bajo control del orden de almacenamiento de la información, pérdida y mal cuidado de la misma.

Por otro lado utilizan algunos softwares para obtener información del proceso de perforación de pozo en tiempo real, como el "Well-Wizard²". Estos softwares son de corto alcance ya que de manera integral no gestionan toda la información requerida durante el trabajo de campo en la perforación de los pozos, además de un funcionamiento limitado, ya que estos están representados por grandes intereses comerciales.

² Sistema de datos Integrado (IDS) que permite acceder a datos históricos dentro del tiempo de perforación del pozo, es un sistema poderoso y flexible que permite a los usuarios configurar monitoreo de tendencia a medida y alarmas de excepciones.



Por ello la dirección de CUPET conjuntamente con la UCI se ha dado la tarea de desarrollar un Sistema para el Manejo Integral de la Perforación de Pozos de Petróleo y Gas (SIPP). Este sistema posee una primera versión que está aplicándose en los pozos de petróleo en perforación en tierra de CUPET.

Debido al inicio del segundo ciclo de desarrollo del sistema se hace necesario replantearse elementos de dicho sistema, como su arquitectura, diseño, modelo de base de datos, y la programación del mismo, con el objetivo de obtener una segunda versión de SIPP. El modelo de base de datos es uno de los elementos a reemplazar del cual se identificaron varias deficiencias.

El modelo no contiene toda la información atómica que se genera en el proceso de perforación. Existen datos redundantes provocando inconsistencia en la información y estructuración de la base de datos. El modelo de base de datos está estructurado en un solo esquema, por lo que existe desorganización en sus tablas y difícil manejo de las mismas. Existen tablas con diferentes funciones que pueden ser separadas en esquemas, ejemplo de ellas, son las tablas nomencladoras del proceso, las que intervienen directamente en la perforación y las que intervienen en la seguridad del sistema.

Por otra parte no se tuvo en cuenta estándares de nomenclaturas en las tablas y campos de las mismas, por lo que no existe uniformidad en la base de datos trayendo consigo desorganización, difícil manejo de datos en la programación del sistema y mal interpretación de los mismos. Además de estos problemas se detecta que no existe una arquitectura de base de datos bien definida, pues no existe una correspondencia entre los distintos niveles de la arquitectura ya sea el nivel externo, el nivel lógico y el nivel interno de la base de datos.



Derivándose así el **problema a resolver**: ¿Cómo contribuir en la gestión de los datos en el Sistema para el Manejo Integral para la Perforación de Pozos de Petróleo y Gas (SIPP) versión 2.0 para que exista mayor integridad, organización e independencia de los mismos? Se plantea como **objeto de estudio**: el proceso de gestión de los datos en el Sistema para el Manejo Integral de la Perforación de Pozos de Petróleo y Gas (SIPP). El **objetivo general** perseguido por este trabajo de diploma es diseñar una base de datos que contribuya a la integridad, organización e independencia de los datos en el Sistema para el Manejo Integral de la Perforación de Pozos de Petróleo y Gas (SIPP) versión 2.0; enmarcando el **campo de acción** en las bases de datos relacionales para la gestión de datos generada en el proceso de perforación de pozos de petróleo. Se tiene como **idea a defender** que con el diseño de una base de datos con alta integridad, organización e independencia de los datos se contribuirá a la gestión de los mismos en el Sistema para el Manejo Integral de la Perforación de Pozos de Petróleo y Gas (SIPP) versión 2.0.

Y para lograr el cumplimiento de estos objetivos se han trazado las siguientes tareas específicas:

- ✓ Investigación de los diferentes procesos de perforación de pozos de petróleo para identificar, describir y definir los procesos del negocio.
- ✓ Investigación del modelo de base de datos de SIPP para valorar la modelación e implementación de la base de datos de SIPP.
- ✓ Investigación de las técnicas y herramientas utilizadas en el desarrollo de la base de datos del SIPP para:
 - Caracterizar gestor de base de datos, herramientas y técnicas empleadas en el diseño.
- ✓ Revisión y selección de las herramientas para prueba de volumen y replicación.
- ✓ Revisión de la Base de Datos actual de SIPP para determinar los datos atómicos que faltan y la existencia de datos redundantes.



- ✓ Descripción de la arquitectura y fundamentación para definir el soporte arquitectónico y estándares de nomenclatura.
- ✓ Diseño e implementación de la Base de Datos para SIPP versión 2.0
- ✓ Utilización de las herramientas para prueba de volumen y replicación.

Métodos Científicos de la Investigación.

Los métodos científicos en los cuales se basa la investigación son los métodos teóricos y empíricos. Estos permiten organizar el trabajo, lo cual posibilita el desarrollo de un amplio conocimiento referente al estado del arte tratado, su evolución en etapas determinadas, así como adquirir de bibliografías elementos importantes para el desarrollo del mismo.

Métodos Teóricos:

Consulta Bibliográfica: Permitió consultar toda la bibliografía referente al objeto de estudio proporcionando información que ayudó a soportar el mismo.

Analítico-Sintético: Permitió analizar teorías y documentos sobre la versión del Sistema Gestor de Base de Datos (SGBD) PostgreSQL utilizado por SIPP versión 1.0, así como características y particularidades manejadas en la conformación del modelo de base de datos relacional que soporta la base de datos de SIPP versión 1.0. Por otra parte logró facilitar el estudio y análisis sobre la arquitectura y normalización de base de datos definida para el sistema SIPP versión 1.0.

Análisis Histórico-Lógico: Permitió el estudio de la trayectoria del SGBD PostgreSQL y de el modelo de base de datos relacional, cuándo fue desarrollado y por qué personalidades, así como su evolución en el tiempo. Se realizaron comparaciones entre las versiones antiguas y las más actuales del SGBD lo que facilitó la obtención de elementos de interés para decidirse por uno u otro para el diseño de la base de datos, SIPP versión 2.0.



Modelación: Método mediante el cual se crea abstracciones con vistas a explicar la realidad. Se utilizó fundamentalmente para la modelación lógica de la base de datos utilizando la herramienta de modelado Visual Paradigm.

Métodos Empíricos:

Entrevista: Con la ayuda de este método se hace una modelación del modelo de clases persistentes en la base de datos donde el cliente plantea los datos atómicos que se le deben prestar una atención específica dentro de la base de datos. Además sirvió para conocer las direcciones de los procesos de negocio dentro de la perforación de pozos de petróleo. Tuvo un papel fundamental en la identificación de los problemas actuales que presenta el sistema planteados por el líder de proyecto, arquitecto y programadores.

Observación: Se utilizó la observación para obtener a través del método el comportamiento real del problema planteado, así como para la recopilación de información realizada con lo que ocurre durante todo el proceso de perforación de pozos de petróleo.

Simulación: Se utilizó con el objetivo de simular un ambiente de replicación para hacerle las pruebas de replicación a base de datos obtenida.

La novedad de la investigación es la creación de un modelo de base de datos con una alta estructuración, organización e independencia de los datos con respecto al modelo que soporta la base de datos de SIPP versión 1.0.

El aporte práctico esperado es el diseño de una base de datos que contribuya a la gestión de los datos en SIPP versión 2.0 con una alta estructuración, organización e independencia de los datos.



La investigación esta estructura en tres capítulos:

Capítulo 1: Fundamentación Teórica. Se realiza un estudio y descripción de los entornos de negocio que rodean al sistema en cuestión, además de los conceptos asociados al dominio del problema. Se realiza una descripción detallada de los problemas que dieron paso a la investigación. Además se explica cómo se desarrolla el proceso de gestión y control de la información de perforación de pozos de petróleo. Por otra parte se caracterizan los Sistemas de Bases de Datos (SBD) así como la arquitectura de los mismos.

Capítulo 2: Descripción y Análisis de la Solución Propuesta. Se realiza el estudio de las diferentes técnicas, así como herramientas para la modelación y el diseño de la base de datos actual del sistema para valorar dicho diseño y técnica aplicada en el mismo. En este capítulo se realizará la solución de la propuesta, incluyendo las descripciones de las herramientas y técnicas de la modelación de la base de datos SIPP versión 2.0. Se realizan los diagramas correspondientes para llegar a la implementación de la base de datos física para su posterior validación. Además se le aplicará la normalización hasta llegar a la tercera forma normal siendo uno de los objetivos de nuestra investigación.

Capítulo 3: Validación del Diseño Realizado. Todo resultado de una investigación debe de estar validado en su conjunto. Por lo que en este capítulo se realizará la validación del diseño realizado, así como las pruebas a la base de datos de volumen y de replicación.



Capítulo 1: Fundamentación Teórica.

1.1. Introducción.

En el presente capítulo se describe detalladamente la situación problemática, especificando los problemas claves que conllevan a la investigación. Se realiza el estudio y descripción de los diferentes entornos de negocio identificados con anterioridad en la primera versión de SIPP. Se describe el proceso de gestión y control de la información de perforación de pozos de petróleo para sentar las bases y entender mejor el objeto de estudio en cuestión. Se caracteriza los Sistemas de Bases de Datos (SBD) y se describe la arquitectura de los mismos. Además de describir el modelo de base de datos relacional, el cual corresponde con el tipo de modelo de la base de datos de SIPP.

1.2. Conceptos asociados al Sistema para el Manejo Integral de Perforación de Pozos de Petróleo y Gas (SIPP).

Es imprescindible, cuando se estudia un fenómeno determinado, describir los sucesos que ocurren dentro del mismo. Describir entonces debe ser el objetivo a perseguir para preparar el escenario de cualquier investigación, para explicar de forma detallada y ordenada la teoría que conlleva a una solución. SIPP tiene una trayectoria muy amplia, donde intervienen muchos conceptos y definiciones que son muy importantes para el sistema. Estos se combinan e intervienen en el proceso de gestión de los datos del sistema, desde el punto de vista del negocio y propios del sistema.

Industria del Petróleo: El diccionario de la lengua española no da una definición concreta del término “industria del petróleo” pero especifica que una industria es el lugar que maneja un conjunto de operaciones destinadas a la obtención, transformación y transporte de materias primas[3]. Otros autores definen la industria del petróleo como una actividad típicamente integrada en forma vertical, lo que significa



que todas las actividades económicas relacionadas con la exportación del petróleo, ya elaborado, se suelen desarrollar bajo la dirección de una misma empresa[4]. Por lo tanto se puede decir que la industria del petróleo es la empresa que maneja gran parte de las operaciones destinadas a la obtención, transformación y comercialización de petróleo y sus derivados.

Procesos de la Industria del Petróleo: Los procesos dentro de la Industria del Petróleo no son más que la secuencia de actividades dentro de la industria que tiene como objetivo principal lograr obtener todos los derivados del hidrocarburo. La perforación es el proceso más importante dentro de la industria petrolera ya que proporciona la certeza de la existencia del hidrocarburo, aunque existe una gran interdependencia entre los procesos, ya que del éxito de uno de ellos, depende el éxito del siguiente. La importancia de este proceso se debe principalmente a que aproximadamente 1 de cada 7 pozos en perforación logra encontrar y producir petróleo. En la actualidad en el proceso de perforación se utilizan técnicas y tecnologías de alta complejidad como son: la perforación utilizando fluido de perforación, perforación direccional (horizontal), el encamisado y cementación de pozo, la utilización de motores, el registro a los pozos, por solo citar algunos. Muchas de estas tecnologías son provistas por compañías de servicio[5].

Geología del Petróleo: En la geología del petróleo se combinan diversos métodos y técnicas exploratorias para seleccionar las mejores oportunidades y encontrar hidrocarburos (Petróleo y Gas). Por lo tanto la Geología del Petróleo comprende las informaciones geológicas de las formaciones y estructuras presentes, la paleontología, el estudio de mapas geológicos y geomorfológicos, estudio de métodos geofísicos que se hayan empleado en el área como métodos potenciales, sismografía y los resultado de las perforaciones exploratorias realizadas en el área que incluyen los estudios accesorios a estas[6].



Proyecto Geológico de Pozo Petrolero: Es el estudio geológico que se realiza a un pozo antes de su perforación el cual queda plasmado en un documento donde se exponen las propiedades y complejidades de las formaciones rocosas que rodean al yacimiento petrofísico.

Yacimiento Petrolero: Estructuras geológicas que se formaron hace millones de años y que contienen hidrocarburos atrapados, líquidos y/o gaseosos, en sus espacios porosos y permeables[7]. Porción de una trampa geológica que contiene aceite, gas o ambos, la cual se comporta como un sistema hidráulico intercomunicado[8].

Registro de Pozo: El registro de pozo es una técnica moderna que consiste en llevar una herramienta hacia diferentes lugares del pozo que se está perforando, para recopilar datos geológicos que corroboren las lecturas que se hacen, con el sistema de monitoreo en tiempo real y el geólogo. Este proceso es de suma importancia ya que se puede pensar que el pozo se encuentra en un estado o formación y este proceso es el que lo corrobora[9].

Petróleo: El petróleo es una mezcla heterogénea de compuestos orgánicos, principalmente hidrocarburos insolubles en agua, que se extrae de lechos geológicos continentales o marítimos. También es conocido como petróleo crudo o simplemente crudo. Es un líquido aceitoso, inflamable, cuyo color varía de incoloro a negro[9].

Pozo de Petróleo y Gas: Entidad donde se realiza la perforación, la cual agrupa un gran número de entidades y trabajadores, como son las compañías de servicio de lodo, direccionales, contratistas, logísticas, entre otras.

UNION CUPET: Es la empresa que se encarga de toda la actividad petrolera desde la exploración hasta la refinación en el país, así como de satisfacer las necesidades del mercado nacional de hidrocarburos, a partir del incremento de la extracción de crudos nacionales[9].



Dirección de Intervención y Perforación de Pozos de Petróleo (DIPP): Empresa que se encuentra ubicada en el municipio Varadero perteneciente a la provincia de Matanzas, aquí se confeccionan una serie de informes, partes y reportes manualmente utilizando el Excel con las informaciones que se reciben diariamente a las ocho de la mañana por correo electrónico de los pozos petroleros en perforación y del Centro de Investigaciones del Petróleo (CEINPET), en caso de no existir la conexión se emiten por vía telefónica. Estos documentos generados contienen información de la perforación diaria, donde se incluyen informaciones de la cantidad de combustible, barrenas, herramientas, camisas y productos químicos utilizados en los pozos, así como las operaciones y actividades que se realizarán en el día, además de la cantidad de dinero invertido. Los mismos son manipulados por las secretarias de la DIPP, aprobada por Jefe de Despacho, y guardada en formato duro y digital para tomar decisiones en momentos posteriores[5].

Centro de Investigaciones del Petróleo (CEINPET): Centro perteneciente a la empresa de CUPET, está localizada en Boyeros, provincia Ciudad de La Habana. Aquí diariamente la información es recibida en forma de reporte a las 7 de la mañana de igual forma al procedimiento descrito anteriormente por la DIPP. La recepción de la información es gestionada manualmente utilizando el Excel para elaborar el parte diario de geología y la confección de las gráficas correspondiente a la columna litológica. [5].

A continuación se especifican los roles principales que intervienen en los procesos anteriormente descritos:

Direccional: Persona que maneja toda la información concerniente a la dirección de las barrenas durante la perforación del pozo, este elabora diariamente el Reporte del Direccional con información obtenida del Well-Wizard (software propietario existente en los pozos de petróleos cubanos que gestiona todos los procesos que ocurren dentro de los mismos)[5].



Químico: Persona que gestiona toda la información relacionada con los productos químicos, así como, los aditivos añadidos al lodo de perforación, este es contratado por empresas extranjeras, diariamente confecciona un reporte con los productos utilizados, la cantidad y el precio de cada uno, lo guarda en formato duro y digital y lo entrega al Supervisor de Pozo[5].

Geólogo: Persona que realiza diariamente un estudio consecuente de las características de las rocas, del terreno, analizando todo tipo de formación, obtiene información del Well-Wizard, elabora el Reporte Diario de Geología, lo guarda en formato duro y digital y lo entrega al Supervisor de Pozo y además de enviarlo al geólogo que labora en la oficina del CEINPET[5].

Supervisor de Pozo: Persona que recibe diariamente información en formato duro y digital del químico, geólogo, direccional y analiza minuciosamente el software Well-Wizard, examina los parámetros recibidos, confeccionando con estos datos una serie de reportes, los cuales guarda en formato duro y digital y envía diariamente a las doce de la noche a la DIPP y a CEINPET[5].

Después de haber conceptualizado los aspectos fundamentales de la perforación de pozo, se puede observar que todos se vinculan con la información por lo que se pasa a definir aspectos asociados al objeto de estudio de la investigación.

Información: Conjunto de datos significativos y pertinentes que describen sucesos o entidades, para que estos datos sean significativos deben de constar de símbolos reconocibles, estar completos y expresar una idea no ambigua[10].

Datos: Elemento que sirve de base a un razonamiento o a una investigación[11]. Son hechos o símbolos que describen sucesos, entidades, condiciones, situaciones o valores[10]. Por lo tanto los datos son elementos que describen sucesos y entidades que sirven de base a razonamientos e investigaciones.



Base de datos (BD): Son recursos que recopilan todo tipo de información, para atender las necesidades de un amplio grupo de usuarios. Su tipología es variada y se caracterizan por una alta estructuración y estandarización de la información[12]. Colección de información organizada de forma que un programa de ordenador pueda seleccionar rápidamente los fragmentos de datos que necesite. Una base de datos es un sistema de archivos electrónico[13]. Se define una base de datos como una serie de datos organizados y relacionados entre sí, los cuales son recolectados y explotados por los sistemas de información de una empresa o negocio en particular[14]. Se puede entonces definir una base de datos como una colección de información organizada y relacionada entre sí para atender las necesidades de un amplio grupo de usuarios, son variadas y se caracterizan por tener una alta estructuración y estandarización de la información.

Sistema Gestor de Bases de Datos (SGBD): Conjunto de programas que permiten crear y mantener una Base de Datos, asegurando su integridad, confidencialidad y seguridad[15]. Es una aplicación que permite a los usuarios crear, definir y mantener las bases de datos[16]. Después de estas definiciones se puede decir que un software que permite la creación, utilización y actualización de los datos almacenados en una o varias bases de datos por uno o varios usuarios desde diferentes puntos de vista es un SGBD.

1.3. Descripción de procesos de negocio para el Sistema para el Manejo Integral de Perforación de Pozos de Petróleo y Gas (SIPP).

Antes de describir cualquier proceso de negocio es necesario saber en qué consiste, y este no es más que un conjunto de tareas relacionadas lógicamente llevadas a cabo para lograr un resultado de negocio definido; se puede definir también como el conjunto de actividades estructurales relacionadas que producen un valor para una organización determinada[3]. La perforación de pozos de petróleo es una actividad que demanda tiempo y recursos financieros. Comienza con el estudio geológico y geofísico del



terreno donde se va a perforar el pozo. Por eso, un equipo de perforación sólo se instala y comienza a perforar cuando geólogos y geofísicos han acordado la locación más apta para la búsqueda de hidrocarburos en el subsuelo. La perforación de un pozo petrolero no es sólo una obra de ingeniería de alta precisión, es un trabajo de atención y tensión para los hombres que integran el equipo[17].

El proceso de perforación incluye diferentes actividades que se desarrollan de forma lineal e incremental durante toda la perforación. Las actividades que se desarrollan dentro del proceso, comienzan con el montaje de campamento del equipo de perforación, donde se registran datos significativos dentro del proceso. Incluye el montaje del equipo de perforación, este es el método de perforación que utiliza actualmente CUPET. Luego se debe asegurar todos los implementos de perforación para lograr que el equipo reúna las condiciones necesarias para comenzar la perforación del pozo. El lodo de perforación es el que brinda vida al avance del pozo, este es un líquido compuesto por agua o petróleo y arcilla con aditivos químicos. El lodo de perforación se utiliza para refrigerar y lubricar la barrena, lubricar la tubería y expulsar del agujero de perforación los fragmentos de roca triturados[18]. La cementación y el encamisado son actividades adheridas al proceso que son de gran complejidad.

Todo lo anteriormente corresponde con las actividades que se realizan en el negocio de la perforación de pozos de petróleo en campo identificadas en la primera versión del sistema de gestión de información SIPP. Cuando se habla de SIPP, se debe entender que el sistema está condicionado por procesos de negocio que soportan su desarrollo de forma satisfactoria, dejando aristas que dan paso a desarrollar otras actividades dentro del proceso de perforación con mayor adaptabilidad que responden a una única línea base del sistema.



Este proceso en SIPP comienza en la Dirección de Intervención y Perforación de Pozos (DIPP). En ella se realizan y se aprueban los proyectos de los pozos a perforar, los cuales se elaboran por un equipo multidisciplinario de diferentes ramas siendo así la responsable de los pozos desde su nacimiento hasta su terminación.

El negocio petrolero en el país todavía tiene mucho por experimentar, y actualmente para darle complemento a todo el proceso de perforación de pozos se necesita la contratación de Compañías de Servicios. Las compañías son las encargadas de brindar servicios de Direccionales, Lodo, Perforación (Contratistas), Registro, Geología y estas son contratadas por la DIPP. Las compañías direccionales son las que se encargan de dirigir el pozo hacia el yacimiento petrofísico y darle la inclinación necesaria para llegar hasta él. Las compañías de servicio de lodo, se encargan del lodo de perforación y de los productos químicos añadidos al mismo. Todas las compañías de servicio que intervienen en la perforación del pozo, emiten diariamente partes que corresponden al desempeño su trabajo en el día. Estos contienen un gran cúmulo de información que se manejan manualmente en formato duro. Estas compañías que muchas veces son extranjeras, y responden por su trabajo íntegramente a la DIPP.

Luego que los pozos entran en fase de producción, pasan a manos de una de las empresas de producción y extracción (Centro, Occidente o Majagua), pero la DIPP continúa siendo partícipe, ya que es la encargada de contratar y monitorear los servicios de Intervención de Pozos, que se realizan con el objetivo de reparar o realizar trabajos de control sobre los pozos que ya están en producción.

Es a partir de todo lo anteriormente expuesto que comienza a ser útil el proceso de gestión de la información en SIPP. Estos procesos son realizados por Compañías de Servicio, las cuales pueden ser cubanas o extranjeras. Cada una de estas compañías emite diariamente un reporte de sus servicios. Los Supervisores son los encargados de dirigir el proceso de perforación, además de construir diariamente los reportes en SIPP, integrando todos los reportes entregados por las compañías de servicio.



1.4. Proceso de gestión y control de la información de perforación de pozos de petróleo.

La gestión está caracterizada por una visión más amplia de las posibilidades reales de una organización para resolver determinada situación o arribar a un fin determinado. Puede asumirse, como la disposición y organización de los recursos de un individuo o grupo para obtener los resultados esperados. Pudiera generalizarse como una forma de alinear los esfuerzos y recursos para alcanzar un fin determinado[19]. Se puede traducir la gestión enfocada a la industria petrolera, específicamente en el proceso de perforación, como la disposición, modificación y la organización de toda la información generada dentro de la perforación del pozo de petróleo.

En todo desarrollo, el control ha ido reforzando una serie de etapas que lo caracterizan como un proceso en el cual las organizaciones deben definir la información y hacerla fluir e interpretarla acorde a sus necesidades para tomar decisiones[19]. Por lo tanto el control de la información en la perforación debe estar acotado a las necesidades de la misma, ser explícito y definir la información haciéndola fluir dentro de todas las etapas de la perforación.

Actualmente el proceso de gestión y control de la información de perforación de pozos de petróleo es complejo y dinámico, que se compone de una serie de etapas que se incorporan unitariamente aunque de manera separable. El proceso comienza con la recolección de la información básica de la perforación de pozos de petróleo, este constituye la línea base de una buena gestión de la información. En la recolección de la información se debe tener en cuenta todo lo referente a la perforación, desde el tipo, operaciones de perforación como son las técnicas de perforación y las operaciones de terminación, recuperación optimizada y rehabilitación. Luego durante el transcurso todos estos datos se combinan con la facilidad de integrarse y garantizar que la información durante la perforación sea lineal e incremental.



Indudablemente SIPP versión 1.0 tiene un papel fundamental en la gestión y el control de la información en este proceso en los pozos de petróleo. Es un sistema de gestión y control con un conjunto de acciones, funciones, medios y responsables que garantizan, mediante su interacción, conocer la situación de un pozo de petróleo en un momento determinado y tomar decisiones para reaccionar ante ellas.

Pero como todo sistema donde se genera gran cúmulo de información debe estar soportado por un medio que almacene toda esa información. En este caso se controla con una base de datos que procesa gran cantidad de información. Esta base de datos que constituye la fuente principal de información del sistema, debe cumplir con una serie de requisitos para su funcionamiento que hoy es objeto de estudio en la investigación por sus deficiencias en cuanto a su integridad. Todo lo anterior refleja que un sistema de gestión y control debe ser entendible, rápido, flexible y seguir la forma de la organización.

1.5. Características y arquitectura de los Sistemas de Base de Datos.

Los datos constituyen la parte esencial de un sistema de información, y son los que justifican su existencia. Para organizar y gestionar datos en el computador se han desarrollado técnicas cuya evolución ha estado determinada principalmente por el desarrollo de la tecnología de los computadores así como por los requisitos y necesidades planteadas por los usuarios[20]. Como se ha analizado en la investigación diferentes autores definen una base de datos como un conjunto de datos que se relacionan entre sí, y que estos están almacenado con carácter más o menos permanente en el computador, por lo tanto se puede considerar una colección de datos variables en el tiempo. Para tener organización en los datos almacenados se alude a la arquitectura de las bases de datos como elemento fundamental en este aspecto.



1.5.1. Características de los Sistemas de Bases de Datos (SBD).

Las bases de datos han profundizado una evolución que se hace continua en nuestros días, pasando de simples sistemas a gestores de estructura de datos que ofrecen muchas posibilidades de manejar la información. Hay tres características importantes inherentes a los sistemas de bases de datos: la separación entre los programas de aplicación y los datos, el manejo de múltiples vistas por parte de los usuarios y el uso de un catálogo para almacenar el esquema de la base de datos.

Las bases de datos deben contar con independencia lógica y física de los datos, esto se refiere a la capacidad de modificar alguna información específica sin que afecte a los demás registros. La independencia de los datos se define como la inmunidad de las aplicaciones a los cambios en la estructura de almacenamiento y en la estrategia de acceso, y constituye el objetivo fundamental de los Sistemas de Base de Datos (SBD)[2]. La independencia lógica es la capacidad de modificar el esquema conceptual sin tener que alterar los esquemas externos ni los programas de aplicación. Por otro lado la independencia física de los datos es la capacidad de modificar el esquema interno sin tener que alterar el esquema conceptual. Dado que la independencia física se refiere sólo a la separación entre las aplicaciones y las estructuras físicas de almacenamiento, es más fácil de conseguir que la independencia lógica.

Otra característica de los SBD es minimizar la redundancia de los datos. Se dice disminuir la redundancia, no eliminarla, pues, aunque se definen las bases de datos como no redundantes, en realidad existe redundancia en un grado no significativo para disminuir el tiempo de acceso a los datos o para simplificar el método de direccionamiento. Lo que se trata de lograr es la eliminación de la redundancia superflua[2]. Cuando se habla de base de datos donde existan datos redundantes existen más posibilidades de inconsistencias, se duplica la información, requiriendo más espacio del necesario.



La integridad de los datos se refiere a las medidas de seguridad que impiden que se introduzcan datos erróneos ya sea por motivos físicos debido a causas externas, o de operación como la introducción de datos incoherentes, esto puede ser mediante encriptación de la información o protección con contraseñas de acceso. Consiste entonces en garantizar la no contradicción entre los datos almacenados de modo que, en cualquier momento del tiempo, los datos almacenados sean correctos, es decir que no se detecte inconsistencia entre los datos. Está relacionada con la minimización de la redundancia, ya que es más fácil garantizar la integridad si se elimina la redundancia[2].

Facilitar las consultas complejas a los datos y brindando seguridad de acceso y auditoría, esto esta combinado con el derecho de acceso a la información contenida en la base de datos por parte de personas o grupos, además debe brindar respaldo y recuperación de información. Los usuarios de una base de datos pueden realizar innumerables solicitudes para resolver muchos problemas diferentes, por lo que un SBD debe ser capaz de realizar búsquedas rápidas por diferentes criterios y permitir que los usuarios planteen sus demandas de una forma simple, aislándolos de las complejidades del tratamiento de los archivos y del direccionamiento de los datos.

Toda base de datos debe proporcionar versatilidad en la representación de la información gestionada permitiendo tener una visión parcial del problema y visión global desde distintos puntos de vista por lo que han de ser capaces de manejar datos ágilmente, ofreciendo un tiempo de respuesta adecuado buscando la solución más simple y que mejor se adecue a las necesidades del usuario.

Un SBD debe ser afinado, esta característica viene dada con la interrogante de cómo los datos se almacenan a nivel físico, ya que los sistemas de base de datos han de ser flexible. Adicionar como característica de los SBD que deben tener un lugar de interacción, el espacio donde se desarrollen cambios, esta característica es denominada interfaz temporal que a veces puede parecer obvia, pero no lo es. Una



buena base de datos debe de estar abierta a cambios, de forma que sea fácil cambiar términos del sistema sin que haya complicaciones en ello.

La seguridad en un SBD garantiza el acceso autorizado a los datos, para interrumpir cualquier intento de acceso no autorizado, ya sea por error del usuario o por mala intención[2]. La seguridad tiene que ser un apéndice de cualquier base de datos. Se ha de evitar que frente a fallos de hardware existan fugas de datos y que existan accesos no autorizados que puedan romper la integridad de los datos.

1.5.2. Arquitectura de los Sistemas de Bases de Datos (SBD).

La arquitectura de un SBD está influenciada en gran medida por el sistema informático subyacente en el que se ejecuta el sistema de base de datos. El procesamiento paralelo dentro de una computadora permite acelerar las actividades del sistema de base de datos, proporcionando a las transacciones una respuesta más rápida, así como la capacidad de ejecutar más transacciones por segundo. La distribución de datos a través de las distintas sedes o departamentos de una organización permite que estos datos residan donde han sido generados o donde son más necesarios, pero continuar siendo accesibles desde otros lugares o departamentos diferentes[21].

La arquitectura de los sistemas de base de datos se divide en tres niveles el nivel interno, nivel externo y el nivel lógico global y conceptual:

- Nivel Interno: Algunos autores lo definen como, el más cercano al almacenamiento físico, o sea, el relacionado con la forma en que los datos están realmente almacenados[2]. Es el nivel real de los datos almacenados y tiene un esquema interno que describe la estructura física de almacenamiento de base de datos. Este nivel es usado por muy pocas personas y lleva asociado una representación de los datos.



- Nivel Externo: Es el nivel de mayor abstracción y tiene varios esquemas externos o vistas de usuario. A este nivel corresponden las diferentes vistas parciales que tienen de la base de datos los diferentes usuarios. En cierto modo, es la parte del modelo conceptual a la que tienen acceso. Es el más cercano a los usuarios, o sea, el relacionado con la forma en que los datos son visto por cada usuario individualmente[2].
- Nivel Lógico Global y Conceptual: Es el nivel medio de abstracción y consta con un esquema conceptual. Se trata de la representación de los datos realizada por la organización, que recoge las vistas parciales de los requerimientos de los diferentes usuarios y las aplicaciones posibles. Se configura como visión organizativa total, e incluye la definición de datos y las relaciones entre ellos.

La correspondencia lógica/interna especifica la forma en que los artículos y campos lógicos se representan en el nivel interno. Si se cambia la estructura de la vista interna, entonces la correspondencia lógica/interna tiene también que cambiar en consecuencia, de modo que el esquema lógico permanezca invariable. Los efectos de estos cambios deben ser aislados por debajo del nivel lógico para que se mantenga la independencia de los datos. Existe una correspondencia externo/lógica entre cada vista externa particular y la vista lógica. Las diferencias que puedan existir entre estos dos niveles son similares a las que pueden existir entre la vista lógica y la interna[2] (Figura 1).

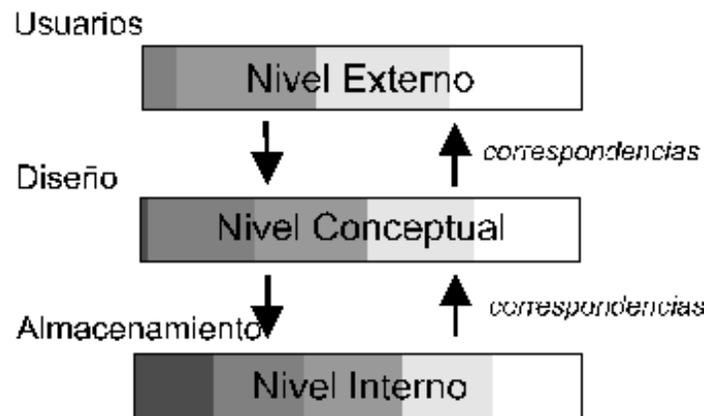


Figura 1. Niveles de la Arquitectura de Base de Datos³

En ocasiones puede encontrarse el nivel conceptual dividido en dos niveles, conceptual y lógico. El primero de ellos corresponde a la visión del sistema global desde un punto de vista organizativo independiente, no informático. El segundo correspondería a la visión de la base de datos expresada en términos del sistema que se va a implantar con medios informáticos.

El objetivo de la arquitectura de tres niveles, es el de separar los programas de la aplicación de la base de datos física. Muchos de los Sistemas Gestores de Bases de Datos (SGBD) no distinguen del todo los tres niveles. Algunos incluyen detalles del nivel físico en el esquema conceptual. En casi todos los SGBD que se manejan vistas de usuario, los esquemas externos se especifican con el mismo modelo de datos que describe la información a nivel conceptual, aunque en algunos se pueden utilizar diferentes modelos de datos en los niveles conceptuales y externos. Sin embargo, es preciso destacar que los tres esquemas no son más que descripciones de los mismos datos pero con distintos niveles de abstracción. Los únicos datos que existen realmente están a nivel físico, almacenados en un dispositivo.

³ Tomado del sitio web hipertexto.com disponible en: http://icesecurity.org/conceptos/b_datos.htm



En un SGBD basado en la arquitectura de tres niveles, cada grupo de usuarios hace referencia exclusivamente a su propio esquema externo. Entonces un SGBD debe ser capaz de transformar cualquier petición expresada en términos de un esquema conceptual, y luego, a una petición en el esquema interno, que se procederá sobre la base de datos, almacenada. Si la petición es de una obtención de datos, será preciso modificar el formato de la información extraída de la base de datos almacenada, para que coincida con la vista externa del usuario. A este proceso de transformar peticiones y resultados de un nivel a otro se denomina correspondencia o transformación[22].

Una buena arquitectura permite establecer el principio de independencia de los datos. Esta independencia puede ser lógica y física. Por independencia lógica se entiende que los cambios en el esquema lógico no deben afectar a los esquemas externos que no utilicen los datos modificados. Por independencia física se entiende que el esquema lógico no se vea afectado por cambios realizados en el esquema interno, correspondientes a modos de acceso.

Si garantizar el control centralizado de la información es una tarea que debe permitir un sistema de base de datos, la arquitectura es un punto crítico en cualquier sistema de base de datos. Una arquitectura organizada y estructurada garantiza una gestión de la información centralizada lo que facilita y clarifica el diseño del sistema. Por lo tanto lograr una arquitectura consistente, precisa, con una organización de los niveles bien definida contribuirá de manera significativa a la integridad, organización e independencia de los datos en una base de datos.



1.6. Modelos de Base de Datos.

Un modelo es el resultado concreto del proceso de generar una representación de sistemas a fin de analizar esos fenómenos o procesos. Un modelo de base de datos es básicamente una descripción de algo conocido como contenedor de datos (algo donde se guarda información), así como de los métodos para almacenar y recuperar información de esos contenedores. Los modelos de datos no son cosas físicas: son abstracciones que permiten la implementación de un sistema eficiente de base de datos; por lo general, se refieren a algoritmos, y conceptos matemáticos[23]. Pueden ser descritos así: “dado los requerimientos de información y proceso de una aplicación de uso intensivo de datos (ejemplo, un sistema de información), construir una representación de la aplicación que capture las propiedades estáticas y dinámicas requeridas para dar soporte a los procesos deseados (ejemplo, transacciones y consultas). Además de capturar las necesidades dadas en el momento de la etapa de diseño, la representación debe ser capaz de dar cabida a eventuales futuros requerimientos”[24].

Se puede definir entonces un modelo de datos como abstracciones de conceptos definidos a partir de principios y conceptos matemáticos que permiten expresar propiedades estáticas y dinámicas de una aplicación y permiten la implementación de un sistema eficiente de base de datos. Los modelos de datos contribuyen a sentar cimientos conceptuales para crear aplicaciones que hacen uso de datos, así como ser las bases formales para las herramientas y técnicas empleadas en el desarrollo y uso de sistemas de información.

Muchas veces las aplicaciones están caracterizadas por tres aspectos fundamentales: propiedades estáticas que son las entidades u objetos, propiedades o atributos de esas entidades, y relaciones entre esas entidades; propiedades dinámicas que son las operaciones sobre entidades, sobre propiedades o relaciones entre operaciones y por



último las reglas de integridad sobre las entidades y las operaciones. El resultado de un modelo de datos es una representación que tiene dos componentes: las propiedades estáticas se definen en un esquema y las propiedades dinámicas se definen como especificaciones de transacciones, consultas e informes [24]. Los modelos se utilizan en todo tipo de ciencias. Su propósito es la de representar una parte del mundo real de forma que sea más fácilmente manipulable. Según lo que se ha descrito los modelos de datos intentan reproducir información real a través de esquemas conceptuales.

Existen tres clasificaciones fundamentales de los modelos de datos estos son: modelos conceptuales, modelos lógicos, y modelos internos. Estos modelos están condicionados por esquemas que extraen del mundo real la información, procesarla y llegarla a plasmar en una base de datos física son procesos lineales que tiene como producto final una base de datos física. Estos esquemas son: el esquema conceptual que representa el modelo de datos de forma independiente del SGBD que se utilizará, el esquema canónico (o de base de datos) que representa los datos en un formato más cercano al del ordenador y por último el esquema interno que representa los datos según el modelo concreto de un sistema gestor de bases de datos[25] (Ver Figura 2).

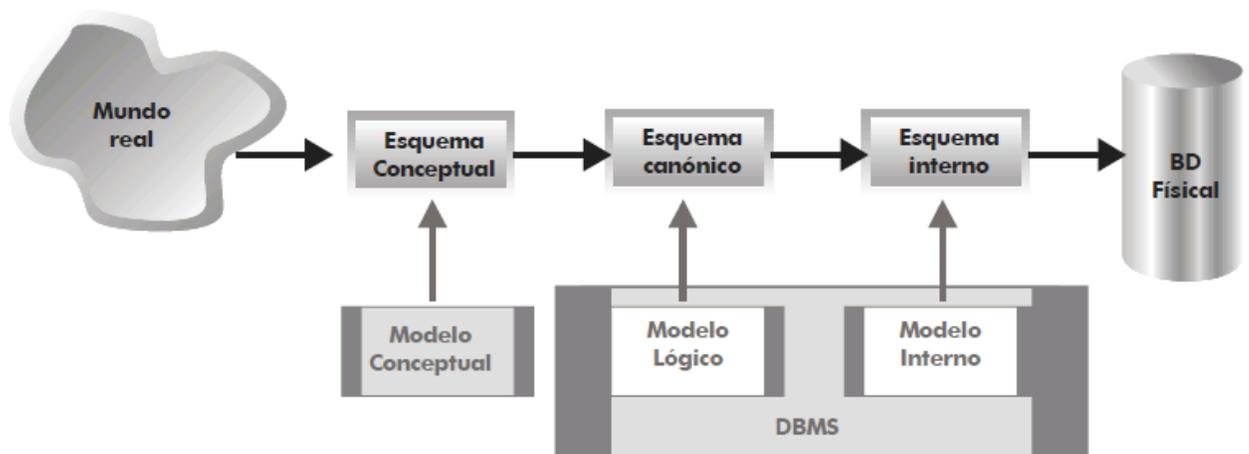


Figura 2. Clasificación de los modelos de datos⁴

⁴ Tomado del libro *Diseño Conceptual de Base de Datos* disponible en: <http://www.jorgesanchez.net/bd/disenioBD.pdf>



Los dos modelos fundamentales de datos son el conceptual y el lógico. Ambos son conceptuales en el sentido de que convierten parámetros del mundo real en abstracciones que permiten entender los datos sin tener en cuenta la física de los mismos. El modelo conceptual se define exteriormente al SGBD, realizándose, manualmente o con la ayuda de algún sistema automatizado, la transformación entre el modelo conceptual y el lógico global. El proceso de modelación conceptual es denominado también modelación semántica, ya que con estos modelos se pretende reflejar, en mayor medida, la semántica o significado de los datos y sus interrelaciones[2].

Algunos ejemplos de modelos conceptuales son: el Modelo Entidad Relación (MER), Modelos RM/T y Modelos semánticos y de modelos lógicos: el Modelo Relacional, Codasyl, Jerárquico[25]. La modelación de base de datos es un proceso fundamental en todo diseño de base de datos, su complejidad depende de cómo este representada la información en el mundo real y como representarla en un modelo para un posterior diseño. Es un proceso lineal donde intervienen diferentes modelos con características propias, y que tienen en común llegar a la representación de datos físicos.

1.6.1. Modelo de Base de Datos Relacional.

El modelo de base de datos relacional es uno de los modelos matemáticos más importantes y actuales para la representación de las bases de datos. Se basa en la teoría matemática de las relaciones, suministrándose por ello una fundamentación teórica que permite aplicar todos los resultados de dicha teoría a problemas como el diseño de sublenguajes de datos y otros[2]. El principio básico de este modelo consiste en representar tanto las entidades, denominadas también tablas, como las asociaciones, con la ayuda de relaciones[26]. La estructura fundamental del modelo relacional es la relación, es decir una tabla bidimensional constituida por filas (tuplas) y



columnas (atributos). Las relaciones representan las entidades que se consideran interesantes en la base de datos. Cada instancia de la entidad encontrará sitio en una tupla de la relación, mientras que los atributos de la relación representan propiedades de la entidad[27]. A partir de los planteamientos se puede definir al modelo de base de datos relacional como un modelo matemático que se basa en la teoría matemática de las relaciones, su estructura fundamental es la relación y plantea como principio básico la representación de entidades y asociaciones con ayuda de las relaciones.

Para poder analizar con mayor profundidad este modelo se debe conocer sus terminologías, que no son más que los elementos que lo componen, y sirven para personificar la información dentro del modelo:

Esquema: Un esquema es la definición de una estructura (generalmente relaciones o tablas de una base de datos), es decir, determina la identidad de la relación y qué tipo de información podrá ser almacenada dentro de ella; en otras palabras, el esquema son los metadatos de la relación[28].

Instancia: Una instancia de manera formal es la aplicación de un esquema a un conjunto finito de datos. Se puede definir como el contenido de una tabla en un momento dado[28].

Entidad: Se trata de cualquier objeto u elemento (real o abstracto) acerca del cual se pueda almacenar información en la base de datos. Una entidad no es una propiedad concreta sino un objeto que puede poseer múltiples propiedades (atributos)[25].

Relación: En el modelo relacional tanto los objetos o entidades, como las relaciones entre ellos, se representan a través de tablas que en la terminología relacional se denominan relaciones.



Cardinalidad: Indica el número de relaciones en las que una entidad puede aparecer. Se anota en términos de cardinalidad mínima, que indica el número mínimo de asociaciones en las que aparecerá cada ejemplar de la entidad (el valor que se anota es de cero o uno), y la cardinalidad máxima que indica el número máximo de relaciones en las que se puede aparecer cada ejemplar de la entidad[25].

Atributos: Describen propiedades de las entidades y las relaciones[25]. Cada atributo se caracteriza por un nombre y por un dominio. El dominio indica qué valores pueden ser asumidos por una columna. A menudo un dominio se define a través de la declaración de un tipo para el atributo[27].

Tupla: Colección no ordenada de elementos diferentes. Para distinguir una tupla de otra, se recurre al concepto de llave primaria, o sea un atributo o conjunto de atributos que permiten identificar unívocamente una tupla en una relación[27].

Por lo que se ha analizado hasta el momento se puede decir que este modelo todos los datos son almacenados en relaciones, y como se define una relación es definida como un conjunto de datos, el orden en que estos se almacenen no tiene la mayor relevancia.

Esto tiene la considerable ventaja de que es más fácil de entender y de ser utilizado por un usuario no experto. La información puede ser almacenada por medio de consultas que ofrecen una amplia flexibilidad y poder administrar la información. Este modelo considera la base de datos como una colección de relaciones. De manera simple, una relación representa una tabla que no es más que un conjunto de filas, cada fila es un conjunto de campos y cada campo representa un valor que interpretado describe el mundo real. Cada fila también se puede denominar tupla o registro y a cada columna también se le puede llamar campo o atributo[28].



Todo modelo de base de datos tiene características distintivas propias que hacen que tenga ventajas sobre otros en este caso el modelo relacional cuenta con las siguientes ventajas:

- ✓ **Simplicidad:** El usuario formula sus demandas en términos de contenido informativo de la base de datos sin tener que atender a las complejidades de la realización del sistema, lo que implica gran independencia de los datos.
- ✓ **Actualización sin dificultades:** Si se tienen relaciones normalizadas, no surgen dificultades grandes en la actualización. Por lo que favorece la normalización por ser más comprensible y aplicable.
- ✓ **Integridad Referencial:** Garantiza la integridad referencial así al eliminar un registro elimina todos los registros relacionados dependientes[2].

Tiene como principal desventaja la dificultad de lograr productividad adecuada en los sistemas, ya que no se emplean los medios técnicos idóneos, tales como las memorias asociativas, siendo necesario simular este proceso.

En el epígrafe 1.2 se define el término base de datos como un conjunto de datos organizados y relacionados entre sí, ¿se podría definir así una base de datos relacional? Por lo que se ha analizado con anterioridad se puede decir que es más que eso se puede definir entonces una base de datos relacional, como el conjunto de relaciones o tablas estructuradas en registros (filas) y campos (columnas), que se vinculan entre sí por un campo en común.



1.6.2. Elementos de una base de datos relacional.

Con anterioridad se describió el modelo relacional y los elementos que lo componen. En una base de datos relacional también la integran elementos que intervienen en su diseño tanto conceptual como lógico. A continuación se analizará estos elementos.

Modelo Entidad Relación (MER): Es un modelo de datos, que responde al tipo de modelo conceptual. Su objetivo es representar relaciones que existan en la vida real entendiendo su semántica. Los cuatro elementos fundamentales de un modelo MER son: las entidades, atributos, interrelaciones y dominio[29].

Entidad: Es el objeto básico representado en el MER. Representa elementos u objetos de la vida real con existencia necesaria en la base de datos. Una entidad puede ser un objeto con existencia física como: una persona, una máquina, una pieza; o bien puede ser un objeto con existencia conceptual: un curso académico, un departamento, un viaje, o cualquier otro elemento del cual se necesite almacenar información. Éstas deben utilizar nombres en singular preferentemente, y que sean descriptores de la entidad. Se representan por un rectángulo con el nombre de la entidad dentro[29].

Atributos: Es la unidad menor de información sobre un objeto almacenado en la BD. Son propiedades que poseen las entidades, por ejemplo, un color, un nombre, identificador, o cualquier otro dato que ofrezca información sobre la entidad. Se representan mediante elipses, o pequeños círculos, conectadas a la entidad mediante una línea recta.

Dominio: Conjunto de valores posibles que puede tomar un atributo determinado[29].

Interrelación: Relación, vínculo o correspondencia entre entidades.



Llave: Atributo (o conjunto de atributos) que identifica a una entidad como un objeto único e irrepitable dentro de la base de datos. Ejemplo: la llave de la entidad PERSONA puede ser el No.CI (número de carnet de identidad); en la entidad PIEZA la llave puede ser el número de serie. Se representa a través de una elipse con el nombre del atributo subrayado dentro, o a través de una pequeña esfera rellena[29].

1.7. Conclusiones Parciales.

En este capítulo han sido tratados aspectos teóricos que son la base fundamental de la investigación. Se describe detalladamente la situación problemática siendo el punto de partida del trabajo de diploma teniendo como resultado, poder fomentar enfoques y fundamentos válidos para esclarecer términos referidos a la implementación del modelo de la base de datos actual de SIPP. Fueron indagados conceptos asociados al sistema y a los procesos de negocio identificados, así como de base de datos y sus relaciones con términos como información y sistema gestor de base de datos; permitiendo ampliar el conocimiento para entender los complejos procesos de negocio de SIPP. Luego se describieron los principales procesos de negocio asociados al sistema y el comportamiento de la gestión de datos en el sistema, dándole complemento al entendimiento y la descripción del flujo de datos en el negocio.

Se describe en este capítulo la arquitectura de base de datos, pudiéndose obtener como resultado, el conocimiento para estructurar la arquitectura de la base de datos de SIPP versión 2.0. Además se clasifican los modelos de datos y analizan sus características principales para analizar el modelo de base de datos relacional en el cual se basa el modelo de datos de la base de datos de SIPP: Se tienen las bases entonces para empezar a realizar la descripción de la propuesta para la base de datos SIPP versión 2.0.



Capítulo 2: Descripción y Análisis de la Solución Propuesta.

2.1. Introducción.

Todo desarrollo de una investigación encaminado a obtener como resultado un producto, en este caso una base de datos, comprende utilizar un conjunto de herramientas que posibiliten su desarrollo. En el presente capítulo se realizará la caracterización de las herramientas tanto del desarrollo como de diseño y el gestor de base de datos a utilizar. Además se hace un estudio y comparación de las herramientas de replicación actuales para escoger la que soporta la replicación de datos. Se describe y analizan aspectos como: incoherencias, redundancia y tablas que no cumplen objetivo en el modelo actual de la base de datos para pasar al diseño de la base de datos versión 2.0.

En otro orden se describe y fundamenta la arquitectura propuesta y se definen los estándares de nomenclatura para la base de datos y el almacenamiento físico de la misma en el sistema gestor de base de datos. Por último se describe las tablas más importantes de la base de datos.

2.2. Sistema Gestor de Base de Datos. PostgreSQL.

Los sistemas de gestión de base de datos son tipos de software muy específico, dedicado a servir de interfaz entre las bases de datos, el usuario y las aplicaciones que lo utilizan. Se compone de un lenguaje de definición de datos y de un lenguaje de consulta. Permiten definir los datos a distintos niveles de abstracción y manipular dichos datos, garantizando la seguridad e integridad de los mismos[30].

PostgreSQL es un sistema de gestión de base de datos basado en el proyecto POSTGRES, de la universidad de Berkeley. Es un potente motor de bases de datos objeto-relacional, que tiene prestaciones y funcionalidades equivalentes a muchos



gestores de base de datos comerciales. Distribuido bajo licencia BSD y con su código fuente disponible libremente. Utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. [31].

Los componentes más importantes de un sistema PostgreSQL son:

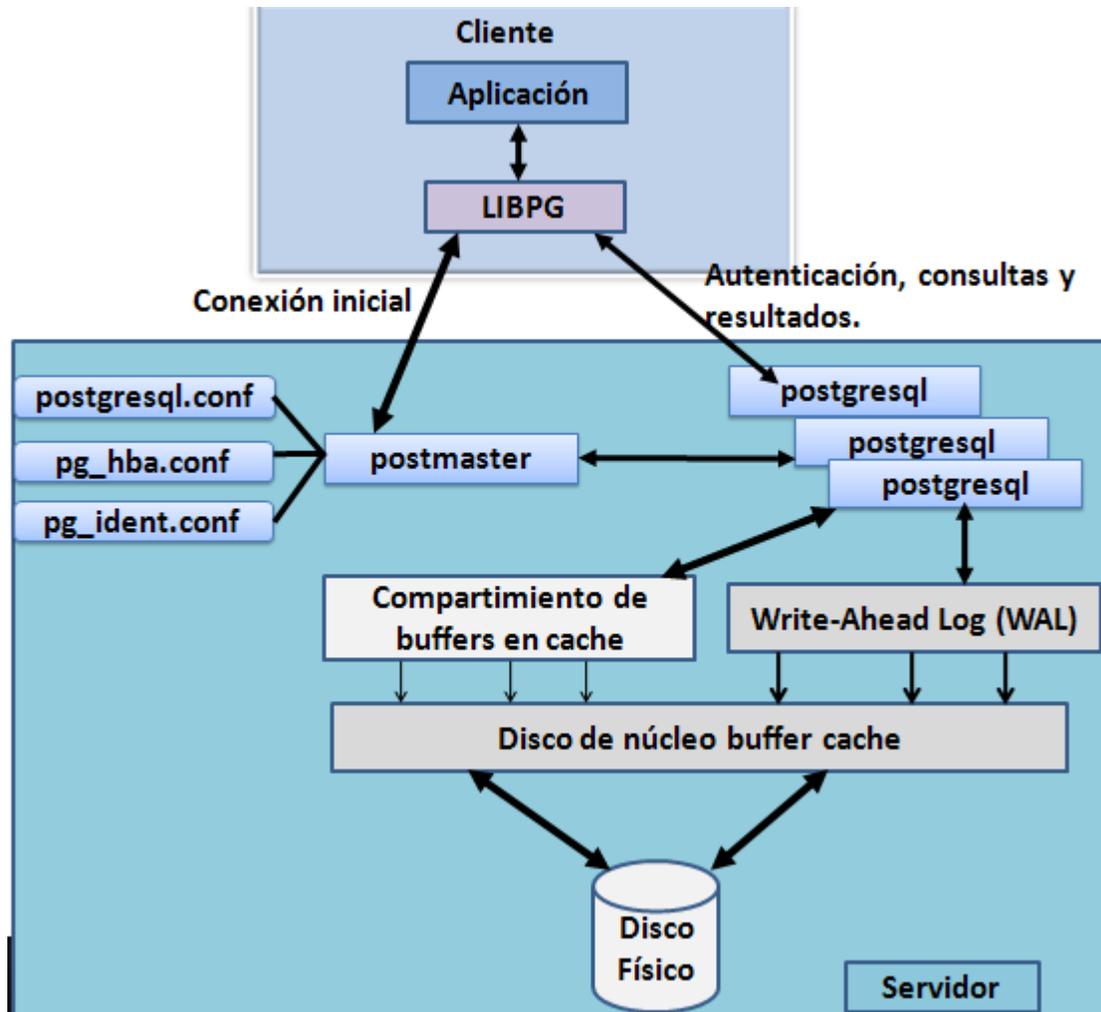


Figura 3. Componentes de PostgreSQL

- **Aplicación cliente:** Esta es la aplicación cliente que utiliza PostgreSQL como administrador de bases de datos. La conexión puede ocurrir vía TCP/IP ó sockets locales.



- **Demonio postmaster:** Este es el proceso principal de PostgreSQL. Es el encargado de escuchar por un puerto/socket por conexiones entrantes de clientes. También es el encargado de crear los procesos hijos que se encarga de autenticar estas peticiones, gestionar las consultas y mandar los resultados a las aplicaciones clientes.
- **Ficheros de configuración:** Los tres ficheros principales de configuración utilizados por PostgreSQL, postgresql.conf, pg_hba.conf y pg_ident.conf.
- **Procesos hijos postgres:** Procesos hijos que se encargan de autenticar a los clientes, de gestionar las consultas y mandar los resultados a las aplicaciones clientes [32].
- **Compartimento de búfer en caché:** Memoria compartida usada por PostgreSQL para almacenar datos en caché.
- **Write-Ahead Log (WAL):** Componente del sistema encargado de asegurar la integridad de los datos (recuperación de tipo REDO).
- **Disco de núcleo buffer cache:** Caché de disco del sistema operativo.
- **Disco:** Disco físico donde se almacenan los datos y toda la información necesaria para que PostgreSQL funcione[31].

PostgreSQL presenta características que lo sitúan como uno de los gestores de base de datos más utilizados a nivel mundial. Los usuarios del mundo lo adquieren con facilidad porque está bajo licencia Berkeley Software Distribution (BSD). Esta licencia tiene pocas restricciones y permite el uso del código fuente en software no libre. El gestor presenta la propiedad de ser indivisible, lo que es lo mismo, presenta atomicidad, lo que asegura que la operación se ha realizado o no, y por lo tanto ante un fallo del sistema, esta no puede quedar a medias. El gestor es consistente, asegurando que una operación no puede afectar a otra, lo que indica que dos transacciones sobre la misma información nunca generarán ningún tipo de error. Tiene durabilidad, una vez realizada una operación en el gestor, esta persistirá, y no se podrá deshacer aunque falle el sistema.



Otra característica y no menos importante es la compatibilidad con las principales familias de sistemas operativos como: Linux, Unix, BSB's, Mac Os, Windows. Mantiene comunidades muy activas y una documentación muy bien organizada, pública y libre. Soporta todas las características de una base de datos (disparadores, procedimientos almacenados, secuencias, relaciones, reglas, tipos de datos definidos por los usuarios, vistas materializadas etc.). Soporta los lenguajes más populares actualmente: PHP, C, C++, Perl, Python etc. y contiene protocolo de comunicación encriptado SSL[33].

PostgreSQL, posee muchas ventajas frente a otros gestores de su tipo. Su costo de adquisición es bajo o nulo por lo que no existen problemas de licencia. Aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas. Ejemplo de su avanzada funcionalidad son las consultas SQL declarativas, control de concurrencia, soporte multiusuario, transacciones, optimización de consultas, herencia y arreglos. A diferencia de otros gestores es altamente extensible, ventaja que le permite soportar tipos de datos complejos como fecha, elementos gráficos, datos sobre redes, cadenas de bit, etc.

Contempla un lenguaje nativo llamado PL/pgSQL y la habilidad para utilizar Perl o Python como lenguaje procedural embebido. Contiene una tecnología MVCC (Control de Concurrencia Multiversión) para evitar bloqueos innecesarios, es decir permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin generar bloqueos[34]. Es soportado gratuitamente por las comunidades de desarrollo, teniendo integrado a su comunidad los mismo desarrolladores que participaron en su implementación. Destacar como desventaja que las búsquedas son lentas a pesar de que crea índices específicos para estas[35].

Después de haber analizado las características, ventajas y desventajas del SGBD PostgreSQL utilizado por SIPP versión 1.0, se concluye mantener la aplicación del mismo en la versión 2.0. Primeramente porque es un SGBD de código abierto de los más potentes en la actualidad. Está distribuido bajo licencia BSD, permitiendo disímiles



usos, haciéndolo altamente comercial y proporciona un gran número de características que lo hacen posicionar como uno de los primeros SGBD con alta aceptación a nivel mundial.

2.3. Herramienta de Diseño. Visual Paradigm.

Las herramientas de diseño aplicado a la informática se pueden definir como el conjunto de aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el costo de los mismos en término de tiempo y dinero. Para confeccionar una base de datos se necesitan herramientas que posibiliten garantizar la seguridad, consistencia e integridad de la información, así como lograr que todo el proceso de su creación sea lo más sencillo y rápido posible.

Visual Paradigm una herramienta CASE que ofrece un entorno de creación de diagramas UML. Esta herramienta es un lenguaje estándar común para todo el equipo de desarrollo que facilita la comunicación. El diseño es centrado en casos de uso y enfocado al negocio, le permite generar software de gran calidad. Tiene capacidad para la ingeniería directa e inversa en Java, C++, PHP, entre otros lenguajes y disponibilidad de múltiples versiones para cada necesidad. Es multiplataforma y muy útil para la generación de código fuente en PHP. Tiene la capacidad de crear esquemas de clase a partir de una base de datos y crear la base de datos a partir del esquema de clases. Incorpora el soporte para el trabajo en equipo, proporcionando que varios desarrolladores trabajen a la vez en el mismo diagrama y vean en tiempo real los cambios hechos por sus compañeros de equipo[20].

Visual Paradigm facilita la realización de los diagramas de modelado siguiendo el estándar UML, estos diagramas son: Diagramas de clases, Casos de Usos, Comunicación, Secuencia, Estado, Actividad, Componente, etc. Tiene la facilidad de integrarse con otros IDE's de desarrollo como NetBeans, JDeveloper y Eclipse. Permite realizar la ingeniería inversa para Java, .Net, XML, Hibernate además de poder exportar



imágenes a jpg y png. Genera el código y la base de datos a partir de los diagramas UML realizado pudiéndose generar también máquina de estado. Soporta aplicaciones web y varios idiomas, además de que coexiste gran compatibilidad entre sus ediciones siendo multiplataforma. Como desventaja se puede decir que esta herramienta no permite su uso en proyectos comerciales, pues es un software propietario, aunque la UCI paga su licencia para su uso.

Visual Paradigm es una herramienta CASE de fácil uso. Tiene la ventaja de ser multiplataforma. Soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientado a objetos, construcción, pruebas y despliegue. Este software de modelado ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a menor costo. Permite dibujar todo los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Dentro de los diagramas que permite crear en el modelado de datos se encuentra el Diagrama de Entidad-Relación (DER), dicho diagrama es usado en el diseño de la propuesta de la base de datos SIPP 2.0.

2.4. Herramienta de Desarrollo. Pg Admin III.

Las herramientas de desarrollo son aquellos programas o aplicaciones que tengan cierta importancia en el desarrollo de un programa. Pg-Admin III es una aplicación gráfica para gestionar el gestor de bases de datos PostgreSQL, siendo la más completa y popular con licencia Open Source (Código Abierto). Está escrita en C++ usando la librería gráfica multiplataforma wxWidgets, lo que permite que se pueda usar en Linux, FreeBSD, Solaris, Mac OS y Windows. Diseñado para responder a las necesidades de todos los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas. La interfaz gráfica soporta todas las características de PostgreSQL y facilita enormemente la administración. La aplicación también incluye un editor SQL con resaltado de sintaxis, un editor de código de la parte del servidor y un agente para lanzar scripts programados. La conexión al servidor puede hacerse mediante conexión



TCP/IP o Unix Domain Sockets (en plataformas * nix) y puede encriptarse mediante SSL para mayor seguridad[36].

Entre las características que presenta se puede encontrar que tiene entradas SQL aleatorias, haciendo más fácil la interacción con el sistema de base de datos. Contiene pantalla de información, ayuda para base de datos, tablas, índices, secuencias, vistas, programas de arranque, funciones y lenguajes. Presenta preguntas y respuestas para configurar usuarios, grupos y privilegios. Pg Admin III mantiene control de revisión con mejoras de generación de script. La interoperabilidad, característica que presenta esta herramienta de desarrollo, pues contiene ayuda para importar y exportar datos y migrar base de datos, además de generar informes predefinidos en la base de datos, tablas, índices, secuencias, lenguaje y vistas.

Por todas las características y ventajas que presenta se continúa defendiendo la idea de utilizar como herramienta de desarrollo Pg Admin III.

2.5. Herramienta para prueba de volumen de la base de datos. EMS Data Generator 2005 para PostgreSQL.

En la ingeniería de software las pruebas de rendimiento son las pruebas que se realizan desde una perspectiva, para determinar lo rápido que se realiza una tarea en un sistema en condiciones particulares de trabajo. Pueden servir para validar y verificar atributos de la calidad del sistema, tales como la escalabilidad, fiabilidad y el uso de recursos. La prueba de carga es el tipo más sencillo de pruebas de rendimiento. Una prueba de carga se realiza generalmente para observar el comportamiento de una aplicación bajo una cantidad de peticiones esperada. Esta carga puede ser el número esperado de usuarios concurrentes utilizando la aplicación y que realizan un número específico de transacciones durante el tiempo que dura la carga.

EMS Data Generator para PostgreSQL es una potente herramienta para la generación de datos de prueba a varias tablas de base de datos PostgreSQL de una vez. La



aplicación asistente permite definir tablas y campos para generar datos, configurar valores de campos, generar campos de tipo carácter por máscara, obtener listas de valores de las consultas SQL y muchas otras características para generar datos de prueba de una forma sencilla y de manera directa. Permite generar datos en un solo toque mediante el uso de plantillas de generación. Otras características que se pueden encontrar en esta herramienta es que tiene interfaz amigable y fácil de utilizar, maneja todos los tipos de datos de PostgreSQL incluyendo matriz, direcciones de red, y tipos geométricos. Tiene la capacidad para establecer los valores nulos para cierto porcentaje de caso y controla automáticamente temas como la integridad referencial para las tablas vinculadas de generación de datos[37].

2.6. Herramientas para replicación de base de datos.

La replicación de datos consiste en el transporte de datos entre dos o más servidores, permitiendo que ciertos datos estén almacenados en más de un sitio, y así aumentar la disponibilidad de los datos y mejorar el rendimiento de las consultas globales[38]. El proceso de replicación de una base de datos consiste en replicar las consultas de actualización en una base de datos maestra sobre una o varias bases de datos esclavas, de manera que se tenga una copia de las mismas a largo tiempo[39]. Por lo tanto se puede afirmar que la replicación de base de datos constituye el proceso por el cual se desea mantener y copiar los datos de una base de datos, de esta manera los datos son transportados y totalmente almacenados.

Las herramientas de replicación de base de datos son imprescindibles para centros de bases de datos y proveedores de alojamiento que desean ofrecer un respaldo de línea de sus datos a sus datos. Es necesario entonces para el sistema SIPP versión 2.0 contar con una herramienta de replicación que transporte todos los datos almacenados en el servidor del área de trabajo en campo de la perforación hacia las oficinas de la DIPP. Primero porque es necesario para los directivos conocer la información en tiempo



real y tomar las decisiones pertinentes en cada situación. Además se lograría la disponibilidad de la información y la integración del sistema tanto en la DIPP con en el pozo en perforación, logrando la consistencia de los datos en tiempo real.

Por la corta visión del negocio y el estudio insuficiente del mismo en la primera versión de la base de datos no se tuvo en cuenta una herramienta de replicación que estuviera presente en el sistema. Para completar este estudio de las herramientas de replicación se pasa a caracterizar algunas de las más reconocidas actualmente.

Para analizar las herramientas de replicación se deben entender algunos términos como:

Replicación asincrónica: Un servidor recibe una respuesta de escritura completada del sistema de almacenamiento cuando la operación de escritura se confirma en el dispositivo de almacenamiento local. En este escenario, el sistema de almacenamiento no espera una confirmación del dispositivo de almacenamiento remoto.

Replicación sincrónica: Un servidor recibe solamente una respuesta de escritura completada del sistema de almacenamiento cuando la operación de escritura de entrada o salida se completa tanto en la ubicación de almacenamiento local como en la ubicación de almacenamiento remoto.

Cluster: Es el conjunto de instancias de bases de datos PostgreSQL que están envueltos en la replicación.

Nodo: Se le llama así a cada una de las bases de datos envueltas en la replicación. Es la base de datos que se encuentra envuelta en el proceso de replicación.

Nodo maestro: Servidor que acepta modificaciones del usuario.

Nodo esclavo: Servidor que recibe modificaciones del maestro.

**Slony-I.**

Es un software que permite hacer replicaciones maestro/esclavo asíncrono, realizando actualizaciones en cascada. Slony-I es un maestro para varios sistemas de replicación esclavos en cascada de apoyo (por ejemplo, un nodo puede alimentar a otro nodo que se alimenta de otro nodo) y de conmutación por error. El panorama para el desarrollo de Slony- I es que es un esclavo de replicación del sistema principal que incluye todas las características y las capacidades necesarias para replicar base de datos de gran tamaño a un número razonable limitado de los sistemas esclavistas. Es un sistema diseñado para su uso en centro de datos y sitios de respaldo, en el modo normal de operación es que todos los nodos están disponibles[40].

Es un replicador maestro a múltiples esclavos mediante disparadores con un esquema avanzado de configuración que tiene adherido un pequeño lenguaje de configuración. Entre sus principales desventajas se puede encontrar que su instalación, configuración y administración es compleja. Además no es viable para conexiones inestables o configuraciones variables.

PyReplica.

PyReplica es un replicador asíncrono maestro-esclavo simple para PostgreSQL basado en Python, usando un disparador maestro en ppython, señales, secuencias, y un script cliente en python. Soporta además de una configuración multi-maestro asíncrona (replicación en ambos sentidos, cada base de datos es un maestro y un esclavo al mismo tiempo)[41]. Entre las características que cuenta esta herramienta se cuenta con la facilidad de la instalación, solamente basada en scripts y sin compilación. Además cuenta con la facilidad de la administración, no hay manejos de comando, y el mantenimiento de la misma. Agregar que la herramienta es multiplataforma (Windows y Linux). Soporta la replicación de valores devueltos por funciones de fecha y secuencias.



Incluye aspectos como la detención de conflictos y monitoreos de conexiones. Algunas de las desventajas de la herramienta son de soporte como: no soporta la replicación de DDL automática, pero puede usarse para propagar órdenes DDL a varios servidores y no soporta ni la replicación de secuencias ni la replicación sincrónica simultánea [42].

SymmetricDS.

SymmetricDS es un software de replicación de datos asincrónica apoya varios suscriptores y sincronización bi-direccional. Utiliza tecnologías web y base de datos para replicar tablas entre bases de datos relacionales en tiempo casi real. Está diseñado para escalar a un gran número de base de datos, trabajar a través de conexiones de bajo ancho de banda y soportar períodos de interrupción de la red. Puede ser instalado como una aplicación web en un servidor de aplicaciones Java o puede ser instalado independientemente incrustado en otra aplicación Java. Una sola instalación de SymmetricDS se denomina nodo. Un nodo es inicializado por un archivo de propiedades y se configura mediante la inserción de los datos de configuración en una serie de tablas en la base de datos. SymmetricDS admite la sincronización a través de plataformas de bases de datos diferentes a través del concepto de base de datos dialectos. Un dialecto es una base de datos de capa de abstracción, SymmetricDS interactúa con el aislamiento, la lógica de sincronización de base de datos principales de los detalles específicos de aplicación[43].

SymmetricDS permite que los cambios de entrada y de salida sean sincronizadas con otras bases de datos. El nodo que inicia la conexión es el cliente y el nodo que recibe la conexión es el servidor. Dado que la sincronización está configurada para extraer o enviar en cualquier dirección, el mismo nodo puede actuar a la vez como cliente o como servidor en diferentes circunstancias.

Después de haber analizado algunas de las herramientas de replicación de datos de las más usadas a nivel mundial, se puede concluir planteando que las tres son poderosas



en cuanto a la replicación de los datos. Las tres cumplen las características de ser de código abierto ventaja que las sitúa como unas de las primeras de su tipo en el mundo en utilidad. Pero Slony-I junto con PyReplica tienen muchas desventajas frente a SymmetricDS primeramente porque su configuración es muy compleja con decenas de archivos y miles de líneas, mientras que SymmetricDS se configura de manera fácil para el usuario, soporta la sincronización entre diferentes plataformas de base de datos, mediante el concepto de *dialectos de base de datos*⁵. También Slony-I y PyReplica son muy inestables en condiciones extremas por lo que son más difíciles de usar y adaptar mientras que SymmetricDS está basado en disparadores, incluye notificaciones de nodos.

Herramienta Propuesta.

Las bases de datos distribuidas son cada vez más usadas por las empresas y suponen una ventaja competitiva frente a los sistemas centralizados, siempre y cuando la empresa en cuestión tenga la necesidad de usar una base de datos de este tipo. Lo más habitual es disponer de varias sedes y tener que manejar información común, para lo cual las bases de datos distribuidas son especialmente útiles.

SymmetricDS constituye una potente herramienta para la replicación de datos, ya sean homogéneos o no, independientemente del gestor de base de datos que se utilice, siempre y cuando soporte la tecnología de trigger y driver JDBC. Es una herramienta en constante evolución con una amplia comunidad, fácil de aprender y con más de una posibilidad para su despliegue, lo que permite usar la forma más adecuada según las características del hardware que se posea. Su configuración es manejada centralmente. Ayudó a la propuesta el hecho de que ya la herramienta se probó en el proyecto con excelentes resultados y además es una herramienta que se maneja actualmente por el centro de asesoría de PostgreSQL en la universidad.

⁵ Un dialecto de base de datos es una capa de abstracción con la cual interactúa SymmetricDS para aislar la lógica de sincronización de los detalles de implementación específicos de cada base de datos.



2.7. Descripción de la Arquitectura propuesta y fundamentación.

La arquitectura constituye la base fundamental en el desarrollo de un software, este favorece a que todos los programadores, diseñadores, y analistas implicados en la creación del mismo lleguen a un intercambio común y todo se desarrolle siguiendo los estándares y patrones ya definidos. En el marco del trabajo con base de datos se puede decir que constituye el conjunto de reglas y normas a respetar para facilitar el trabajo de desarrollo y mantenimiento de los objetos en la base de datos. Dichas normas abarcan desde las puramente tecnológicas hasta las más triviales como puede ser nomenclaturas de objetos o convenciones de tamaño para un tipo de datos específico según la función que se realiza.

Se hace necesario para poder desarrollar la base de datos lograr la definición de un perfil tecnológico de la misma. Con la investigación que se realiza en el epígrafe anterior se marcan pautas de las herramientas que serán utilizadas para el diseño y la implementación de la misma. Se mantiene, como en la primera versión, el sistema gestor de base de datos PostgreSQL y como herramienta de diseño el Visual Paradigm

por todas las características y ventajas que presentan. Como cliente de base de datos se escoge Pg Admin III por estar presente en la primera versión de la base de datos y las particularidades que presenta.

El ambiente donde será explotado el producto tiene algunas características no elegibles por los desarrolladores por lo que hay que adecuarse a las políticas específica de la institución, sobre todo en mantener la seguridad. El control de acceso al servidor se realiza mediante los archivos de configuración de PostgreSQL (postgresql.conf y pg_hba.conf). Estos archivos regulan todas las variables necesarias para controlar las direcciones IP con permisos para acceder al servidor de datos, los usuarios y la cantidad de conexiones aceptadas.



No se especifica en la arquitectura definida de la primera versión de la base de datos ningún límite de concurrencia. Sin embargo es necesaria la definición de este parámetro dentro de la misma, para lograr una cohesión en la seguridad del sistema de base de datos. Debido al poco volumen de personal existente en la DIPP y en los pozos en perforación que interactúan con la base de datos, se puede inferir una concurrencia máxima de 100 usuarios, asumiendo el peor caso de que todos los trabajadores se conecten a la vez. Los permisos en la base de datos esquemas y objetos asociados, se realizarán a través de la creación de roles y grupos de roles, para así darle permisos de lectura y escritura en dependencia de la responsabilidad y funcionalidad necesaria.

2.7.1. Soporte Arquitectónico.

El modelo propuesto cuenta con 110 tablas, distribuidas en seis esquemas. El esquema para las tablas nomencladoras del proceso, el cual contiene 38 tablas. El esquema para las tablas directamente implicadas en el proceso de perforación, el cual contiene 49 tablas. El esquema que almacena los datos iniciales del pozo, llamado proyecto_pozo, compuesto por 6 tablas. Se incluye el esquema encamisado y cementación, que responde a las actividades de encamisado de intervalos y cementación de los mismos, conformado por 9 tablas. El esquema montaje conformado por 2 tablas, que manejan los costos en dicha fase y por último el esquema de seguridad con 6 tablas que corresponden al registro, trazabilidad y seguridad del sistema. Todos los esquemas responden a los diferentes módulos del sistema: Nomencladores, Administración, Supervisor, Geólogo y DIPP.



2.7.2. Estándares de Nomenclatura.

Base de Datos.

El nombre de la base de datos se escribe con todas las letras en minúsculas. En caso de ser un nombre compuesto las palabras se separan por “_”. Existen dos entornos que interactúan con el servidor de base de datos. El primer entorno y el más importante ya que soporta un volumen grande de datos es el servidor de la DIPP el cual tiene por nombre “*dipp*” y la base de datos “*sipp*”. El segundo entorno es en los pozos en perforación, donde el servidor está en la máquina de los supervisores y la nomenclatura para las mismas es *sipp_<<códigopozo>>*. Ejemplo: En el caso de un pozo donde su código definido por la DIPP sea vd1004 será *sipp_vd1004*.

Esquemas.

Los esquemas en que está distribuidas las tablas son cinco. Se nombran *sqm_<<nombredelesquema>>*. Ejemplo: En el caso del esquema nomencladores sería *sqm_nomencladores*.

Tablas.

El nombre a emplear para las tablas se escribe con todas las letras en minúsculas, sin acentos y las palabras separadas por el símbolo “_”. Ejemplo: *informacion_geologica_plan*.

Tablas Nomencladores.

Las tablas que representen nomencladores del sistema comienzan su nombre por el prefijo *n* seguida por su nombre según la regla de nomenclatura para las tablas. Ejemplo: *n_tipo_diametro*.

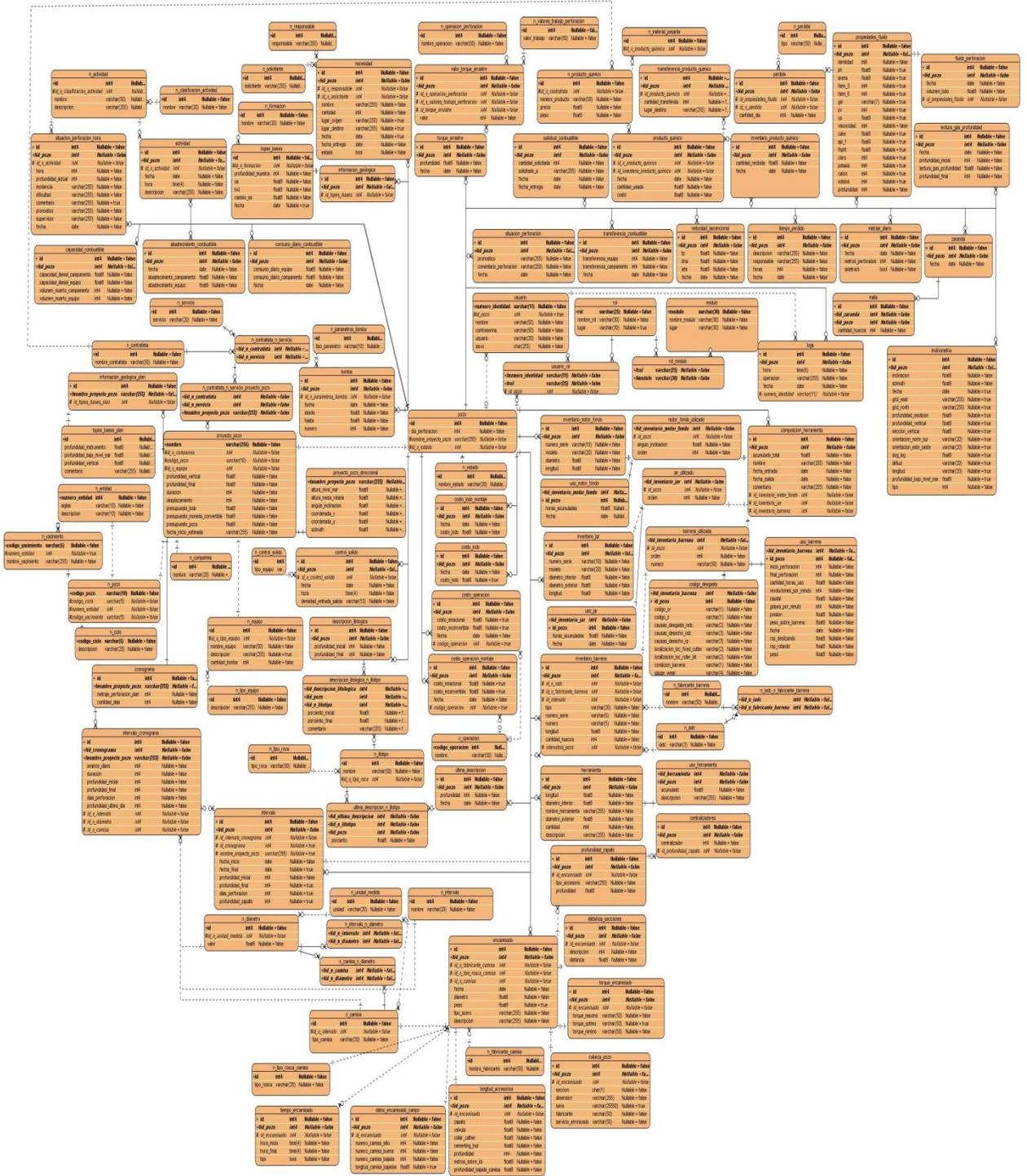
**Campos.**

El nombre a emplear para los campos debe ser intuitivo, evitar las abreviaturas y escribirse con todas las letras minúsculas comenzando por el prefijo correspondiente y las palabras separadas por el símbolo “_”. Si el campo es un identificador de llave primaria debe ser id en caso de que no exista otro atributo único que pueda identificar a la tabla. Si es un identificador de llave foránea debe estar constituido por id y el nombre de la tabla de la cual se extrae la llave, separados por “_”. Si la tabla de la cual se extrae la llave foránea no tiene como identificador el campo id entonces se mantiene como llave foránea el nombre del campo identificador de la tabla.

Ejemplo: *id_n_actividad*.



2.8.2. Diagrama Entidad-Relación de la base de datos propuesta.



**2.9. Conclusiones parciales.**

En este capítulo se realiza la propuesta del diseño de la base de datos del Sistema para el Manejo Integral de la Perforación de Pozos de Petróleo y Gas (SIPP) versión 2.0, con el objetivo de darle respuesta al problema de la investigación. En dicha propuesta se definió el soporte arquitectónico que tendría la base de datos además de los estándares de nomenclatura para la misma. Se redujo la redundancia de la información debido a que no existen datos repetidos innecesariamente con lo cual también aumenta la integridad de los mismos. Como resultado del actual diseño se logró obtener una base de datos donde existe integridad, organización e independencia de los datos cumpliendo con las exigencias y objetivos planteados en la investigación.



Capítulo 3: Validación teórica del diseño.

3.1. Introducción.

En el presente capítulo se aborda la validación teórica y funcional del diseño de la base de datos de SIPP (versión 2.0). Inicialmente se recogen aspectos como la integridad, seguridad, redundancia de la información y normalización de base de datos. En un segundo momento se tratan aspectos como las pruebas realizadas a la base de datos para comprobar su correcto y eficiente funcionamiento.

3.2. Integridad.

El concepto de integridad, proviene del término latino *integrítas*, se refiere a la cualidad de íntegro. Aquello íntegro es algo que no carece de ningunas de sus partes, hace referencia a lo intachable[45]. La integridad en una base de datos se refiere a la corrección y exactitud de la información contenida. Una base de datos determinada podría estar sujeta a cualquier cantidad de restricciones de integridad de una complejidad arbitraria[46]. Se puede definir la integridad en término de base de datos como el conjunto de seguridades que son utilizadas para mantener los datos correctos.

Cuando se utilizan sentencias INSERT, DELETE o UPDATE, la integridad de datos puede verse afectada; ya que se puede añadir datos no válidos o modificarse datos existentes en forma incorrecta, con lo que la integridad no se cumple[45]. La exigencia de la integridad de datos garantiza la calidad de los datos y que la información almacenada en las tablas sea válida, coherente y exacta.

Una vez que se define la estructura de datos del modelo relacional es decir una vez que se determina el modelo conceptual, se definen las reglas de integridad que los datos almacenados en dicha estructura deben cumplir para garantizar que seas correctos. Al definir cada atributo sobre un dominio se impone una restricción sobre el conjunto de



valores permitidos para cada atributo. Hay además dos reglas de integridad muy importantes que son restricciones que se deben cumplir en todas las bases de datos relacionales. Estas reglas son la de integridad de entidad y las de integridad referencial. Antes de definir estas reglas es preciso conocer el concepto de NULO y de DOMINIO[46].

NULO: Es un indicador que le dice al usuario que el dato falta o es aplicable. Por conveniencia un dato que falte se dice que tiene valor NULO, pero el valor NULO no es un tipo de dato real. En vez de ello es una señal o un recordatorio de que el valor falta o es desconocido[46].

DOMINIO: Posibles valores que puede tener un campo. Un dominio no es más que un tipo de dato, posiblemente un tipo simple definido por el sistema o por el usuario. Esta característica, usada en forma correcta se convierte en un mecanismo de control, restricción y validación de los datos a ingresar[46].

Dos pasos importantes en el diseño de las tablas son la identificación de valores válidos para una columna y la determinación de cómo forzar la integridad de los datos en la columna. La integridad de datos pertenece a una de las siguientes categorías:

- **Integridad de entidad.**

La integridad de entidad define una fila como entidad única para una tabla determinada. La integridad de entidad exige la integridad de las columnas de los identificadores o la clave principal de una tabla, mediante índices y restricciones únicas, o restricciones de llaves primarias[47]. Las restricciones de entidades aseguran la integridad de las entidades que son modeladas por el sistema. En el nivel más simple, la existencia de una clave principal es una restricción de entidad que impone la regla “cada entidad debe estar identificada de forma única”. En esta no está permitido que algún componente de clave primaria acepte valores nulos[46].



Las razones de esta regla son:

- Las tuplas en las relaciones base representan entidades en la realidad.
- Las entidades en la realidad son identificables por definición.
- Sus contrapartes en la base de datos también deben ser identificables.
- Los valores de la clave primaria sirven como identificadores en la base de datos.
- Los valores de clave primaria no pueden ser nulos.

En el caso de la base de datos diseñada para SIPP (versión 2.0) cada entidad reflejada en el modelo relacional confeccionado tiene definida su llave primaria que no puede ser nula ni repetir.

- **Integridad de dominio**

La integridad de dominio viene dada por la validez de las entradas para una columna determinada. Puede exigir la integridad de dominio para restringir el tipo mediante tipos de datos, el formato mediante reglas y restricciones CHECK, o el intervalo de valores posibles mediante restricciones de llave foránea, restricciones CHECK, definiciones por defecto, definiciones no nulas y reglas[47]. Una definición bien adecuada de restricciones de dominio no sólo permite probar valores insertados en la base de datos. También permite probar consultas para asegurar que las comparaciones que se hacen tienen sentido.

En la base de datos se cumple la integridad de dominio en sus diferentes tablas. Un ejemplo de esto es en la tabla *metraje_diario*, donde existe un campo que define un momento crítico en el sistema y es cuando existe sidetrack⁶. El dominio para este campo admite solamente el conjunto de valores siguientes: verdadero, para especificar que hubo sidetrack y falso para cuando continúa la dirección del pozo pronosticada,

⁶ Técnica utilizada en la perforación direccional. El objetivo es poder sortear una pesca de herramienta perdida en el pozo. Puede ocasionarse en zonas donde existen cambios inesperados de configuración geológica cambiando la trayectoria direccional original del pozo.



evitando que el usuario introduzca datos no adecuados que afecte la integridad del sistema. (Figura 4)

metraje_diario		
+ id	int4	Nullable = false
+#id_pozo	int4	Nullable = false
fecha	date	Nullable = false
metros_perforados	int4	Nullable = false
sidetrack	bool	Nullable = false

Figura 4. Tabla donde se evidencia la integridad de dominio.

▪ Integridad referencial

La integridad referencial es una propiedad deseable en las bases de datos, garantiza que una entidad siempre esté relacionado con entidades válidas, es decir que existan en la misma base de datos. La integridad referencial supone que todos los datos sean correctos, sin repeticiones, datos perdidos o relaciones mal resueltas[45].

La regla de integridad referencial define que la base de datos no puede contener valores de claves foráneas sin concordancia. Esta regla se aplica a las claves foráneas. Si en una relación hay laguna clave foránea, entonces sus valores deben coincidir con los valores de la clave primaria a la que hace referencia, o bien, puede ser completamente nulo[46].

La integridad referencial también vigila que se cumplan las siguientes reglas:

- No se podrá introducir un valor en la tabla relacionada si antes no ha sido introducida en la tabla principal.
- No se puede eliminar un registro de una tabla principal si existen registros coincidentes en la tabla relacionada.
- No se puede cambiar un valor de la clave primaria en la tabla principal si el registro tiene registros relacionados.
- Integridad definida por el usuario.



Un ejemplo de aplicación de este concepto en el modelo de base de datos propuesto es el siguiente. Se asegura que los atributos *id_n_compannia*, *codigo_pozo* y *id_n_equipo* de la entidad *proyecto_pozo* sean llaves foráneas debido a que provienen y son llaves primarias de las entidades *n_compannia*, *n_pozo* y *n_equipo* respectivamente. Además cumplen las restricciones siguientes:

- Los números y códigos identificadores de *id_n_compannia*, *codigo_pozo* y *id_n_equipo* en la entidad *proyecto_pozo* tienen los mismos dominios que en sus identidades de origen.
- Al introducir números identificadores de *id_n_compannia*, *id_n_equipo* en la tabla *proyecto_pozo*, éstos tienen que encontrarse en su entidad origen, es decir *proyecto_pozo* no puede tener un número identificador *id_n_compannia* y *id_n_equipo* que no estén presente en la tablas *n_compannia* y *n_equipo* respectivamente. (Ver)

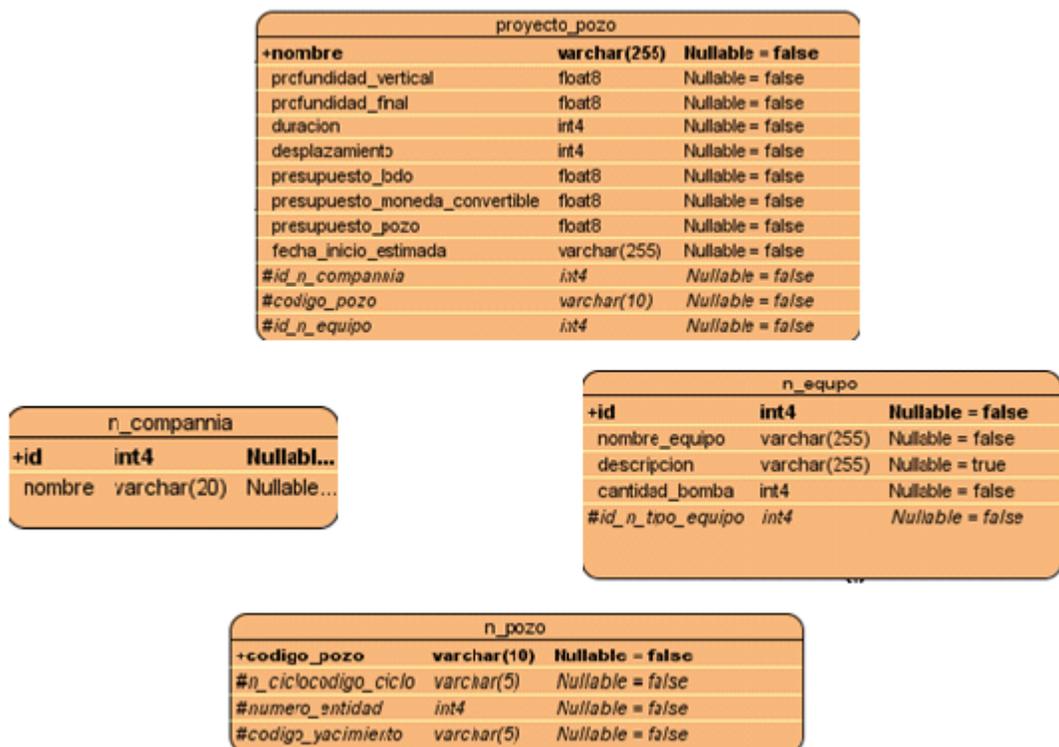


Figura 5. Tablas donde se evidencia la integridad referencial.



Por lo planteado anteriormente se comprueba que en el modelo de datos se cumple con la regla de integridad referencial, que es una de las más importantes dentro de la integridad de la base de datos.

- **Integridad definida por el usuario**

La integridad definida por el usuario permite definir reglas de empresa específicas que no pertenecen a ninguna otra categoría de integridad. Todas las categorías de integridad admiten la integridad definida por el usuario. Esto incluye todas las restricciones de nivel de columna y nivel de tabla en CREATE TABLE, procedimientos almacenados y desencadenadores. El negocio modelado no define ninguna regla particular para las entidades persistentes por lo que la base de datos SIPP (versión 2.0) no fue modelada bajo ninguna restricción de usuario.

3.3. Normalización de la base de datos.

La normalización es la expresión formal del modo de realizar un buen diseño y provee los medios necesarios para describir la estructura lógica de los datos en un sistema de información[2]. La normalización es el proceso mediante el cual se transforman datos complejos a un conjunto de estructuras de datos más pequeñas, que además de ser simples y más adaptables, son más fáciles de mantener. También se puede entender la normalización como una serie de reglas que sirven para ayudar a los diseñadores de bases de datos a desarrollar un esquema que minimice los problemas de lógica. Cada regla está basada en la que antecede[48]. Se puede concluir planteando que la normalización es el proceso en el cual se va comprobando el cumplimiento de una serie de reglas, o restricciones; cada regla que se cumple aumenta el grado de normalización ayudando a evitar anomalías en la actualización y mejorando la independencia de los datos.



El diseño de base de datos conceptual proporciona el esquema de relaciones y sus restricciones de integridad que sirve como punto inicial para el proceso de implementación de la base de datos. El proceso de normalización es un estándar que básicamente, es un proceso de conversión de las relaciones evitando:

- La redundancia de los datos: repetición de datos en un sistema.
- Anomalías de actualización: inconsistencias de los datos como resultado de datos redundantes y actualizaciones parciales.
- Anomalías de borrado: pérdidas no intencionadas de datos debido a que se han borrado otros datos.
- Anomalías de inserción: imposibilidad de adicionar datos en la base de datos debido a la ausencia de otros datos.

- **Proceso de Normalización**

El proceso de normalización va reduciendo un conjunto de relaciones, a una forma más sencilla y simple. Consiste en ir descomponiendo relaciones en otras relaciones con un mayor grado de sencillez en términos de las operaciones que se ejecutan sobre ellas. El proceso incluye varias fases que se realizan en orden. La realización de la 2da. Fase supone que se ha concluido la 1ra y así sucesivamente. Tras completar cada fase se dice que la relación está en: 1FN, 2FN, 3FN, FNBC, 4FN y 5FN. Se dice que una determinada relación está en una forma normal si satisface un cierto conjunto de restricciones.

Para comenzar a estudiar formas normales es necesario conocer algunos conceptos relacionados con el proceso.

- Dependencia funcional: Dada una relación R , se dice que el atributo y de R es funcionalmente dependiente del atributo x de R , si y solo si, cada valor x en R



tiene asociado a él, precisamente, un valor de y en R en cualquier momento del tiempo[2].

- Dependencia funcional completa: El atributo y es funcionalmente y completamente dependiente del atributo x , si es funcionalmente dependiente de x y no es funcionalmente dependiente de algún subconjunto de x .

➤ Formas Normales

Existen básicamente seis formas normales incluyendo la Forma Normal BOYCE-COOD. Cada una de estas formas tiene sus propias reglas. Cuando una base de datos se conforma a un nivel, se considera normalizada a esa forma de normalización. No siempre es una buena idea tener una base de datos conformada en el nivel más alto de normalización, puede llevar a un nivel de complejidad que pudiera ser evitado si estuviera en un nivel más bajo de normalización.

En general, las primeras tres formas normales son suficientes para cubrir las necesidades de la mayoría de las bases de datos. El creador de estas tres primeras formas normales o reglas fue Dr. Edgar Frank Codd⁷.

➤ Primera Forma Normal (1FN)

Formalmente una relación está en 1FN si cumple la propiedad de que sus dominios no tienen elementos que a su vez, sean conjuntos. Los valores que puede tomar un atributo de una relación son los elementos contenidos en su correspondiente dominio. Si se permitiera que un elemento del dominio de un atributo fuera un conjunto, entonces dicho atributo pudiera tomar como valor ese conjunto de valores.

⁷ Brillante matemático y científico de la computación. Nacido en Inglaterra, la mayor parte de su vida la pasó en Estados Unidos trabajando y desarrollando sus ideas que culminaron en una serie de informes técnicos acerca de una nueva manera de organizar y acceder a los datos. Fue creador de las tres primeras formas normales en 1972 y en 1974 conjuntamente con BOYCE la FNBC.



Eso implicaría que en una posición fila, columna habría un conjunto de valores y no un único valor, como debe ocurrir en el modelo relacional[2].

Se puede traducir lo dicho anteriormente como: Una tabla está en primera Forma Normal si y solo si:

- Todos los atributos son atómicos. Un atributo es atómico si los elementos del dominio son indivisibles, mínimos.
- La tabla contiene una clave primaria.
- La tabla no contiene atributos nulos.
- Si no incluye ningún grupo repetitivo (un grupo repetitivo es un atributo que contiene un conjunto de valores y no un único valor).

Lo que se persigue en realidad al normalizar a este nivel es:

- Eliminar los grupos repetitivos de las tablas individuales.
- Crear una tabla separada por cada grupo de datos relacionados.
- Identificar cada grupo de datos relacionados con una clave primaria.

➤ **Segunda Forma Normal (2FN)**

Una relación R se dice que está en **2FN** si está en **1FN** y si, y solo si, los atributos no llaves (ni primarias, ni candidatas) de R , si los hubiese, son funcional y completamente dependientes de la llave primaria de R [2]. La regla de la 2da forma normal establece que todas las dependencias parciales se deben eliminar y separar dentro de sus propias tablas. Una dependencia parcial es un término que describe a aquellos que no dependen de la llave primaria de la tabla para identificarlos.

Lo que se persigue en realidad al normalizar a este nivel es:

Crear tablas separadas para aquellos grupos de datos que se aplican a varios registros.



- Relacionar estas tablas mediante una clave externa.

➤ **Tercera Forma Normal (3FN)**

Una relación R está en 3FN si está en 2FN y si, y solo si, los atributos no llaves son independientes de cualquier otro atributo no llave primaria. Esto es lo mismo que decir que se deben eliminar las dependencias transitivas de atributos no llaves respecto a la llave primaria, estando ya la relación en 2FN[2].

Una tabla está normalizada en esta forma si todas las columnas que no son llave funcionalmente dependientes por completo de la llave primaria y no hay dependencias transitivas. Una dependencia transitiva es aquella en la cual existen columnas no llave que dependen de otras columnas que tampoco son llave. Cuando las tablas están en 3ra. Forma Normal se previenen errores de lógica cuando se insertan o borran registros. Cada columna en una tabla está identificada de manera única por la llave primaria, y no deben datos repetidos. Esto provee un esquema limpio y elegante, que es fácil de trabajar y expandir.

Lo que se persigue en realidad al normalizar a este nivel:

- Creando tablas para los atributos no llaves que no dependen transitivamente de la llave primaria (y los atributos que no se analizan por ser atributos llaves pertenecientes a claves candidatas).
- Creando una tabla para los atributos no llaves que dependen transitivamente de la llave primaria a través de otro atributo o conjunto de atributos no llave primaria (que no son parte de la llave primaria). La llave primaria de la relación así formada será el atributo o conjunto de atributos a través de los cuales existe la dependencia transitiva.



3.4. Análisis de la redundancia de la información.

La redundancia hace referencia al almacenamiento de los mismos datos varias veces en diferentes lugares. La redundancia de datos puede provocar problemas como:

- Incremento del trabajo: Como un mismo dato está almacenado en dos o más lugares, esto hace que cuando se graben o actualicen los datos, deban hacerse en todos los lugares a la vez.
- Desperdicio de espacio de almacenamiento: Ya que los mismos datos están almacenado en varios lugares distintos, ocupando así más bytes del medio de almacenamiento. Este problema es más evidente en grandes bases de datos.
- Inconsistencia de datos: Esto sucede cuando los datos redundantes no son iguales entre sí. Esto puede suceder, por ejemplo, cuando se actualiza el dato en un lugar pero el dato duplicado en otro lugar no actualizado.

Por todo lo antes planteado se puede llegar a la conclusión que la base de datos SIPP (versión 2.0) no cuenta con redundancia de datos puesto que a la misma además de aplicársele las formas normales hasta la tercera forma normal se hizo énfasis en el diseño del modelo entidad-relación de la base de datos para precisamente evitar la redundancia de los datos, exceptuando la redundancia de datos controlada, que se emplea para mejorar el rendimiento en las consultas a las bases de datos y disminuir el tiempo de acceso a los datos o para simplificar el método significativo de direccionamiento.



3.5. Análisis de la seguridad y recuperación de la base de datos.

Los datos constituyen un recurso valioso que debe ser estrictamente controlado gestionado al igual que cualquier otro recurso corporativo. La seguridad garantiza el acceso autorizado a los datos, para interrumpir cualquier intento de acceso no autorizado, ya sea por error del usuario o por mala intención[2]. Las bases de datos, en particular, deben tener un sistema de seguridad sólido para controlar las actividades que pueden realizarse y determinar qué información puede verse y cuál puede modificarse. Un sistema de seguridad sólido, asegura la protección de datos sin tener en cuenta cómo los usuarios obtienen el acceso a la base de datos.

➤ Seguridad y Control de Acceso

La seguridad de la base de datos está implementada en varios niveles:

El nivel 0 es el que se encarga de las maquina (host) y los usuarios. Es decir, permite configurar que máquina/s y/o usuario/s se pueden conectar a la base de datos. Utilizando las opciones del fichero de configuración del SGBD `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 qué base de datos se pueden conectar. Utilizando las opciones del fichero de configuración del SGBD `pg_ident.conf`.

El nivel 2 es el que se encarga de las tablas. Es el que permite configurar a qué tablas pueden acceder. Utilizando los comandos `GRANT` para dar permisos y `REVOKE` para eliminar los permisos.

La seguridad debería conservarse por el mantenimiento de los usuarios de las bases de datos. Dependiendo del tamaño de un sistema de base de datos y de la cantidad de trabajo requerido para administrar los usuarios de base de datos, el administrador de seguridad debería ser el único usuario con los privilegios requeridos para crear, alterar



o borrar un usuario de la base de datos. Por otro lado, debería haber un número de administradores con privilegios para administrar los usuarios de base de datos. Sólo personas que gozan de confianza deberían tener privilegios totales para administrar los usuarios de base de datos.

Para la seguridad de la solución propuesta es necesario el acceso por otro usuario a la base de datos que no sea el que brinda PostgreSQL por defecto “postgres”. Para ello se realiza la creación de usuarios:

- **superadministrador:** El usuario tiene todos los privilegios de acceso y creación de copias de seguridad de la base de datos, así como el mantenimiento de la misma. En cualquier circunstancia o pérdida de la información será el encargado del restablecimiento inmediato de la información.
- **administrador:** El usuario solo tiene los privilegios que le serán concedidos por el superadministrador, solo tendrá privilegios de acceso a las tablas referentes a la aplicación en los módulo/s que utiliza.
- **trabajo:** Usuario para el uso de los desarrolladores. Tiene permisos para insertar, actualizar, eliminar datos de las tablas así como ejecutar funciones y obtener los datos en cuestión.

Se debe configurar, de acuerdo con las políticas de acceso permitidas, las conexiones locales y remotas mediante los archivos de configuración: `pg_hba.conf` y `postgresql.conf` (Anexo1).



➤ **Backups y recuperación de los datos.**

Existen dos formas de realizar el aseguramiento de los datos.

1. Compactado del directorio de datos.

- Detener el servicio de PostgreSQL.
- Realizar la copia en *.rar o *.zip del directorio data de la carpeta de instalación de PostgreSQL.
- Iniciar el servicio de PostgreSQL.
- Copiar el *.rar o *.zip en algún soporte. (CD, disco).

Para recuperar la copia: descomprimir en el directorio data de la carpeta de instalación de PostgreSQL.

Ventajas:

- Es rápido y sencillo.

Desventajas:

- Incompatibilidad de versiones.

2. Volcado (dump)

Para que no exista pérdida de información PostgreSQL permite realizar salvadas automáticas a petición del cliente para ello utiliza el comando `pg_dump` el cual indica al SGBD que va a realizar la salvada de la Base de Datos. Se le ha de proveer una serie de parámetros, el nombre de la Base de Datos, el usuario y la contraseña para conectarse a la misma, el nombre que tomará el archivo de salvada y donde se guardará la misma. Las desventajas de este proceso residen en la lentitud que puede alcanzar con bases de datos de gran volumen y en el consumo de recursos del sistema.

Si se desea restaurar salvadas, se utiliza el comando `pg_restore`, indicándole la salvada que se desea restaurar.



Considerando que la base de datos es hasta el momento pequeña, de las que se puede hacer copias de seguridad con rapidez, se propone hacer salvadas totales (salvas completas hacia toda la información de la base de datos) con la siguiente frecuencia:

- Realizar salvadas diarias.
- El borrado de las salvadas se hará semanalmente.
- El almacenamiento de las salvadas de la base de datos se realizará en el caso del entorno DIPP en la máquina del informático encargado del sistema. Y en el caso del entorno Pozo se almacenará en el servidor destinado para el sistema. El nombre para el guardado de la salva será la siguiente manera: `sipp_v2.0_<<fecha: formato día_mes_año>>_<<hora: formato 24 horas>>`
- En caso de momentos críticos de la base de datos ya sean de entrada de datos continuos en el sistema o de prueba por los desarrolladores se realizaría una salva antes y después de ese momento.

3.6. Análisis de la Replicación de datos

Como se analizó en el capítulo anterior la réplica de datos es un proceso que permite copiar y distribuir idénticamente tablas desde una base de datos hacia una o múltiples sitios de una red. Este proceso asegura que los datos estén disponibles en el lugar necesario para ser utilizado en el momento indicado; algo necesario en el sistema SIPP para lograr la disponibilidad de la información en tiempo casi real en la DIPP sobre los pozos en perforación en campo.

El entorno de réplica puesto en práctica para la base de datos diseñada es Maestro-Maestro, porque permite sincronizar los cambios entre todas las partes que intervienen en la réplica. Este entorno tiene altos beneficios para el sistema ya que permite mantener sitios recuperables ante posibles desastres o caídas, así como proveer sistemas con alta disponibilidad y para balancear la carga de consulta a través de las distintas ubicaciones.



Después de realizar la configuración de la herramienta SymmetricDS para las tablas del modelo propuesto se simuló un ambiente de replicación servidor_dipp-servidor_pozo y se obtuvieron resultados satisfactorios. La replicación de datos es continua aún sin conexión la base de datos guarda los eventos debido a la propiedad de la herramienta de emplear la técnica de replicación basada en disparadores. La función del disparador es ejecutar una acción cuando ocurre un evento en la base de datos, los eventos pueden ser inserción, actualización o eliminación.

Se incluyen reglas de replicación y establecen prioridades para no dar paso a conflictos de actualización estableciendo canales de replicación para los distintos esquemas del modelo. Se logra la integridad y disponibilidad de la información en ambos servidores.

Por todo lo antes planteado se puede llegar a la conclusión que el sistema de replica propuesto para la base de datos es ampliamente factible para ponerlo en práctica en el sistema SIPP. Para la configuración y aspectos a tener en cuenta se anexa a esta investigación el manual de configuración de la herramienta confeccionado específicamente para la base de datos de SIPP.

3.7. Análisis de Prueba de volumen.

Con el objetivo de obtener un llenado voluminoso de la base de datos, y realizarle pruebas de la misma sin violar la integridad de los datos, se generó un determinado volumen de información en las tablas, de tal forma se simula el flujo de información en la base de datos. Es importante destacar que para las tablas que incluyen llaves foráneas el software hace una búsqueda en el catálogo de llaves para realizar las validaciones.

El software utilizado para la prueba fue el EMS Data Generator para PostgreSQL, esta selección estuvo respaldada al hecho de que es un poderoso utilitario para generar datos de prueba a tablas de bases de datos PostgreSQL de una vez. La elección estuvo dada



al hecho de que no existen herramientas similares libres de fácil acceso que tenga tanta flexibilidad en su uso. Las alternativas libres no están lo suficientemente descritas en la red, y las alternativas propietarias que existen no contiene el mismo grado de libertad de uso o presentan limitaciones para el llenado integro de la base de datos o el volumen de datos que puede generarse de una vez.

Las tablas fueron llenadas con un rango de datos equivalentes al que tendrá la base de datos después de la culminación del pozo de petróleo más largo que se ha perforado en el país.

El uso de esta herramienta posibilitó chequear la integridad de la base de datos. En la siguiente tabla se muestra los resultados alcanzados por la prueba de llenado voluminoso de la base de datos.

Resultados de las Pruebas de Volumen.		
Registros insertados por tablas.	Tiempo de ejecución del software.	Registro total en la base de datos.
2000	5 minutos	210000
1000	2.31 minutos	105000
100	0.19 minutos	10500

En la tabla se muestran tres momentos de carga de datos, identificándose el número de registros insertados por tabla, el tiempo de ejecución que tardo del software para el llenado de datos y cantidad de registros totales en la base de datos después de terminada cada carga. En los tres momentos se completó la prueba de carga intensiva y la escalabilidad de la base de datos.



Es necesario tener presente, que los resultados obtenidos al realizar las pruebas de carga intensiva a la base de datos solo muestran un estimado en realidad, y que estas no pueden ser tomadas como reales o absolutas. Existen numerosos factores que influyen en la rapidez y estabilidad de la base de datos, como por ejemplo: la cantidad de usuarios conectados simultáneamente y el grado de concurrencia de las solicitudes hechas por los mismos, las cuales hay que tener en cuenta siempre, por lo que la mejor prueba que se puede realizar a cualquier sistema es la interrelación directa con el usuario.

3.8. Trazabilidad de las acciones.

La trazabilidad de las acciones es la habilidad de trazar o dejar huellas de los movimientos y procesos por los que pasa un determinado producto, la capacidad que tiene una organización o sistema para rastrear, reconstruir o establecer relaciones entre objetos monitoreados, para identificar y analizar situaciones específicas o generales en los mismos[49].

En el sistema SIPP es importante mantener el control sobre todas las acciones que los usuarios realizan sobre la base de datos, por ejemplo, si un usuario actualiza un registro, debe quedar constancia de que acción ejecutó, la fecha, la hora y el área donde realizó el cambio. Para así de esta forma lograr el seguimiento de todas las acciones ejecutadas, conformando el historial de estas.

Con el objetivo del seguimiento de las trazas en la base de datos se crea la tabla llamada logs, en la cual se almacena la hora, fecha y operación así como el pozo asociado al usuario. Teniendo así archivada las trazas en esta tabla.



3.9. Análisis de los resultados obtenidos.

Luego de haber confeccionado el modelo lógico y físico de la base de datos para el sistema SIPP versión 2.0 y realizar las pruebas de validación se puede decir que se cumplió el objetivo propuesto dándole respuesta al problema planteado. La base de datos obtenida contiene una lata estructuración de los datos y estándares de nomenclatura que posibilitan una organización y uniformidad propia del modelo, algo con lo que no contaba el modelo de base de datos de la anterior versión.

Se logra un sistema de replicación asimétrico basado en disparadores que logra la disponibilidad de información en ambas partes involucradas en el negocio. El modelo propuesto en la versión 1.0 contó con un proceso insuficiente de normalización algo que en el diseño de propuesta actual está completamente solucionando llevando el modelo hasta una 3FN evitando las anomalías de actualización y una mayor independencia de los datos. El modelo obtenido contiene 110 tablas dividido en 6 esquemas logrando obtener una correspondencia entre los distintos niveles de la arquitectura.



3.10. Conclusiones Parciales.

En este capítulo se realizó la validación teórica y funcional de la base de datos del Sistema para el Manejo Integral de la Perforación de pozos de Petróleo y Gas (SIPP) versión 2.0. En un primer momento fueron definidas un conjunto de restricciones de integridad para garantizar la validez de los datos introducidos o modificados en la misma. Además fueron aplicadas las tres primeras formas normales con el objetivo de eliminar problemas generados en el diseño inicial, teniendo en cuenta aspectos para asegurar la seguridad de la base de datos en el sistema. En un segundo momento se realizó pruebas a la base de datos con la herramienta EMS Datos Generator, para el llenado voluminoso de la información y SymmetricDS para el sistema de replicación de datos; a través de ambas pruebas se confirmó la integridad, seguridad y adecuado funcionamiento de la base de datos.



Conclusiones Generales.

Luego de haber concluido el diseño e implementación de la base de datos versión 2.0 para el sistema, se puede decir que se obtuvieron los resultados esperados.

- Se obtuvo un diagrama de clases persistentes donde las relaciones entre las tablas fueron construidas adecuadamente.
- Se logró un diseño del modelo lógico de la base de datos normalizado hasta la 3FN y minimización de la información redundante en el mismo.
- Se implementó el modelo físico de la base de datos con restricciones de integridad que garantizan que los datos involucrados o modificados sean los correctos.
- Se logra un sistema de replicación para lograr la disponibilidad e integridad de los datos en las partes involucradas en el negocio.

Logrando darle solución al problema a resolver con el cumplimiento del objetivo propuesto.

**Bibliografía.**

1. Rojo, Á.S. *UNIVERSIA. Bases de datos de información y documentación en Internet*. 2009 [citado 2010 25/12/2010]; Disponible en: <http://biblioteca.universia.net/ficha.do?id=38515924>.
2. García, R.M.M., *Sistemas de Bases de Datos*. Segunda Edición ed, ed. J.Q. Pantoja. 2005, Ciudad de la Habana: Editorial Pueblo y Educación.
3. Española, R.A.d.I.L. (2005) *Diccionario de la lengua española*. WordReferencie.com
4. Venezuela, G.B.d. *La industria petrolera en Venezuela*. 2011 [citado 2011 15/01/2011]; Disponible en: http://www.pac.com.ve/index.php?option=com_content&view=article&catid=64&Itemid=87&id=4524.
5. González, M.H.Á., *Diseño de la Base de Datos del Sistema de Información de Perforación de Pozos (SIPP)*. 2009, Universidad de las Ciencias Informáticas (UCI). Ciudad de la Habana.
6. Geológico, M. *Geología del Petróleo*. 2010 [citado 2011 16/01/2011]; Disponible en: <http://www.mundogeologico.com.ve/geologia%20del%20petroleo.html>.
7. Garza, F.R.d.I., *Simulación de Yacimientos Petroleros*. 2005.
8. Ramírez., R.B. *Ingeniería de Yacimientos*. 2010 [citado 2011 20/01/2011]; Disponible en: <http://molten.latinclicks.info/clasificayac/>.
9. SIPP, *Glosario de Términos*. 2009: Ciudad de la Habana.
10. Dambrosio (2010) *El concepto de datos*.
11. *Definición de datos*. 2010 [citado 2011 28/01/2011]; Disponible en: <http://www.definiciones.com.mx/definicion/D/dato/>.
12. Molina, M.P. (2004) *Base de datos*.
13. *Preguntas frecuentes de la web*. 2011 [citado 2011 28/02/2011]; Disponible en: <http://www.masadelante.com/faqs/base-de-datos>.
14. Valdés, D.P. (2010) *¿Qué son las Bases de Datos?*
15. Collector, G. *Sistema Gestor de base de datos SGBD*. 2004 [citado 2011 01/02/2011]; Disponible en: http://www.error500.net/garbagecollector/archives/categorias/bases_de_datos/sistema_gestor_de_base_de_datos_sqbd.php.

16. *Introducción a las Bases de Datos*. 2010 23/01/2011 [citado: 2011; Disponible en: <http://pdf.rincondelvago.com/bases-de-datos.html>].
17. Hernández, G. *Petróleo.Perforación*. 2003 [citado 2011 26/03/2011]; Disponible en: <http://www.gustato.com/petroleo/Petroleo2.html>.
18. Kraus, R.S. (2006) *Petróleo: Prospección y Perforación*.
19. Solán, O.G. (2005) *Los sistemas de control de gestión estratégica de las organizaciones*.
20. Rodríguez, D.S.D., *Sistema para el Control de los Recursos del Proyecto*. 2008, Universidad de las Ciencias Informáticas: Ciudad de la Habana.
21. Acosta, M.d.J.V. (2008) *Bases de Datos. Instrucciones para operación académica*.
22. Andrés, M.M.M., ed. *Arquitectura de Base de Datos*. 2001: España.
23. García, U. *Definición Modelo de Datos*. 2008 [citado 2011 02/02/2011]; Disponible en: <http://www.mitecnologico.com/Main/DefinicionModeloDeDatos>.
24. Ortiz, A.M. (2000) *Base de Datos. Modelo de Datos*.
25. Sánchez, J., *Diseño Conceptual de Bases de Datos*. 2004.
26. Morales., S.L.R.M.c., *Sistema Gestor de Base de Datos Relacionales*. 2010.
27. Quiroz, J. (2003) *El modelo relacional de las bases de datos*. Boletín de Política Informática Núm 6,2003
28. *Modelo Relacional*. 2004 [citado 2010 02/02/2011]; Disponible en: <http://www.mitecnologico.com/Main/ModeloRelacional>.
29. Perón, Y.Y., *Diseño de la Base de Datos para el Sistema de Captura y Catalogación de Medias*. 2009, Universidad de las Ciencias Informáticas: Ciudad de la Habana. p. 10.
30. CAVSI (2010) *¿Qué es un Sistema Gestor de Bases de Datos o SGBD?*
31. Martínez, R. *PostgreSQL -es*. 2009-2011 [citado 2011 26/03/2011]; Disponible en: http://www.postgresql.org.es/sobre_postgresql.
32. Anderson, J., *CakePHP A Rapid Development Framework*, L.E. Masters, Editor. 2006.
33. Quiñones, E. (2008) *Introducción a PostgreSQL*. Asociación Peruana de Software Libre.
34. Padrón, Y.F., *Plataforma Video Web.*, in *Departamento de Señales Digitales*. 2009, Universidad de las Ciencias Informáticas (UCI). Ciudad de la Habana. p. 26-27.
35. EIR, C.y.S. (2009) *PostgreSQL*.
36. Anónimo. *Acerca de la guía de Ubuntu. pgAdmin III*. 2008 [citado 2011 27/03/2011]; Disponible en: http://www.guia-ubuntu.org/index.php?title=PgAdmin_III.
37. WIN, P. *EMS Data Generator 2005 for PostgreSQL*. 2002 [citado 2011 27/03/2011]; Disponible en:

http://pcwin.com/Business_Finance/EMS_Data_Generator_2005_for_PostgreSQL/download.htm.

38. Morel, D.E.C. (2008) *Replicación de Datos en SQL Server*.
39. DigiSoft. *Replicación de Datos 2009* [citado 2011 04/04/2011]; Disponible en:
http://www.digisoftve.com/es/index.php?option=com_content&view=article&id=9&Itemid=12.
40. Salvador, U.d. *Replicación modelado y minería de datos.Replicación en PostgreSQL Slony-I*. 2011 [citado 2011 04/04/2011]; Disponible en:
<http://basesdedatosues.blogspot.com/2010/06/replicacion-postgresql-slony-i.html>.
41. Reingart, M. *PyReplica.Descripción General*. 2009 [citado 2011 04/04/2011]; Disponible en: <http://www.sistemasagiles.com.ar/trac/wiki/PyReplicaEs>.
42. Reingart, M. (2010) *PyReplica.Sistema de replicación simple para PostgreSQL programado en Python*.
43. Henson, E.L.-C. (2008) *SymmetricDS User guide*.
44. Acces, M.O. *Conceptos básicos del diseño de base de datos*. 2011 [citado 2011 29/03/2011]; Disponible en: <http://office.microsoft.com/es-mx/access-help/conceptos-basicos-del-diseno-de-una-base-de-datos-HA001224247.aspx>.
45. Wordpress. *Definición.de*. 2008 [citado 2011 30/04/2011]; Disponible en: <http://definicion.de/integridad/>.
46. Savedra, J. (2009) *Integridad Informática enfocada a las bases de datos*.
47. Server, C.d.D.d.S. *Integridad de los datos*. 2009 [citado 2011 30/04/2011]; Disponible en: <http://msdn.microsoft.com/es-es/library/ms184276.aspx>.
48. MySQL, C.d. (2003) *Normalización de bases de datos*.
49. Macías, D.F., *Diseño de una base de datos para controlar la información de Unión de Jóvenes Comunistas en la UCI*. 2008, Universidad de las Ciencias Informáticas (UCI): Ciudad de la Habana. p. 67.