Universidad de las Ciencias Informáticas

Facultad 6

Título: “Diseño e Implementación de un almacén de datos para la red nacional de Genética Médica”.

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

Autor(es): Yaneisy Contreras Martinez

Alicia Guilarte González

Tutor(es): Ing. Yadira Robles Aranda

Ing. Haymeé Llerena Esperon

Ing. Alfonso Claro Arceo

Lic. Yanelis Benítez Fernández

Ciudad de la Habana

Junio 2010

Todos y cada uno de nosotros paga puntualmente su cuota de sacrificio consciente de recibir el premio en la satisfacción del deber cumplido, conscientes de avanzar con todos hacia el Hombre Nuevo que se vislumbra en el horizonte.

Ernesto Che Guevara

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Yaneisy Contreras Martinez

_____
Firma del Autor

Alicia Guilarte González

_____
Firma del Autor

Yadira Robles Aranda

_____
Firma del Tutor

Haymeé Llerena Esperon

_____
Firma del Tutor

Yanelis Benítez Fernández

_____
Firma del Tutor

Alfonso Claro Arceo

_____
Firma del Tutor

DATOS DE CONTACTO

Tutores:

Ing. Yadira Robles Aranda
Correo electrónico: yrobles@uci.cu
Universidad de las Ciencias Informáticas, Ciudad de La Habana, Cuba.

Ing. Haymeé Llerena Esperón
Correo electrónico: hllerena@uci.cu
Universidad de las Ciencias Informáticas, Ciudad de La Habana, Cuba.

Ing. Alfonso Claro Arceo
Correo electrónico: aclaro@uci.cu
Universidad de las Ciencias Informáticas, Ciudad de La Habana, Cuba.

Lic. Yanelis Benítez Fernández
Correo electrónico: jhanelis@uci.cu
Universidad de las Ciencias Informáticas, Ciudad de La Habana, Cuba.

AGRADECIMIENTOS

Agradezco a mi madre por ser la persona con la que siempre he podido contar, por todo su amor y dedicación, a mi padre por ser el ejemplo a seguir para toda persona que quiera ser alguien importante en la vida, por todo su apoyo y confianza.
A mi hermano Rafelito por ser más que un hermano, por ser como un padre para mí y por cuidar a mi mamita todos estos años que he estado lejos.
A toda mi familia por su preocupación y cariño.
A mis tutores por brindarme su ayuda cuando los necesite.
A las personas de DATEC especialmente a Lázaro, que sin él no se hubiese terminado toda la investigación a tiempo.
A Yaneisy y Sulay por haber estado a mi lado en todo momento, por su apoyo y sus consejos de amigas.
A todos mis amigos que han estado junto a mí en estos 5 años, dándome su apoyo incondicional, ustedes son más que mis amigos, son mis hermanos y aunque pase el tiempo y la distancia nos separe nunca voy a olvidar todos los buenos y malos momentos que viví junto a ustedes.
A mi novio por todo su amor y comprensión.
A todos aquellos que de una forma u otra han hecho posible la realización de este trabajo.

Alicia

A mi abuelo que ha sabido confiar en mí, que me ha educado en todo lo correcto ensañándome a esforzarme y no amedrentarme ante las dificultades. Por haber dado con amor todo cuanto pudo para que lograra este éxito que hoy es de nosotros.
A mi padre que siempre me ha apoyado en todas mis decisiones, por haberme brindado sus consejos y ayuda siempre, por saberme enseñar el camino a seguir en la vida y ser mi inspiración eterna.
A mi hijo que aunque aún no ha nacido me ha dado fuerzas para seguir adelante.
A mi hermano, tíos, primos y esposo por su dedicación, confianza y amor; por convertirme en un ser cada día mejor, por alentarme, darme seguridad ante todos los acontecimientos de mi vida y estar a mi lado en cada momento.
A mis compañeros de grupo que han compartido conmigo las distintas etapas de la carrera y en ese transitar, hemos aprendido a querernos como verdaderos amigos.
A todos los profesores por haberme dotado de los conocimientos necesarios para mi formación como profesional y especialmente a mis tutores.
A Alicia, Sulay, Dayneris y Livan por haber sido mi brazo derecho durante los 5 años de la carrera, por su dedicación y confianza, por haberme enseñado los valores que encierra el concepto de amistad.
Y a todas las personas que de una forma u otra han propiciado el resultado de este trabajo.

Yaneisy

DEDICATORIA

A mis padres y mi hermano por ser las personas más importante de mi vida, a ellos les debo todo lo que soy, gracias por su amor y su confianza.

Alicia

A mi abuela Josefa y mi mamá, por ser mis guías y ejemplo y más que eso, por haberme dado la oportunidad de existir.

Yaneisy

RESUMEN

La informatización de los procesos está presente en la mayoría de las empresas permitiendo el manejo de los datos en forma centralizada. En un mundo cada vez más acelerado y competitivo, una estructura adecuada de almacenamiento que permita obtener información operacional, es una necesidad esencial en el proceso decisivo de los negocios.

El presente trabajo de diploma permitirá el desarrollo de un datawarehouse para proporcionar el acceso a la información actual y los datos históricos obtenidos desde una base de datos, permitiendo su análisis desde diversas perspectivas y con grandes velocidades de respuesta. Surge por la necesidad que presenta la red nacional de Genética Médica de poder almacenar datos de manera eficiente para su explotación y análisis, permitiendo obtener los resultados esperados para el proceso de toma de decisiones. Para su desarrollo se utilizan herramientas libres y su construcción está basada en la metodología Hefesto, la estructura lógica propuesta, el diseño y la implementación son consecuentes con ésta. Se realizan pruebas de rendimiento para determinar la velocidad de respuesta al ejercer consultas y obtener resultados.

Este datawarehouse brinda grandes beneficios al sistema de salud nacional, sobre todo a los genetistas encargados de interactuar con éste, permitiéndoles gestionar datos guardados en diversos formatos, fuentes y tipos, para luego depurarlos e integrarlos, además de almacenarlos en un solo destino.

PALABRAS CLAVES: base de datos, datawarehouse, hefesto, información.

TABLA DE CONTENIDO

INTRODUCCIÓN

Las tecnologías en el mundo de hoy presentan un gran auge dado el desarrollo científico técnico alcanzado, es por ello que la informatización de la sociedad ha crecido a nivel mundial y esto trae consigo que aumente la capacidad de generación y almacenamiento de la información. Por lo antes señalado las empresas y organizaciones mundiales se han visto en la necesidad de encontrar una nueva solución que almacene esta gran cantidad de datos y a su vez, poder extraer información realmente útil para las mismas.

En estos momentos los datawarehouse son el centro de atención de las grandes empresas, ya que constituyen uno de los soportes fundamentales para el proceso de toma de decisiones gerenciales, de ahí la importancia de que los datos guardados en ellos sean confiables y con calidad, estos almacenan información categorizándola o estructurándola de forma que favorezcan el análisis de los datos y puedan proporcionar análisis históricos.

Cuba está poniendo todo su empeño, -a pesar de su estado de país subdesarrollado-, en estar en contacto directo con el desarrollo de los nuevos avances tecnológicos y los diferentes procesos por los que ha transitado la informática; es por eso que prepara a su población en el estudio de esta ciencia, al mismo tiempo que trabaja en la utilización de estos medios para los principales sectores de la economía del país, sobre todo en la salud.

En las últimas décadas han surgido grandes avances en la genética y la tecnología biomédica, que abren nuevos y vastos espacios a la investigación y brindan novedosas herramientas en esta rama. Un resultado relevante ha sido la creación del Centro Nacional de Genética Médica (CNGM), institución que se ha expandido formando la red nacional de Genética Médica la cual está compuesta por 184 centros ubicados en todos los municipios del país, que coordinados por el CNGM, conducen el programa nacional para el diagnóstico, manejo y prevención de enfermedades genéticas y defectos congénitos [1]. Esta red de instituciones desarrolla acciones asistenciales, docentes y de investigación en el campo de los problemas de salud de carácter genético en la población cubana. Su objetivo es desarrollar proyectos de investigación e innovación, en el campo de la genética médica, inmunología y disciplinas afines.

El bloqueo económico, comercial y financiero impuesto por el gobierno de los Estados Unidos, priva a Cuba del acceso a la tecnología más avanzada en tan promisorio campo, lo cual limita de manera considerable la labor investigativa del CNGM. A pesar de eso, el altísimo nivel profesional de los especialistas cubanos y un sistema de salud en función del bienestar del pueblo, ponen al país en condiciones de realizar estudios genéticos propios de naciones del primer mundo [2].

La Universidad de las Ciencias Informáticas (UCI) es una de las instituciones que desde su creación ha desempeñado un papel importante en la batalla de informatizar todos los campos de la economía del país. La facultad 6 conjuntamente con el CNGM cuenta con la aplicación alasMEDIGEN que está compuesta por nueve módulos, los cuales son: Registro Cubano de Historias Clínicas (RECUHCL), Registro Cubano de Gemelos (RECUGEM), Registro Cubano de Discapacitados (RECUDIS), Registro Cubano de Retrasos Mentales (RECURM), Registro Cubano de Malformaciones Congénitas (RECUMAC), Registro Cubano de Enfermedades Genéticas (RECUEGEN), Registro Cubano de Enfermedades Comunes, Registro Cubano de Anomalías Cromosómicas y Teleconsulta Genética.

El desarrollo de esta aplicación ha permitido desde su inicio, almacenar electrónicamente datos, tanto internos como externos de defectos, diagnóstico y tratamiento de los trastornos hereditarios. Este control que se tiene sobre los pacientes se extiende a todos los rincones del país, por tanto, al transcurrir los años, los datos almacenados aumentan considerablemente, por lo que los genetistas encargados de interactuar con estos sistemas de información se enfrentan de forma más agudizada, a problemas relacionados con el considerable tiempo que tienen que dedicar a la obtención de información. A esto se le suma la dificultad o imposibilidad de expresar una consulta compleja y la obtención de resultados de forma eficiente, además no se pueden obtener datos históricos, o sea, solo los valores actuales de los datos y no de la última vez que se modificó, ni de las transformaciones hechas en años anteriores.

La red nacional de Genética Médica necesita una aplicación que esté preparado óptimamente con herramientas adecuadas para la explotación y análisis de los datos, que permita obtener el conocimiento necesario y las facilidades de apoyo en el proceso de toma de decisiones.

Para dar solución a la problemática anteriormente planteada, se definió el modelo conceptual de un almacén de datos para la red nacional de Genética Médica, atendiendo a las necesidades de este centro,

se va a realizar el diseño de una estructura lógica y su implementación tomando como base este modelo conceptual.

A partir de la situación descrita anteriormente se tiene como **problema a resolver** ¿Cómo obtener e implementar una estructura lógica a partir de un modelo conceptual propuesto para generar conocimiento?

La presente investigación tiene como **objeto de estudio**: el proceso de desarrollo de los datawarehouse.

A partir del objeto de estudio se determinó el **campo de acción**: el proceso de diseño e implementación de los datawarehouse.

Por lo antes planteado, se definió como **objetivo de la investigación**: realizar el diseño y la implementación de un datawarehouse para la red nacional de Genética Médica que facilite el apoyo a la toma de decisiones. A partir de aquí se derivan los siguientes **objetivos específicos**:

- Elaborar el modelo lógico de la estructura del datawarehouse para la red nacional de Genética Médica.
- Implementar el modelo multidimensional del datawarehouse.
- Validar la solución desarrollada mediante la realización de pruebas de volumen y carga.

Para desarrollar el trabajo de diploma, se proponen las siguientes **tareas de la investigación**:

- Estudio de las tendencias de tecnologías y herramientas para el desarrollo de los datawarehouse.
- Diseño de tablas de dimensiones.
- Diseño de tablas de hechos.
- Realización de uniones.
- Determinación de jerarquías.
- Implementación del modelo multidimensional utilizando el gestor de base de datos PostgreSQL.
- Validación de la solución desarrollada mediante la realización de pruebas de volumen y carga.

El presente trabajo está estructurado por **tres capítulos**, distribuidos de la siguiente manera:

Capítulo 1: Fundamentación teórica

En este capítulo se realizará un estudio del estado del arte, conceptos, tecnologías y metodologías que son utilizadas para el desarrollo de los datawarehouse, así como sus características, arquitectura, herramientas, ventajas y desventajas.

Capítulo 2: Descripción de la Solución

En este capítulo se mostrarán las estructuras multidimensionales modeladas para el datawarehouse, basándose en el análisis realizado anteriormente según los describe la metodología utilizada. Con la aplicación de esta metodología se lleva a cabo la construcción de un modelo lógico y su implementación a partir del modelo conceptual propuesto.

Capítulo 3: Análisis de la solución propuesta

En este capítulo se hace énfasis en aspectos tales como la normalización, las pruebas de volumen y carga, análisis de los tiempos de respuesta, así como en hacer pruebas para valorar el rendimiento con la concurrencia de los usuarios.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

En el presente capítulo se realizará un estudio del estado del arte, conceptos y tecnologías de los datawarehouse. Se tomará como base la metodología propuesta por una investigación previa que ha sido realizada seleccionando la más apropiada para el desarrollo del datawarehouse. Se exponen sus características, arquitectura, estructura, herramientas, ventajas y desventajas.

1.1 Datawarehouse

Las mejores decisiones de negocio son la clave para el éxito en el mercado competitivo de hoy. Las organizaciones buscan que sus tomas de decisiones sean entendibles por el volumen y complejidad de los datos disponibles. Facilitar estos datos a una amplia audiencia de usuarios, es uno de los cambios más significativos para los profesionales de la información.

Generalmente, los sistemas transaccionales u OLTP usan estructuras normalizadas, en las cuales se optimizan las inserciones y actualizaciones de artículos e incluso algunas selecciones, pero es menos probable que el sistema se organice de forma tal que produzca reportes eficientes para datos resumidos con cierta jerarquía. Una solución al problema sería los almacenes de datos conocidos también como datawarehouse, estos usan los datos relevantes de fuentes existentes y los integran en una estructura que ha sido optimizada para las selecciones.

Un datawarehouse se caracteriza por integrar y depurar información de una o más fuentes para luego procesarla, permitiendo su análisis desde diversas perspectivas y con grandes velocidades de respuesta. La creación de un datawarehouse representa en la mayoría de las ocasiones el primer paso desde el punto de vista técnico, para implantar una solución completa y fiable de Inteligencia de Negocio.

Existen diferentes definiciones sobre datawarehouse, la más conocida fue propuesta por Inmon uno de los primeros autores en escribir sobre el tema de los datawarehouse en 1992: Un datawarehouse es una colección de datos orientados a temas, variante en el tiempo, integrados y no-volátiles, organizados para soportar necesidades empresariales [3].

Un datawarehouse para que tenga éxito necesita un conjunto de objetivos que incluya los factores críticos de la empresa, estos son:

- Soporte en la toma de decisiones estratégicas: El datawarehouse proporciona datos detallados y acumulados que pueden ser utilizados para el análisis de tendencias, comparación del rendimiento y análisis estadístico.
- Rapidez de respuesta a las consultas: Mediante OLAP se dispone de rápidas respuestas a las consultas. Los tiempos de respuesta pueden pasar de días a minutos.
- Calidad de los datos: La calidad de los datos puede ser mejorada cuando la fuente de estos es analizada desde el datawarehouse.
- Hacer accesible la información de la organización: La información contenida en el datawarehouse debe ser navegable, fácilmente comprendida por los usuarios, y sobre todo de acceso rápido.
- Hacer que la información de la organización sea consistente: La información de un departamento de la organización puede ser contrastada con la información de otro departamento. Si dos mediciones tienen el mismo nombre, entonces significan lo mismo, por el contrario, si dos mediciones representan conceptos diferentes, deben llamarse de distinto modo de manera que toda la información sea correcta y esté al día.

Después de haber conocido más acerca de los datawarehouse y sus objetivos es que se entiende la razón por la cual la mayoría de las organizaciones lo utilizan, sin embargo existen algunas que aún almacenan sus datos en sistemas operacionales y una de las razones es el desconocimiento de las diferencias entre ellos.

1.2 Sistemas Operacionales vs Datawarehouse

La información almacenada en las bases de datos se orientó desde un primer momento a sistemas de procesamiento transaccional en línea OLTP (On Line Transaction Processing) de un modo tal que los procesos se diseñaron fundamentalmente para introducir información en los sistemas, pero no para extraerla de ellos. A medida que ha ido creciendo el volumen de información almacenada, ha crecido también la dificultad de acceder a ella de un modo sencillo y eficiente.

Los datawarehouse están orientados a procesos de consultas en contraposición con los procesos transaccionales, sus tablas pueden no estar normalizadas y se admite redundancia en los datos. Un datawarehouse no se encuentra en la tercera forma normal, lo que le permite mayor rapidez a la hora de seleccionar los datos, en contraposición con un OLTP que es la mejor opción para insertar, actualizar y

eliminar. El OLTP, normalmente, está formado por un número mayor de tablas, cada una con pocas columnas, mientras que en un datawarehouse el número de tablas es menor, pero cada una de éstas tiende a ser mayor en número de columnas. Los OLTP son continuamente actualizados por otros sistemas del día a día, mientras que los datawarehouse son actualizados en *batch* de manera periódica. Un datawarehouse, normalmente, almacena muchos meses o años de datos para análisis históricos de la información, un OLTP, normalmente, almacena algunas semanas.

Las aplicaciones de OLTP están organizadas para ejecutar las transacciones para los cuales fueron hechos, como por ejemplo: mover dinero entre cuentas, una devolución de inventario. Por otro lado, un datawarehouse está organizado en base a conceptos, como por ejemplo: clientes, facturas, productos.

Otra diferencia radica en el número de usuarios. Normalmente, el número de usuarios de un datawarehouse es menor al de un OLTP. Es común encontrar que los sistemas transaccionales son accedidos por cientos de usuarios simultáneamente, mientras que los datawarehouse sólo por decenas.

1.3 Características de los Datawarehouse

Para el desarrollo de un datawarehouse es necesario tener en cuenta las características que lo identifican, estas son:

Orientado a temas: Una primera característica del datawarehouse es que la información se clasifica en base a los aspectos que son de interés para la empresa. Siendo así, datos tomados en contraste con los clásicos procesos orientados a las aplicaciones [4].

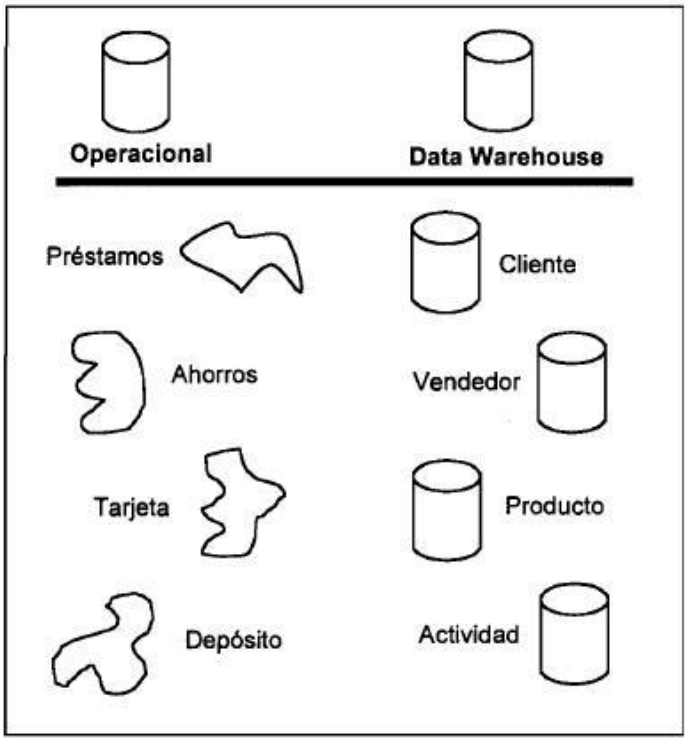


Figura 1. Los datawarehouse están orientando a temas

El ambiente datawarehouse se organiza alrededor de sujetos tales como clientes, vendedores, productos y actividades. Las aplicaciones están relacionadas con el diseño de la base de datos y del proceso. La diferencia entre la orientación de procesos, funciones de las aplicaciones y orientación a temas, radica en el contenido de los datos a nivel detallado. En el datawarehouse se excluye la información que no es usada por el proceso de sistemas de soporte de decisiones, mientras que la información orientada a las aplicaciones, contiene datos para satisfacer de inmediato los requerimientos funcionales y de proceso, que pueden ser usados o no por el analista de soporte de decisiones.

En resumen, orientado a temas, significa que está organizado con relación a las principales materias de la empresa. Los datos se organizan por temas para facilitar su acceso y entendimiento por parte de los usuarios finales. Por ejemplo, todos los datos personales sobre los pacientes pueden ser consolidados en una única tabla del datawarehouse. De esta forma, las peticiones de información sobre pacientes serán más fáciles de responder dado que toda la información reside en el mismo lugar.

Integrado: El aspecto más importante del ambiente del datawarehouse es que la información encontrada en el interior está integrada. Esta integración de datos se muestra de muchas maneras: en convenciones de nombres consistes, en la medida uniforme de variables, en la codificación de estructuras, en atributos físicos de los datos, fuentes múltiples y otros. En la figura 2, se muestra el contraste de la integración encontrada en el datawarehouse con la carencia de integración del ambiente de aplicaciones y sus respectivas diferencias [4].

- Codificación: los diseñadores de aplicaciones codifican el campo GÉNERO en varias formas. Estos, representan GÉNERO como una “M” y una “F”, otros como un “1” y un “0”, otros como una “X” y una “Y” e inclusive, como “masculino” y “femenino”. No importa como el GÉNERO llega al datawarehouse, probablemente “M” y “F” son tan buenas como cualquier otra representación. Lo importante es que el GÉNERO debe llegar al datawarehouse en un estado integrado y uniforme. Por lo tanto, cuando el GÉNERO se carga en el datawarehouse desde una aplicación, donde ha sido representado en formato “M” y “F”, los datos deben convertirse al formato del datawarehouse que no precisamente debe ser el mismo.
- Medida de los atributos: los diseñadores de aplicaciones obtienen las unidades de medida en una variedad de formas. Un diseñador almacena los datos en centímetros, otros en pulgadas, otros en millones de pies cúbicos por segundo y otros en yardas. Al dar medidas a los atributos, la transformación traduce las diversas unidades usadas en las diferentes bases de datos para transformarlas en un estándar común. Cualquiera que sea la fuente, cuando la información llegue al datawarehouse necesita ser medida de la misma manera.
- Convenciones de nombramiento: el mismo elemento es frecuente referido por nombres diferentes en las diversas aplicaciones. El proceso de transformación asegura que se utilice preferentemente el nombre del usuario.
- Fuentes múltiples: el mismo elemento puede derivarse desde fuentes múltiples. En este caso, el proceso de transformación debe asegurar que la fuente apropiada sea usada, documentada y movida al depósito.

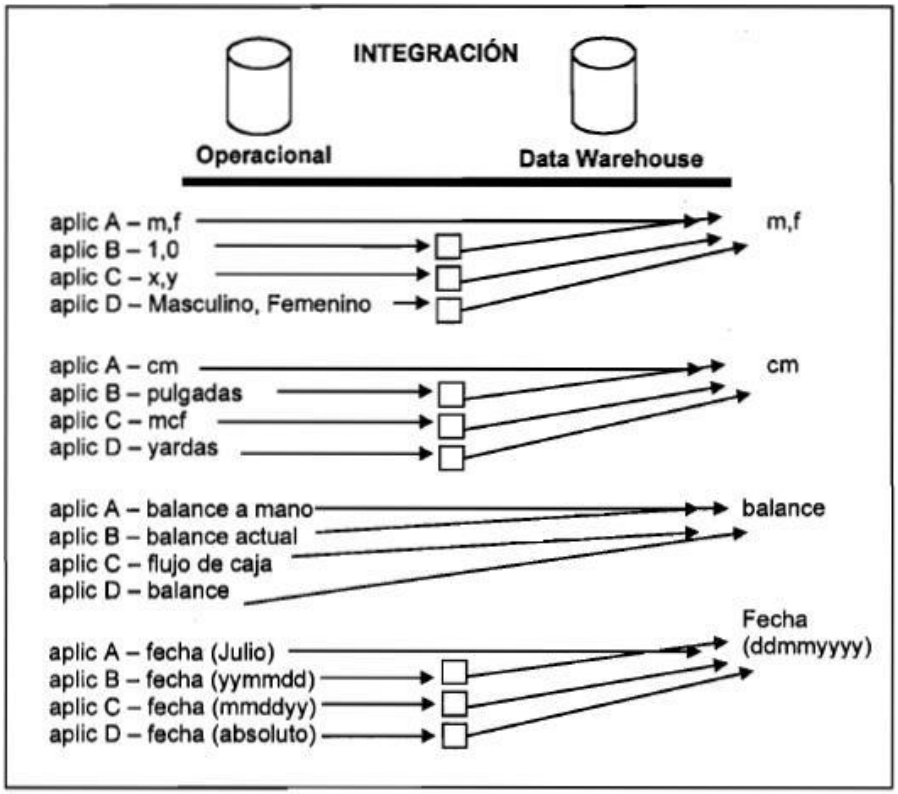


Figura 2. Los datawarehouse son integrados

Tal como se muestra en la figura 2, los puntos de integración afectan casi todos los aspectos del diseño como son: las características físicas de los datos, la incompatibilidad de tener más de una fuente de datos, el problema de estándares de denominación inconsistentes, pero cualquiera que sea el diseño el resultado es el mismo, la información necesita ser almacenada en el datawarehouse en un modelo globalmente aceptable y singular, aun cuando los sistemas operacionales almacenan los datos de manera diferente.

Histórico: Esta característica básica de los datos en un depósito, es muy diferente de la información encontrada en el ambiente operacional. En estos, la información se requiere en el momento de acceder. En otras palabras, en el ambiente operacional, se accede a una unidad de información y se espera que los valores requeridos se obtengan a partir del momento de acceso.

Como la información en el datawarehouse es solicitada en cualquier momento, los datos encontrados en el depósito se llaman de “tiempo variante”. Los datos históricos son de poco uso en el procesamiento

operacional. La información del depósito, debe incluir los datos históricos para usarse en la identificación y evaluación de tendencias.

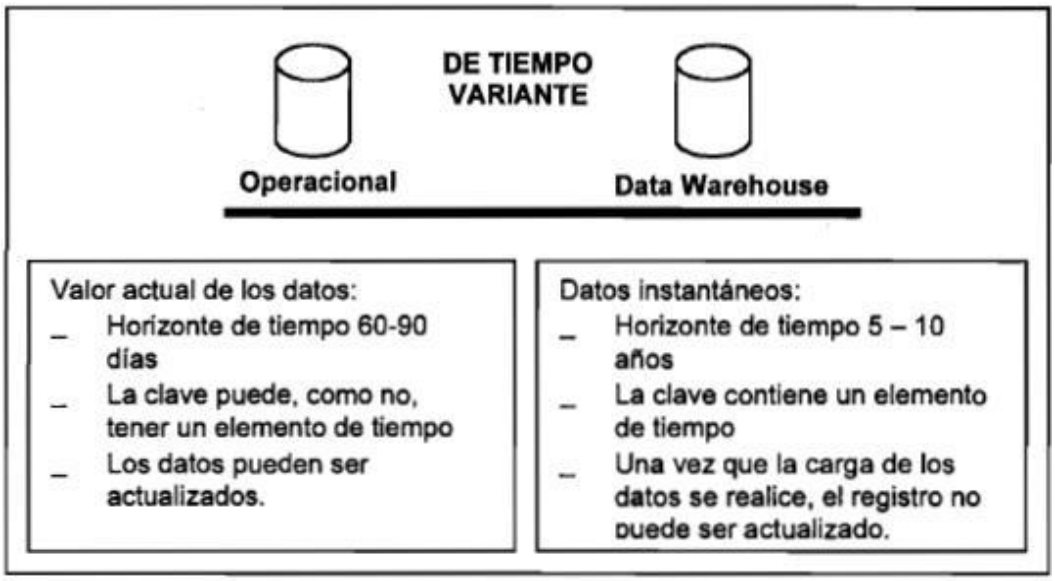


Figura 3. Los datawarehouse son de tiempo variante

El tiempo variante se muestra de varias formas:

- La más simple es que la información representa los datos sobre un horizonte de tiempo largo (desde 5 a 10 años). El horizonte de tiempo representado para el ambiente operacional es mucho más corto desde valores actuales hasta setenta a noventa días.
- Las aplicaciones que tienen un buen rendimiento y están disponibles para el procesamiento de transacciones, deben llevar una cantidad mínima de datos y algún grado de flexibilidad. Por ello, las aplicaciones operacionales tienen un corto horizonte de tiempo, debido al diseño de aplicaciones rígidas.
- Se muestra el tiempo variante en el datawarehouse, como una estructura clave. Cada estructura clave en éste contiene, implícita o explícitamente, un elemento de tiempo como día, semana, mes o año.
- La información una vez registrada correctamente no puede ser actualizada. La información contenida en el datawarehouse es para todos los propósitos prácticos.

No volátil: La información solo es útil cuando es estable. Los datos operacionales cambian momento a momento, pero la perspectiva esencial para el análisis y la toma de decisiones, requiere de una base de datos estable [4].

En la figura 4, se muestra que la actualización (insertar, modificar, borrar), se hace regularmente en el ambiente operacional sobre una base de registro. Hay dos únicos tipos de operaciones en el datawarehouse: la carga inicial de datos y el acceso a los mismos. No hay actualización en el depósito como una parte normal del proceso.

En el nivel de diseño del datawarehouse, la necesidad de ser precavido para actualizar las anomalías no es un factor primordial, ya que no se hace actualización de datos. Esto significa que en el nivel físico de diseño, se pueden tomar libertades para optimizar el acceso a los datos, particularmente al usar la normalización física.

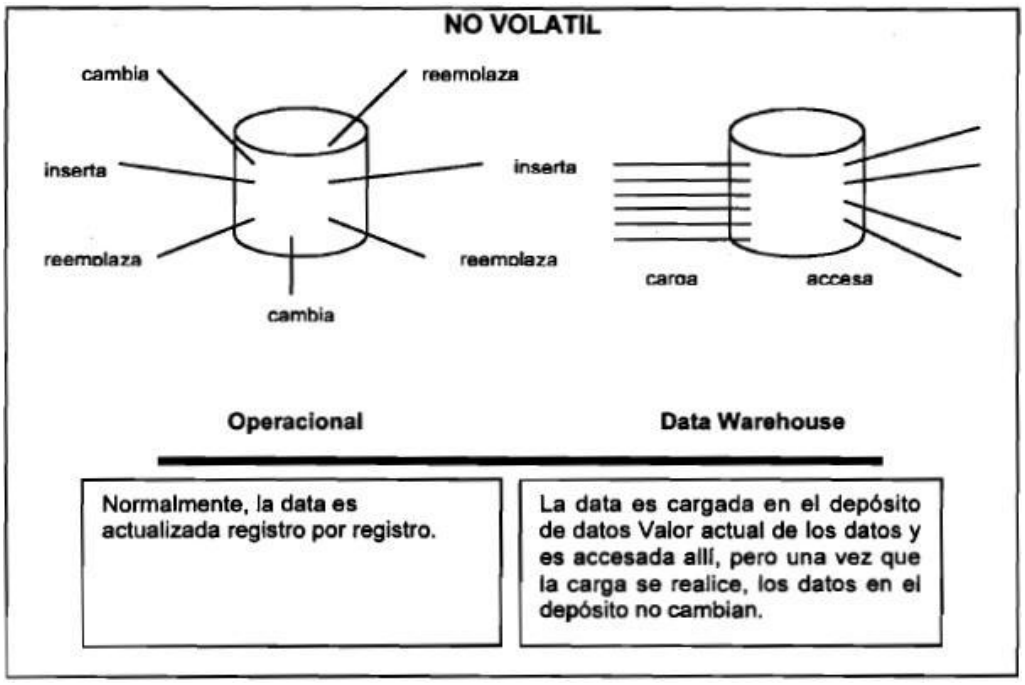


Figura 4. Los datawarehouse no son volátiles

Se debe considerar lo siguiente:

- Los datos se filtran cuando pasan por el ambiente operacional al depósito. Existen datos que nunca salen del ambiente operacional. Solamente los datos que se necesitan ingresan al ambiente del datawarehouse.
- El horizonte de tiempo de los datos difiere de un ambiente a otro. La información en el ambiente operacional es reciente con respecto a la del datawarehouse.
- El datawarehouse contiene un resumen de la información que se encuentra en el ambiente operacional.
- Los datos experimentan la transformación fundamental cuando pasan al datawarehouse. La mayor parte de los datos se alteran significativamente al ser seleccionados y movidos hacia él. Dicho de otra forma, la mayoría de los datos se alteran física y radicalmente cuando se mueven al depósito. No son los mismos datos que residen en el ambiente operacional desde el punto de vista de integración.

En resumen, las actualizaciones, inserciones, borrados y cambios, son regularmente operaciones en el ambiente operacional, pero la manipulación básica de los datos que ocurre en el datawarehouse es muy simple, solamente hay dos clases de operaciones que ocurren en él, la carga y el acceso de datos.

Ventajas y Desventajas del Datawarehouse

Entre las ventajas de un datawarehouse se encuentran:

- Proporciona una herramienta para la toma de decisiones en cualquier área funcional, basándose en información integrada y global del negocio.
- Facilita la aplicación de técnicas y estadísticas de análisis para encontrar relaciones ocultas entre los datos del almacén; obteniendo un valor añadido para el negocio de dicha información.
- Mejora la entrega de información, es decir, información completa, correcta, consistente, oportuna y accesible; información que los usuarios necesitan, en el momento adecuado y en el formato apropiado.
- Proporciona la capacidad de aprender de los datos del pasado y de predecir situaciones futuras en diversos escenarios.
- Simplifica dentro de la empresa la implantación de sistemas de gestión integral de la relación con el cliente.

- Permite reaccionar rápidamente a los cambios del mercado.
- Aumenta la competitividad en el mercado.
- Se consolida información de diferentes sistemas de origen, sin importar si estos provienen de la misma o varias fuentes.
- Existe consistencia de la información ya que se logra consolidar varios departamentos en uno solo. Es más fácil la toma de decisiones con la información consolidada que separada.
- El hecho de tener información ya almacenada y consolidada hace más fácil realizar el análisis de la misma.
- Realizar un datawarehouse provee las ventajas de utilizar información de múltiples fuentes de información sin importar la compatibilidad de ambas.
- Un datawarehouse ayuda a tener mejores tiempos de respuestas y mejora el proceso de producción.

Entre las desventajas de un datawarehouse se encuentran:

- Requiere una gran inversión, debido a que su correcta construcción no es tarea sencilla y consume muchos recursos, además, su implementación implica desde la adquisición de herramientas de consulta y análisis, hasta la capacitación de los usuarios.
- Existe resistencia al cambio por parte de los usuarios.
- No todos los usuarios confiarán en el datawarehouse en una primera instancia, pero sí lo harán una vez que comprueben su efectividad y ventajas. Además, su correcta utilización surge de la propia experiencia.

Una vez obtenido el conocimiento necesario sobre los beneficios que nos proporciona el uso del datawarehouse se hace imprescindible descubrir más a fondo cuales son los componentes que permiten el correcto almacenamiento y obtención de los datos requeridos para desarrollar un buen negocio.

1.4 Estructura del Datawarehouse

Hay niveles diferentes de esquematización y detalle que delimitan el datawarehouse, entre ellos se encuentran:

Detalle de datos actuales: En gran parte el interés más importante radica en el detalle de los datos actuales debido a que estos reflejan las ocurrencias más recientes. Son voluminosos, ya que se almacenan al más bajo nivel de detalle; casi siempre se almacenan en discos, permitiendo el fácil acceso, aunque su administración sea costosa y compleja.

Detalle de datos antiguos: Los datos antiguos son aquellos que se almacenan sobre alguna forma de almacenamiento masivo. No se accede a ellos frecuentemente y se almacenan a un nivel de detalle consistente con los datos detallados actuales.

Datos ligeramente resumidos: Los datos ligeramente resumidos son aquellos que provienen desde un bajo nivel de detalle encontrado, hasta un nivel de detalle actual. Uno de los puntos que el diseñador tiene para construirlo es que la unidad de tiempo se encuentre sobre la esquematización hecha.

Datos completamente resumidos: El siguiente nivel de datos encontrado en el datawarehouse es el de los datos completamente resumidos. Estos datos son compactos y fácilmente accesibles.

Metadata: El componente final del datawarehouse es el de la metadata. De muchas maneras la metadata se sitúa en una dimensión diferente al de otros datos del datawarehouse, debido a que su contenido no es tomado directamente desde el ambiente operacional. La metadata juega un rol especial y muy importante en el datawarehouse y es usada como:

- Un directorio para ayudar al analista a ubicar los contenidos del datawarehouse.
- Una guía para el mapeo de datos de cómo se transforma, del ambiente operacional al ambiente del datawarehouse.
- Una guía de los algoritmos usados para la esquematización entre el detalle de datos actual, con los datos ligeramente resumidos y estos, con los datos completamente resumidos [5].

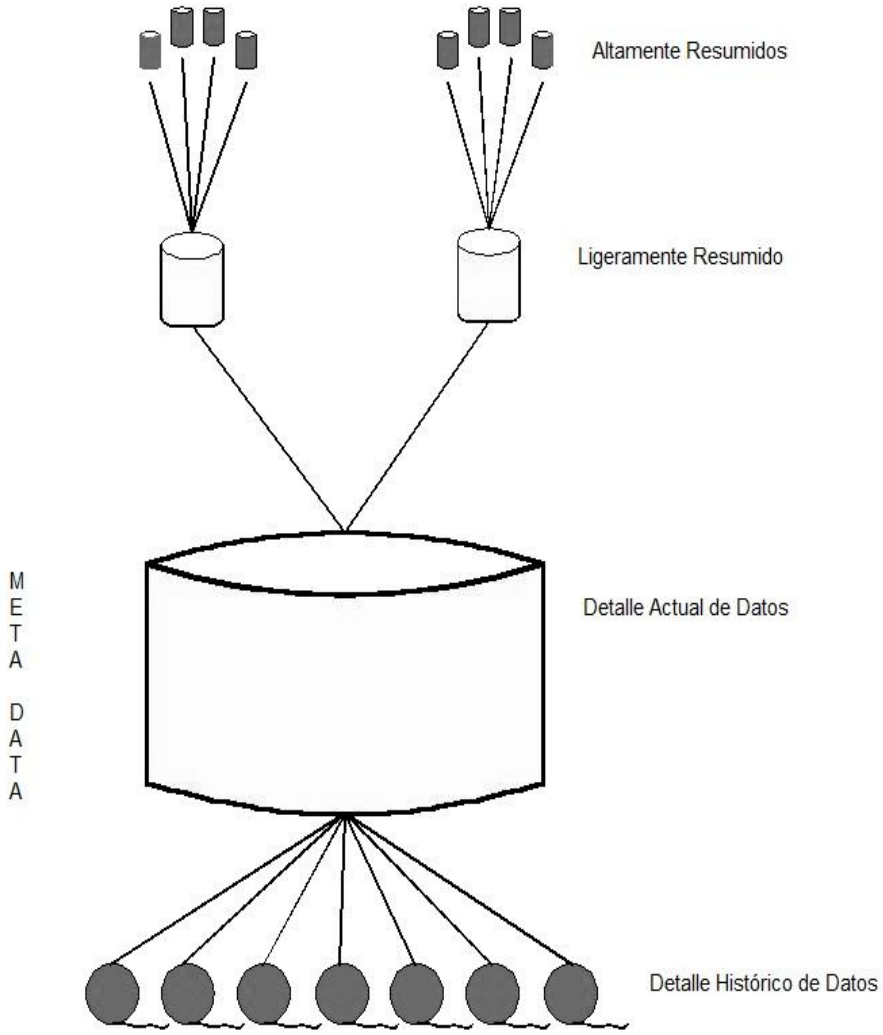


Figura 5. Estructura de los datos de un datawarehouse.

1.5 Arquitectura del Datawarehouse

La arquitectura del datawarehouse es la manera de mostrar la estructura global de los datos, así como su procesamiento y presentación, de forma tal que los usuarios finales dispongan de una mejor visión de los datos.

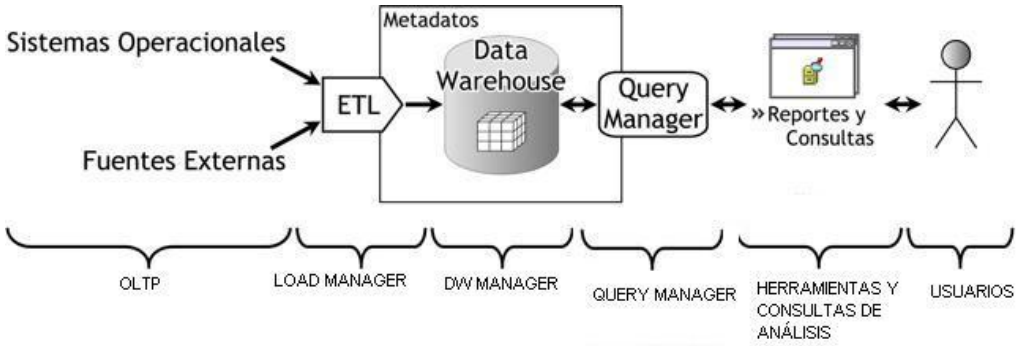


Figura 6. Arquitectura de un datawarehouse.

Como puede evidenciarse en la figura 6 los datawarehouse están formados por diversos elementos entre ellos encontramos los OLTP que representan toda aquella información que genera la empresa en su accionar diario, además, de las fuentes externas con las que puede llegar a disponer.

Otro elemento que integra la arquitectura es Load Manager, aquí es donde se realiza el proceso de extracción, transformación y carga (ETL), se extraen datos de las diversas fuentes, se transforman para resolver posibles problemas de inconsistencias entre los mismos y finalmente, después de haberlos depurado se procede a su carga en el depósito de datos.

El repositorio central del datawarehouse conocido como DW Manager también forma parte de esta arquitectura, el mismo posee las siguientes características:

- Los datos fuentes son transformados y almacenados en un modelo adecuado para la toma de decisiones.
- Gestiona el depósito de datos a través de tablas de hechos y tablas de dimensiones, y lo organiza en torno a una base de datos multidimensional.
- Realizan funciones de definición y manipulación del depósito de datos, para poder soportar todos los procesos de gestión del mismo.
- Tiene como objetivo conseguir una mayor eficiencia en las consultas al no tener que manejar todo el grueso de los datos.

El Query Manager es la parte de la arquitectura encargada de recibir las consultas del usuario, las aplica a la estructura de datos correspondiente y devuelve los resultados obtenidos.

Las herramientas de consulta y análisis son sistemas que permiten al usuario realizar la exploración de datos del datawarehouse. Utiliza una interfaz gráfica amigable que le permite al usuario generar consultas que son enviadas desde la herramienta de consulta y análisis al Query Manager, éste realiza la extracción de información al DW Manager y devuelve los resultados obtenidos a la herramienta que se los solicitó. Después de ejecutar esta operación los datos son mostrados al usuario de forma entendible.

Y por último, los usuarios que también forman parte de la arquitectura del datawarehouse, estos son los encargados de tomar las decisiones del negocio, es por ello que se hace énfasis en la integración y limpieza de los datos, para conseguir que la información extraída posea toda la calidad requerida.

Es importante resaltar que tanto la arquitectura como la estructura de los datawarehouse son importantes en el funcionamiento del mismo, debido a que estos almacenan información que debe estar organizada y estructurada en función de poder realizar búsqueda de datos que permitan la toma de decisiones.

1.6 Formas de Almacenamiento de Datos

El procesamiento analítico en línea OLAP (On Line Analytic Processing), es el componente más poderoso de los datawarehouse, ya que es el motor de consultas especializado de éste. Las herramientas OLAP requieren que los datos estén organizados dentro del depósito en forma multidimensional, por esto es que utilizan los cubos multidimensionales. Además, a través de este tipo de herramientas, se puede analizar el negocio desde diferentes escenarios históricos y proyectar como se ha venido comportando y evolucionando en un ambiente multidimensional, o sea, mediante la combinación de diferentes perspectivas, temas de interés o dimensiones.

ROLAP

El procesamiento analítico relacional en línea ROLAP (Relational On Line Analytic Processing), cuenta con todos los beneficios de un sistema gestor de base de datos relacional a los cuales se les provee de extensiones y herramientas para poder utilizarlo como un sistema gestor de datawarehouse. Este tipo de organización física se implementa sobre tecnología relacional, pero disponen de algunas facilidades para mejorar el rendimiento.

Entre las características más importantes y sobresalientes de ROLAP, se encuentran las siguientes:

- Almacena la información en una base de datos relacional.
- Posee tres capas lógicas: de almacenamiento, de análisis y de presentación.
- Utiliza índices de mapas de *bits*.
- Utiliza índices de *join*.
- Posee optimizadores de consultas.
- Cuenta con extensiones del SQL (*drill-up*, *drill-down*).
- Soporta grandes volúmenes de datos.

MOLAP

El procesamiento analítico multidimensional en línea MOLAP (Multidimentional On Line Analytic Processing), tiene el objetivo de almacenar físicamente los datos en estructuras multidimensionales de manera que la representación externa y la interna coincidan. Para ello, se dispone de estructuras de almacenamiento específicas (*arrays*) y técnicas de compactación de datos que favorecen el rendimiento del depósito de datos. Las principales características de MOLAP son:

- Posee tecnología optimizada para consultas y análisis, basada en el modelo multidimensional.
- Cuenta con un motor especializado.
- Proporciona herramientas limitadas y propietarias.
- No es adecuada para muchas dimensiones.
- Construye y almacena datos en estructuras multidimensionales.
- Almacenamiento en estructura multidimensional
- Mayor rapidez de respuestas.

HOLAP

El procesamiento analítico híbrido en línea HOLAP (Hybrid On Line Analytic Processing), constituye la unión entre MOLAP y ROLAP, combinando estas dos implementaciones para almacenar algunos datos en un motor relacional y otros en una base de datos multidimensional.

1.7 Estado Actual de los Datawarehouse

1.7.1 Nivel Internacional

Las empresas que utilizan datawarehouse son fundamentalmente aquellas que manejan grandes volúmenes de datos relacionados con clientes, compras, *marketing*, transacciones y operaciones. Los principales sectores donde se ha implantado datawarehouse son los siguientes [7]:

- Empresas de telecomunicaciones. Disponen de datos de millones de clientes: circuitos, facturas mensuales, volúmenes de llamadas, servicios utilizados, equipamiento vendido, configuraciones de redes. Telefonía móvil es un claro ejemplo de este tipo de compañías destacándose Jazztel, Vodafone, France Telecom. Estas empresas utilizan el datawarehouse para operar en un mercado creciente competitivo, no regulado y global que, a su vez, atraviesa profundos cambios tecnológicos.
- Empresas de transporte: Aerolíneas, Transporte de Cargas, Transporte de Pasajeros. Entre ellas British Airways, Union Pacific, Air France. En estas empresas se utilizan los datawarehouse para almacenar y acceder a meses o años de datos de clientes y sistemas de reservas para realizar actividades de mercadeo, planeamiento de capacidad, monitoreo de ganancias, proyecciones y análisis de ventas y costos, programas de calidad y servicios de clientes.

  Las empresas de transporte de cargas llevan datos históricos de años, millones de cargamentos, capacidades, tiempos de entrega, costos, ventas, equipamiento. Las aerolíneas utilizan los datawarehouse para los programas de viajeros frecuentes, para compartir información sobre los fabricantes de naves, para la administración de transporte de cargas y administración de inventarios.
- Turismo: Centrales de Reservas, Cadenas Hoteleras, Agencias de Viajes.
- Empresas de fabricación de bienes de consumo masivo. Entre ellas Coca-Cola, Adidas, Nike, 3M, Bosh Siemens, prácticamente todas las empresas de fabricación de automóviles. En este comercio se utilizan grandes sistemas de procesamiento paralelo masivo para acceder a meses o años de historia transaccional tomada directamente de los puntos de venta de cientos, o miles de sucursales. Con esta información detallada se efectúan en forma más precisa y eficiente actividades de compra, fijación de precios y manejo de inventarios. Las promociones y las ofertas de cupones son seguidas, analizadas y corregidas. Modas y tendencias son cuidadosamente administradas a efectos de maximizar utilidades y reducir costos de inventario.
- Entidades Financieras: BBVA, Caja Madrid, Caja Extremadura entre otras.

- La compañía Teradata, especializada en soluciones de datawarehouse, ha anunciado que la empresa especializada en productos para la belleza y salud Australian Pharmaceutical Industries, ha actualizado su sistema Teradata con la aplicación Datawarehouse Appliance, éste tiene el fin de añadir visibilidad dentro de sus ventas y procesos de inventario. La compañía cuenta con cerca de 4 mil farmacias en toda Australia.

Es importante resaltar que las empresas Sybase y Sun han unido fuerzas para desarrollar The Enterprise Datawarehouse Reference Architecture que abrió la posibilidad de la creación y desempeño del datawarehouse más grande del mundo. Esta solución suple las necesidades de: simplicidad, flexibilidad, administración y protección de la inversión.

De forma general se puede decir que estas empresas que utilizan datawarehouse cuentan con mayor aceptación de sus productos, apenas éste comienza a ser fuente primaria de información las personas tienen mayor confianza en las decisiones empresariales que se toman. El datawarehouse hace lo posible para aprovechar el valor potencial de los recursos de información de la empresa y convertir ese valor potencial en valor verdadero. Éste extiende el alcance de la información para que se acceda directamente en línea, lo que a la vez contribuye en su capacidad para operar con mayor efectividad las tareas rutinarias.

Los usuarios del datawarehouse acceden a una riqueza de información multidimensional, es presentado coherentemente como una fuente única, confiable y disponible para ellos por medio de sus estaciones de trabajo. Las decisiones empresariales se hacen más rápidas por personas más informadas y el tiempo perdido esperando información que finalmente es incorrecta o no encontrada, se elimina.

1.7.2 Nivel Nacional

Con el actual avance de las tecnologías, las empresas cubanas aspiran a un mejor manejo de sus datos y mejor visión de los mismos. Un ejemplo de tal desempeño es el datawarehouse implantado por la Corporación CIMEX que se dedica fundamentalmente a la exportación e importación de mercancías. Éste centra su atención en la actividad del comercio, principalmente en la gestión de inventario, permitiendo una gestión de compra–venta eficiente, con una finalidad fundamental: disminuir los costos, sin afectar al

cliente, permitiendo prestaciones eficientes y con la calidad requerida, aumentando las ganancias o utilidades de las empresas.

En el XIII Concurso Nacional de Computación y en la Feria de Informática del 2002 se presentó un Almacén de Datos para CUBACEL con buenos resultados obtenidos a partir de su implantación.

La UCI cuenta con el Centro de Tecnología de Gestión de Datos DATEC (Data Tecnology Center), en el que se han desarrollado algunos datawarehouse como es el de la Oficina Nacional de Estadística (ONE). Actualmente se lleva a cabo el desarrollo de uno para la propia universidad, éste no está completo pero ya cuenta con los data mart Portadores energéticos y Gestión de proyectos, posteriormente se desarrollarán los otros para ser integrados. En el curso 2008-2009, estudiantes de la facultad siete de este mismo centro universitario desarrollaron un datawarehouse para el control del Recursos Humanos de la Salud en Cuba y estudiantes de la facultad seis desarrollaron uno para el proyecto Sistemas de Gestión de Información de Laboratorios específicamente para el módulo Análisis Químico.

Como se ha podido apreciar, Cuba ha estado al tanto de los últimos cambios tecnológicos con la implantación de estos datawarehouse. Para dar cumplimiento a los objetivos trazados en su realización se tuvo en cuenta una serie de pasos que conformaron su estructura.

1.8 Metodología para el Desarrollo de un Datawarehouse

1.8.1 Metodología Hefesto

La metodología que se utiliza es HEFESTO, cuya propuesta está fundamentada en una investigación previa titulada “Análisis de un almacén de datos para la red nacional de Genética Médica” en la cual se realizó una comparación de las metodologías existentes llegando a la conclusión que ésta que se propone es la más apropiada para realizar el análisis, diseño e implementación de este datawarehouse, debido a que es una metodología sencilla, ordenada, explícita y efectiva, que permite desarrollar un datawarehouse, guiándose por pasos lógicos relacionados sólidamente durante todas las etapas del proceso de confección.

Las características de esta metodología son [8]:

- Los objetivos y resultados esperados en cada fase se distinguen fácilmente y son sencillos de comprender.
- Se basa en los requerimientos del usuario, por lo que su estructura es capaz de adaptarse con facilidad y rapidez ante los cambios en el negocio.
- Reduce la resistencia al cambio, ya que involucra al usuario final en cada etapa para que tome decisiones respecto al comportamiento y funciones del datawarehouse.
- Utiliza modelos conceptuales y lógicos.
- Es independiente del tipo de ciclo de vida que se emplee para contener la metodología.
- Es independiente de las herramientas que se utilicen para su implementación.
- Es independiente de las estructuras físicas que contengan el datawarehouse y de su respectiva distribución.
- Cuando se culmina con una fase, los resultados obtenidos se convierten en el punto de partida para llevar a cabo el paso siguiente.
- Se aplica tanto para *data mart* como para datawarehouse.

En la figura 7 se muestra la secuencia de pasos que sigue esta metodología.

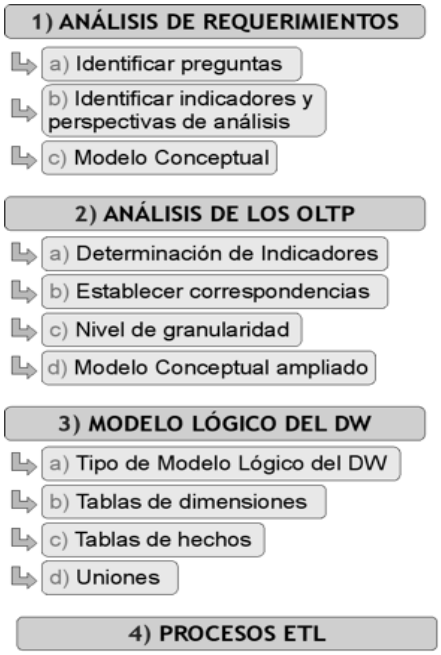


Figura 7. Descripción de la metodología Hefesto.

1.9 Herramientas para el Desarrollo de un Datawarehouse

Para el desarrollo de un datawarehouse es necesario tener en cuenta las herramientas con las que se va a realizar el diseño y la implementación del mismo, a continuación se mencionan alguna de ellas: herramientas CASE, Sistemas Gestores de Bases de Datos (SGBD) y herramientas que permitan realizar pruebas.

Herramientas CASE

Algunas de las herramientas CASE que se utilizan para el diseño de un datawarehouse son:

- Erwin: es una herramienta para el diseño de base de datos, que brinda productividad en su diseño, generación, y mantenimiento de aplicaciones. Desde un modelo lógico de los requerimientos de información, hasta el modelo físico perfeccionado para las características específicas de la base de datos diseñada, permite visualizar la estructura, los elementos importantes, y optimizar el diseño de la base de datos. Erwin soporta principalmente bases de datos relacionales SQL y bases de datos que incluyen Oracle, Microsoft SQL Server, Sybase.
- Erecase: Es una herramienta para el diseño de bases de datos que utiliza como modelo conceptual el modelo Entidad Relación Extendido (ERE). Como característica novedosa permite la validación estructural del diagrama ER basándose en los extremos máximos y mínimos de las relaciones. Esta nueva herramienta CASE se crea para el diseño de bases de datos con el objetivo de permitir la validación estructural de los diagramas Entidad Relación Extendido (ERE).
- DBDesigner es un sistema totalmente visual de diseño de bases de datos, que combina características y funciones profesionales con un diseño simple, muy claro y fácil de usar, a fin de ofrecer un método efectivo para gestionar las bases de datos. Está disponible tanto para sistemas GNU/Linux como para Microsoft Windows 2000/XP. Tiene características interesantes como la posibilidad de poder hacer ingeniería inversa: generar el modelo ER a raíz de una base de datos existente, también permite la sincronización del trabajo con servidores de bases de datos (para ir aplicando los últimos cambios que se hacen sobre el modelo relacional directamente a su base de datos) y dispone de dos modos principales: el Modo diseño (*Design Mode*) y el Modo Consulta (*Query Mode*).

Después de haber realizado un análisis de las herramientas existentes para el diseño de un datawarehouse se llega a la conclusión que la herramienta que cumple con todas las exigencias es **DBDesigner.**

Entre las principales características que avalan esta decisión se encuentran [9]:

- Modelados realizados en XML.
- Permite hacer ingeniería inversa desde bases de datos como MySql, Oracle, Microsoft SQL Server y cualquier base de datos ODBC.
- Modo de diseño y de consulta: permite realizar por una parte el modelado de la base de datos y por otra realizar consultas sobre las tablas y construir consultas en SQL para PHP, Kylix y otros lenguajes de programación.
- Permite generar ficheros .sql con las sentencias necesarias para crear el modelo plasmado en la parte gráfica.
- Colocación automática de las llaves foráneas.
- Disponible para Linux y Windows.
- Es software libre y licenciado bajo la GNU GPL. Esto significa que se pueden descargar ejecutables así como el código fuente del programa y usarlo de forma gratuita.

Sistemas Gestores de Bases de Datos

Los SGBD son un tipo de software muy específico, dedicados a servir de interfaz entre la base de datos, el usuario y las aplicaciones que lo utilizan. Se compone de lenguajes de definición, manipulación, consulta y seguridad de datos. El propósito general de los SGBD es el de manejar de manera clara, sencilla y ordenada un conjunto de datos [10].

A continuación se realiza un estudio comparativo entre dos potentes SGBD, donde se exponen sus principales características:

MySQL

MySQL es un sistema de gestión de bases de datos relacional que tiene gran aceptación debido a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración.

Las principales características de este gestor de bases de datos son las siguientes [11]:

- Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP).
- Soporta hasta 32 índices por tabla.
- Gestión de usuarios y contraseñas, manteniendo un muy buen nivel de seguridad en los datos.

Sin embargo este SGDB es lento con grandes bases de datos y algo muy importante carece de integridad referencial.

PostgreSQL

PostgreSQL es un potente SGDB, que tiene prestaciones y funcionalidades equivalentes a muchos gestores de bases de datos comerciales. Es más completo que MySQL ya que permite métodos almacenados, restricciones de integridad y vistas.

A continuación se enumeran las principales características de este gestor de bases de datos [11]:

- Soporta distintos tipos de datos: fecha, monetarios, elementos gráficos, cadenas de *bits*.
- Incorpora una estructura de datos *array*.
- Incorpora funciones de diversa índole: manejo de fechas, geométricas, orientadas a operaciones con redes.
- Permite la declaración de funciones propias, así como la definición de disparadores.
- Soporta el uso de índices, reglas y vistas.
- Incluye herencia entre tablas (aunque no entre objetos, ya que no existen), por lo que a este gestor de bases de datos se le incluye entre los gestores objeto-relacionales.

Entre las ventajas que ofrece se destacan:

- Instalación ilimitada: Es frecuente que las bases de datos comerciales sean instaladas en más servidores de lo que permite la licencia. Con PostgreSQL, nadie puede demandarlo por violar acuerdos de licencia, puesto que no hay costo asociado a la licencia del software.
- Estabilidad y confiabilidad legendaria: En contraste a muchos sistemas de bases de datos comerciales, es extremadamente común que compañías reporten que PostgreSQL nunca ha presentado caídas en varios años de operación de alta actividad.

- Extensible: El código fuente está disponible para todos sin costo.
- Diseñado para ambientes de alto volumen: PostgreSQL usa una estrategia de almacenamiento de filas para conseguir mejor respuesta en ambientes de grandes volúmenes.

Luego del estudio de estos dos SGBD se ha seleccionado para la realización del datawarehouse **PostgreSQL** debido a las características anteriormente expuestas y además, es más rápido con grandes volúmenes de información y tiene la capacidad de comprobar la integridad referencial, así como también la de almacenar procedimientos en la propia base de datos.

Para realizar este análisis solo se tomó en cuenta estos SGBD debido a que los restantes son propietarios y los que pudieran quedar libres no poseen las ventajas que los sistemas anteriormente mostrados brindan.

Después de haber escogido **PostgreSQL** como SGDB, se necesita seleccionar la herramienta con la cual se va a trabajar, tomando como base que se debe utilizar una herramienta libre se ha elegido **PgAdmin III**, escrita en C++ usando la librería gráfica, lo que permite que se pueda usar en sistemas operativos como Linux, Mac OS X y Windows, entre otros. Es capaz de gestionar versiones a partir de la PostgreSQL 7.3 ejecutándose en cualquier plataforma. PgAdmin III está diseñado para responder a las necesidades de todos los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas.

Herramientas para realizar pruebas a un datawarehouse

Las pruebas constituyen un elemento de gran importancia dentro del datawarehouse, nos dan una medida del futuro funcionamiento del mismo. Para realizar las pruebas es necesario utilizar herramientas robustas que nos permitan analizar correctamente los resultados, entre ellas se encuentra el **Data Generator para PostgreSQL**.

Ésta es una poderosa herramienta para generar datos de prueba a tablas de bases de datos PostgreSQL. La aplicación asistente le permite definir tablas y campos, configurar valores de rangos, obtener listas de valores desde consultas SQL.

Entre las características de Data Generator para PostgreSQL se encuentran:

- Amigable interfaz de usuario.
- Soporte para todos los tipos de datos de PostgreSQL, incluyendo los tipos *array*, dirección de red y geométricos.
- Diferentes tipos de generación por cada campo, incluyendo lista, azar, generación incremental de datos.
- Capacidad para usar resultados de consultas SQL como lista de valores para la generación de datos.
- Control automático sobre integridad referencial para la generación de datos a tablas vinculadas.

Otra de las herramientas a utilizar es **JMeter**, considerado un generador de carga diseñado para la realización de pruebas de carga y estrés, se caracteriza por su versatilidad y estabilidad. Se ejecuta sobre la máquina virtual de Java por lo que es multiplataforma. Genera carga por diversos protocolos, ya sea, FTP, HTTP, HTTPS, SQL. Realiza carga variable, en niveles de concurrencia, número de veces, tiempo y su característica principal radica en que pertenece a la familia de software libre. Muestra los resultados de las pruebas en una amplia variedad de informes y gráficas. Además facilita a una rápida detección de los cuellos de botella existentes debido al tiempo de respuesta excesivo.

Ventajas de la herramienta JMeter:

- De las herramientas gratis, es la más completa y útil para las pruebas de carga.
- Tiene una estructura en árbol que le da potencia, permitiendo que los usuarios que la utilice poner los límites a la hora de diseñar el plan de prueba. Brinda mayor cantidad de variantes para recoger los resultados obtenidos que el resto de las herramientas gratis, lo que permite hacer un análisis exhaustivo de las pruebas realizadas.

Conclusiones del capítulo

En este capítulo se realizó un estudio de las tendencias, tecnologías y herramientas actuales para el diseño e implementación de los datawarehouse, de esta manera se determinó utilizar como herramienta CASE DBDesigner para realizar el diseño del datawarehouse, PostgreSQL como SGBD, Data Generator para PostgreSQL y Jmeter para realizar las pruebas, todas ellas pertenecientes a la familia de software libre. Además se definió como metodología a utilizar: Hefesto y se puntualizaron los principales componentes de la arquitectura de un datawarehouse.

CAPÍTULO 2: ANÁLISIS DE LA SOLUCIÓN PROPUESTA

En este capítulo se muestran las estructuras multidimensionales modeladas para el datawarehouse, basándose en el análisis realizado anteriormente según lo describe la metodología utilizada. Con la aplicación de esta metodología se lleva a cabo la construcción de un modelo lógico y su implementación a partir del modelo conceptual propuesto.

2.1 Arquitectura de la Solución

Para su descripción la arquitectura se divide en 3 secciones: el componente de presentación, el repositorio central y el de carga de datos. En cada una de las secciones existe un conjunto de herramientas que soportan el proceso.

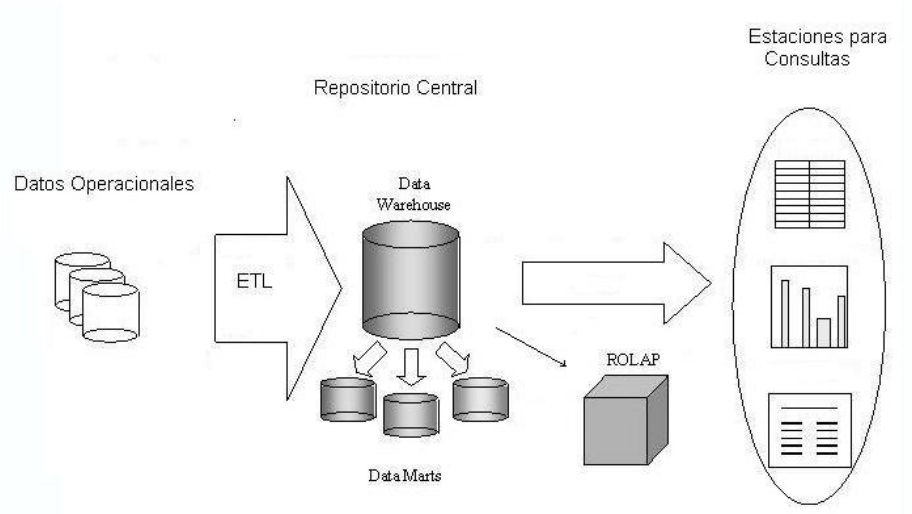


Figura 8. Arquitectura de la solución.

El componente más importante y sobre el cual se basa el sistema es el repositorio central, la estructura del mismo está compuesta por el SGBD PostgreSQL que es donde va a estar desplegado el datawarehouse. Las estaciones para consulta y administración son aquellas donde los usuarios finales realizarán la búsqueda de la información requerida para la toma de decisiones.

Por último y no menos importante se encuentra el componente de administración y carga de datos que aunque no constituye alcance de nuestra investigación se propone la utilización de la herramienta Pentaho Data Integration para todo el proceso ETL ya que brinda la posibilidad de abrir, limpiar e integrar esta

valiosa información y ponerla en manos del usuario, además evita grandes cargas de trabajo manuales frecuentemente difícil de mantener y de desplegar, es importante resaltar que esta herramienta es una iniciativa de la comunidad de Open Source. Las rutinas de ETL se encontrarán diseñadas para ser utilizadas cada vez que se desee adicionar datos al repositorio siempre y cuando esos datos se encuentren en un formato similar al utilizado para la carga del histórico.

Es importante resaltar, aunque no sea alcance de esta investigación, que se debe implementar el proceso analítico en línea utilizando ROLAP dado que la base de datos que se construye como parte del datawarehouse estará soportada por el sistema gestor de base de datos PostgreSQL.

2.2 Modelado Multidimensional

Aquí se encuentra una de las diferencias entre los sistemas operacionales y los datawarehouse, cada uno de ellos es sostenido por un modelo de datos diferentes. Los sistemas operacionales se sustentan en el Modelo Entidad Relación (MER) y el datawarehouse trabaja con el Modelo Multidimensional. Los datawarehouse gestionan el depósito de datos y lo organizan en torno a una base de datos multidimensional que tal y como lo indica su nombre, almacena los datos en diversas dimensiones que conforman un cubo multidimensional, en donde el cruce de los valores de los atributos de cada dimensión a lo largo de las abscisas, determina un hecho específico que se define como datos instantáneos en el tiempo, que son filtrados, agrupados y explorados a través de condiciones definidas en las tablas de dimensiones.

En la siguiente figura puede verse la representación de los datos en un cubo multidimensional.

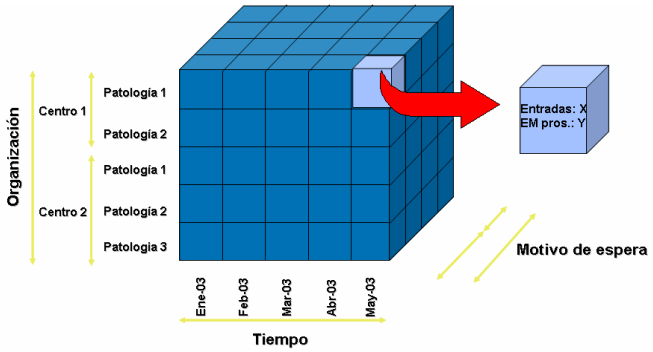


Figura 9. Cubo multidimensional.

Las bases de datos multidimensionales implican tres variantes de modelado:

- Esquema en estrella (Star Scheme).
- Esquema copo de nieve (Snowflake Scheme).
- Esquema constelación o copo de estrellas (Starflake Scheme).

Esquema en estrella: es la técnica de diseño más popular usada para un datawarehouse. Es un paradigma en el cual un único objeto en el centro (conocido como tabla de hechos) está conectado radialmente con otros objetos circundantes llamados tabla de dimensiones formando una estrella.

Éste es el más simple de interpretar y optimiza los tiempos de respuesta ante las consultas de los usuarios. Este modelo es soportado por casi todas las herramientas de consulta y análisis, y los metadatos que representan la información acerca de los datos, son fáciles de documentar y mantener.

Entre las características del esquema en estrella se encuentran:

- Posee los mejores tiempos de respuesta.
- Existe paralelismo entre su diseño y la forma en que los usuarios visualizan y manipulan los datos.
- Simplifica el análisis.
- Facilita la interacción con herramientas de consulta y análisis.

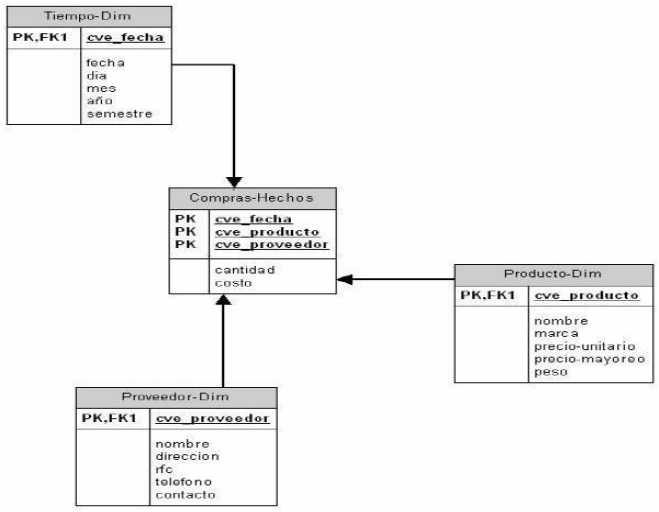


Figura 10. Modelo de estrellas con 3 dimensiones y una tabla de hechos.

Esquema Copo de Nieve: es una extensión del esquema en estrella donde cada punta de la estrella se explota en más puntas y su denominación se debe a que el diagrama del esquema se asemeja a un copo de nieve.

Entre las características del copo de nieve se encuentran:

- Posee mayor complejidad en su estructura.
- Hace una mejor utilización del espacio.
- Las dimensiones están normalizadas, por lo que requiere menos esfuerzo de diseño.
- Puede desarrollar clases de jerarquías fuera de las dimensiones, que permiten realizar análisis de lo general a lo detallado y viceversa.

A pesar de todas las características y ventajas que trae aparejada la implementación del esquema copo de nieve, existen dos grandes inconvenientes de ello:

- Si se poseen múltiples dimensiones, cada una de ellas con varias jerarquías, se creará un número de dimensiones bastante considerable, que pueden llegar al punto de ser inmanejables.
- Al existir muchas uniones y relaciones entre tablas, el desempeño puede verse reducido.

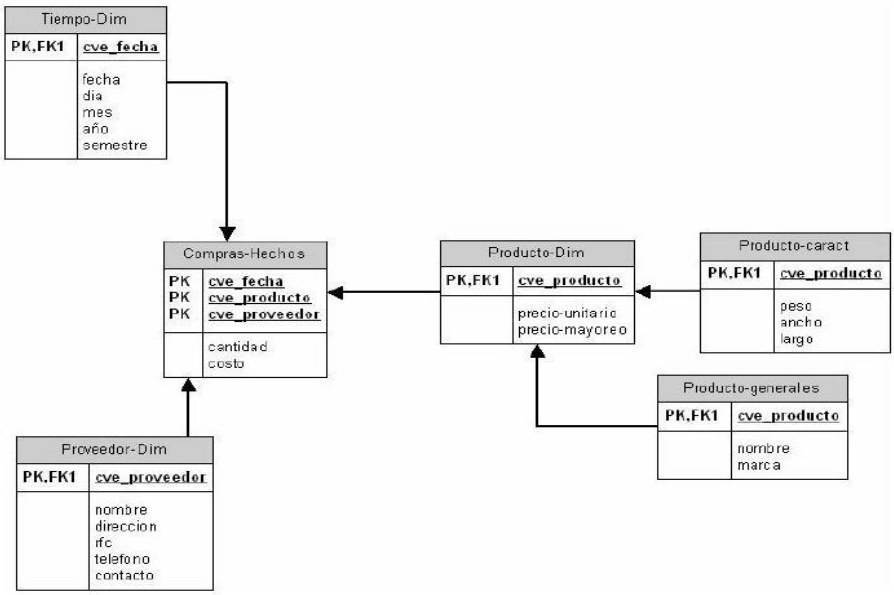


Figura 11. Modelo Copo de Nieve.

Esquema Constelación: Este modelo está compuesto por una serie de esquemas en estrella, y está formado por una tabla de hechos principal y por una o más tablas de hechos auxiliares. Dichas tablas yacen en el centro del modelo y están relacionadas con sus respectivas tablas de dimensiones. No es necesario que las diferentes tablas de hechos compartan las mismas tablas de dimensiones, ya que las tablas de hechos auxiliares pueden vincularse con solo algunas de las tablas de dimensiones asignadas a la tabla de hechos principal, y también pueden hacerlo con nuevas tablas de dimensiones. Su diseño es muy similar a la del esquema en estrella. Con éste se podrán analizar más aspectos claves del negocio con un mínimo esfuerzo de diseño, además contribuye a la reutilización de dimensiones debido a que una misma dimensión puede utilizarse para varias tablas de hechos.

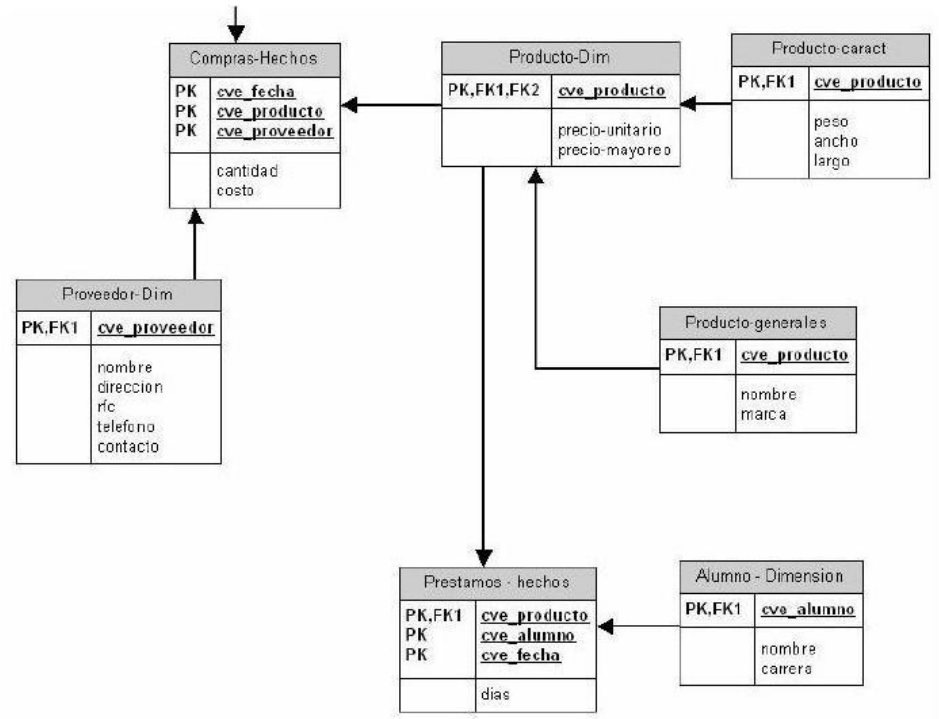


Figura 12. Esquema de Constelación.

El esquema que se utiliza para la representación del datawarehouse es el **esquema en estrella**, debido a que contiene una estructura de depósito de datos que se adapta mejor a los requerimientos y necesidades del usuario.

2.3 Modelo Lógico de la Estructura del Datawarehouse

Para conformar la estructura del datawarehouse se toma como base el modelo conceptual propuesto en la investigación anteriormente realizada. Este modelo está guiado por una secuencia de pasos que propone la metodología Hefesto, teniendo en cuenta el orden de las tareas a seguir por dicha metodología corresponde realizar el desarrollo del modelo lógico.

Es necesario tener una estandarización de los nombres para que el diseño de este modelo lógico muestre una mejor visión, o sea, organizar la forma en que se van a denominar las estructuras con el fin de que quede documentado para su utilización por los inmersos en la arquitectura.

Por esta razón si la tabla es una dimensión, al nombre le preceden las letras “dim” ejemplo dim_enfermedades. En caso de ser una tabla de hecho se le antepone al nombre las siglas “hech”, ejemplo, hech_casos_genetica. Al finalizar este paso queda completamente estructurada la nomenclatura utilizada dentro del datawarehouse. Una vez definidas es que se comienza con la implementación de la estructura física.

2.3.1 Tablas de Dimensiones

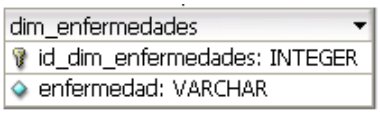
Las tablas dimensiones o tablas *lock_up* almacenan un conjunto de valores que están relacionados a una dimensión particular. Éstas almacenan los valores que se utilizan en las tablas de hechos. Definen como están los datos organizados lógicamente y proveen el medio para analizar el contexto del negocio.

Para realizar estas tablas se tomará en cuenta las perspectivas extraídas del modelo conceptual, este paso se le aplicará por igual a todos los tipos de esquemas lógicos. Lo primero que se realiza es la creación de las dimensiones del mismo, para ello se toma cada perspectiva con sus atributos relacionados y se les realizará el siguiente proceso:

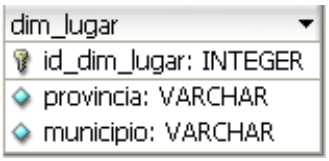
- Se elegirá un nombre que identifique la dimensión.
- Se añadirá un campo que represente su clave principal.
- Se redefinirán los nombres de los atributos si no son suficientemente explícitos.

Perspectiva "Persona"

Perspectiva "Enfermedades"

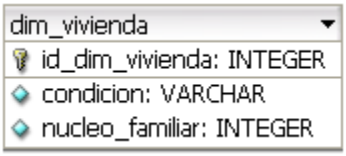


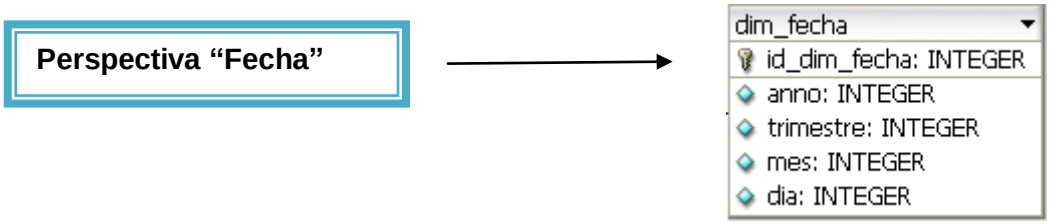
Perspectiva "Lugar"



Perspectiva "Conducta Social"

Perspectiva "Vivienda"





Finalmente quedan diseñadas las tablas dimensiones, cada una de ellas cuenta con una clave primaria que la identifica, en este caso los atributos que empiezan su nombre con la palabra *id,* son aquellos que dentro de cada tabla actúan como llaves primarias.

2.3.2 Tablas de Hechos

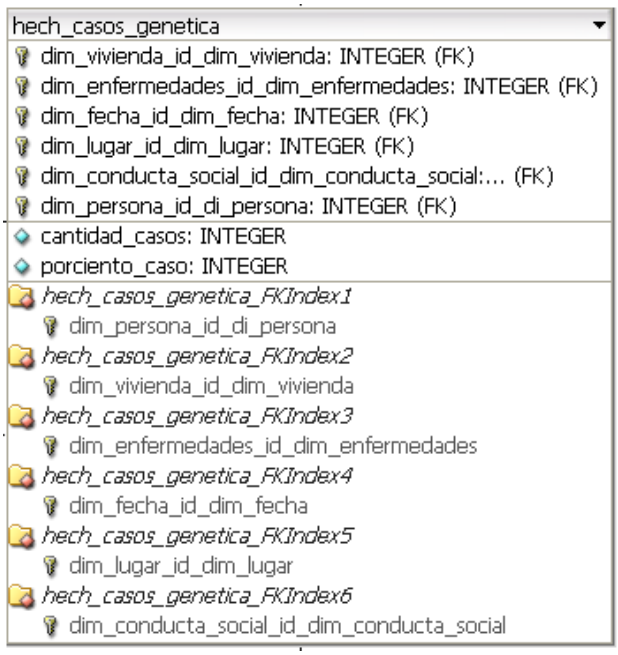
Una tabla de hecho es una representación de un proceso de negocio. Cada datawarehouse incluye una o varias tablas de hechos. Una característica esencial de las tablas de hechos es que contienen datos numéricos (hechos) que se pueden resumir para proporcionar información sobre el historial de las operaciones de la organización. Cada tabla de hechos también incluye un índice de varias partes que contiene, como claves externas, las claves primarias de las tablas de dimensiones relacionadas así como los atributos de los registros de hechos.

La clave de la tabla hecho recibe el nombre de clave compuesta o concatenada debido a que se forma de la composición (o concatenación) de las llaves primarias de las tablas dimensionales a las que está unida. Así entonces, se distinguen dos tipos de columnas en una tabla de hecho: columnas *fact* y las columnas *key*, donde la columna *fact* es la que almacena alguna medida de negocio y una columna *key* forma parte de la clave compuesta de la tabla [8].

En este paso, se definirán las tablas de hechos, que son las que contendrán los indicadores seleccionados en el modelo lógico. Para el esquema en estrella se realizará de la siguiente forma:

- Se le asigna un nombre a la tabla de hecho, el cual representará el negocio enfocado.
- Se le asignará una clave primaria que será la combinación de todas las llaves primarias de las dimensiones que se usarán para realizar las consultas.
- Se renombrarán los hechos o indicadores si es que no llegasen a ser lo suficientemente explícitos.

A continuación se muestra la tabla de hechos: hech_casos_genetica, donde las columnas *key* están ubicadas en la parte superior conformando la unión de todas las llaves primarias de las dimensiones con la que se relaciona y la columna *fact* está conformada por los atributos cantidad_casos y porciento_casos.



2.3.3 Uniones y Jerarquías

Para todos los tipos de esquemas, se realizan las uniones correspondientes entre sus tablas de dimensiones y sus tablas de hechos, en este caso para realizar la unión del proceso se utiliza el **esquema estrella**.

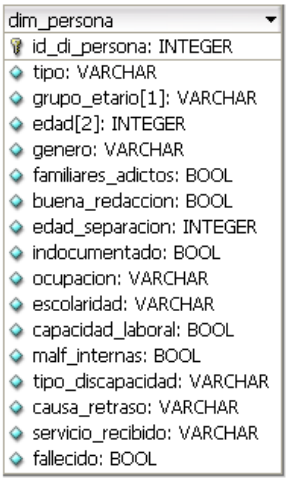
Dentro de las dimensiones es posible definir jerarquías, las cuales son grupos de atributos que siguen un orden preestablecido. Una jerarquía implica una organización de niveles dentro de una dimensión, con cada nivel representando el total agregado de los datos del nivel inferior. Una dimensión típica soporta una o más jerarquías naturales.

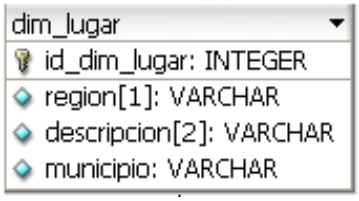
Una jerarquía puede pero no exige contener todos los valores existentes en la dimensión. Se debe evitar caer en la tentación de convertir en tablas dimensionales separadas cada una de las relaciones muchos-a-uno presentes en las jerarquías. Esta descomposición es irrelevante en el planeamiento del espacio ocupado en disco y solamente dificulta el entendimiento de la estructura para el usuario final.

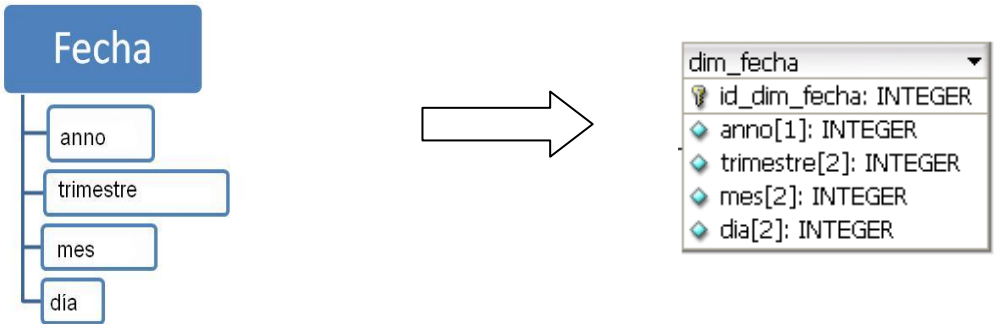
Las jerarquías poseen las siguientes características:

- Pueden existir varias en una misma dimensión.
- Están compuestas por dos o más niveles.
- Se tiene una relación de uno a muchos entre atributos consecutivos de un nivel superior y uno inferior.

A continuación se muestran las jerarquías de los atributos de algunas de las dimensiones que componen al datawarehouse:







Una vez analizadas las dimensiones y sus jerarquías, el diseño del modelo lógico queda de la siguiente forma:

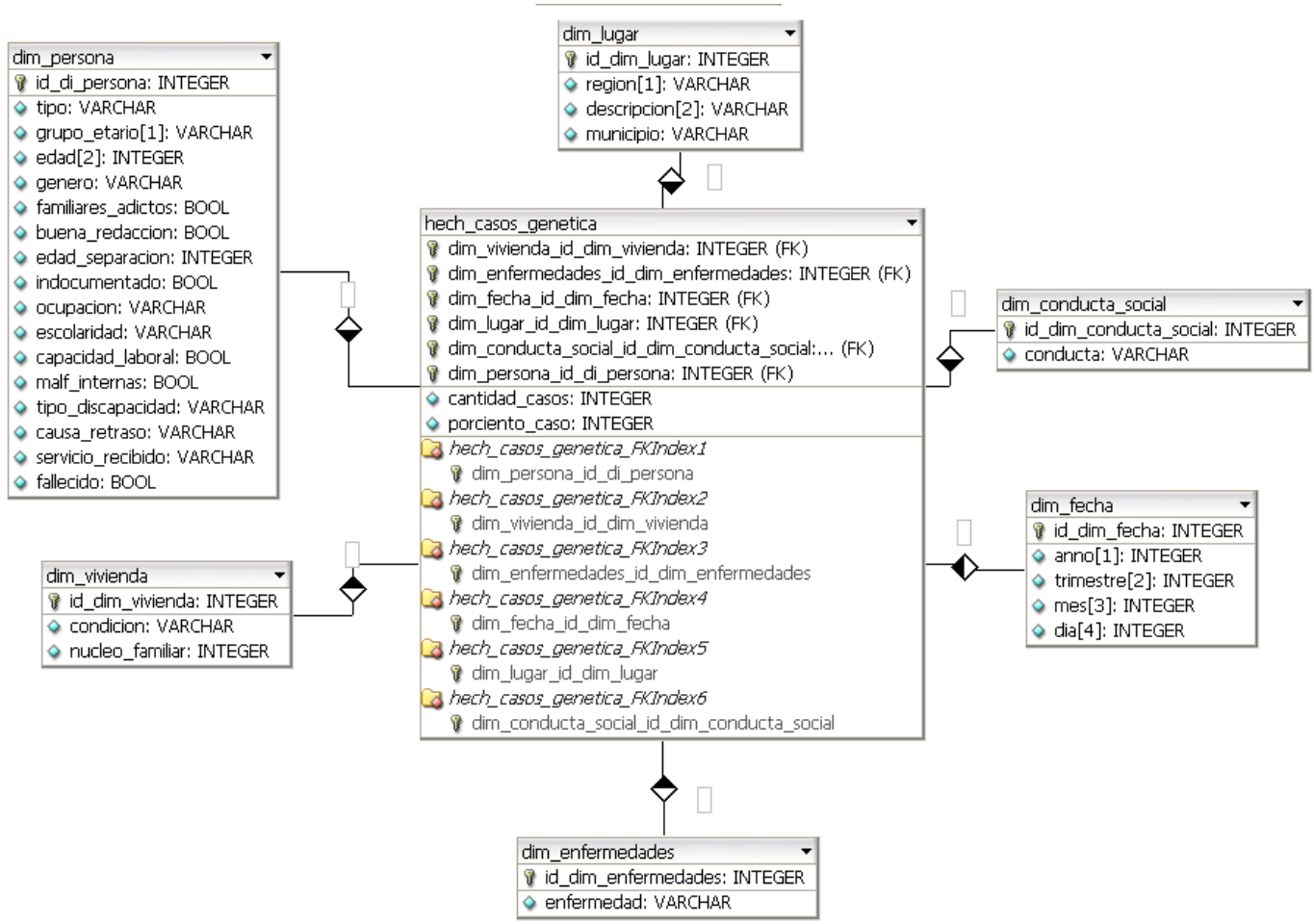


Figura 13. Modelo lógico del diseño del datawarehouse con las jerarquías.

Después de haber concluido el diseño del modelo lógico se procede a la implementación del mismo, para esto es imprescindible tener en cuenta el SGDB que soportará el datawarehouse.

2.4 Implementación del modelo multidimensional

Para realizar la implementación del modelo lógico se extrae el script de la herramienta DBDesigner que fue la escogida para realizar el diseño del datawarehouse y luego se ejecuta en el SGBD, antes de ser desplegado es necesario revisar que las estructuras dimensionales estén correctamente diseñadas. La figura 14 ilustra las tablas dimensiones y hechos dentro del SGBD.

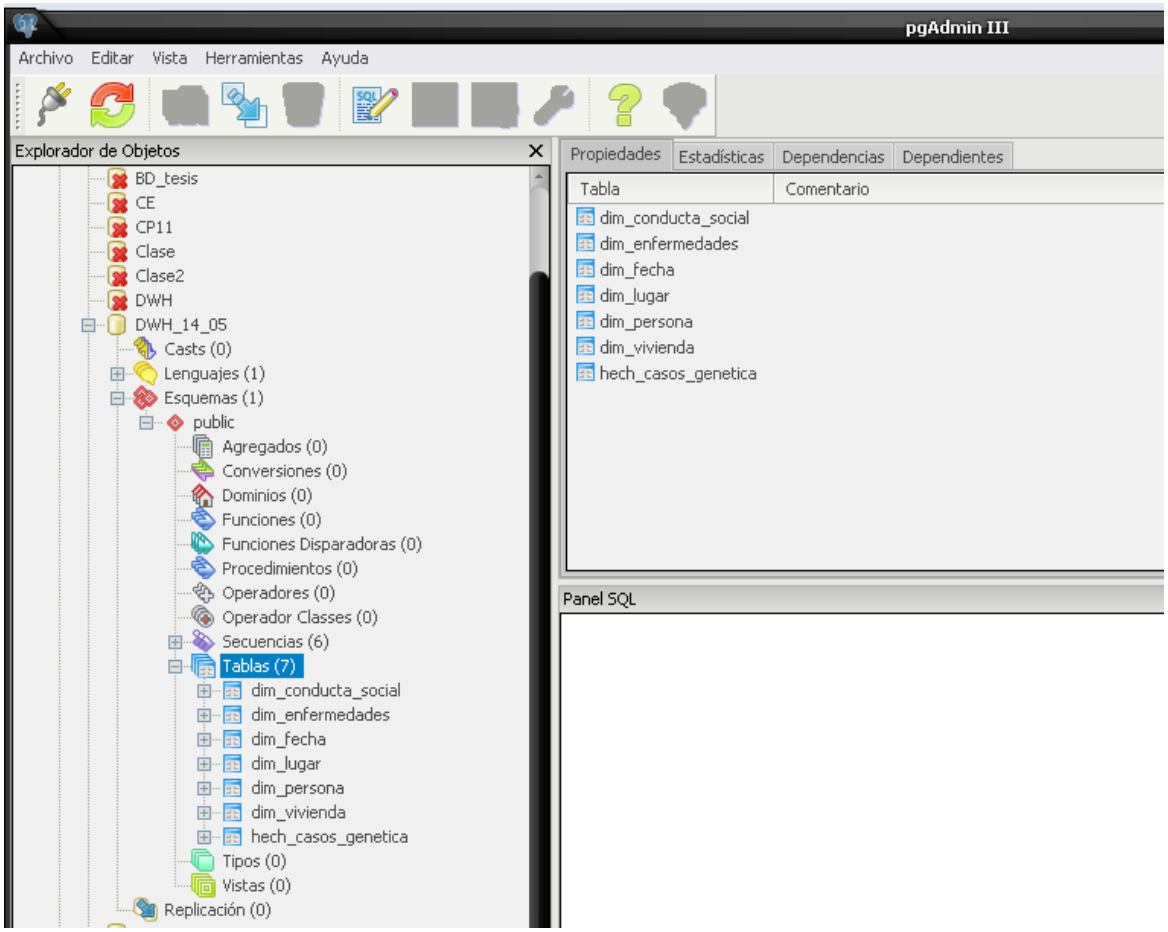


Figura 14. Tablas hechos y dimensiones en el PgAdmin III.

Es importante destacar que en la solución se corrigió la utilización de algunos nombres de atributos en las dimensiones y la corrección de tipos de datos, específicamente en la cantidad de caracteres que soportaban.

Conclusiones del capítulo

En este capítulo se desarrolló el diseño e implementación del datawarehouse siguiendo los pasos de la fase de diseño de la metodología Hefesto, donde se determinaron 6 tablas de dimensiones y 1 tabla de hechos, se definieron las uniones y las jerarquías y luego se desarrolló la implementación de este modelo lógico mediante el SGBD PostgreSQL.

CAPÍTULO 3: ANÁLISIS DE LOS RESULTADOS

En este capítulo se hace énfasis en aspectos como la normalización, las pruebas de volumen y carga, el análisis de los tiempos de respuesta, así como en hacer pruebas para valorar el rendimiento con la concurrencia de usuarios. Este análisis resulta de vital importancia como el diseño y la implementación del propio datawarehouse. La calidad de estos resultados se ve reflejada al transcurrir el tiempo cuando los datos históricos comienzan a aumentar considerablemente, entonces es cuando se necesita observar los beneficios de los tiempos de respuesta, el dinamismo en la elaboración de las consultas, los conocimientos que puedan ser extraídos de la información almacenada y la efectiva preparación de los usuarios finales, para poder garantizar el éxito del datawarehouse.

3.1 Normalización

Un diseño normalizado a menudo almacenará diferentes pero relacionadas piezas de información en tablas lógicas separadas (llamadas relaciones). Si estas relaciones están almacenadas físicamente como archivos de disco separados, puede ser lento terminar una consulta del datawarehouse que tome información de varias relaciones (una operación unión). Si muchas relaciones son unidas, puede ser prohibitivamente lento. En otras palabras, en un datawarehouse normalizado, las estructuras de datos son no redundantes y representan las entidades básicas y las relaciones descritas por los datos (por ejemplo productos, comercio y transacción de ventas). Pero un procesamiento analítico en línea (OLAP) típico de consultas que involucra varias estructuras, requiere varias operaciones de unión para colocar los datos juntos.

Ralph Kimball en su famoso libro The Data Warehouse Toolkit plantea claramente que las tablas dimensionales no tienen que estar normalizadas sino deben permanecer como tablas planas, puesto que las tablas dimensionales normalizadas destruyen la habilidad de la presentación tabulada. Los espacios en disco salvados por la normalización de las tablas dimensionales, son típicamente menores que un porcentaje del espacio total de disco necesario para el esquema completo. Los esfuerzos para normalizar cualquiera de las tablas en un datawarehouse dimensional solamente con el objetivo de salvar espacio en disco, son una pérdida de tiempo.

3.2 Calibración de los Datawarehouse

A partir de un estimado razonable que se realiza en cuanto al tamaño del datawarehouse, se tiene una concepción aproximada de la dimensión espacial que alcanzaría éste. Por tal razón, se realizará un análisis de cada una de las tablas de dimensiones y hechos propuestas para calcular la cantidad de unidades, la cantidad de filas implicadas en cada una de las tablas hasta llegar al número de *bytes* que serán ocupados por concepto de tamaño, teniendo en cuenta que los datawarehouse crecen considerablemente debido al aumento de la información.

Filas aproximadas por cada dimensión:

- Tabla dim_conducta_social: 3
- Tabla Dimensión dim_enfermedades: 9
- Tabla Dimensión dim_fecha: 600
- Tabla Dimensión dim_lugar: 33
- Tabla Dimensión dim_persona: 50 000
- Tabla Dimensión dim_vivienda: 400

Filas aproximadas por cada tabla de hechos:

- Tabla hech_casos_genetica: 50 000

Total de Campos Claves en la tabla de hechos:

- Tabla hech_casos_genetica: 6

Total de Campos Medidas en las tablas de hechos:

- Tabla hech_casos_genetica: 2

Total de Campos en las tablas de dimensiones y de hechos:

- Tabla hech_casos_genetica: 8
- Tabla dim_conducta_social: 2
- Tabla dim_enfermedades: 2
- Tabla dim_fecha: 5
- Tabla dim_lugar: 3

- Tabla dim_persona: 15
- Tabla dim_vivienda: 3

Tamaño de las Tablas de Dimensión y Hecho:

- Tabla hech_casos_genetica: 2968 Kb
- Tabla dim_conducta_social: 8192 bytes
- Tabla dim_enfermedades: 8192 bytes
- Tabla dim_fecha: 40 Kb
- Tabla dim_lugar: 8192 bytes
- Tabla dim_persona: 7 464 Kb
- Tabla dim_vivienda: 24 Kb

3.3 Pruebas y Análisis del Rendimiento

En la ingeniería del software, las pruebas de rendimiento son las pruebas que se realizan desde una perspectiva, para determinar lo rápido que realiza una tarea un sistema en condiciones particulares de trabajo. También pueden servir para validar y verificar otros atributos de la calidad del sistema, tales como fiabilidad y uso de los recursos.

Se puede encontrar un conjunto de varias formas para validar el uso de un sistema informático, como ejemplo de ellas se pueden mencionar: integración, pruebas de unidad, funcionalidad, sistema, volumen, carga, estrés; las que más impactan en el desarrollo de un datawarehouse son las pruebas que tengan relación con el rendimiento, capacidad y concurrencia.

Dentro del desarrollo de los datawarehouse la realización de las pruebas es un paso importante para garantizar el éxito de la solución informática. El mecanismo que se seguirá para ello será la realización de pruebas de volumen y carga las cuales validarán la utilización del datawarehouse. En este punto se analizan los rendimientos del sistema que se ha construido, al dar respuesta a distintos pedidos de información accediendo a la base de datos que se encuentra en el servidor PostgreSQL.

En este análisis se propone determinar el tiempo que demora el usuario en recuperar la información almacenada en el datawarehouse mediante consultas de distintos grados de complejidad, sobre una

cantidad determinada de filas, demostrando cuán óptimo y fácil se recupera la información de las estructuras dimensionales.

Fuente de datos a reportar: Datawarehouse en PostgreSQL.
Tipo de consulta a realizar: Consultas SQL.
Características del Hardware del Servidor:

- Hardware: 512 MB de memoria RAM, 320 Gb de capacidad de disco duro HDD, procesador Intel Pentium IV a 3.00 GHz de velocidad.
- Software: SO Microsoft Windows XP Profesional, PostgreSQL.

3.3.1 Pruebas de Volumen y Carga

La prueba de volumen está sujeta al elemento de prueba a grandes cantidades de datos para determinar si se alcanzan los límites que hacen fallar al software. También identifica la carga máxima continua o volumen que el elemento de prueba puede manejar por un período dado.

La prueba de volumen permite verificar que la aplicación funciona adecuadamente bajo los siguientes escenarios:

- Máximo número de clientes conectados (o simulados), todos ejecutando distintas o la misma función (peor caso de desempeño) por un período extendido.
- Máximo tamaño de base de datos (actual o escalado) y múltiples consultas ejecutadas simultáneamente.

El datawarehouse diseñado se pobló con datos aleatorios generados por una herramienta llamada Data Generator para PostgreSQL. Es una herramienta de software libre que colma el datawarehouse de una cantidad determinada de datos. Se configuró esta generación de datos con valores arbitrarios, pero coincidentes en cuanto a sus tipos y volúmenes con los datos reales que maneja la entidad. El uso de este generador se pudiera considerar una prueba más, ya que si existe inconsistencia en el diseño del datawarehouse, éste no comienza el poblado de datos hasta tanto no quede un correcto diseño.

Al introducir los datos no se presentaron problemas de límite de capacidad, ni se detectaron desbordamientos de matrices, columnas, atributos, tipos de datos, ni peticiones excesivas de memoria.

Las llaves autogeneradas no se salieron del rango especificado, ni se detectaron problemas con los tipos de datos definidos en el paso de diseño. Lo anteriormente planteado garantiza que el SGBD utilizado y el diseño implementado soportan completamente el almacenamiento de los niveles de información requeridos para la puesta en producción del datawarehouse para la red nacional de Genética Médica. En el caso de las tablas dimensiones la herramienta se demoró solo 2 segundos cada 100 datos generados (Ver Anexo 1) y la tabla de hechos se demoró solo 4 segundos cada 100 datos generados (Ver Anexo 2).

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, que realizan un número específico de transacciones durante el tiempo que dura la carga. Esta prueba puede mostrar los tiempos de respuesta de todas las transacciones importantes de la aplicación. El propósito principal de una prueba de carga es simular el acceso de muchos usuarios a un servidor al mismo tiempo.

En resumen, las pruebas de carga consisten en someter a una aplicación y/o base de datos a un régimen de carga de trabajo similar al esperado en la explotación real del sistema. El objetivo de estas pruebas es buscar consultas mal diseñadas, consultas candidatas a optimización, la necesidad de índices adicionales, código mal diseñado, tiempo de demora de respuesta de magnitudes inaceptables, hardware insuficiente, problemas de control de concurrencia.

Para realizar las pruebas de carga se utiliza la herramienta JMeter por la facilidad de su uso y las funcionalidades que brinda. Esta herramienta posee dos tipos de generación de carga, indirecta, es decir, a través de una aplicación y directa que basa fundamentalmente su utilización en consultas grabadas en la traza o log del servidor de base de datos. La que se va a utilizar para las pruebas del datawarehouse es la directa configurada específicamente para la realización de consultas sobre el servidor.

Prueba No.1

- Tablas involucradas: Tabla de hech_casos_genetica.
- Objetivo de la consulta: Obtener todos los datos almacenados en la tabla hech_casos_genetica.
- Consulta: SELECT * FROM hech_casos_genetica.
- Cantidad de Usuarios:

5 usuarios concurrentes

	Media (seg)	Mediana (seg)	Min (seg)	Max (seg)
Resultados	3,73	3,39	3,00	4,70

10 usuarios concurrentes

	Media (seg)	Mediana (seg)	Min (seg)	Max (seg)
Resultados	8,38	8,50	3,78	10,78

Gráfico de relación entre las pruebas.

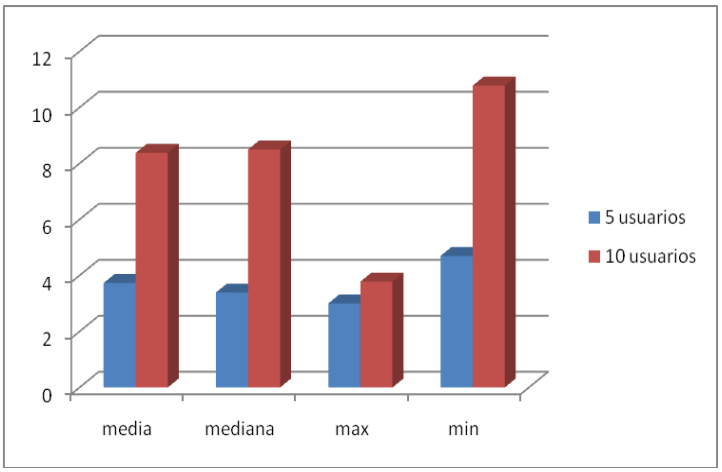


Gráfico 1. Representación de la Prueba 1

Prueba No.2

- Tablas involucradas: Tabla de hech_casos_genetica.
- Objetivo de la consulta: Obtener todos los datos de la tabla hech_casos_genetica que sean del municipio Santiago de Cuba y tienen como identificador el número 1.
- Consulta: SELECT * FROM hech_casos_genetica WHERE hech_casos_genetica."id_Lugar" = 1.
- Cantidad de Usuarios:

5 usuarios concurrentes

	Media (seg)	Mediana (seg)	Min (seg)	Max (seg)
Resultados	0,06	0,06	0,04	0,07

10 usuarios concurrentes

	Media (seg)	Mediana (seg)	Min (seg)	Max (seg)
Resultados	0,06	0,06	0,06	0,06

Gráfico de relación entre las pruebas.

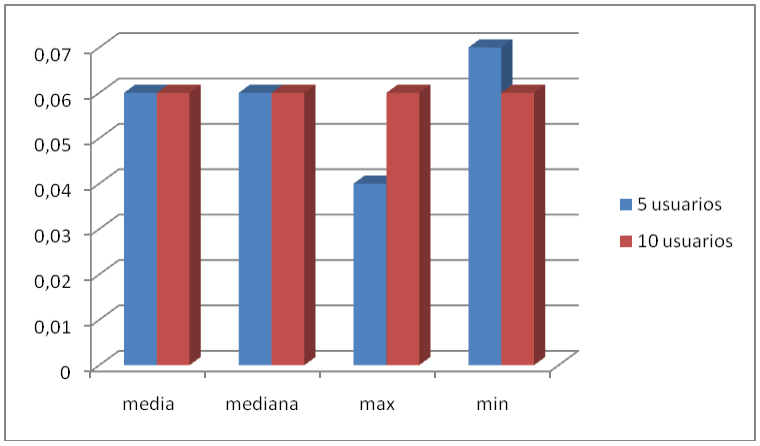


Gráfico 2. Representación de la Prueba 2

Prueba No.3

- Tablas involucradas: Tabla de dim_enfermedades, Tabla de dim_persona, Tabla de hech_casos_genetica.
- Objetivo de la consulta: Obtener la cantidad de gemelos con enfermedad de Cáncer.
- Consulta: SELECT count(hech_casos_genetica."id_Persona") FROM public.dim_persona INNER JOIN public.hech_casos_genetica ON (public.dim_persona."id_Persona"= public.hech_casos_genetica."id_Persona") INNER JOIN public.dim_enfermedades ON (public.hech_casos_genetica."id_Enfermedades"= public.dim_enfermedades."id_Enfermedades") WHERE public.dim_persona.tipo = 'gemelo' AND dim_enfermedades.enfermedad='cancer'.
- Cantidad de Usuarios:

5 usuarios concurrentes

	Media (seg)	Mediana (seg)	Min (seg)	Max (seg)
Resultados	0,03	0,03	0,01	0,04

10 usuarios concurrentes

	Media (seg)	Mediana (seg)	Min (seg)	Max (seg)
Resultados	0,03	0,03	0,01	0,04

Gráfico de relación entre las pruebas.

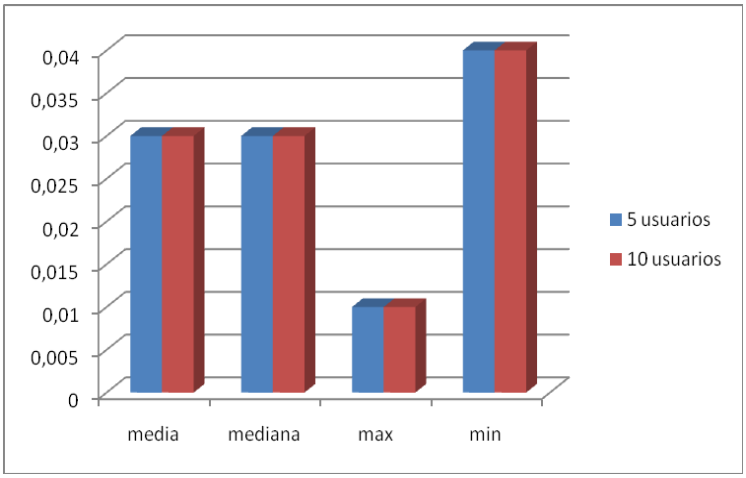


Gráfico 3. Representación de la Prueba 3

Prueba No.4

- Tablas involucradas: Tabla de dim_enfermedades, Tabla de dim_persona, Tabla de hech_casos_genetica.
- Objetivo de la consulta: Obtener la cantidad de personas por cada tipo.
- Consulta: SELECT public.dim_persona.tipo, count(public.hech_casos_genetica."id_Persona") AS "Cantidad" FROM public.dim_persona INNER JOIN public.hech_casos_genetica ON (public.dim_persona."id_Persona" = public.hech_casos_genetica."id_Persona") INNER JOIN public.dim_enfermedades ON (public.hech_casos_genetica."id_Enfermedades"= public.dim_enfermedades."id_Enfermedades") GROUP BY public.dim_persona.tipo.
- Cantidad de Usuarios:

5 usuarios concurrentes

	Media (seg)	Mediana (seg)	Min (seg)	Max (seg)
Resultados	0,81	0,71	0,60	1,17

10 usuarios concurrentes

	Media (seg)	Mediana (seg)	Min (seg)	Max (seg)
Resultados	1,94	2,18	0,40	2,60

Gráfico de relación entre las pruebas.

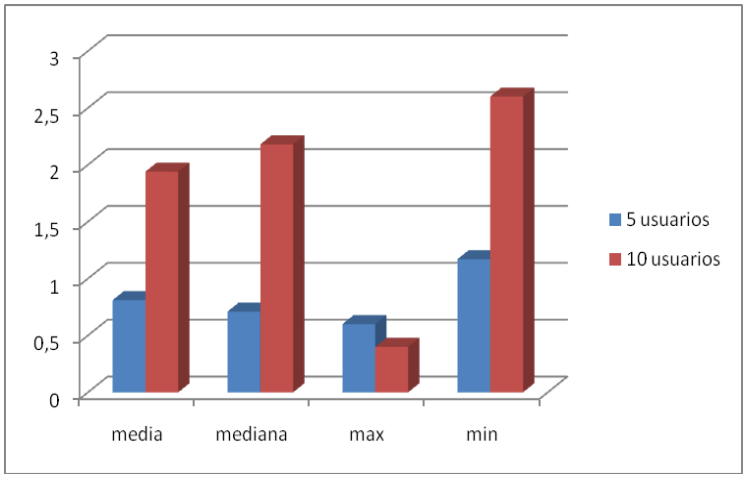


Gráfico 4. Representación de la Prueba 4

Prueba No.5

- Tablas involucradas: Tabla de dim_persona
- Objetivo de la consulta: Obtener el identificador de las personas mayores de 5 años que tienen familiares adictos.
- Consulta: SELECT public."dim_persona"."id_Persona" FROM public."dim_persona" WHERE "dim_persona".familiares_adictos = true AND "dim_persona".edad > 5.
- Cantidad de Usuarios:

10 usuarios concurrentes

	Media (seg)	Mediana (seg)	Min (seg)	Max (seg)
Resultados	0,27	0,29	0,10	0,48

15 usuarios concurrentes

	Media (seg)	Mediana (seg)	Min (seg)	Max (seg)
Resultados	0,57	0,51	0,29	1,06

Gráfico de relación entre las pruebas.

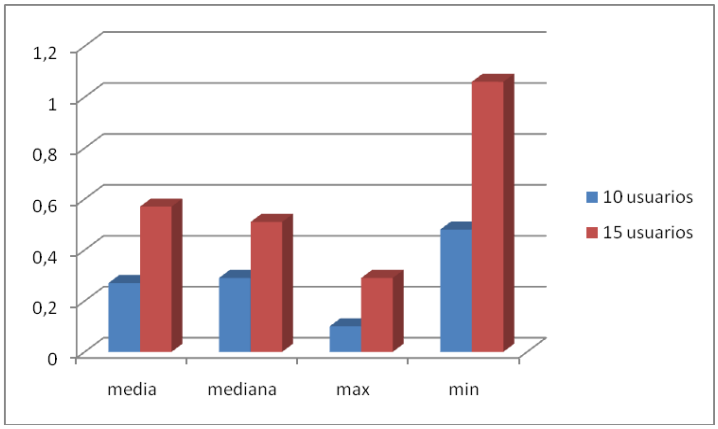


Gráfico 5. Representación de la Prueba 5

Prueba No.6

- Tablas involucradas: Tabla de dim_fecha, Tabla de hech_casos_genetica.
- Objetivo de la consulta: Obtener el identificador de las personas discapacitadas que se han registrado en el año 2010.
- Consulta: SELECT public.hech_casos_genetica."id_Persona" FROM public.hech_casos_genetica INNER JOIN public.dim_fecha ON (public.hech_casos_genetica."id_Fecha" = public.dim_fecha."id_Fecha") INNER JOIN public.dim_persona ON (public.hech_casos_genetica."id_Persona" = public.dim_persona."id_Persona") WHERE public.dim_fecha.anno = '2010' AND public.dim_persona.tipo = 'Discapacitado'.

- Cantidad de Usuarios:

10 usuarios concurrentes

	Media (seg)	Mediana (seg)	Min (seg)	Max (seg)
Resultados	0,42	0,42	0,23	0,67

15 usuarios concurrentes

	Media (seg)	Mediana (seg)	Min (seg)	Max (seg)
Resultados	0,94	0,99	0,18	1,67

Gráfico de relación entre las pruebas.

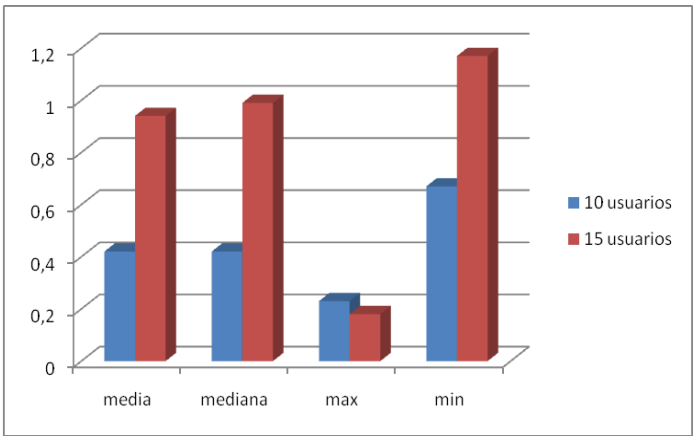


Gráfico 6. Representación de la Prueba 6

Prueba No.7

- Tablas involucradas: Tabla dim_vivienda, Tabla de dim_persona, Tabla de hech_casos_genetica.
- Objetivo de la consulta: Obtener el identificador de de la personas que conviven en sus viviendas con más de 7 personas.
- Consulta: SELECT public.dim_persona."id_Persona" FROM public.hech_casos_genetica

INNER JOIN public.dim_vivienda ON (public.hech_casos_genetica."id_Vivienda" = public.dim_vivienda."id_Vivienda") INNER JOIN public.dim_persona ON (public.hech_casos_genetica."id_Persona" = public.dim_persona."id_Persona") WHERE public.dim_vivienda.nucleo_familiar > 7.

- Cantidad de Usuarios:

10 usuarios concurrentes

	Media (seg)	Mediana (seg)	Mín (seg)	Max (seg)
Resultados	0,40	0,39	0,18	0,62

15 usuarios concurrentes

	Media (seg)	Mediana (seg)	Mín (seg)	Max (seg)
Resultados	0,65	0,68	0,23	1,03

Gráfico de relación entre las pruebas.

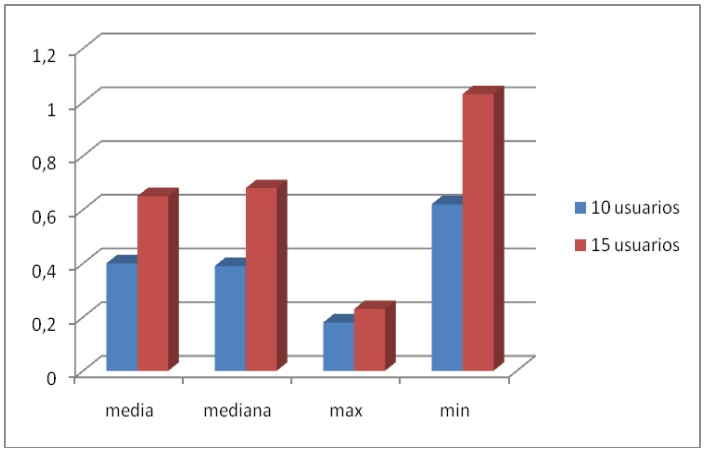


Gráfico 7. Representación de la Prueba 7

Prueba No.8

- Tablas involucradas: Tabla dim_conducta_social, Tabla hech_casos_genetica
- Objetivo de la consulta: Obtener el identificador y conducta social de todas las personas que sean agresivos.

- Consulta: SELECT public."hech_casos_genetica"."id_Persona", public.dim_conducta_social.conducta FROM public."dim_conducta_social" INNER JOIN public."hech_casos_genetica" ON (public."dim_conducta_social"."id_Conducta_Social"= public."hech_casos_genetica"."id_Conducta_Social") WHERE "dim_conducta_social".conducta = 'agresivo'.
- Cantidad de Usuarios:

5 usuarios concurrentes

	Media (seg)	Mediana (seg)	Min (seg)	Max (seg)
Resultados	0,22	0,23	0,20	0,25

10 usuarios concurrentes

	Media (seg)	Mediana (seg)	Min (seg)	Max (seg)
Resultados	0,74	0,73	0,35	1,20

Gráfico de relación entre las pruebas.

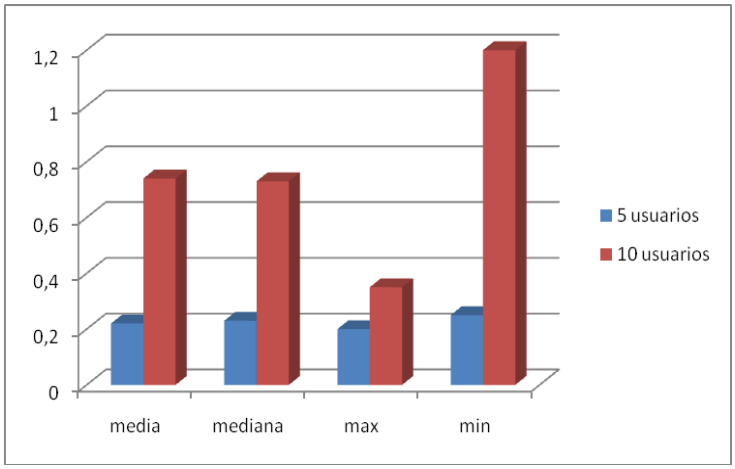


Gráfico 8. Representación de la Prueba 8

Como puede apreciarse los resultados se enmarcan en 4 variables aportadas por la herramienta JMeter, de significativo valor para la presentación de los resultados de las pruebas: la media, valor de la suma

aritmética de los tiempos de respuesta dividido entre 2; la mediana, valor de la variable que deja el mismo número de datos antes y después que él, una vez ordenados estos, de acuerdo con esta definición el conjunto de datos menores o iguales que la mediana representará el 50% de los datos, los restantes que sean mayores que la mediana representarán el otro 50% del total de datos de la muestra; el valor mínimo, que se refiere al tiempo de respuesta menor de todos los usuarios que hicieron peticiones concurrentes y el valor máximo que, similarmente, es el tiempo mayor de respuesta a todos los usuarios.

En las gráficas anteriores, los tiempos de respuestas oscilan dependiendo de la cantidad de filas que se recuperen en la consulta y la cantidad de tablas a la que se accede para dar respuesta a éstas; las pruebas 1, 4 y 6 son las que mayor cantidad de datos devuelven, por esa razón sus tiempos de respuestas son más elevados que los restantes. Por otra parte se encuentran las consultas que acceden a varias tablas para dar respuesta a las peticiones, es importante resaltar que si la consulta accede a la tabla de hechos, el tiempo de respuesta será considerablemente mayor que una consulta que acceda solo a las tablas de dimensiones, como es el caso de la prueba 4 y 5, la primera involucra la tabla de hechos y dos tablas de dimensiones y la otra sólo una tabla de dimensiones, por esa razón existe una gran diferencia entre los tiempos de respuestas siendo mayores los de la prueba 4. Otro aspecto a tener en cuenta es la cantidad de usuarios que acceden al datawarehouse ya que los tiempos de respuestas para 5 usuarios son mucho menores que los de 15 usuarios.

En general los resultados obtenidos son satisfactorios, destacándose los mayores tiempos de respuestas en la consulta 1 ya que ésta devuelve mayor cantidad de datos que las restantes consultas debido a que su objetivo es obtener todos los valores almacenados en la tabla de hechos dim_casos_genetica convirtiéndose en un proceso un poco más lento.

Conclusiones del capítulo

En este capítulo se realizaron las pruebas de rendimiento, las cuales garantizaron una evolución de las estructuras creadas en función de mantener los niveles de servicio a partir de los requerimientos de almacenamiento. Las pruebas de volumen validaron la infraestructura de hardware y software propuestas, garantizando la capacidad de gestión de los datos almacenados. Las pruebas de carga resultaron un elemento fundamental en el proceso de optimización y demostraron que los tiempos de respuestas fueron aceptables.

CONCLUSIONES

La investigación cumplió los objetivos planteados arribando así a las siguientes conclusiones:

- Se obtuvo el diseño del modelo lógico de la estructura del datawarehouse para la red nacional de Genética Médica guiado por la secuencia de pasos de la metodología Hefesto.
- La implementación del modelo multidimensional en el SGDB proporcionó un datawarehouse listo para almacenar los datos que necesita la red nacional de Genética Médica.
- Las pruebas realizadas permitieron validar la solución propuesta obteniendo resultados satisfactorios en cada una de ellas.

RECOMENDACIONES

Con el propósito de enriquecer la propuesta realizada en este trabajo, se recomienda:

- Realizar el proceso ETL utilizando la herramienta Pentaho Data Integration.
- A partir de las necesidades presentadas se recomienda después que se haya realizado el proceso ELT, desarrollar el proceso de consulta y análisis de los datos para que finalmente el datawarehouse brinde los servicios necesarios a los genetistas que conforman la red nacional de Genética Médica.

REFERENCIAS BIBLIOGRÁFICAS

1. Centro Nacional de Genética Médica. [En línea] 9 de Noviembre de 2009. http://www.sld.cu/sitios/genetica/.

2. Granma Internacional Digital. [En línea] 12 de Noviembre de 2009. http://www.granma.cu/espanol/2009/octubre/lun12/insolito-genetica-medica.html.

3. **Febles, Juan Pedro.** Importancia de la utilización de un Data Warehouse (DW) en las empresas. [En línea] 3 de Febrero de 2010. http://www.bibliociencias.cu/gsdl/collect/libros/index/assoc/HASH0106/b6fac6b9.dir/doc.pdf.

4. **Herrera, Cristhian.** Todo lo que querias saber sobre Data Warehouse (I). [En línea] 3 de Febrero de 2010.http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=datawarehouse#2.4.Caracter%C3%ADsticas%20del%20Datawarehouse|outline.

5. Manual de Construcción de un Data Warehouse. [En línea] 3 de Febrero de 2010. http://www.ongei.gob.pe/publica/metodologias/Lib5084/14.HTM.

6. Data Warehousing, SQL Server. [En línea] 3 de Febrero de 2010. http://www.sqlmax.com/dataw1.asp.

7. Empresas que utilizan Data Ware House. [En línea] 4 de Febrero de 2010. http://www.navactiva.com/web/es/atic/aseso/desarrollo/asesor1/2005/60564.php?fecha=2010-04.

8. Caracteristias/Dataprix. [En línea] 1 de Marzo de 2010. http://www.dataprix.com/data-warehousing-y-metodologia-hefesto/-metodologia-hefesto/53-caracteristicas.

9. DBDesigner y MySql. [En línea] 12 de Marzo de 2010. http://webamedida.net/index.php?option=com_content&task=view&id=1&Itemid=4.

10. 4.3 SGBD/Dataprix. [En línea] 12 de Marzo de 2010. http://www.dataprix.com/data-warehousing-y-metodologia-hefesto/i-data-warehousing-investigacion-y-sistematizacion-concepto-14.

11. PostGreSGL. [En línea] 13 de Marzo de 2010. http://www.netpecos.org/docs/mysql_postgres/index.html.

BIBLIOGRAFÍA

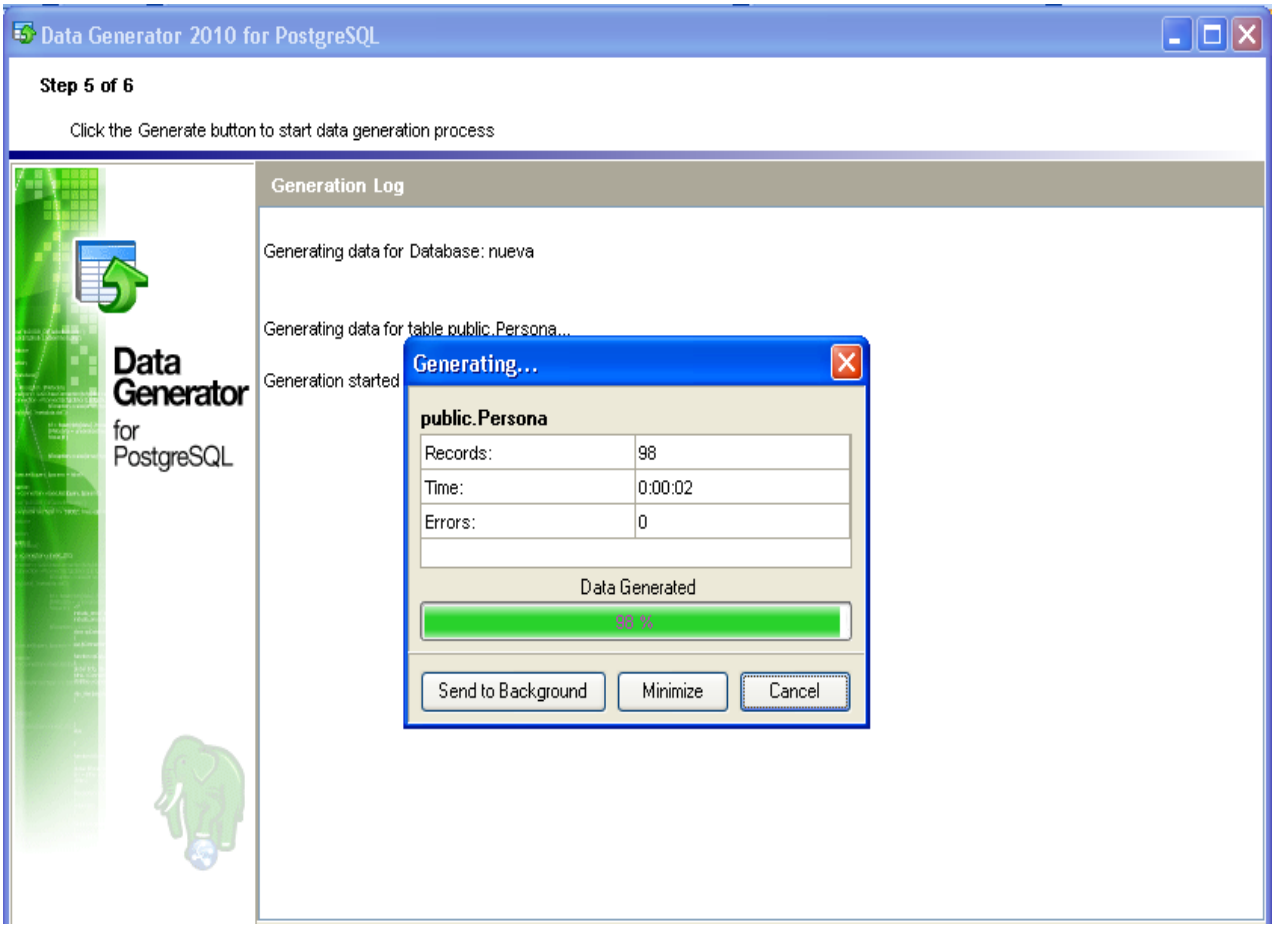
1. Centro Nacional de Genética Médica. [En línea] 9 de Noviembre de 2009.http://www.sld.cu/sitios/genetica/.
2. Granma Internacional Digital. [En línea] 12 de Noviembre de 2009. http://www.granma.cu/espanol/2009/octubre/lun12/insolito-genetica-medica.html.
3. **Febles, Juan Pedro.** Importancia de la utilización de un Data Warehouse (DW) en las empresas.[En línea] 3 de Febrero de 2010. http://www.bibliociencias.cu/gsdl/collect/libros/index/assoc/HASH0106/b6fac6b9.dir/doc.pdf.
4. **Herrera, Cristhian.** Todo lo que querias saber sobre Data Warehouse (I). [En línea] 3 de Febrero de 2010.http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=datawarehouse#2.4.Caracter%C3%ADsticas%20del%20Datawarehouse|outline.
5. Manual de Construcción de un Data Warehouse. [En línea] 3 de Febrero de 2010. http://www.ongei.gob.pe/publica/metodologias/Lib5084/14.HTM.
6. Data Warehousing, SQL Server. [En línea] 3 de Febrero de 2010. http://www.sqlmax.com/dataw1.asp.
7. Empresas que utilizan Data Ware House. [En línea] 4 de Febrero de 2010.http://www.navactiva.com/web/es/atic/aseso/desarrollo/asesor1/2005/60564.php?fecha=2010-04.
8. Caracteristias/Dataprix. [En línea] 1 de Marzo de 2010. http://www.dataprix.com/data-warehousing-y-metodologia-hefesto/-metodologia-hefesto/53-caracteristicas.
9. DBDesigner y MySql. [En línea] 12 de Marzo de 2010. http://webamedida.net/index.php?option=com_content&task=view&id=1&Itemid=4.
10. 4.3 SGBD/Dataprix. [En línea] 12 de Marzo de 2010. http://www.dataprix.com/data-warehousing-y-metodologia-hefesto/i-data-warehousing-investigacion-y-sistematizacion-concepto-14.
11. PostGreSGL. [En línea] 13 de Marzo de 2010. http://www.netpecos.org/docs/mysql_postgres/index.html.
12. Data Warehousing y metodología Hefesto . [En línea] 11 de Marzo de 2010. http://www.dataprix.com/es/data-warehousing-hefesto.
13. Manual de metodología HEFESTO para Datawarehousing. [En línea] 11 de Marzo de 2010. http://www.dataprix.com/es/manual-metodologia-hefesto-para-datawarehousing.

14. **Bernabeu, Darío.** Sistematización de conceptos y metodología HEFESTO para la construcción de un Data Warehouse. [En línea] 14 de Marzo de 2010. http://www.bi-spain.com/articulo/56125/data-warehouse/sistematizacion-de-conceptos-y-metodologia-hefesto-para-la-construccion-de-u.
15. **Mejía, Daniel.** Aplicaciones de Ingeniería de Software. [En línea] 8 de Mayo de 2010. http://delfin.mxl.uabc.mx/~angelica/Pruebas.pdf.
16. Almacén de datos. [En línea] 23 de Octubre de 2009. http://etl-tools.info/es/bi/almacendedatos_arquitectura.htm.
17. Almacenes de datos. [En línea] 23 de Octubre de 2009. http://www.rhernando.net/modules/tutorials/doc/bd/dw.html.
18. Beneficios del Data Warehouse - El Nuevo Diario - Managua, Nicaragua. [En línea] 6 de Noviembre de 2009. http://impreso.elnuevodiario.com.ni/2005/07/11/informatica/39878 .
19. Bases de datos en castellano. Data Warehousing bases de datos . [En línea] 12 de Enero de 2010. http://www.programacion.com/bbdd/tutorial/warehouse/.
20. Data Warehouse - Manual para la Construcción. [En línea] 16 de Marzo de 2010. http://www.elprisma.com/apuntes/curso.asp?id=5132.
21. Análisis de Dimensiones y Hechos. Modelo Lógico. [En línea] 16 de Marzo de 2010. http://churriwifi.wordpress.com/2010/04/22/15-3-analisis-dimensiones-hechos/.
22. **Garía, María Isabel Guzmán.** *Implementación de un datawarehouse para el soporte de toma de decisiones.* Guatemala : s.n., 2001.
23. **Mateos, Alberto Oliva.** *Aplicación de Seguridad en Servicios Web XML para dispositivos móviles mediante la implementación de un perfil SAML.Tomo II.* 2006.
24. Modelo multidimensional. [En línea] 14 de Marzo de 2010. http://www.inf.udec.cl/revista/edicion4/cwolff.htm.
25. **Prieto, José Abásolo.** Powerpoints de Modelo multidimensional en base de datos gratis. [En línea] 15 de Marzo de 2010. http://www.acis.org.co/fileadmin/Base_de_Conocimiento/XXV_Salon_de_Informatica/IntegracionDatos - JoseAbasolo.ppt.
26. Conceptos básicos del diseño de una base de datos. [En línea] 16 de Marzo de 2010. http://office.microsoft.com/es-es/access/HA012242473082.aspx.
27. **Eduardo.** Normalizacion de base de datos. [En línea] 3 de Mayo de 2010. http://www.mysql-hispano.org/page.php?id=16.

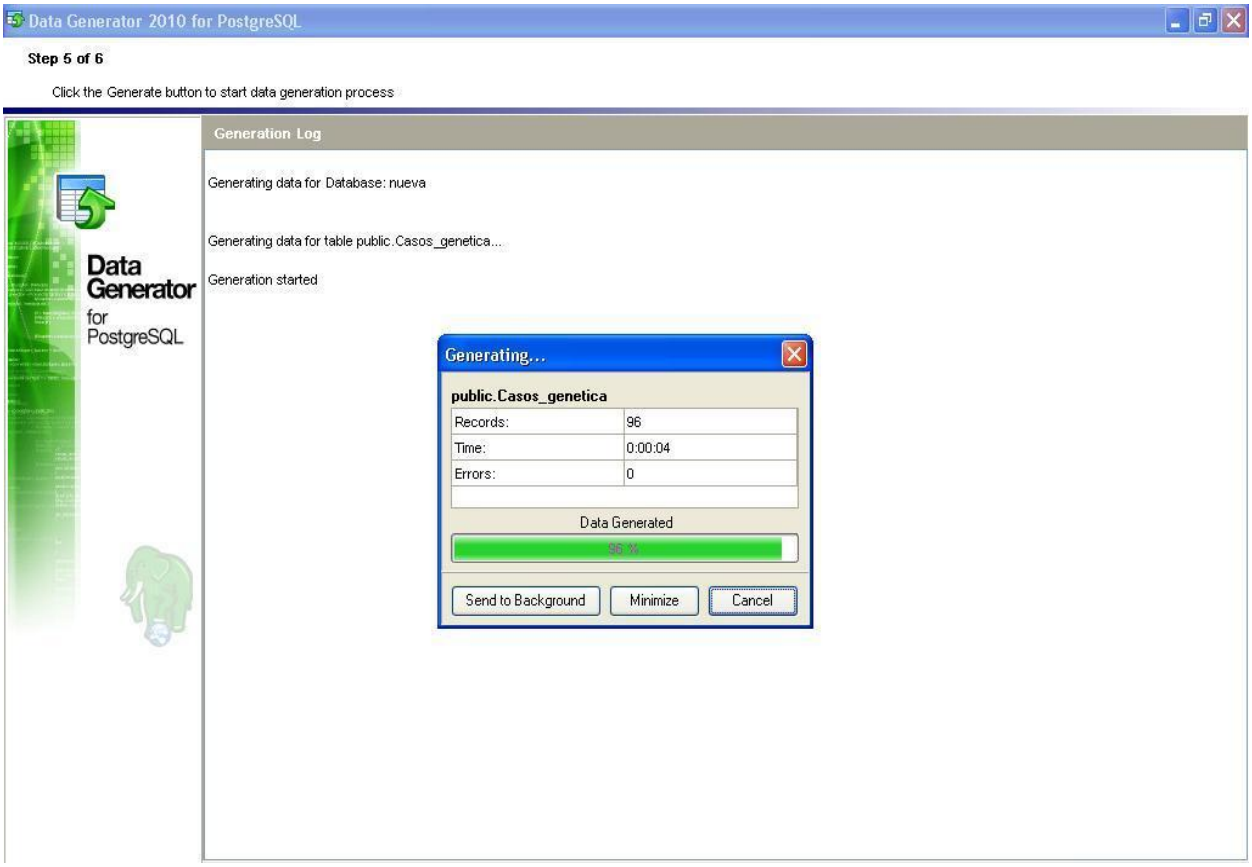
28. Normalizacion de Base de Datos. [En línea] 4 de Mayo de 2010. http://www.trucostecnicos.com/trucos/ver.php?id_art=278.
29. Definición de Pruebas de rendimiento/ Definicion-es.com. [En línea] 4 de Mayo de 2010. http://www.atrsoft.com/esl/content/view/full/348.

ANEXO

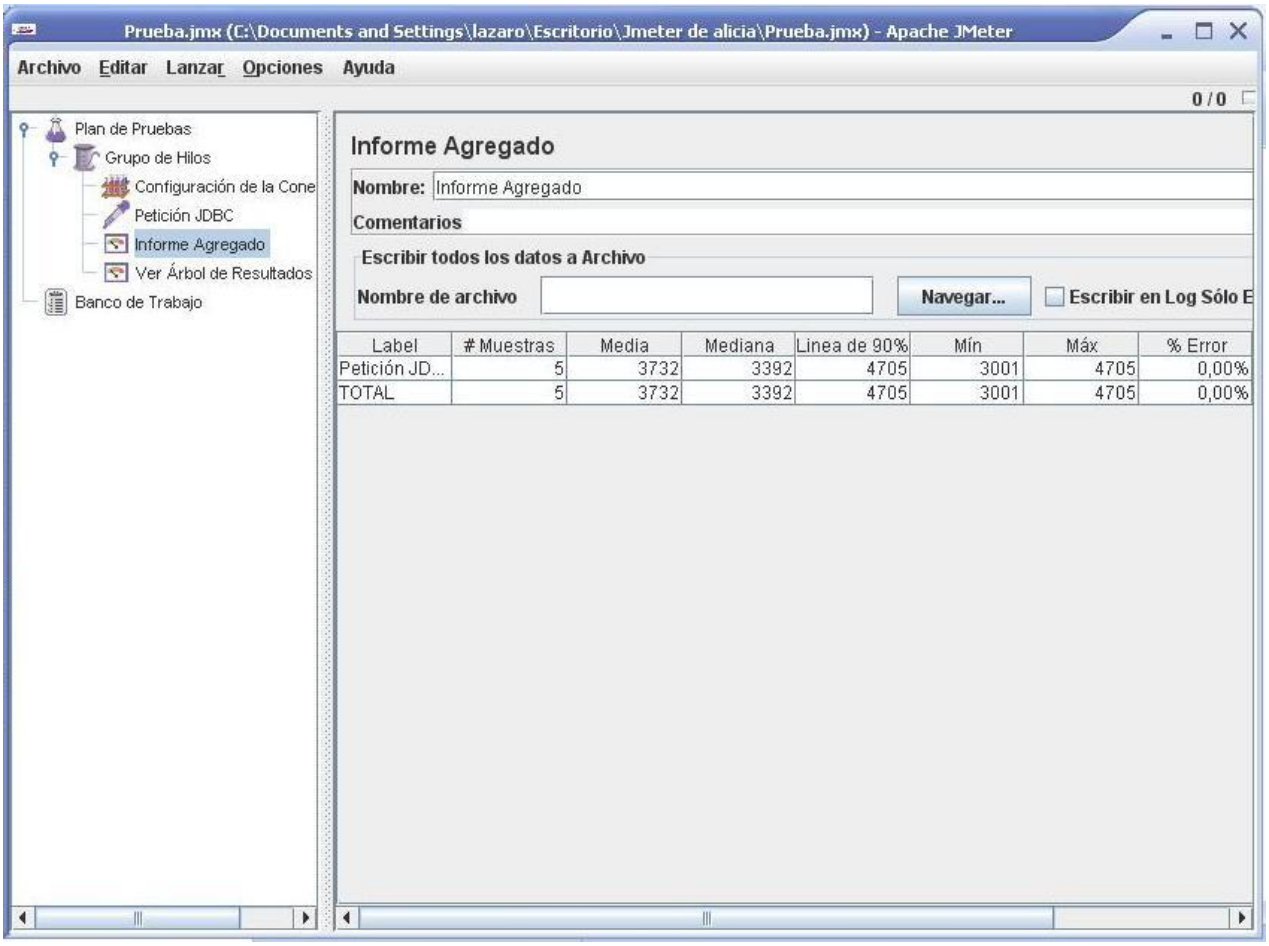
Anexo 1 Resultado de las Pruebas de Volumen para las tablas de dimensiones.



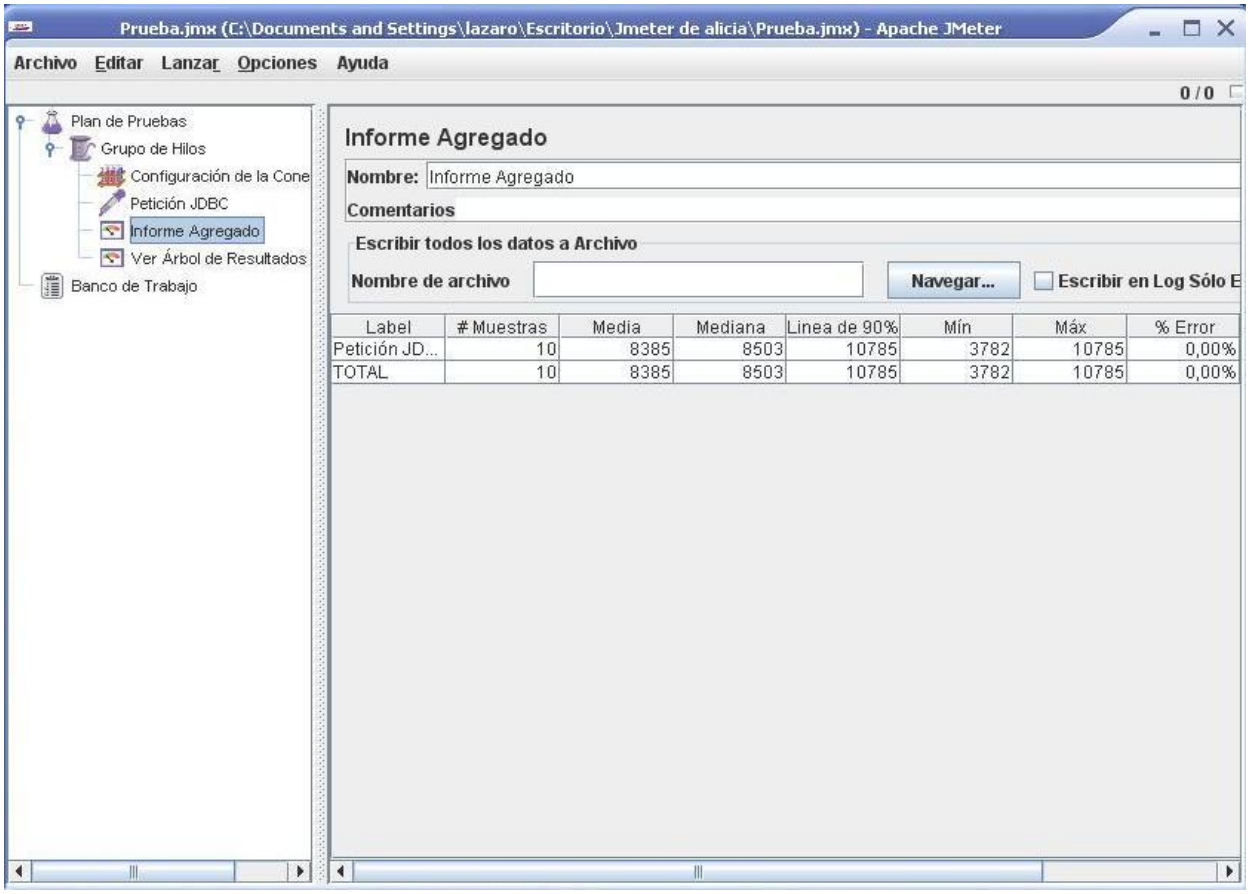
Anexo 2 Resultado de las Pruebas de Volumen para las tablas de hechos



Anexo 3 Resultado de las Pruebas de Carga (Prueba #1 con 5 usuarios)



Anexo 4 Resultado de las Pruebas de Carga (Prueba #1 con 10 usuarios)



GLOSARIO DE TÉRMINOS

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

Hardware: conjunto de elementos materiales que componen un ordenador. En dicho conjunto se incluyen los dispositivos electrónicos y electromecánicos, circuitos, cables, tarjetas, periféricos de todo tipo y otros elementos físicos.

Data mart: es una base de datos departamental, especializada en el almacenamiento de los datos de un área de negocio específica. Se caracteriza por disponer la estructura óptima de datos para analizar la información al detalle desde todas las perspectivas que afecten a los procesos de dicho.

Inteligencia de negocio: altamente relacionado a la buena planeación y estrategia comercial de cualquier organización de cualquier índole y tamaño. Se refiere al uso de los datos de una empresa para facilitar la toma de decisiones mediante la comprensión del funcionamiento actual y la anticipación de acciones para dar una dirección operativa óptima a la empresa.

Batch: característica típica de ciertos procesos, que indica una serie de tareas que se ejecutan de forma sucesiva en el ordenador y se consideran como una unidad.

Base de datos: la base de datos es un conjunto de información que está almacenada en forma sistemática, de manera tal que los datos que la conforman puedan ser utilizados en forma fragmentada cuando sea necesario.

Open Source: es el término con el que se conoce al software distribuido y desarrollado libremente. El código abierto tiene un punto de vista más orientado a los beneficios prácticos de compartir el código que a las cuestiones morales y/o filosóficas las cuales destacan en el llamado software libre.

OLAP: Procesamiento analítico en línea.

MOLAP: Procesamiento analítico multidimensional en línea.

ROLAP: Procesamiento analítico relacional en línea.

HOLAP: Procesamiento analítico híbrido en línea.

OLTP: Procesamiento transaccional en línea.

ETL: Extracción, Transformación y Carga de datos.

SGBD: Sistema Gestor de Bases de Datos.