

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS



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

Propuesta de un Data Mart para el análisis de la deuda externa en el Banco Nacional de Cuba

Autor(es): Néstor García Sánchez
Angel Luis Batista Santiesteban

Tutor(es): Ing. Yanetsi Mestre Morales
Ing. Erick Fleitas Chang

Ciudad de la Habana, Junio de 2010

FRASE

“El aspecto fundamental en el cual la juventud debe señalar el camino es precisamente en el aspecto de ser vanguardia en cada uno de los trabajos que le compete.”

A handwritten signature in black ink, appearing to be 'le' with a small dash below it.

DECLARACION DE AUTORIA

Declaramos que somos los únicos autores de este trabajo titulado: Propuesta de un Data Mart para el análisis de la deuda externa en el Banco Nacional de Cuba y autorizamos a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año_____.

Angel Luis Batista Santiesteban
Autor

Néstor García Sánchez
Autor

Ing. Yanetsi Mestre Morales
Tutora

Ing. Erick Fleitas Chang
Tutor

AGRADECIMIENTOS

Néstor:

Primeramente agradecer a mis padres Martha y Octavio y a mi hermano Yusvani por el amor, la amistad, la cohesión y por estar a mi lado en todos los momentos de mi vida.

A mi familia, en especial a tía Nancy y a mi prima Nélide que siempre me han brindado unidad, valores y principios éticos que ha servido para mi formación como persona.

A mi compañero de trabajo Angel Luis Batista por su inconmensurable tesón frente a las tareas de la tesis y por ser un amigo entrañable.

A mis hermanos de crianza Osvaldo, Yasmani y Maily por ser como una familia para mi.

A mis grandes amigos Armín, José Antonio (Pepex) y Yohairo por la entrega incondicional de su amistad hacia mi persona.

A mis compañeros del preuniversitario Fabián, Alberto, Alejandro, Jordán, y Alain, a la pandilla de Cojímar y en especial Yasel, Fernando, Lázaro, Bauza y Toni.

A todas las personas que me han proporcionado apoyo y solidaridad en el transcurso de estos cinco años de carrera.

A nuestros tutores Yanetsi Mestre Morales y Erick Fleitas Chang por brindarnos su ayuda y apoyarnos en todo momento.

A la revolución y a nuestro comandante por confiar en mí y por poner en nuestras manos toda la tecnología con la que se pudo llevar a cabo el presente trabajo.

Angel:

Agradezco a todas las personas (familiares y amigos) que de una forma u otra contribuyeron con mi formación, especialmente:

El mayor agradecimiento es a mi mamá (Madelayne Santiesteban) por ser mi ejemplo a seguir, por confiar siempre en mí y guiarme en todo momento, por impulsarme cada vez a ser mejor, a esforzarme y a dar el máximo para superarme, gracias por todo su apoyo y sacrificio.

A mis abuelos, especialmente a mi abuela Luz María y mi abuelo Oliberto, a mi tía Damaris por tenerme como un hijo más para ella, en general mis dos familias por tanta preocupación y apoyo sentimental y fraternal.

A mis tíos y primos (Albertico, Pipo, Luisito, Adriana, Leo) por brindarme su confianza, por darme fuerzas y ayudarme cuando lo necesite y a los cuales agradezco los momentos compartidos.

A mi hermano Rubencito por tenerme siempre presente y velar por mí, por quererme tanto, por querer seguir mis pasos, por ser mi razón de ser y al cual quiero con la vida.

A mi padrastro, a Inés y a Elia el cual ya no se encuentra con nosotros, gracias por brindarme su ayuda y por poner fe en mí para lograr este sueño.

A mi amigo, hermano y compañero de tesis Néstor García, te doy las gracias por tu dedicación, comprensión y paciencia para soportarme todo este tiempo, sinceramente te agradezco por confiar en mí para hacer este trabajo, el cual es el mayor sueño como estudiante.

A mis amigos y hermanos Adriel, Yandy y Carlos por confiar en mí, por brindarme su apoyo en los buenos y malos momentos, porque en 5 años de universidad hemos hecho una gran amistad que quedará para siempre.

Agradecerle a la familia de Yandy que me acogió en su casa como un integrante más y la cual me ayudó, me dio fuerzas y me aconsejó cuando lo necesité.

A todos mis compañeros y amigos, en especial a Dalgys, Lismary, Taimé, Magda, Surima, Jenny, Yarida, Daynelis, Javier, Yohairo, Iris, Armín, Elide, Pepe (Otilio), Pepex (José Antonio), José A. Brizuela, Reynal, Alain y Roberto por permitirnos disfrutar al máximo estos maravillosos años juntos, por ser partícipes de nuestros logros, desconciertos, malos y buenos momentos, marcando una etapa inolvidable en nuestras vidas. En fin, por ser protagonistas de mis mejores recuerdos durante toda la Universidad.

A mis compañeros del barrio y preuniversitario Rolandito, Luis A, Leonela, Yelena, Yamila, Raciél, Liudmila, a mi primo y hermano Yordanis y a mis hermanos Yinet y Yoel.

A nuestra tutora Yanetsi Mestre Morales y Erick Fleitas Chang por ayudarnos y apoyarnos durante el desarrollo de la tesis, muchas gracias.

A René por ayudarme en cada momento con el desarrollo de la tesis.

A la revolución y a nuestro comandante por crear este proyecto futuro y por poner en nuestras manos toda la tecnología con la que se pudo llevar a cabo el presente trabajo.

En fin a todas esas personas que me hicieron esforzar para vencer los obstáculos. A todos muchas gracias por haber ocupado un espacio de mi vida y carrera.

DEDICATORIA

Queremos dedicarles el presente trabajo de manera extraordinaria a nuestras familias y en especial a nuestros padres.

RESUMEN

En el presente trabajo de diploma se tiene como objetivo desarrollar un Data Mart para consolidar la información relacionada con el Banco Nacional de Cuba (BNC), específicamente sobre las Deudas externas. El resultado de la implementación del data mart viene como consecuencia a dificultades en la toma de decisiones alrededor del análisis de la deuda externa en la estructura de datos del SABIC (Sistema automatizado para la Banca Internacional de Comercio). El Data Mart pondrá en manos de los directivos de manera organizada una gran cantidad de información que se maneja diariamente en el Banco Nacional de Cuba. El almacén de datos está confeccionado mediante la metodología DWEP, utilizando herramientas que brinda la Plataforma Pentaho y en el mismo se podrá acceder a información de todo lo que esté relacionado con una deuda, esto va desde la cantidad de presupuesto que se pueda contraer, hasta el país con la que se contrae y dentro de este se podrá saber que banco contrajo la deuda, en que moneda se hizo el pedido y el día, el mes, el trimestre, el semestre y el año en el que se efectuó la deuda.

Palabras Claves

Data Mart, Deudas, Sabic, DWEP, Pentaho

ÍNDICE DE CONTENIDO

INTRODUCCIÓN.....	1
Capítulo 1. Conceptos, herramientas y metodologías para el diseño del Data Mart	4
1.1 Introducción.....	4
1.2 Data Warehouse.....	4
1.2.1. Objetivos fundamentales de un Data Warehouse.....	6
1.2.2. Data Warehouse y Data Mart.....	7
1.2.3 Métodos más usados en la construcción de Data Warehouse.....	7
1.3 Proceso básico del Data Warehouse. (12).....	11
1.4 Sistemas Operacionales e Informacionales. (13).....	13
1.4.1 Sistemas Operacionales	13
1.4.2 Objetivos de los Sistemas Operacionales	14
1.4.3 Sistemas Informacionales	14
1.4.4 Objetivos de los Sistemas Informacionales.....	14
1.4.5 Sistemas Operacionales vs. Sistemas Informacionales	14
1.5 Arquitectura conceptual de los datos.....	15
1.5.1 Datos Representativos.....	19
1.6 Tecnología de análisis multidimensional de los datos. OLAP	19
1.7. Modelos de almacenamiento de datos	21
1.7.1. ROLAP (Relational On Line Analytic Processing).....	21
1.7.2. MOLAP (Multidimensional On Line Analytic Processing).....	21
1.7.3. HOLAP (Hybrid Online Analytical Process)	22
1.8. Modelamiento multidimensional (26)	22
1.8.1 Esquema en Estrella.....	22
1.8.2 Esquema Copo de Nieve	24
1.8.3 Esquema Constelación	24
1.9 Propuestas de metodologías de diseño.....	25
1.10. Herramientas propuestas para el desarrollo del Data Mart	26
1.10.1. La plataforma Pentaho Business Intelligence.....	26
1.10.2. Kettle 3.0.3.....	27
1.10.3. Schema Workbench 3.0.3.....	27
1.10.4 PostgreSQL	27
1.10.5 Rational Rose	28
1.11 Conclusiones.....	29
Capítulo 2. Diseño del Data Mart	30
2.1 Introducción.....	30
2.2 Descripción detallada del Data Mart	30
2.3 Aplicación del método DWEP.....	30
2.4 Requerimientos	31
2.5 Análisis.....	31
2.6 Proceso de diseño de un Data Mart.....	31
2.7 Modelado de datos	31
2.7.1 Modelo de datos conceptual	31
2.7.2 Modelo de datos lógico	32
2.7.3 Modelo de datos físico	32
2.8 Esquema conceptual del Almacén de datos.....	32

2.9 Mapeo de datos.....	35
2.10 Conclusiones.....	38
Capítulo 3. Implementación del Data Mart	39
3.1 Introducción.....	39
3.2 Proceso ETL.....	39
3.3 Edificación del sistema de almacenamiento de datos.....	39
3.4 Configuración de la extracción y transformación de datos.	39
3.5 Cubo de Datos	41
3.6 Creación de los cubos de Datos	41
3.7 Mondrian.	42
3.8 Despliegue de la solución propuesta.	43
3.9 Esquema físico de la fuente.	44
3.10 Esquema físico del almacén de datos	44
3.11 Diagrama de integración de transporte.....	44
3.12 Conclusiones.....	44
CONCLUSIONES	45
RECOMENDACIONES	46
BIBLIOGRAFÍA.....	47
GLOSARIO DE TÉRMINOS.....	50
ANEXOS.....	51

ÍNDICE DE FIGURAS

Fig. 1.1 Top-Down.	9
Fig. 1.2 Bottom-Up.....	9
Fig. 1.3 Proceso ETL.	13
Fig. 1.4 Arquitectura de datos de una sola capa	16
Fig. 1.5 Arquitectura de dos capas.....	16
Fig. 1.6 Arquitectura de tres capas.....	17
Fig. 1.7 Drill-down.....	20
Fig. 1.8 Drill-up	20
Fig. 1.9 Esquema en Estrella.	23
Fig. 1.10 Esquema Copo de Nieve.....	24
Fig. 1.11 Esquema Constelación.	24
Fig. 2.1 Esquema conceptual del almacén de datos	33
Fig. 2.2 Esquema conceptual del almacén de datos TH_Deudas.	33
Fig. 2.3 Esquema conceptual de datos, Dimension_Banco.....	34
Fig. 2.4 Esquema conceptual del DWH.....	35
Fig. 2.5 Esquema Lógico de la fuente (SCS)	36
Fig. 2.6 Mapeo nivel 0.....	36
Fig. 2.7 Mapeo de datos nivel 1	37
Fig. 2.8 Mapeo 2 de la tabla Banco y País (nivel 2)	37
Fig. 2.9 Mapeo 1 de la tabla banco y país (nivel 3)	38
Fig. 3.1 Transformación ETL_Banco.....	40
Fig. 3.2 Transformación ETL_Tiempo	40
Fig. 3.3 Transformación ETL_Hecho_Deuda	41
Fig. 3.4 Creación del cubo de datos, dimensiones y jerarquías.....	42
Fig. 3.5 Cliente de Píivot que muestra el cubo de Datos.....	43

INTRODUCCIÓN

El desarrollo del mundo actual ha puesto dinamismo en la actividad económica en cuanto a la modernización de nuevos sistemas, en el cual la informática juega un papel sorprendentemente creciente en el desarrollo científico y tecnológico y se ha retomado la importancia del tratamiento computacional de la información estratégica para la toma de decisiones.

La habilidad para convertir los datos acumulados durante años por los sistemas informáticos en información estratégica, integrada y actualizada es vital para las organizaciones, surgiendo sistemas con nuevas tecnologías que permiten a los usuarios analizar los datos en busca de información que les ayude a tomar decisiones claves para su negocio. Hoy en día existen diversos tipos que dan soporte a la toma de decisiones, pero el que ha tenido más auge a escala mundial ha sido el Data Warehouse (DWH) o Almacenes de Datos, puesto que provee un ambiente para hacer un mejor uso de la información administrada por diversas aplicaciones operacionales y/o externas al brindar la información en el nivel de detalle adecuado para posteriores análisis, proporcionándole a la empresa una enorme ventaja competitiva.

Durante el transcurso de los años en Cuba, el desarrollo del software ha ido avanzando y mejorando en cuanto a calidad, rapidez y complejidad, por lo que se han creado y modificado nuevos proyectos para informatizar todas las esferas de la economía. Uno de los sectores destacados es el Sistema Bancario y en especial el Banco Nacional de Cuba (BNC), el cual se ha ido informatizando gradualmente, en el mismo se generan diariamente datos como producto de las transacciones que se realizan, el control y la organización de este flujo constante de datos, se torna un poco difícil y no se cuenta con herramientas softwares que permitan el manejo de grandes volúmenes de datos y búsqueda de información útil y relevante que enriquezca las decisiones de los directivos que necesitan respuestas efectivas y concretas para una correcta toma de decisiones.

El BNC es el encargado de controlar la deuda externa del país. El proceso de análisis de las deudas actualmente en el BNC se realiza mediante reportes obtenidos del sistema informático SABIC (Sistema automatizado para la Banca Internacional de Comercio), donde los datos almacenados en base de datos relacionales están cambiando constantemente.

Con ello se lleva un control estricto de las deudas, pero no posibilita que se realice una correcta toma de decisiones por parte de los directivos, al no contar con una herramienta de software que publique información almacenada sobre situaciones anteriores en función de definir guías para acciones futuras.

A partir de la situación descrita anteriormente se selecciona como **problema a resolver**: ¿Cómo apoyar a los directivos del Banco Nacional de Cuba en el proceso de toma de decisiones asociado al análisis de la deuda externa?

Se define como **objeto de estudio** las metodologías de diseño de un Data Mart.

El **campo de acción** está centrado en el proceso de implementación de un Data Mart para el análisis de la deuda externa del Banco Nacional de Cuba.

El **objetivo general** de esta investigación es desarrollar un Data Mart para apoyar el proceso de toma de decisiones del Banco Nacional de Cuba respecto al análisis de la deuda externa.

Para dar cumplimiento al objetivo anteriormente planteado se definen las siguientes **tareas investigativas**:

- Efectuar un estudio sobre las herramientas, tecnologías y tendencias actuales, propuestas para el desarrollo de Data Warehouse, para identificar y seleccionar las posibles a utilizar en la solución del problema planteado.
- Analizar el proceso de gestión de deudas en el Banco Nacional de Cuba, así como la base de datos del sistema informático Sabic para derivar los requerimientos del sistema.
- Realizar un análisis de los gestores de bases datos y seleccionar el más adecuado para la construcción del Data Mart.
- Obtención de los artefactos que se generan en los flujos de trabajo: Requerimientos, Análisis y Diseño e Implementación, para lograr una mejor comprensión del Data Mart a desarrollar.
- Realizar el proceso de extracción, transformación y carga de datos para filtrar, limpiar, homogenizar y agrupar la información proveniente de la fuente de datos del Data Mart.
- Realizar el procesamiento analítico en línea OLAP para analizar el negocio desde diferentes escenarios históricos y proyectar como se ha comportado y evolucionado en un ambiente multidimensional.

Organización del Documento

El presente documento se estructura en resumen, tres capítulos de contenidos, conclusiones, recomendaciones, referencias bibliográficas y los anexos correspondientes.

En el capítulo 1 se recogen conceptos, objetivos y características de los Data Warehouse, así como las metodologías, tecnologías y software empleados en el diseño, además se realiza un análisis de las diferencias y funcionalidades de las principales herramientas que dan soporte a los data Warehouse.

En el capítulo 2 se da una descripción detallada del Data Mart y se realiza el diseño del mismo usando la metodología DWEP (Data Warehouse Engineering Process).

En el capítulo 3 se lleva a cabo la implementación del Data Mart, se definen los tipos de modelamiento de datos, se conciben los esquemas físicos del Data Mart, se lleva cabo el proceso de extracción, transformación y carga de datos (ETL), la creación de los cubos de Datos y las consultas de datos OLAP.

Capítulo 1. Conceptos, herramientas y metodologías para el diseño del Data Mart

1.1 Introducción

En el presente capítulo se realiza un análisis detallado sobre los conceptos, objetivos y principales características sobre los Data warehouse. Se aborda todo lo referente a los procesos de extracción, transformación y carga en los Datawarehouse, los tipos de modelamiento de datos, se puntualiza que son los sistemas operacionales e informacionales así como precisar las principales tecnologías, software, herramientas y tendencias que se están empleando en el mundo actual para el diseño e implementación del mismo.

1.2 Data Warehouse

Los sistemas relacionales no están preparados óptimamente para extraer conocimiento de los datos almacenados, estos por mucho que se les mejore y que se les quiera adaptar, no son los encargados de esa nueva fase de consulta, para eso ha surgido una nueva forma de organización de los datos, los Data Warehouse (DWH) también conocido como Almacén de Datos, que se han convertido en el centro de la arquitectura para los sistemas de información a partir de la década de los '90 hasta la fecha.

Durante los últimos años ha existido un gran incrementado en la utilización de los Data Warehouse, pues desde su surgimiento han facilitado el análisis y comprensión de los datos de las empresas, mejorando el porcentaje sobre la toma de decisiones efectivas ejecutadas por los directivos. Los principales protagonistas sobre este tema fueron Ralph Kimball considerado el principal promotor del enfoque dimensional para el diseño de almacenes de datos y William H. Inmon conocido como el padre de los Data Warehouse, ambos han desarrollado sus propios enfoques, modelos y arquitectura de los Data Warehouse.

Hoy en día con el avance de las tecnologías un mayor número de personas se dedican a estudiar este tema exponiéndose varios criterios y definiciones sobre Data Warehouse, tomando como ejemplo estos criterios se puede decir que un DWH es una base de datos que almacena información proveniente de diferentes bases de datos operacionales y/o externas, con el objetivo de consolidar dicha información y hacerla disponible para que los usuarios finales puedan fácilmente ejecutar consultas, confeccionar reportes y realizar análisis para la toma de decisiones. A continuación se exponen otras definiciones planteadas por diversos autores:

Los Data Warehouse surgen con la promesa del manejo y control de la información. Ellos aseguran una vista única de los datos, que pueden provenir de diversas fuentes. Gracias a esto, los usuarios finales no se ven en la necesidad de aprender y utilizar múltiples sistemas de acceso y manipulación de los datos. Un Data Warehouse facilita la comprensión de los datos, transformándolos en información útil, teniendo como bandera el apoyo a la Toma de Decisiones. (1)

El Data Warehouse o Almacén de Datos. Surgió con el objetivo de hacer consultable la información que se tiene de una empresa tanto de meses como de años anteriores. Es un almacenamiento simple, completo y consistente de datos obtenidos desde una variedad de fuentes, disponibles para el usuario final de forma tal que puedan entenderlos y utilizarlos. Facilita la integración de sistemas de aplicación no integrados. Un Data Warehouse soporta el procesamiento informático al proveer una plataforma sólida, a partir de los datos históricos para hacer el análisis que se necesita de la información. El objetivo fundamental de un Data Warehouse, es asistir al usuario en el entendimiento del pasado y contar con los elementos para la planeación del futuro de corto, mediano y largo plazo. Es actualmente, el centro de atención de las grandes instituciones, porque provee un ambiente para que las organizaciones hagan un mejor uso de la información que está siendo registrada por las diversas aplicaciones operacionales. (2)

La definición más formal fue la propuesta por William H. Inmon (3) en donde plantea que: Un Data Warehouse es una colección de datos orientados a temas, integrados, no volátiles y variante en el tiempo, organizados para dar soporte al proceso de ayuda a la toma de decisiones, puesto que permiten analizar la información consolidada según diferentes puntos de vista.

➤ **Integrado**

Los datos almacenados en el data Warehouse deben integrarse en una estructura consistente, por lo que las inconsistencias presentes entre los diversos sistemas operacionales deben ser eliminadas.

➤ **Orientado a temas**

Los datos son organizados por temas para facilitar su acceso y entendimiento por parte de los usuarios finales. Por ejemplo, todos los datos sobre personas pueden ser consolidados en una única tabla del data Warehouse. De esta forma, las peticiones de información sobre personas serán más fáciles de responder dado que toda la información reside en el mismo lugar.

➤ **Variante en el tiempo**

Los datos no se actualizan, sino que se almacena el historial de ellos, es decir, el conjunto de valores que el dato ha tenido a lo largo de su historia, asociando a cada dato una referencia de tiempo, con el fin de poder identificar los distintos valores que dicho dato ha ido tomando a lo largo de su ciclo de vida.

➤ **No volátil**

El almacén de información de un DataWarehouse existe para ser leído y no modificado. La información es por tanto permanente, una vez cargado los datos en el DWH, deben mantenerse invariables, permitiéndose sólo realizar dos tipos de operaciones: la carga inicial de datos y el acceso a los mismos. No existe el proceso de actualización de datos, sólo de carga.

1.2.1. Objetivos fundamentales de un Data Warehouse.

Principales objetivos de un Data Warehouse (4):

- **Hacer que la información sea accesible:** Los contenidos del Data Warehouse son entendibles ya que los niveles de la información son correctos y obvios y navegables, el acceso a ellos es caracterizado por el rápido desempeño. Estos requerimientos no tienen fronteras y tampoco límites fijos.
- **Hacer que la información sea consistente:** Significa que toda la información es contabilizada y completada.
- **Es información adaptable y elástica:** El Data Warehouse está diseñado para cambios continuos. Cuando se le hacen nuevas preguntas al Data Warehouse, los datos existentes y las tecnologías no cambian ni se corrompen. Cuando se agregan datos nuevos al Data Warehouse, los datos existentes y las tecnologías tampoco cambian ni se corrompen.
- **Es la fundación de la toma de decisiones:**

El Data Warehouse tiene los datos correctos para soportar la toma de decisiones. Solo hay una salida verdadera del Data Warehouse: las decisiones que son hechas después de que el Data Warehouse haya presentado las evidencias. La original etiqueta que preside el Data Warehouse sigue siendo la mejor descripción de lo que queremos construir: un sistema de soporte a las decisiones.

Un Data Warehouse se puede decir que sería de gran utilidad para las empresas puesto a que desarrollan varias aportaciones sobre los mismos, como son:

- ❖ 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 estadísticas de análisis y modelización para encontrar relaciones ocultas entre los datos del almacén; obteniendo un valor añadido para el negocio de dicha información.
- ❖ 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.
- ❖ Supone una optimización tecnológica y económica en entornos de Centro de Información, estadística o de generación de informes con retornos de la inversión espectaculares.

1.2.2. Data Warehouse y Data Mart.

Un concepto importante en el contexto de DWH, lo constituye la definición de **Data Mart**, el mismo no es más que un Data Warehouse con función departamental o regional, consta de las mismas características que un DWH y brinda las mismas facilidades, pero está orientado a una sola actividad y no a satisfacer las necesidades de toda la empresa. Por tanto, no se puede pensar en Data Mart en los términos de un DWH más pequeño, porque no es su tamaño lo que lo define sino su objetivo en la organización.

Un 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 departamento. Un data mart puede ser alimentado desde los datos de un Data Warehouse, o integrar por sí mismo un compendio de distintas fuentes de información. (5)

1.2.3 Métodos más usados en la construcción de Data Warehouse.

Los modelos propuestos por William H. Inmon y Ralph Kimball para llevar a cabo el diseño de un DWH son los más aplicados en la actualidad, coincidiendo en que un Data Mart o un data Warehouse independiente no satisface las necesidades que tienen las compañías a escala corporativa de acceder inmediatamente y con facilidad a sus datos, pero sus criterios difieren en cuanto al modelo de datos y a las arquitecturas.

Ralph Kimball propone como modelo de datos al modelo dimensional, el más popular en las soluciones que se implementan de manera práctica, el cual facilita a los usuarios finales las consultas y el análisis. Se caracteriza por ser sencillo de crear, extremadamente estable en presencia de cambios, además de mostrarse muy intuitivo y comprensible. (6)

También William H. Inmon reconoce al modelo dimensional como el mejor para el desarrollo de los Data Marts por las ventajas brindadas, pero propone la construcción del Data Warehouse basado en el modelo entidad relación.

La idea de Inmon se basa en que el modelo entidad relación es mucho más rico y adaptable que el dimensional (7). En cuanto a la arquitectura William H. Inmon en su libro “Building the Data Warehouse” (8) plantea que la construcción del Data Warehouse no debe ser sustituida por la implementación de varios Data Marts.

Resaltando que la excusa para no desarrollar un almacén de datos la mayoría de las veces es por no contar con un gran presupuesto, la sustitución de este por los Data Marts trae desventajas puesto que están diseñados para un área particular de la empresa, lo que trae consigo diferencias entre las estructuras de datos de los mismos, que al integrarlos en el Data Warehouse algunos no serán reusables, ni flexibles, ni útiles para la reconciliación que se necesita. Inmon manifiesta que el proceso de construcción del Data Warehouse parte de los sistemas operacionales existentes, creándose áreas de diferentes temas, cuando existan una cierta cantidad de estas, el Data Warehouse inicia el proceso de población de las áreas de una manera integrada, una vez concluido se comienza a dar respuestas a las inquietudes de los usuarios; empezando así el florecimiento del nivel departamental a medida que se tienen más datos en el Data Warehouse, y es en este punto del desarrollo cuando se centra la atención en las cuestiones de los diferentes departamentos, para definir y crear los Data Warehouse departamentales, los Data Marts.

Ralph Kimball en desacuerdo con la arquitectura propuesta por William H. Inmon resalta en su libro “The Data Warehouse ETL Toolkit, Practical Techniques for Extracting, Cleaning, Conforming and Delivering Data” (9), que los Data Marts están basados en los datos de la fuente y no en la visión departamental, en otras palabras que el Data Mart es sólo una parte de un producto orientado a la compañía, los cuales deben consistir en una continua pirámide de estructuras dimensionales idénticas, comenzando siempre con los datos atómicos. Plantea también que la idea de construcción de un Data Warehouse centralizado no es realista, siendo más real construirlo en un ambiente descentralizado e incremental, porque las empresas están en constante cambio, adquiriendo nuevas fuentes de datos y necesitando nuevas perspectivas, propone además, centrarse en trazar estrategias adaptables e incrementales basándose en una idealista visión de controlar toda la información antes de construir el Data Warehouse. Por esta razón manifiesta que el proceso de construcción de un almacén de datos parte de los sistemas operacionales existentes, creando los diferentes Data Marts basados en la información de dichas fuentes, para luego de tenerlos desarrollados y funcionales se comience con la construcción del Data Warehouse basado en la información que estos contienen. En la actualidad este método es el más usado, gracias a las diferentes ventajas que proporciona, permitiendo a las empresas acometer los proyectos de manera separada y de esta forma reducir los efectos negativos que tendría fracasar en un intento por construir un Data Warehouse.

A continuación se muestran las arquitecturas planteadas por William H. Inmon y Ralph Kimball con una breve explicación. (10)

Arquitectura Top-Down planteada por William H. Inmon

- Top-Down: primero se define el DWH y luego se desarrollan, construyen y cargan los DM a partir del mismo.

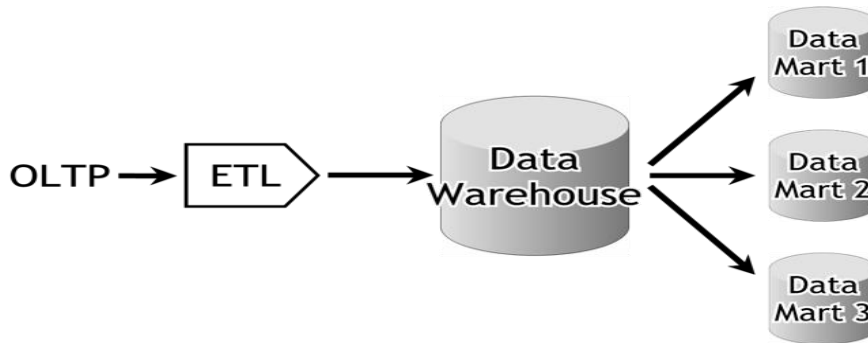


Fig. 1.1 Top-Down.

Como se puede apreciar, el DWH es cargado a través de procesos ETL y luego este alimenta a los diferentes DM, cada uno de los cuales recibirá los datos que correspondan al tema o departamento que traten. Esta forma de implementación cuenta con la ventaja de no tener que incurrir en complicadas sincronizaciones de hechos, pero requiere una gran inversión y una gran cantidad de tiempo de construcción.

Arquitectura Bottom-Up planteada por Ralph Kimball

- Bottom-Up: en esta arquitectura, se definen previamente los DM y luego se integran en un DWH centralizado.

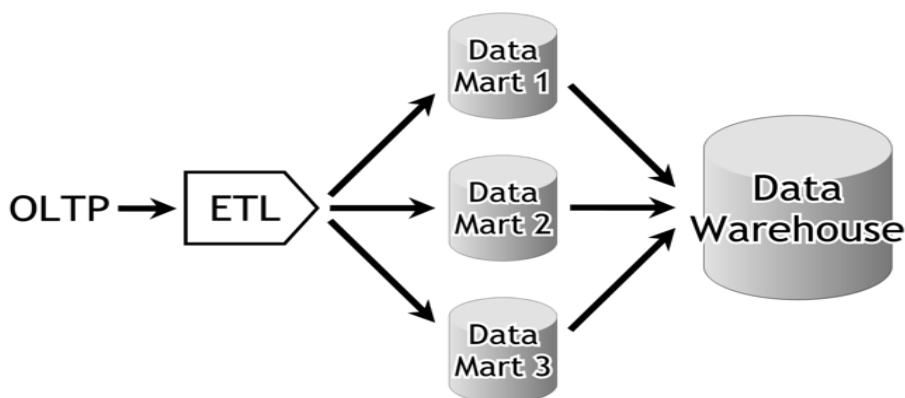


Fig. 1.2 Bottom-Up

Los DM se cargan a través de procesos ETL, los cuales suministrarán la información adecuada a cada uno de ellos. En muchas ocasiones, los DM son implementados sin que exista el DWH, ya que tienen sus mismas características pero con la particularidad de que están enfocados en un tema específico. Luego de que hayan sido creados y cargados todos los DM, se procederá a su integración con el depósito. La ventaja que trae aparejada este modelo es que cada DM se crea y pone en funcionamiento en un corto lapso de tiempo y se puede tener una pequeña solución a un costo no tan elevado. Luego que todos los DM estén puestos en marcha, se puede decidir si construir el DWH o no. El mayor inconveniente está dado en tener que sincronizar los hechos al momento de la consolidación en el depósito.

1.2.4 Ventajas y desventajas de los Data Warehouse. (11)

1.2.4.1 Ventajas

A continuación se comentan algunas de las ventajas más sobresalientes que ofrecen los DWH y que ejemplifican de mejor modo sus características y cualidades:

- ❖ Transforma datos orientados a las aplicaciones en información orientada a la toma de decisiones.
- ❖ Integra y consolida diferentes fuentes de datos y departamentos empresariales, que anteriormente formaban islas, en una única plataforma sólida y centralizada.
- ❖ Provee la capacidad de analizar y explotar las diferentes áreas de trabajo y de realizar un análisis inmediato de las mismas.
- ❖ Permite reaccionar rápidamente a los cambios del mercado.
- ❖ Aumenta la competitividad en el mercado.
- ❖ Elimina la producción y el procesamiento de datos que no son utilizados ni necesarios, producto de aplicaciones mal diseñadas o ya no utilizadas.
- ❖ 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.

Logra un impacto positivo sobre los procesos empresariales. Cuando los usuarios tienen acceso a una mejor calidad de información, la empresa puede lograr por sí misma: aprovechar el enorme valor potencial de sus recursos de información y transformarlo en valor verdadero; eliminar los retardos de los procesos empresariales que resultan de información incorrecta, inconsistente y/o inexistente; integrar y optimizar procesos a través del uso compartido e integrado de las fuentes de información; permitir al usuario adquirir mayor confianza acerca de sus propias decisiones y de las del resto, y lograr así, un mayor entendimiento de los impactos ocasionados.

- ❖ Aumento de la competitividad de los encargados de tomar decisiones.
- ❖ Los usuarios pueden acceder directamente a la información en línea, lo que contribuye a su capacidad para operar con mayor efectividad en las tareas rutinarias o no. Además, pueden tener a su disposición una gran cantidad de valiosa información multidimensional, presentada coherentemente como fuente única, confiable y disponible en sus estaciones de trabajo. Así mismo, los usuarios tienen la facilidad de contar con herramientas que les son familiares para manipular y evaluar la información obtenida en el DWH, tales como: hojas de cálculo, procesadores de texto, software de análisis de datos, software de análisis estadístico, reportes, etc.
- ❖ Permite la toma de decisiones estratégicas y tácticas.

1.2.4.2 Desventajas

A continuación se exponen algunas de las desventajas que pueden tener los DWH:

- ❖ Requiere una gran inversión, debido a que su correcta construcción no es tarea sencilla y consume muchos recursos, además, su misma implementación implica desde la adquisición de herramientas de consulta y análisis, hasta la capacitación de los usuarios.
- ❖ Resistencia al cambio por parte de los usuarios.
- ❖ Los beneficios del almacén de datos son apreciados en el mediano y largo plazo. Este punto deriva del anterior, y básicamente se refiere a que no todos los usuarios confiarán en el DW 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.
- ❖ Si se incluyen datos propios y confidenciales de clientes, proveedores, etc., el depósito de datos atentará contra la privacidad de los mismos, ya que cualquier usuario podrá tener acceso a ellos.
- ❖ Infravaloración de los recursos necesarios para la captura, carga y almacenamiento de los datos.
- ❖ Infravaloración del esfuerzo necesario para su diseño y creación.
- ❖ Incremento continuo de los requerimientos del usuario.

1.3 Proceso básico del Data Warehouse. (12)

Para comprender íntegramente el concepto de Data warehouse, es importante entender uno de sus procesos básicos, denominado ETL (Extracción, Transformación y Carga).

Extracción: Este es el primer paso de obtener la información hacia el ambiente del Data warehouse. La extracción convierte los datos a un formato preparado para iniciar el proceso de transformación. Una parte del proceso de extracción es la de analizar los datos extraídos, de lo que resulta un chequeo que verifica si los datos cumplen la pauta o estructura que se esperaba, de no ser así, los datos son rechazados. Un aspecto importante que se debe exigir a la tarea de extracción es que esta produzca un impacto mínimo en el sistema origen. Debido a esto se han tomado diferentes medidas, por ejemplo, si los datos a extraer son muchos, el sistema de origen se podría ralentizar e incluso colapsar, provocando que este no pueda utilizarse con normalidad para su uso cotidiano, por esta razón, en sistemas grandes, las operaciones de extracción suelen programarse en horarios o días donde este impacto sea nulo o mínimo. **En fin no es más que la obtención de información de las distintas fuentes tanto internas como externas.**

Transformación: Una vez que la información es extraída hacia el área de tráfico de datos, se comienza con la limpieza de la información, es decir, seleccionar únicamente los campos necesarios para el Data warehouse, combinar fuentes de datos, haciéndolas coincidir por los valores de las llaves, creando nuevas llaves para cada registro de una dimensión. La fase de transformación aplica una serie de reglas de negocio o funciones sobre los datos extraídos para convertirlos en datos que serán cargados. Algunas fuentes de datos requerirán alguna pequeña manipulación de los datos. **En resumen se define como filtrado, limpieza, depuración, homogeneización y agrupación de la información.**

Carga: Al final del proceso de transformación, los datos están en forma para ser cargados. Los Data warehouse mantienen un historial de los registros de manera que se pueda hacer una auditoría de los mismos y disponer de un rastro de toda la historia de un valor a lo largo del tiempo. **Es decir que este proceso se encarga de la organización y actualización de los datos y los metadatos en la base de datos.**

A continuación, se explica en síntesis el accionar del proceso ETL, y cuál es la relación existente entre sus diversas funciones. En la siguiente figura se puede apreciar mejor lo antes descrito:

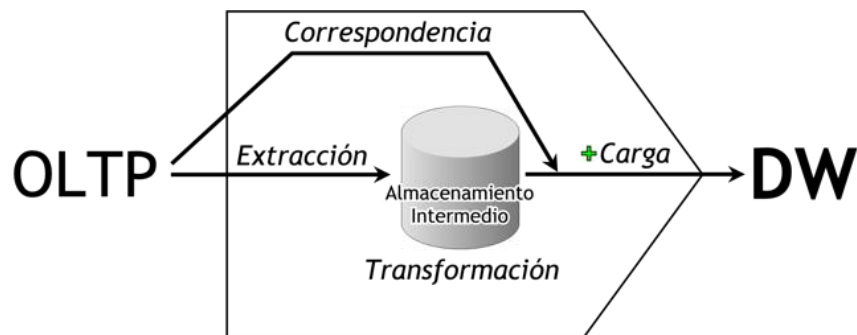


Fig. 1.3 Proceso ETL.

Los pasos que se siguen son:

1. Se extraen los datos relevantes desde los OLTP y se depositan en un almacenamiento intermedio.
2. Se integran y transforman los datos, para evitar inconsistencias.
3. Se cargan los datos desde el almacenamiento intermedio hasta el DWH. Si existiesen correspondencias directas entre datos de los OLTP y el DWH, se procede también a su respectiva carga.

1.4 Sistemas Operacionales e Informacionales. (13)

1.4.1 Sistemas Operacionales

Los sistemas que se utilizan para manipular los negocios, generalmente se orientan a una actividad específica y no ofrecen una vista global de la empresa. Cuando se necesita realizar análisis de la actividad de la empresa para tomar decisiones o proyectar nuevas estrategias de negocio, se carece de datos históricos que apoyen estas tareas. Por todo esto es necesario entonces, diferenciar los sistemas con que se trabaja en cuanto a sus finalidades. Los Sistemas Operacionales son aquellos que permiten realizar las funciones de la empresa y los Sistemas Informacionales son los que permiten estudiar el comportamiento de la empresa.

Los sistemas operacionales o sistemas de producción se utilizan para el funcionamiento de los negocios en tiempo real. Se basan en datos actuales del estado del negocio, llamados datos operacionales y se diseñan en principio para manipular un gran número de transacciones simples de lectura-escritura. Las características de cada uno de ellos, sus similitudes y diferencias serán abordadas a continuación.

1.4.2 Objetivos de los Sistemas Operacionales

Los sistemas operacionales tienen un papel fundamental para cualquier organización, ya que garantizan la automatización de los procesos y el flujo de la información a través de la organización. El diseño y la implementación de los sistemas de producción están dirigidos a cumplir los siguientes objetivos:

- Apoyar las funciones diarias de la organización.
- Brindar servicios de oficina.
- Entregar la información de manera automatizada.
- Asegurar la calidad y la protección de la información.

1.4.3 Sistemas Informacionales

Los sistemas informacionales se utilizan para administrar y controlar la empresa. Se basan en datos estables en un momento en el tiempo o periódicos, llamados datos históricos, y se diseñan, principalmente, para ejecutar consultas que involucran perspectivas de los datos, complejas y de solo lectura. Se apoyan en los datos que sustentan el proceso de toma de decisiones en una organización.

1.4.4 Objetivos de los Sistemas Informacionales

El propósito principal de los sistemas informacionales consiste en facilitar que la empresa se rija por la información. Un sistema informacional, automatizado o no, es la herramienta esencial que garantiza esta forma de dirección.

Los objetivos fundamentales de un Sistema Informacional se describen a continuación.

- Una fuente única de información
- Disponibilidad de la información distribuida
- La información en el contexto de los negocios
- Entrega automatizada de información
- Calidad y protección de la información

1.4.5 Sistemas Operacionales vs. Sistemas Informacionales

Si se comparan los aspectos fundamentales que distinguen a ambos sistemas se puede decir que la particularidad más importante entre ambos es el tipo de procesamiento de la información que realizan, ya sea transaccional o analítico, teniendo cada una ventajas y desventajas que permiten realizar una comparación entre las mismas.

Los sistemas operacionales utilizan el Procesamiento de Transacciones en Línea (On-Line Transaction Processing, OLTP) sustenta las operaciones diarias. Se conoce también como procesamiento operacional, pues describe los requerimientos para un sistema que trabaja en un ambiente operacional. OLTP se refiere a un tipo de cómputo en el cual el énfasis se realiza en el procesamiento de las transacciones tal y como son recibidas por las aplicaciones.

Los sistemas informacionales utilizan el Procesamiento Analítico en Línea (On-Line Analytical Processing, OLAP) sustenta el estudio del comportamiento del negocio y su proyección. Se conoce también como procesamiento para la toma de decisiones. La funcionalidad OLAP se caracteriza por un análisis dimensional y dinámico de los datos, ayuda al usuario a sintetizar la información a través de vistas personalizadas, análisis históricos y pronósticos.

Sistemas Operacionales	Sistemas Informacionales
Funciones diarias	Decisiones estratégicas a largo plazo
Constantes cambios en los datos	Datos raramente modificados
Procesamiento repetitivo	Procesamiento heurístico
Fundamentalmente, datos actuales	Fundamentalmente, datos históricos
Datos detallados	Datos detallados y resumidos
Orientados a aplicaciones	Orientados a entidades
Patrón de uso predefinido	Patrón de uso impredecible
Para manejo de transacciones	Para realizar análisis y toma de decisiones
Servicios de oficina	Servicios administrativos

Tabla 1.1 Tabla comparativa de sistemas operacionales e informacionales.

1.5 Arquitectura conceptual de los datos

La estructura que reúne todos los componentes de un Data Warehouse es conocida como arquitectura, esta es la forma de representar la organización total de datos, comunicación, procesamiento y presentación. Como dice Paulraj Ponía "...la arquitectura incluye todo lo que se necesita para preparar y guardar los datos. Por otro lado, también contiene todos los recursos para distribuir la información desde el data warehouse. Está compuesta más allá de reglas, procedimientos y funciones que permiten al almacén de datos trabajar y cumplir los requisitos de la empresa... Define las normas, medidas, diseño general, y técnicas de apoyo." (14)

Para resolver los problemas en la actualidad se hace uso de diferentes arquitecturas conceptuales de datos que se definen a nivel lógico. En un Data Warehouse suele representarse varias capas a través de las cuales circulan los datos, de modo que estos se obtienen a partir de los de la capa previa (15), nombrándose de acuerdo al número de capas que abarcan.

En la arquitectura de Datos de **Una Sola capa** la información se guarda sólo una vez en el Data Warehouse, almacenándose únicamente los datos de tiempo real, sobre los cuales actúan sistemas informacionales y operacionales, esto puede traer disputas ya que ambos sistemas actúan sobre el mismo conjunto de datos y quizás en el momento en que se precisen no estén disponibles para los fines operacionales porque pueden estar siendo consultados y mientras esto sucede no es posible realizar actualizaciones. (16)

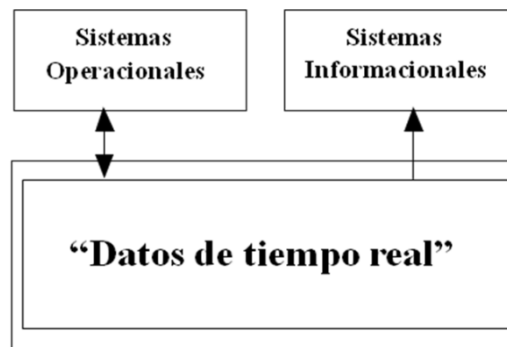


Fig. 1.4 Arquitectura de datos de una sola capa

En la arquitectura de Datos de **Dos Capas** se perfecciona lo antes expuesto, conteniendo además de una capa inferior, donde se contemplarán los datos de tiempo real utilizados por las aplicaciones operacionales en modo lectura/escritura, una superior, para el almacenamiento de los datos derivados utilizados por las aplicaciones informacionales, éstos pueden ser una copia directa de los datos de tiempo real o pueden obtenerse mediante la aplicación de un algoritmo, aumentando de esta forma los requerimientos de almacenamiento debido a la duplicación de información pero garantizando el acceso de los sistemas operacionales en cualquier instante de tiempo.

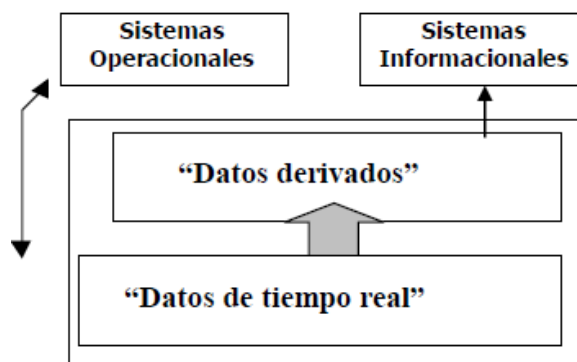


Fig. 1.5 Arquitectura de dos capas

La transformación llevada a cabo de los datos de tiempo real a datos derivados requiere de una capa intermedia para solucionar los problemas de inconsistencias, realizando el procesamiento de los distintos conjuntos de datos de tiempo real adecuadamente, a esta capa se le conoce como capa de datos reconciliados y esta nueva arquitectura recibe el nombre de Arquitectura de Datos de **Tres Capas**.

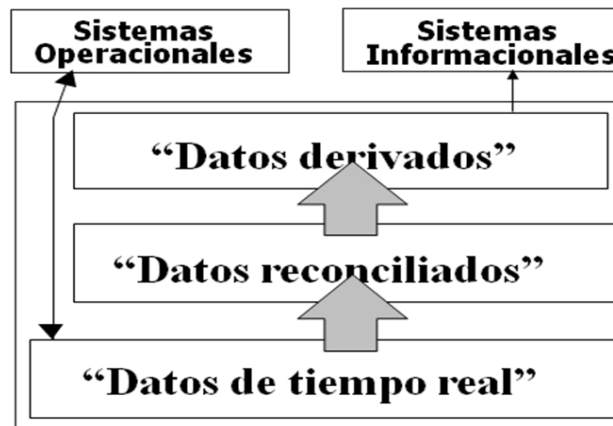


Fig. 1.6 Arquitectura de tres capas

Como planteaba anteriormente para realizar el diseño de un Data Warehouse se puede seleccionar cualquiera de las arquitecturas anteriormente mencionadas de acuerdo a las características y necesidades de la empresa.

El desarrollo de los Data Warehouse actualmente lleva un crecimiento acelerado debido a que es una tecnología muy entendible y para comprender cómo se relacionan todos sus componentes es necesario contar con un modelo de Arquitectura Data Warehouse, considerando su estructura en un marco de ocho niveles.

1. **Nivel operacional:** Se definen los orígenes de datos del almacén, es decir las diversas bases de datos operacionales y fuentes externas de donde serán extraídos.
2. **Nivel de acceso a la información:** Incluye las herramientas de consultas, análisis, generadores de informes y herramientas de datamining, teniendo como finalidad la manipulación, análisis y presentación de los datos de acuerdo a los requerimientos de los usuarios. Cuya finalidad es servir de soporte a las decisiones gerenciales.
3. **Nivel de acceso a los datos:** Es el encargado de la conexión entre el nivel de acceso a la información y el nivel operacional, siendo SQL el lenguaje de datos estándar para el intercambio de los mismos.

4. **Nivel de directorio de datos o Metadata:** Es un repositorio para almacenar y gestionar los metadatos. Este repositorio y su diseño son aspectos de suma importancia para el éxito de un data warehouse, aunque su valor en los proyectos de desarrollo ha sido subestimado (17). Su importancia radica en el hecho de que todo el conocimiento sobre la creación de un data warehouse es almacenado en el mismo y de aquí que los metadatos sean los responsables de guiar los procesos de extracción, limpieza y carga de los datos dentro del almacén además de ayudar a que las herramientas de consulta y los generadores de informe funcionen correctamente (18), (19). Los metadatos se refieren a la información sobre la estructura, el contenido y las interdependencias que existen entre los componentes del data warehouse. Estos describen los tipos de datos, las definiciones físicas y lógicas de los mismos, las consultas e informes predefinidos, las reglas de validación y negocio, las definiciones de las fuentes de datos, las rutinas de transformación y de proceso, etc. En definitiva, se refieren a cualquier cosa que define un objeto del data warehouse.
5. **El Nivel de gestión de proceso:** Es el encargado de la programación de diversas tareas que deben realizarse para construir y mantener el Data Warehouse y la información del directorio de datos. Es el encargado de controlar varios procesos con el propósito de conservar actualizado el Data Warehouse.
6. **El Nivel de mensaje de la aplicación:** Tiene que ver con el envío de información alrededor de la red de la empresa y puede usarse para recolectar las transacciones o los mensajes y entregarlos a una ubicación segura en un tiempo seguro.
7. **Nivel de Data Warehouse:** Es donde ocurre el almacenamiento físico de datos, usada principalmente para usos estratégicos, de manera que los datos almacenados sean flexibles y fáciles de acceder. En algunos casos puede verse el Data Warehouse simplemente como una vista lógica, pues puede no involucrar almacenamiento de datos.
8. **Nivel de organización de datos:** Es el componente final de la arquitectura Data Warehouse, conocido también como gestión de copia o réplica, incluye procesos para combinar, cargar datos para el depósito, resumir, acceder a la información desde bases de datos operacionales y/o externas, permite además el análisis de calidad de datos y filtros que identifican modelos y estructura de datos dentro de la data operacional existente.

Para el desarrollo satisfactorio y una mejor solución a nuestro problema se escogerá la arquitectura de dos capas debido a que la misma cuenta con las características y propiedades necesarias que satisfacen estrictamente todos los pasos a seguir en el diseño e implementación del Data Mart a continuación se hace referencia a los tipos de datos que contiene dicha arquitectura.

1.5.1 Datos Representativos

Datos de tiempo real

- ✓ Datos actuales que representan el funcionamiento presente de la empresa.
- ✓ Se acceden en modo lectura/escritura mediante transacciones predefinidas.
- ✓ Se crean, manipulan y usan a través de aplicaciones operacionales o de producción.
- ✓ Tradicionalmente se encuentran en bases de datos o en archivos.
- ✓ Actualmente muchos de estos datos se encuentran fuera del mundo de las bases de datos relacionales, sobre sistemas caducos, con estructuras pobres o muy complejas y, en ocasiones, no explotan lo suficiente las capacidades de los ambientes relacionales o están mal diseñados.
- ✓ No están directamente controlados por el área de sistemas informacionales.
- ✓ Existen en una amplia variedad de formatos y ubicaciones.

Datos derivados

- ✓ Se obtienen a través de transformaciones aplicadas a los datos de tiempo real.
- ✓ Se utilizan para administrar la empresa en modo de solo lectura.
- ✓ Se refieren a períodos o puntos específicos en el tiempo.

1.6 Tecnología de análisis multidimensional de los datos. OLAP

El procesamiento analítico en línea OLAP (On Line Analytic Processing), es la componente más poderosa de los DWH, pues es el motor de consultas especializado del almacén de datos. OLAP, es una tecnología de software para el análisis en línea, administración y ejecución de consultas, que permiten inferir información del comportamiento del negocio. Su principal objetivo es el de brindar rápidas respuestas a complejas preguntas, para interpretar la situación del negocio y tomar decisiones. Cabe destacar que lo realmente interesante en OLAP, no es la ejecución de simples consultas tradicionales, sino la posibilidad de utilizar operadores tales como drill-up, drill-down, para explotar profundamente la información.

Este tipo de herramientas puede analizar el negocio desde diferentes escenarios históricos y proyectar cómo se ha venido comportando y evolucionando en un ambiente multidimensional, o sea, mediante la combinación de diferentes perspectivas, temas de interés o dimensiones. Esto permite deducir tendencias, por medio del descubrimiento de relaciones entre las perspectivas que a simple vista no se podrían encontrar sencillamente.

Drill-down: Permite apreciar los datos en un mayor detalle, bajando por la jerarquía de una dimensión. Esto brinda la posibilidad de introducir un nuevo nivel o criterio de agregación en el análisis, disgregando los grupos actuales. (20)

Drill-down es ir de lo general a lo específico. Gráficamente:

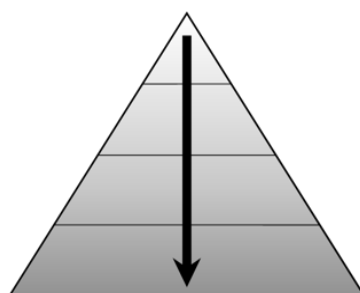


Fig. 1.7 Drill-down

Drill-up: Permite apreciar los datos en menor nivel de detalle, subiendo por la jerarquía de una dimensión. Esto brinda la posibilidad de quitar un nivel o criterio de agregación en el análisis, agregando los grupos actuales. (21)

Drill-up es ir de lo específico a lo general. Gráficamente:

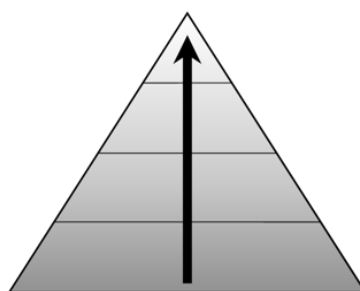


Fig. 1.8 Drill-up

Las herramientas OLAP requieren que los datos estén organizados dentro del depósito en forma multidimensional, por lo cual es que utilizan los cubos multidimensionales. Además de las características ya descritas, se pueden enumerar las siguientes:

- ❖ Permite recolectar y organizar la información analítica necesaria para los usuarios y disponer de ella en diversos formatos, tales como tablas, gráficos, reportes, etc.
- ❖ Soporta análisis complejos de grandes volúmenes de datos.
- ❖ Complementa las actividades de otras herramientas que requieran procesamiento analítico en línea.
- ❖ Presenta al usuario una visión multidimensional de los datos (matricial).
- ❖ Es transparente al tipo de tecnología que soporta el DWH, ya sea ROLAP, MOLAP o HOLAP (se explican en el próximo epígrafe).
- ❖ Permite definir de forma flexible las dimensiones que se quieren analizar, sus restricciones, jerarquías y combinaciones.
- ❖ No tiene limitaciones con respecto al número máximo de dimensiones permitidas.
- ❖ Permite a los usuarios, analizar la información basándose en más criterios que un análisis de forma tradicional.
- ❖ Al contar con muestras grandes, se pueden explorar mejor los datos en busca de respuestas.
- ❖ Permiten realizar agregaciones y combinaciones de los datos de maneras complejas y específicas, con el fin de realizar análisis más estratégicos.

1.7. Modelos de almacenamiento de datos

En general existen varios modelos de almacenamiento de datos en los sistemas de información de los DWH. Las opciones de almacenamiento físico afectan al rendimiento, a los requisitos de almacenamiento y a las ubicaciones de almacenamiento de las particiones y de sus cubos primarios. Una de estas opciones es el modo de almacenamiento de la partición. Una partición puede tener uno de estos modos de almacenamiento básicos: MOLAP, ROLAP, HOLAP. (22)

1.7.1. ROLAP (Relational On Line Analytic Processing)

Este tipo de organización física se implementa sobre tecnología relacional, pero disponen de algunas facilidades para mejorar el rendimiento. Es decir, ROLAP cuenta con todos los beneficios de un Sistema Gestor de Base de Datos(SGBD) Relacional a los cuales se les provee extensiones y herramientas para poder utilizarlo como un Sistema Gestor de DWH, mucho más lenta que las otras dos estrategias de almacenaje.(23)

1.7.2. MOLAP (Multidimensional On Line Analytic Processing)

Usa unas bases de datos multidimensionales para proporcionar el análisis, su principal premisa es que el OLAP está mejor implantado almacenando los datos multidimensionalmente. El almacenaje de MOLAP, provee excelente rendimiento y compresión de datos. Tiene el mejor tiempo de respuesta, dependiendo solo en el porcentaje y diseño de las agregaciones del cubo. En general este método, es muy apropiado para cubos con uso frecuente por su rápida respuesta. (24)

1.7.3. HOLAP (Hybrid Online Analytical Process)

Un desarrollo un poco más reciente ha sido la solución OLAP híbrida (HOLAP, procesamiento analítico en línea híbrido), la cual combina las arquitecturas ROLAP y MOLAP para brindar una solución con las mejores características de ambas: desempeño superior y gran escalabilidad. Un tipo de HOLAP mantiene los registros de detalle (los volúmenes más grandes) en la base de datos relacional, mientras que mantiene las agregaciones en un almacén MOLAP separado. Los cubos almacenados como HOLAP, son más pequeños que los MOLAP y responden más rápidos que los ROLAP. HOLAP es generalmente usado para cubos que requieren rápida respuesta, para sumarizaciones basadas en una gran cantidad de datos.

El modo de almacenamiento que se utilizará en el Data Mart todavía no se ha precisado, el mismo se definirá más adelante cuando el Data Mart esté estructurado y definido en el diseño del mismo. (25)

1.8. Modelamiento multidimensional (26)

Las bases de datos multidimensionales, proveen una estructura que permite, a través de la creación y consulta a una estructura de datos determinada para tener acceso flexible a los datos, para explorar y analizar sus relaciones, y consiguientes resultados.

Los Data Warehouse 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 tributos de cada dimensión a lo largo de las abscisas, determinan un hecho específico. Los cálculos que se aplican sobre las dimensiones son matriciales, los cuales se procesan dando como resultado reportes tabulares. Las bases de datos multidimensionales implican tres variantes posibles de modelamiento, que permiten realizar consultas de soporte de decisión:

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

1.8.1 Esquema en Estrella

El esquema en estrella, consta de una tabla de hechos central que contiene los hechos, medidas o indicadores que serán utilizados por los analistas de negocio para apoyar el proceso de toma de decisiones, donde los hechos son datos instantáneos en el tiempo, y varias tablas de dimensiones relacionadas a esta, a través de sus respectivas claves. Este esquema es el más simple de interpretar y optimiza los tiempos de respuesta ante las consultas de los usuarios y es soportado por casi todas las herramientas de consulta y análisis y los metadatos son fáciles de documentar y mantener.

En la siguiente figura se puede apreciar un esquema en estrella estándar:

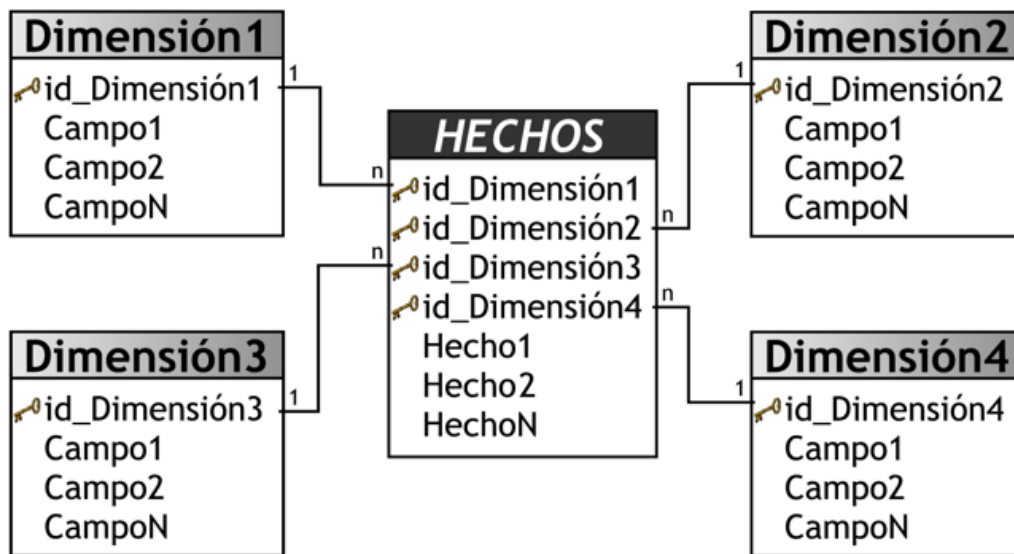


Fig. 1.9 Esquema en Estrella.

A continuación se destacan algunas características de este modelo:

- ❖ Posee los mejores tiempos de respuesta.
- ❖ Su diseño es fácilmente modificable.
- ❖ 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.

1.8.2 Esquema Copo de Nieve

Este esquema representa una extensión del modelo en estrella donde las dimensiones se organizan en jerarquías de dimensiones.

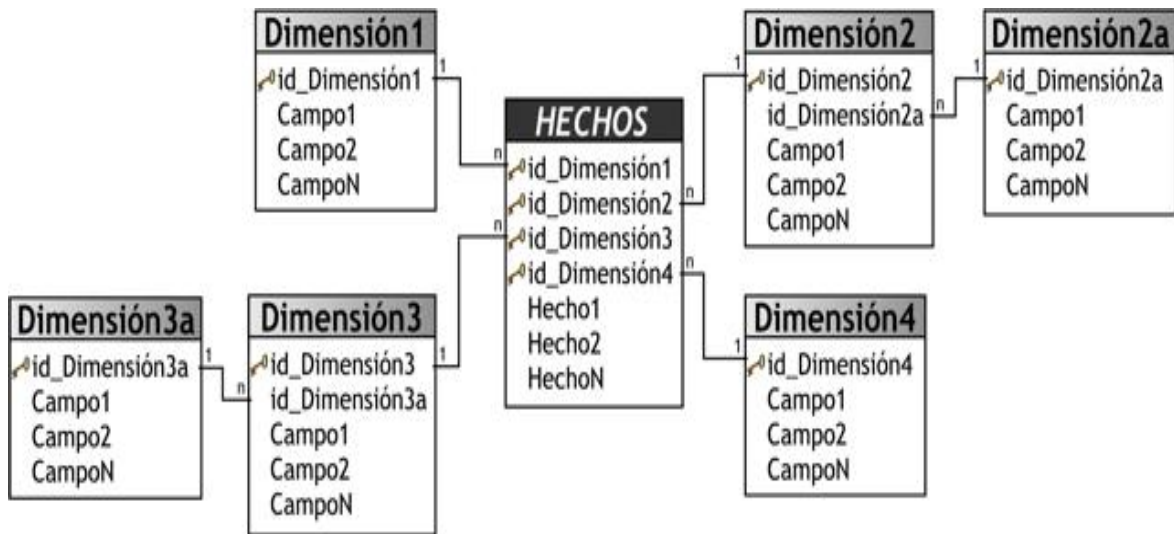


Fig. 1.10 Esquema Copo de Nieve.

1.8.3 Esquema Constelación

Este modelo está compuesto por una serie de esquemas en estrella, y tal como se puede apreciar en la siguiente figura, está formado por una tabla de hechos principal y por una o más tablas de hechos auxiliares las cuales pueden ser sumalizaciones de la principal. Dichas tablas yacen en el centro del modelo y están relacionadas con sus respectivas tablas de dimensiones.

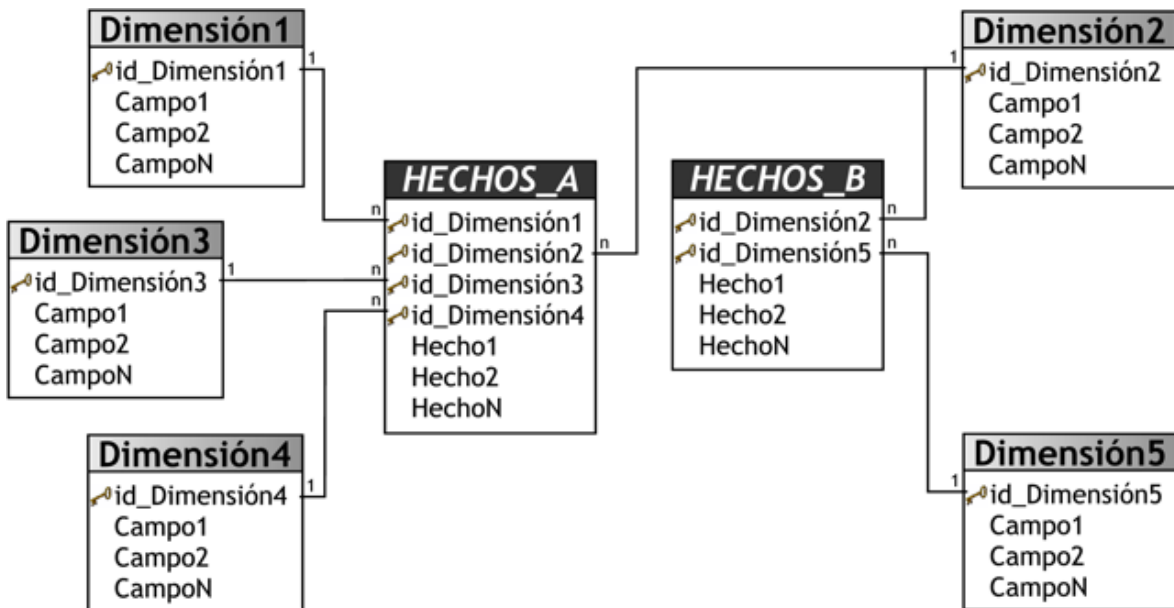


Fig. 1.11 Esquema Constelación.

1.9 Propuesta de metodologías de diseño

En la actualidad existen varias propuestas de modelos y métodos para llevar a cabo el diseño de los almacenes de datos, tales como (27), (28), (29), (30), (31), (32), sin embargo ninguno de estas propuestas se han aceptados como un modelo estándar para realizar el modelado multidimensional, pues no cubren todas las fases y transformaciones necesarias. Después de haber hecho un estudio exhaustivo de las metodologías existentes, se escogió la metodología DWEP (Data Warehouse Engineering Process) para realizar el diseño del Data Mart, a continuación se explican sus características.

1.9.1.2 Metodología DWEP

Es un método orientado a objetos, independiente de cualquier implementación específica, ya sea relacional, multidimensional, orientado a objetos, etc. permite la representación de todas las etapas del diseño de un Data Warehouse, está basada en UML que es el Lenguaje Unificado de Modelado (33) y RUP que es el Proceso Unificado de Desarrollo de Software. Consta de varias etapas y niveles como se plantea a continuación.

Etapas:

- **Origen:** Define los orígenes de datos del almacén de datos, como los sistemas OLTP, fuentes de datos externas, etc.
- **Integración:** Define el mapeo entre los orígenes de datos y el propio almacén de datos.
- **Almacén de Datos:** Define la estructura del almacén de Datos.
- **Adaptación:** Define el mapeo entre el almacén de datos y las estructuras empleadas por el cliente.
- **Cliente:** Define las estructuras concretas que son empleadas por los clientes para acceder al almacén de datos, como Data Mart o aplicaciones OLAP.

Niveles:

- **Conceptual:** Define el almacén de datos desde un punto de vista conceptual, es decir, desde el mayor nivel de abstracción y contiene únicamente los objetos y relaciones más importantes.
- **Lógico:** Abarca aspectos lógicos del diseño del almacén de datos, como la definición de las tablas y claves, la definición de los procesos ETL (Extraction, Transformation and Loading), etc.
- **Físico:** Define los aspectos físicos del almacén de datos, como el almacenamiento de las estructuras lógicas en diferentes discos o la configuración de los servidores de bases de datos que mantienen el almacén de datos.

Esta metodología muestra una mayor integridad en el diseño de los Data warehouse, al abarcar una serie de modelos basada en los paquetes de UML, brinda una mayor trazabilidad para diseño del data mart desde el modelo conceptual hasta el físico y cumple con las características y requerimientos esenciales para el diseño íntegro del mismo.

1.10. Herramientas propuestas para el desarrollo del Data Mart

1.10.1. La plataforma Pentaho Business Intelligence

La plataforma Pentaho Business Intelligence (BI) o Inteligencia de Negocio cubre muy amplias necesidades de análisis de los datos y de los informes empresariales. Las soluciones de Pentaho están escritas en Java y tienen un ambiente de implementación también basado en Java. Pentaho es una solución muy flexible para cubrir una amplia gama de necesidades empresariales, tanto las típicas como las sofisticadas y específicas al negocio. (34)

Incluye herramientas para realizar consultas, generación de informes y reportes, análisis interactivo, tableros de mando, ETL/integración de datos, data mining (minería de datos), y un servidor para la plataforma de BI que lo ha convertido en la suite BI de software libre más popular del mundo. Productos de Pentaho se utilizan en el Sistema de Mando Aéreo de US Army, Lifetime Networks, Terra Industries y Sun Microsystems. (35)

Los módulos que se utilizan de la plataforma Pentaho BI son: (36)

Análisis - Pentaho Análisis suministra a los usuarios un sistema avanzado de análisis de información. Con uso de las tablas dinámicas, generadas por Mondrian, el usuario puede navegar por los datos, ajustando la visión de los datos, los filtros de visualización, añadiendo o quitando los campos de agregación. Además, con el Microsoft Excel Analysis Services, se puede analizar los datos dinámicos en Microsoft Excel (usando la conexión a OLAP **server Mondrian**).

Mondrian es un motor de ROLAP desarrollado en Java que permite analizar grandes conjuntos de datos almacenados en un Data Warehouse. Posee un alto desempeño y análisis interactivo de volúmenes de información de cualquier tamaño, además explora dimensionalmente los datos y realiza cálculos avanzados utilizando el lenguaje MDX (multidimensional expressions o expresiones multidimensionales). (37)

Para acceder a las funcionalidades que brinda Mondrian se requiere de un cliente, es el caso de **JPivot** o **JRubik**.

JPivot: Es una librería de Java Server Pages (JSP) personalizados que presentan tablas OLAP y permite a los usuarios realizar operaciones tales como Drill Down y Drill Up además de consultas OLAP por medio del lenguaje MDX. Está diseñado para trabajar con motores OLAP como el de Mondrian por lo cual no utiliza las interfaces de programación de aplicaciones o (APIs) de Mondrian directamente sino que implementa su propio modelo OLAP. (38)

JRubik: Está basado en los componentes de JPivot y también se puede conectar a fuentes OLAP basadas en Mondrian. Las consultas son realizadas mediante el lenguaje MDX y cuenta con componentes similares a los de JPivot, solo que su interfaz no está basada en JSP sino en Swing, por lo que es una aplicación Desktop. (39)

Integración de Datos - Se realiza con una herramienta Kettle ETL (Pentaho Data Integration) que permite implementar los procesos ETL.

1.10.2. Kettle 3.0.3

Kettle es una herramienta de código abierto compuesta por cuatro componentes fundamentales: SPOON para el diseño gráfico de las transformaciones, PAN para la ejecución de los trabajos y las transformaciones, CHEF para el diseño de la carga de datos y KITCHEN para la ejecución de los trabajos batch diseñados con CHEF. (40)

1.10.3. Schema Workbench 3.0.3

El Schema Workbench de Mondrian es un diseñador de interfaces que permite crear y configurar un cubo OLAP. El motor de Mondrian procesa las consultas MDX utilizando ROLAP. Los esquemas con los que trabaja son simplemente ficheros XML con toda la información acerca de los datos, los cuales serán utilizados por Mondrian para formar la estructura del cubo. Los ficheros XML pueden ser considerados como estructuras semejantes a las creadas anteriormente durante el diseño de la tabla de hechos y las dimensiones que conforman el almacén. Es importante conocer que no es necesario construir un cubo físico, solo basta con crear el modelo con los metadatos o sea con la información acerca de los datos. (41)

1.10.4 PostgreSQL

Está considerado el SGBD de software libre más avanzado del mundo, PostgreSQL proporciona un gran número de características que normalmente sólo se encontraban en los SGBD comerciales de alto calibre tales como Oracle. Es un SGBD capaz de manejar complejas rutinas y reglas. Su avanzada funcionalidad se pone de manifiesto con las consultas SQL declarativas, el control de concurrencia multiversión, soporte multiusuario, transacciones, optimización de consultas, herencia y valores no atómicos (atributos basados en vectores y conjuntos). Sirve de soporte a los lenguajes más populares como PHP, C, C++, Java, Python, Ruby, entre otros, además de soportar un número ilimitado de bases de datos. En resumen, tanto MySQL como PostgreSQL cuentan con un magnífico soporte para la Web. El uso de MySQL se ve favorecido cuando el factor velocidad es primordial, mientras que PostgreSQL inclina la balanza a su favor cuando se busca la estabilidad, integridad y seguridad del sistema, cualidades estas últimas deseadas para dar solución al problema de la investigación.

Por tanto, se selecciona **PostgreSQL** como SGBD atendiendo a las características planteadas anteriormente.

1.10.5 Rational Rose

Rational Rose es una herramienta software para el modelado visual mediante UML de sistemas software. Permite Especificar, Analizar y Diseñar el sistema antes de codificarlo, mantiene la consistencia de los modelos del sistema software, chequea la sintaxis UML, permite generar documentación automáticamente y generar código a partir de lo modelos. Entre otras de sus características están:

- ❖ Soporte para análisis de patrones ANSI C++, Rose J y Visual C++ basado en "Design Patterns: Elements of Reusable Object-Oriented Software".
- ❖ Característica de control por separado de componentes modelo que permite una administración más granular y el uso de modelos.
- ❖ Soporte de ingeniería Forward y/o reversa para algunos de los conceptos más comunes de Java 1.5.
- ❖ La generación de código Ada, ANSI C ++, C++, CORBA, Java y Visual Basic, con capacidad de sincronización modelo- código configurables.
- ❖ Soporte Enterprise Java Beans™ 2.0.
- ❖ Capacidad de análisis de calidad de código.
- ❖ El Add-In para modelado Web provee visualización, modelado y las herramientas para desarrollar aplicaciones de Web.
- ❖ Modelado UML para trabajar en diseños de base de datos, con capacidad de representar la integración de los datos y los requerimientos de aplicación a través de diseños lógicos y físicos.
- ❖ Capacidad de crear definiciones de tipo de documento XML (DTD) para el uso en la aplicación
- ❖ Integración con otras herramientas de desarrollo de Rational.
- ❖ Capacidad para integrarse con cualquier sistema de control de versiones SCC-compliant, incluyendo a Rational ClearCase.
- ❖ Publicación web y generación de informes para optimizar la comunicación dentro del equipo.

1.11 Conclusiones

En este capítulo se realizó un estudio teórico y conceptual sobre Data Warehouse, a partir de un conjunto de libros y artículos especializados. Sobre la base de dicho estudio se decidió detallado usar para la implementación del Data Mart el enfoque de Ralph Kimball, pues es menos costoso, es más funcional, se pueden priorizar las áreas más críticas, además su estructura de datos ofrece una mayor facilidad al usuario para la exploración y búsqueda de información. Para el desarrollo del Data Mart se seleccionó la arquitectura de datos de dos capas, como gestor de Base de Datos se escogió PostgreSQL, como herramienta de desarrollo para el Data Mart se seleccionó la Suite de BI Pentaho por ser de código abierto, ser actualmente la líder en los temas de depósitos de datos y ser una plataforma orientada a la solución y centrada en procesos.

Capítulo 2. Diseño del Data Mart

2.1 Introducción

Llevar a cabo el proceso de diseño de un Data Mart constituye una tarea bastante difícil y engorrosa puesto que en ella están implícitas un conjunto de elementos arquitectónicos, los cuales se deben conjugar satisfactoriamente para lograr un ensamblado adaptable a las necesidades de la empresa, así como también constituye un hito importante la efectiva definición de los procesos de negocio, dimensiones, medidas, cubos de datos, granularidad y modelos dimensionales, para entonces efectuar eficientes procesos de extracción, transformación y carga de los datos, hasta la presentación al cliente de la información requerida. Es por eso que entran a jugar un conjunto de factores claves para el logro de tales metas, como un dominio amplio y extenso de lo que necesita saber el cliente, el cómo debe ser mostrada la información para que se puedan realizar a partir de esta los análisis precisos y las deducción correctas de los datos almacenados, en tanto que estos adquieren la connotación de información una vez que se muestra en forma de reporte, y dicha información se convierte en conocimiento, pues se aprovecha en el proceso de toma de decisiones gerenciales.

2.2 Descripción detallada del Data Mart

Se pretende construir un Data Mart sobre el control de la deuda externa del Banco Nacional de Cuba para facilitar el proceso de toma de decisiones de los directivos, el mismo debe permitir análisis dinámicos del comportamiento de las deudas del BNC con otros bancos extranjeros. Una deuda está relacionada con un concepto de cuenta, con un tipo de moneda específica y con un banco de un país determinado. Un país puede tener varios bancos con los cuales Cuba está endeudada. La deuda se incrementa a partir de los intereses que se generan en el tiempo. La historia de las deudas dígame importe e intereses, se encuentra almacenada en la base de datos relacional que es gestionada mediante el SABIC. El BNC se encarga de decidir qué deuda saldar partiendo de su importe y los intereses que se generen en el tiempo. El Data Mart permitirá el análisis del importe de las deudas de varios bancos, que pertenecen a uno o varios países en el tiempo, para ver en qué medida aumenta o disminuye a partir de los intereses y de los pagos que se realizan respectivamente.

2.3 Aplicación del método DWEP

El DWEP es un método global para llevar a cabo el diseño de todas las fases de los almacenes de datos, incluyendo las fuentes de datos operacionales, los procesos ETL y el propio esquema del almacén de datos. A continuación se mostrarán las fases del ciclo de vida del Data Mart para el análisis y control de las deudas del BNC con sus respectivos diagramas.

Para la realización de dichos diagramas se hace uso de un Perfil UML para Modelación Dimensional que permite modelar las principales propiedades dimensionales de un almacén de datos. Este perfil es creado por Sergio Lujan Mora y Juan Trujillo, (42), el cual permite llevar a cabo el modelado del DWH utilizando la metodología DWEP (Data Warehouse Engineering Process).

2.4 Requerimientos

En el flujo de Requerimiento se definen las principales medidas, las agregaciones más importantes, las dimensiones, etc. Estos requisitos se obtuvieron mediante el análisis de los reportes sobre el control de deudas solicitados por los clientes en el flujo de trabajo Requerimiento en la fase de Inicio del proyecto SAGEB y analizando la base de datos del SABIC.

2.5 Análisis

El flujo de trabajo análisis tiene como entrada los requisitos obtenidos en el flujo anterior y los objetivos del mismo son refinar y estructurar estos requerimientos y definir las fuentes de datos operacionales que servirán como surtidor al Data Mart, en este caso será la Base de datos del SABIC montada en SQL server 2000. La modelación de la fuente de datos en diferentes niveles de detalles se realizará en el esquema conceptual (SCS) y en el esquema lógico (SLS).

2.6 Proceso de diseño de un Data Mart

El proceso de diseño del almacén de datos puede dividirse en tres etapas secuenciales: diseño conceptual, diseño lógico y diseño físico, (43). Pero las características de los almacenes de datos hacen que las estrategias de diseño para las bases de datos operacionales generalmente no sean aplicables para el diseño de DWH. En la etapa de diseño conceptual se construye un esquema conceptual de la realidad a partir de los requerimientos y/o bases fuentes, a partir de él se genera un esquema lógico, que es dependiente del tipo de modelo y tecnología de sistema de gestión de base de datos. Por último, en la etapa de diseño físico se implementa el esquema lógico en el manejador de bases de datos elegido.

2.7 Modelado de datos

Tanto en las bases de datos tradicionales como en los Data Warehouse existen tres niveles de modelado de datos: conceptual, lógico y físico. Las diferencias entre los DWH con las bases de datos operacionales en cuanto al tipo de consultas y rendimiento esperado, hacen que las estrategias de diseño y los modelos de datos utilizados para el DW sean diferentes.

2.7.1 Modelo de datos conceptual

El modelo conceptual captura la información fundamental acerca de las entidades del dominio del problema y sus relaciones. Este modelo es más cercano al espacio del problema que al espacio de la solución.

2.7.2 Modelo de datos lógico

El modelo lógico describe los datos en detalle, generalmente incluye todas las entidades y relaciones entre ellas, sus atributos y tipos de datos, así como las llaves primarias y extranjeras, sin tener en cuenta cómo ellos se implementarán físicamente en la base de datos. Es un puente entre el nivel conceptual y el físico.

2.7.3 Modelo de datos físico

Este modelo describe las estructuras de almacenamiento y los métodos usados para tener un acceso efectivo a los datos.

En el mundo de hoy en las empresas se torna un poco distintivo las características durante la construcción de los DWH en cuanto a la arquitectura definida por los especialistas durante su elaboración debida a que la misma necesita ser construido según la necesidad de la compañía. A continuación se podrá ver lo referido acerca de la arquitectura.

2.8 Esquema conceptual del Almacén de datos.

El principal objetivo del esquema conceptual del almacén de datos departamental es seleccionar los puntos relevantes para la correcta toma de decisiones y determinar cómo utilizarlos en medidas y/o dimensiones.

La metodología DWEP propone dividir el proceso del diseño conceptual del almacén de datos en tres niveles:

Nivel 1: Definición del Modelo. Un paquete representa un esquema estrella de un modelo multidimensional. En este nivel, una dependencia entre dos paquetes indica que los esquemas estrellas comparten al menos una dimensión.

Nivel 2: Definición de un esquema estrella. Un paquete representa un hecho o una dimensión de un esquema estrella. En este nivel, una dependencia entre dos paquetes de dimensión indica que las dimensiones comparten al menos un nivel en sus correspondientes jerarquías.

Nivel 3: Definición de un hecho o una dimensión. Se compone de un conjunto de clases que representan los niveles jerárquicos en un paquete de dimensión o el esquema estrella completo en el caso de un paquete de hecho.

El Data Mart está compuesto por una tabla de hecho: Tbl_Deuda, y por cuatro dimensiones: Dimension_Banco, Dimension_Cuenta, Dimension_Moneda y Dimension_Tiempo.

En la Figura siguiente se muestra el esquema conceptual del Data Mart en el nivel 1, en él se representa el diagrama en estrella que forma al Data Mart: Schema_Deuda.

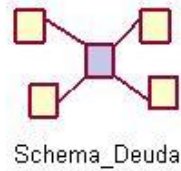


Fig. 2.1 Esquema conceptual del almacén de datos

En la Fig. 2.2: Se muestra el esquema conceptual del paquete Schema_Deuda, en él se observa la tabla de hechos TH_Deudas y las dimensiones que integran este esquema estrella: Dimension_Cuenta, Dimension_Moneda, Dimension_Tiempo y Dimension_Banco.

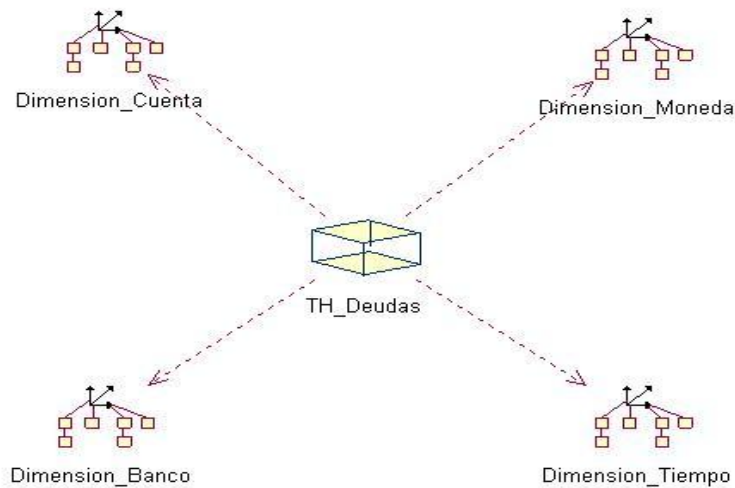


Fig. 2.2 Esquema conceptual del almacén de datos TH_Deudas.

En la siguiente Fig. 2.3 se muestra el esquema conceptual del almacén de datos Dimension_Banco con sus respectivos atributos.

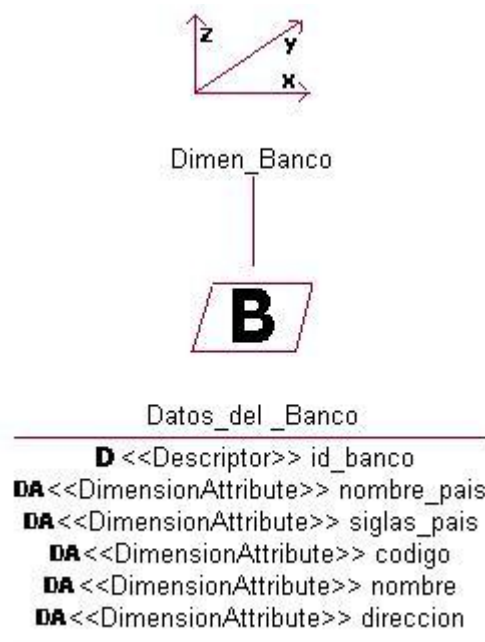


Fig. 2.3 Esquema conceptual de datos, Dimension_Banco

En los anexos 3.4, 3.5, y 3.6 se muestran los contenidos de las restantes dimensiones: Dimension_Moneda, Dimension_Cuenta, y Dimension_Tiempo respectivamente

También se expone en la Fig. 2.4 el contenido del esquema conceptual en el cual la tabla hecho es definida con sus correspondientes medidas (importe e interés) y se representa además las clases dimensiones con sus niveles de jerarquías.

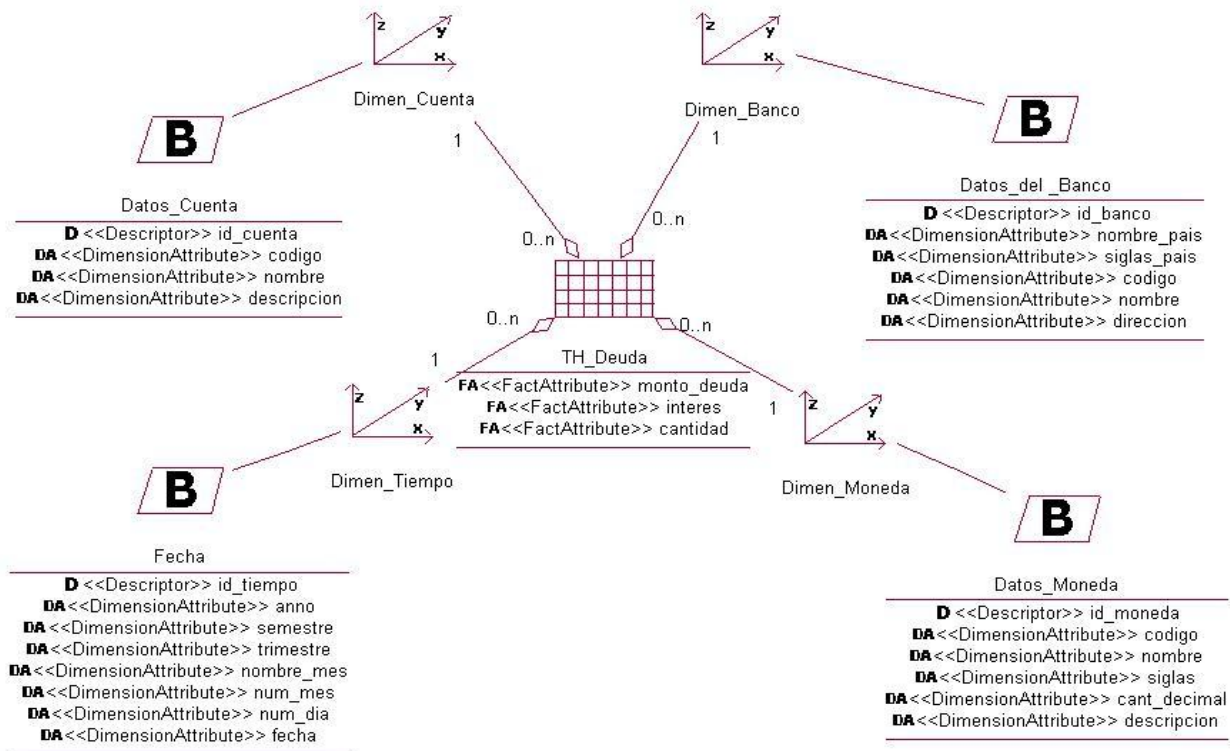


Fig. 2.4 Esquema conceptual del DWH

2.9 Mapeo de datos.

Para el proceso de colmar un almacén de datos departamental se necesita llevar a cabo el proceso de Mapeo de datos. Para representar este flujo de datos desde las diversas fuentes hacia el Data Mart se utilizará el diagrama de mapeo de datos (Data Mapping), con varios niveles de detalle. Debido a que este puede tornarse muy complejo, el autor (LUJAN-MORA 2005) en su propuesta lo divide en cuatro niveles:

Nivel de base de datos o Nivel 0: En este nivel cada esquema del almacén de datos se representa mediante un paquete. Los mapeos entre los diferentes esquemas se modelan en un único paquete de mapeo, que encapsula todos los detalles.

Nivel de flujo de datos o Nivel 1: Este nivel describe las relaciones de datos a nivel individual entre las fuentes de datos hacia los respectivos destinos en el almacén de datos.

Nivel de tabla o Nivel 2: Mientras que el diagrama de mapeo en el nivel 1 describe las relaciones entre las fuentes y los destinos de datos mediante un único paquete, el diagrama de mapeo de datos en el nivel de tabla detalla todas las transformaciones intermedias que tienen lugar durante ese flujo.

Nivel de atributo o Nivel 3: En este nivel, el diagrama de mapeo de datos captura los mapeos existentes a nivel de atributo.

Esquema lógico de la fuente (SLS)

Para realizar el esquema lógico de la fuente se toma como entrada el esquema conceptual de la fuente visto anteriormente, en ellos se representan las tablas banco, la tabla cuenta, la tabla moneda, la tabla país y la tabla diario con sus respectivos atributos. Observar la Fig. 2.5

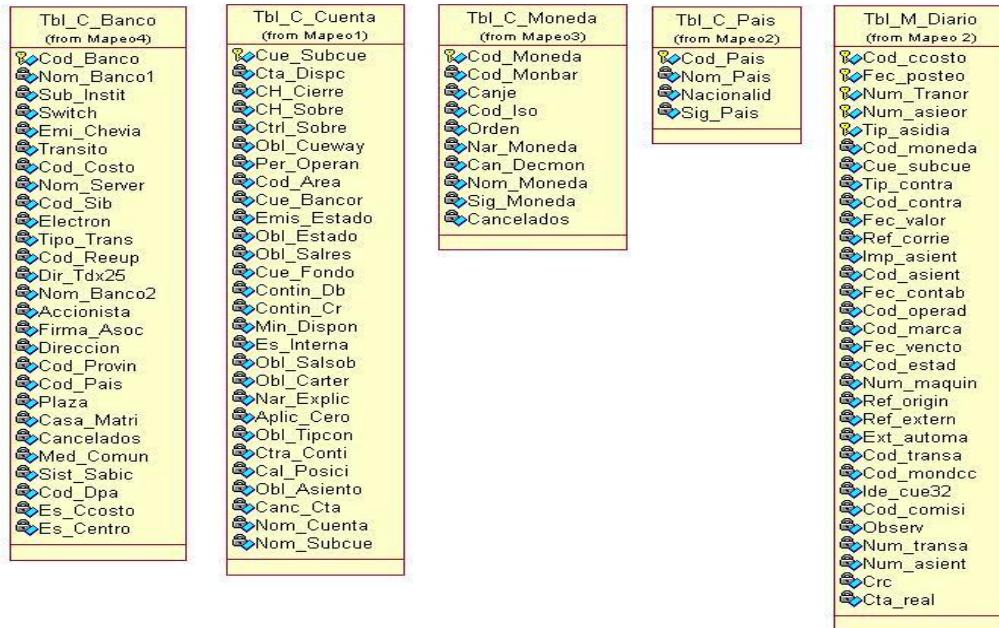


Fig. 2.5 Esquema Lógico de la fuente (SCS)

En la Fig. 2.6 se puede consultar el nivel 0 del mapeo de datos representado por un paquete llamado Data Mapping relacionado con el esquema conceptual de la fuente (SCS) y el esquema conceptual del DWH.

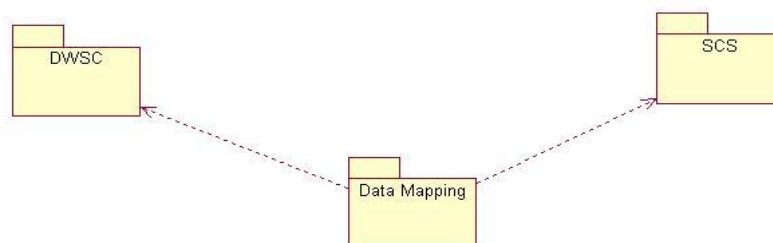


Fig. 2.6 Mapeo nivel 0

El Data mart para el análisis de las deudas del BNC posee 5 tablas que se quieren poblar: 1 de hecho y 4 de dimensiones, por lo que existen 4 escenarios (nivel 1) dentro del paquete Mapping, uno para cada uno de las tablas. Este nivel 1 representa el flujo de datos existente entre la fuente y el destino de datos en el contexto de cada escenario, descrito en la Fig. 2.7.



Figura 2.7 Mapeo de datos nivel 1

Para adquirir una visión más detallada del Mapeo de datos se puede acudir a la figura 2.8 que muestra las peculiaridades del Mapeo de las tablas C_Banco y C_Pais, del cual se necesitan mapear los atributos Nom_Pais, Sig_Pais, Cod_Banco, Nom_Banco1 y Dirección(nivel 2).

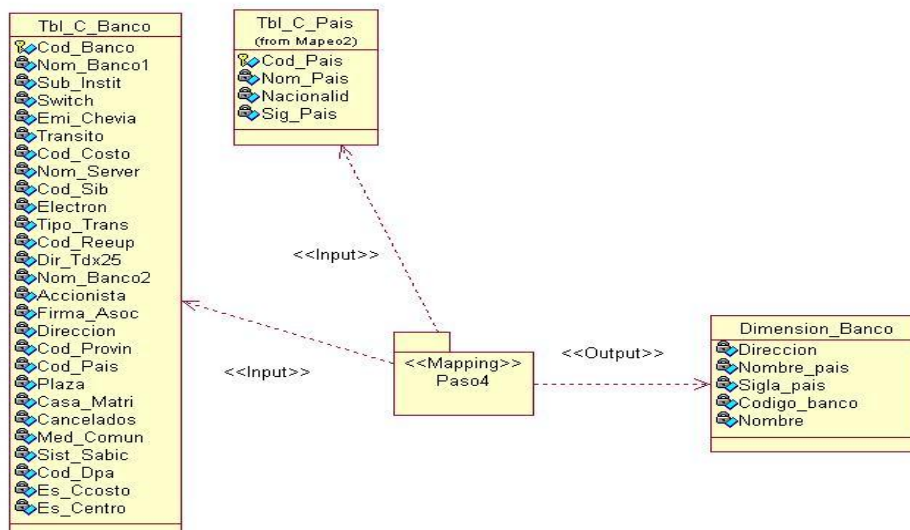


Fig. 2.8 Mapeo 2 de la tabla Banco y País (nivel 2)

En los anexos 3.10, 3.11 y 3.13 se muestran los contenidos de las restantes tablas: la tabla _moneda, la tabla _banco y la tabla _Tiempo respectivamente.

En la Fig. 2.9 se realiza el mapeo (nivel 3) de la tbl_C_Banco y la tbl_Pais de donde se mapean los atributos Nom_Pais, Sig_Pais, Cod_Banco, Nom_Banco1 y Dirección.

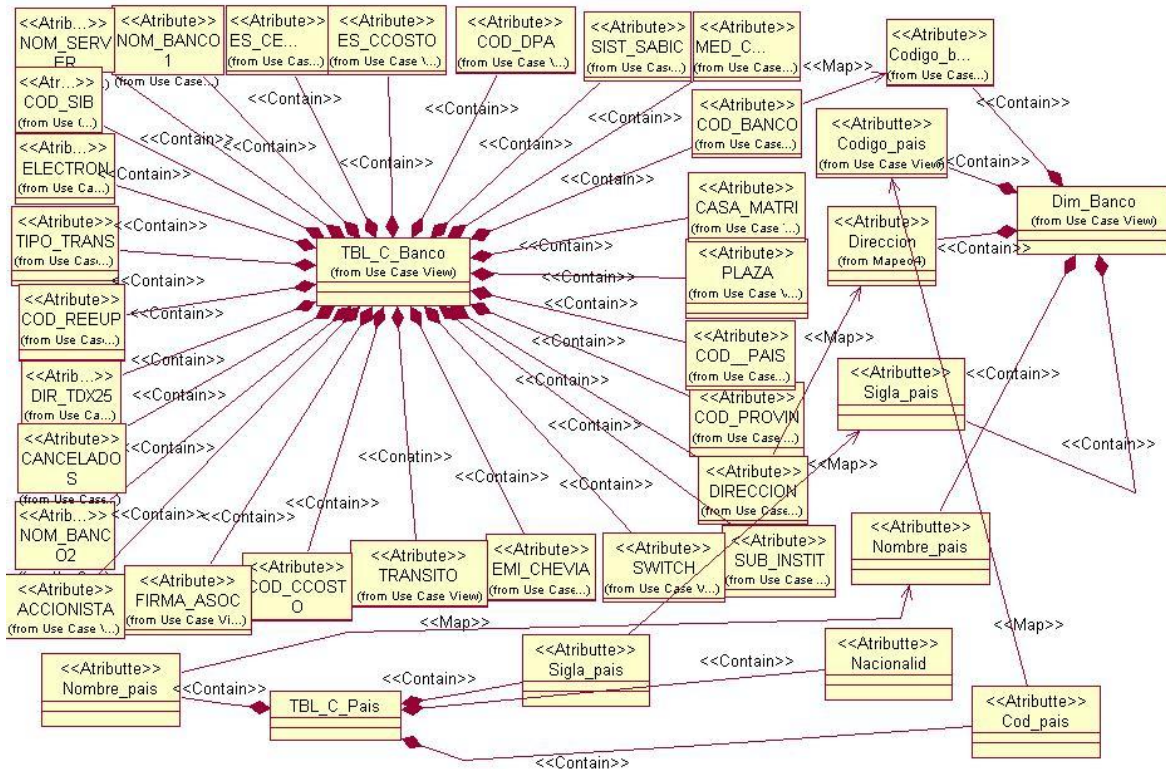


Fig. 2.9 Mapeo 1 de la tabla banco y país (nivel 3)

En los anexos 3.14, 3.15 y 3.16 se muestran los contenidos de los restantes mapeos de tablas: la tabla_moneda, la tabla _banco y la tabla _Tiempo respectivamente.

2.10 Conclusiones

En este capítulo se describió de forma detallada el Data Mart asociado al control de las deudas externas del BNC, se hizo uso de la metodología DWEP para el diseño y la implementación del Data Mart, de la cual se explican algunos pasos que se realizaron durante el diseño.

Capítulo 3. Implementación del Data Mart

3.1 Introducción

Con la introducción a este capítulo se procederá a la implementación detallada del Data Mart destinado al control y análisis de las deudas en el Banco Nacional de Cuba, en este capítulo se procederá a construir las estructuras físicas del Data Mart, se definirán los tipos de modelamiento de datos ya sea relacional o multidimensional, se conciben los esquemas físicos del DM, se llevan cabo las ETL y se generaran los cubos de Datos.

3.2 Proceso ETL

Primeramente se construirá la base de datos que soportará las dimensiones y contendrá los datos después de su extracción y posterior a esto se llevará a cabo las transformaciones de datos desde los sistemas operacionales que se desean integrar. Para llevar a cabo este proceso se trabajó con la herramienta descrita anteriormente: la suite Pentaho, Kettle.

3.3 Edificación del sistema de almacenamiento de datos.

Como primer paso se necesita el montaje en el Gestor de base de datos PostgreSQL utilizando el diseño elaborado y terminado anteriormente, lo que conlleva a crear todas las tablas con sus respectivos atributos y relaciones y de esta manera construir una estructura de almacenamiento adecuado para los procesos de Transformación y Carga de los datos. (44)

3.4 Configuración de la extracción y transformación de datos.

Debido a que los datos deberán ser extraídos, transformados, depurados y cargados del sistema operacional al Data Mart, es imprescindible conocer cómo se realizarán cada una de estas actividades. En este sentido, se hace necesario controlar desde dónde y en qué tiempo se realizará la extracción de los datos, qué herramienta se utilizará, en este caso el Kettle de Pentaho así como dominar sus potencialidades, de qué tablas y cuáles datos se extraerán. En cuanto a la transformación de los datos, esta se hará de acuerdo a las reglas o parámetros que se hayan definido en el negocio.

En la siguiente transformación se comienza utilizando el paso del Kettle **Entrada Tabla**, ver anexo 3.18, para especificar la tabla y los campos orígenes. En el caso de esta transformación, luego de tener definida la entrada de los datos se hace una búsqueda de varios valores en otra tabla mediante el paso **Búsqueda en Base de Datos**, ver anexo 3.19. Una vez que se tienen todos los campos necesarios de la base de datos origen, se procede a filtrar las filas por la columna Cod_Pais mediante el paso **Filtrar Filas**, ver anexo 3.20, para seleccionar solamente todos los bancos extranjeros, luego se crea una secuencia mediante el paso **Añadir Constante**, ver anexo 3.21, donde los identificadores de las dimensiones son valores generados por una secuencia que se encuentra en la base de datos.

Cuando se tienen todos los datos de la dimensión seleccionados se procede a mapearlos con los del paso **Salida Tabla**, ver anexo 3.22, mediante el paso **Selecciona/Renombra Valores**, ver anexo 3.23, para así de esta forma terminar la transformación.

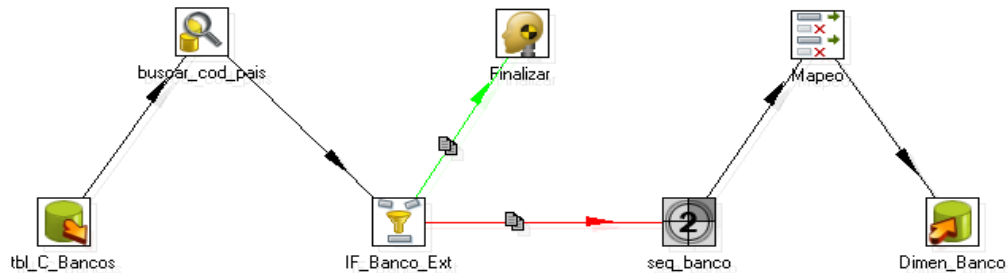


Figura: 3.1 Transformación ETL_Banco

La transformación ETL _Tiempo es la encargada de cargar las fechas de la tabla M_Diario de la fuente de datos. Estas fechas representan el año, mes, y día de cuándo se contrajo la deuda, se utiliza un nuevo paso llamado **Valor de Java Script Modificado** componente para validar los campos de la fecha y obtener el semestre, trimestre y el nombre del mes, luego se obtiene la secuencia de la dimensión tiempo para generar su identificador en la base datos, seguido a este paso se procede a mapear los datos y por último se crea el paso de salida **Insertar/Actualizar Tiempo** para que agrupe todas las fechas idénticas.

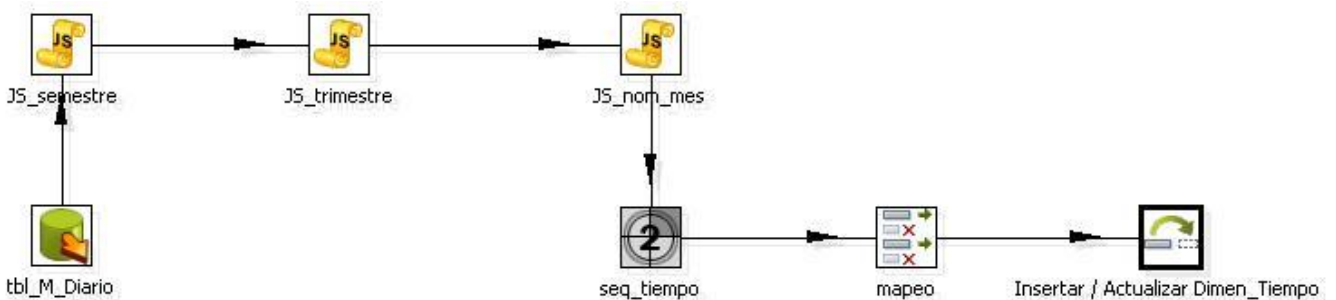


Figura: 3.2 Transformación ETL_Tiempo

Una de las mayores complejidades de los procesos ETL no es más que la carga de los datos en la tabla de hechos. En dicha transformación se necesita utilizar Script de Java para obtener algunos valores mediante una función de java script y con ellas poblar la tabla de hechos y relacionar todas sus dimensiones. Como se observa en la figura siguiente.

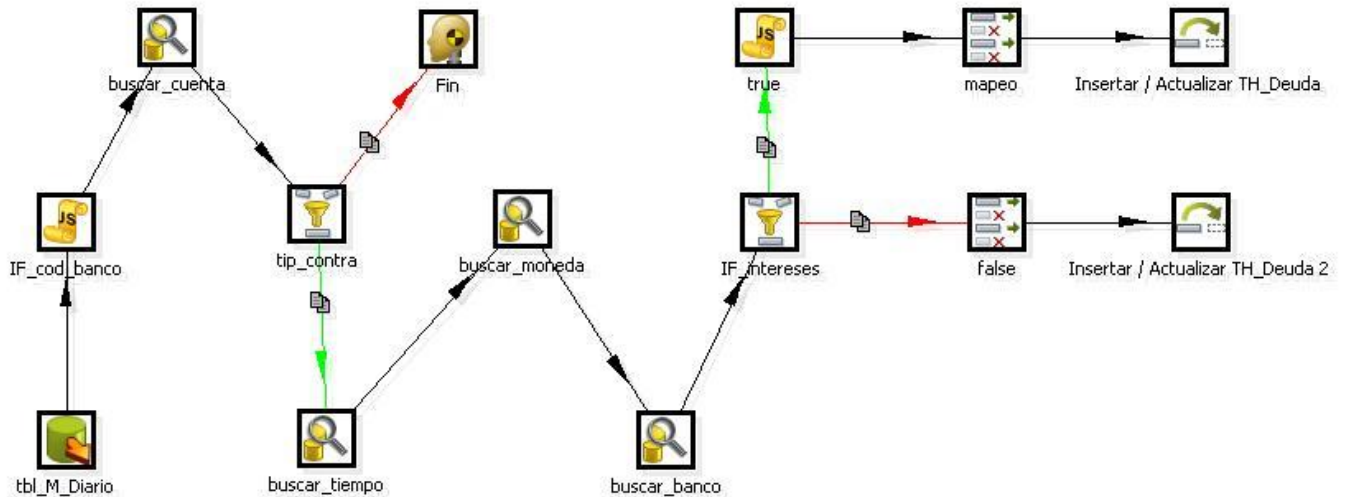


Figura: 3.3 Transformación ETL_Hecho_Deuda

En los anexos (3.25 y 3.26) se muestran los contenidos de las restantes transformaciones respectivamente.

3.5 Cubo de Datos

Un cubo es una estructura multidimensional que contiene dimensiones y medidas. Las dimensiones definen su estructura mientras que las medidas le proporcionan los valores numéricos de interés para el usuario final. Cada celda del cubo es definida por la intersección de las dimensiones, y el valor numérico almacenado constituye un agregado que se almacena en cada celda (45)

3.6 Creación de los cubos de Datos

Para la creación de los cubos de datos primeramente se necesita tener instalado la Suite de Pentaho con todos sus componentes, del cual se utilizará una de sus principales herramientas para el diseño y creación de los cubos de datos el Shema Workbench.

El Data Mart que se propone como solución cuenta con un solo cubo de datos llamado Deuda .En la Figura 3.4 se puede apreciar cómo se va construyendo el cubo de datos. Primeramente se diseñan la tabla de hechos y seguidamente las dimensiones con todos los niveles de jerarquías que posee.

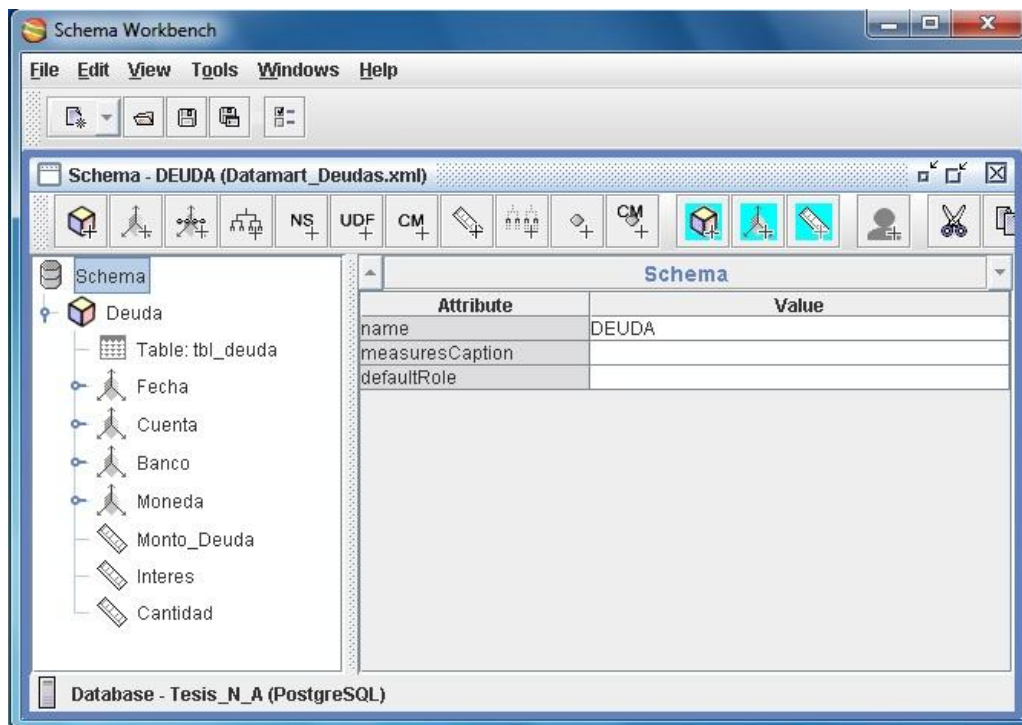
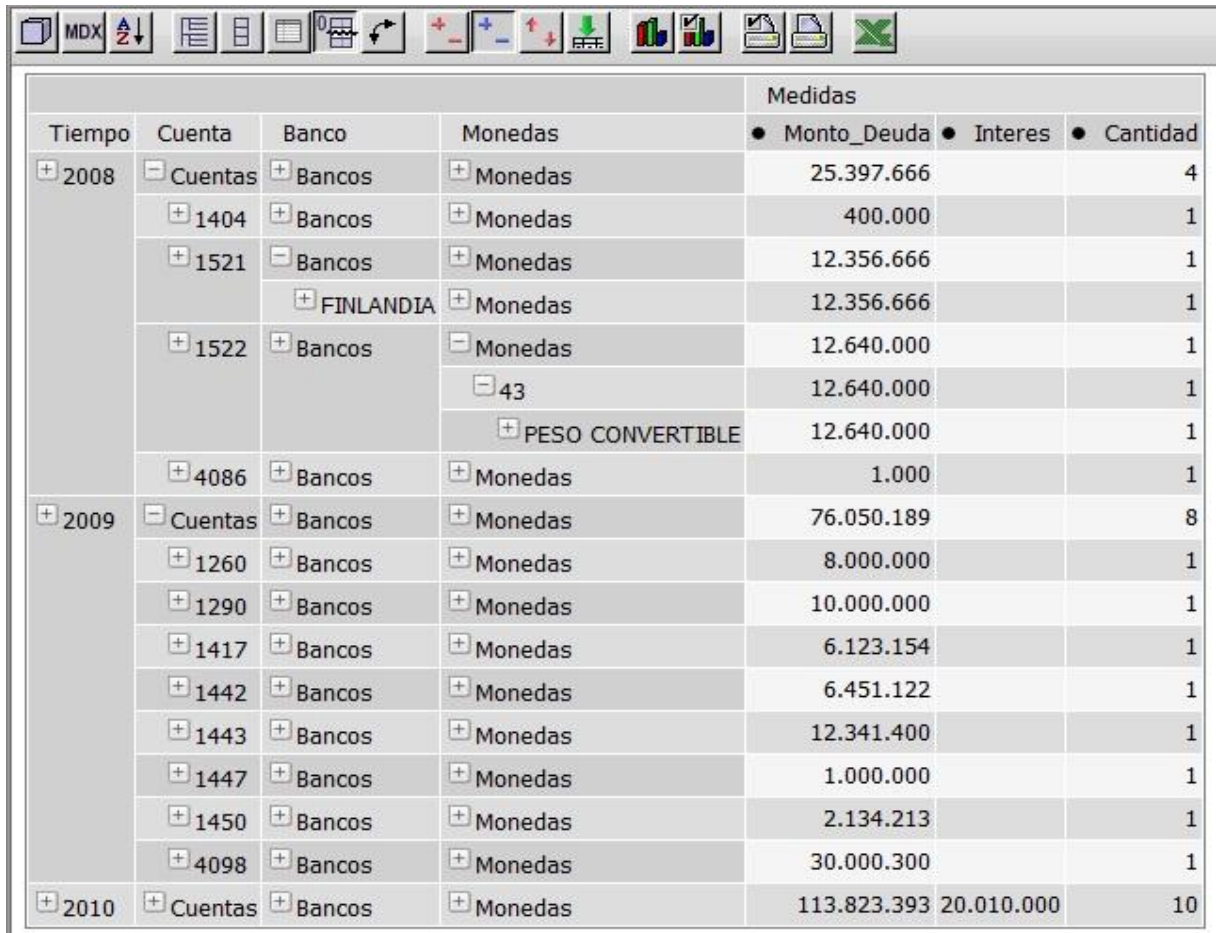


Figura 3.4 Creación del cubo de datos, dimensiones y jerarquías

3.7 Mondrian.

Al tener diseñado el cubo, se almacena en un archivo XML que se cargará posteriormente en el Mondrian, herramienta utilizada por el usuario para realizar el Proceso Analítico en Línea (OLAP). En la figura siguiente se muestra el cliente JPivot, disponible a través de la web, cuya página principal muestra un menú con todas las opciones que permite el filtrado por los campos que se desee, ajustar la visión de los datos, el empleo de filtros de visualización, añadir o quitar campos, observar la información desde diferentes perspectivas, cuyos resultados serán mostrados mediante tablas o gráficos y exportarlos a formato PDF o EXCEL.



Tiempo	Cuenta	Banco	Monedas	Medidas		
				● Monto_Deuda	● Interes	● Cantidad
+ 2008	- Cuentas	+ Bancos	+ Monedas	25.397.666		4
	+ 1404	+ Bancos	+ Monedas	400.000		1
	+ 1521	- Bancos	+ Monedas	12.356.666		1
		+ FINLANDIA	+ Monedas	12.356.666		1
	+ 1522	+ Bancos	- Monedas	12.640.000		1
			- 43	12.640.000		1
			+ PESO CONVERTIBLE	12.640.000		1
	+ 4086	+ Bancos	+ Monedas	1.000		1
+ 2009	- Cuentas	+ Bancos	+ Monedas	76.050.189		8
	+ 1260	+ Bancos	+ Monedas	8.000.000		1
	+ 1290	+ Bancos	+ Monedas	10.000.000		1
	+ 1417	+ Bancos	+ Monedas	6.123.154		1
	+ 1442	+ Bancos	+ Monedas	6.451.122		1
	+ 1443	+ Bancos	+ Monedas	12.341.400		1
	+ 1447	+ Bancos	+ Monedas	1.000.000		1
	+ 1450	+ Bancos	+ Monedas	2.134.213		1
+ 4098	+ Bancos	+ Monedas	30.000.300		1	
+ 2010	+ Cuentas	+ Bancos	+ Monedas	113.823.393	20.010.000	10

Figura 3.5 Cliente de Pívor que muestra el cubo de Datos

En la figura 3.5 se puede observar las relaciones entre los de datos de cada dimensión y medida representada, facilitando el análisis de los mismos a un nivel más complejo. En el anexo 3.32 se muestra representado gráficamente por el JPivot el contenido de cada medida paralelo al año en que se sostuvo el pedido del Importe.

3.8 Despliegue de la solución propuesta.

Los diagramas que ofrece la metodología DWEP para este flujo son: el esquema físico de la fuente de datos, el esquema físico del almacén de datos y el diagrama de integración de transporte.

3.9 Esquema físico de la fuente.

Este esquema detalla, desde el punto de vista físico, el origen de los datos del Data Mart. Se muestra en el anexo 3.33, este está constituido por un servidor llamado SABIC_SERVER, dentro del cual se encuentra el componente database DEUDA_DMart. El nodo SABIC_SERVER tiene una serie de valores que permiten describir sus características particulares. El estereotipo <<Server>> define una computadora que ejecuta funciones servidoras, en nuestro caso el servidor de base de datos del proyecto, <<Database>> para especificar la base de datos, <<Disk>> es utilizado para representar un disco físico de la computadora, <<BusInterno>> para definir el tipo de comunicación entre dos elementos y <<Tablespace>> para definir en que tablespace se encuentra la base de datos de la fuente.

3.10 Esquema físico del almacén de datos

En este esquema se muestran los aspectos físicos de la implementación del almacén de datos, el mismo se divide en dos partes: el diagrama de componentes y el diagrama de despliegue. El diagrama de componentes es mostrado en el anexo 3.34 en el mismo se observa que el Data Mart para el análisis de la deuda externa del Banco Nacional de Cuba cuenta con dos tablespace: Hechos y Dimensiones, el primero contiene las tabla de hecho: Tbl_Deuda y el segundo alberga las dimensiones: Dim_Banco, Dim_Cuenta, Dim_Moneda, Dim_Tiempo.

En el diagrama de despliegue son especificados los diferentes aspectos relativos a la configuración del Software y del Hardware, por otro lado la distribución de las estructuras lógicas definidas previamente también son representadas. En la figura 3.35 se puede observar la configuración del servidor que alberga al DM.

3.11 Diagrama de integración de transporte.

Este diagrama muestra las estructuras físicas de los procesos ETL utilizados para poblar el almacén de datos. En el anexo 3.36 se muestran los dos servidores de base de datos necesarios para completar los procesos: el de la fuente, y el del almacén de datos y el servidor de aplicación del Pentaho donde se ejecutan los procesos.

3.12 Conclusiones

Durante el desarrollo de este capítulo se realizó de forma íntegra la implementación del Data Mart donde se hizo uso de la metodología escogida en conjunto con las herramientas de desarrollo y se creó una secuencia de pasos sobre la fase de implementación, ya sea en la creación de las ETL como en la creación del cubo de datos.

CONCLUSIONES

Con este trabajo de diploma se desarrolló la propuesta de un Data Mart para el análisis de la deuda externa en el Banco Nacional de Cuba con el objetivo de apoyar a los directivos del mismo en el proceso de toma de decisiones.

Para dar cumplimiento a esto:

Se efectuó un estudio sobre las herramientas, tecnologías y tendencias actuales, propuestas para el desarrollo de Data Warehouse.

Se definieron y materializaron los artefactos generados usando la metodología DWEP.

Se llevó a cabo el proceso de extracción, transformación y carga de datos para filtrar, limpiar, homogenizar y agrupar la información proveniente de la fuente de datos del Data Mart.

Se realizó el procesamiento analítico en línea OLAP para analizar el negocio desde diferentes escenarios históricos y proyectar como se ha comportado y evolucionado en un ambiente multidimensional.

Al tratarse el desarrollo de este Data Mart como un proceso iterativo e incremental, se podrán incluir en el futuro nuevas necesidades, procesos de negocios y cambios que surjan en el Banco Nacional.

RECOMENDACIONES

- Desarrollar y motivar a mayor escala el estudio y la investigación sobre los Data Mart.
- Proponer la construcción de almacenes de datos para otros módulos del Banco Nacional de Cuba.
- Incentivar la idea de la construcción de los Data Mart a los proyectos productivos que se desarrollan en la universidad.
- Proponer la presentación del trabajo en eventos científicos.

BIBLIOGRAFÍA

1. Autores, Colectivo de. Importancia de la utilización de un Data Warehouse (DW) en las empresas. <http://www.bibliociencias.cu/gsd/collect/libros/index/assoc/HASH0106/b6fac6b9.dir/doc.pdf>
2. Idem [1]. [Online]
3. INMON, W. H. *Building the data warehouse*. New York, USA, QED Information Sciences, Inc, 1992. 272p. 0471569607
4. Data Warehousing. [En línea] <http://www.sqlmax.com/dataw1.asp>
5. Data mart: http://www.sinnexus.com/business_intelligence/datamart.aspx
6. KIMBALL, R. and J. CASERTA. *The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming, and Delivering Data*. Indianapolis y Canadá, Wiley Publishing, Inc., 2004. 525 p. 0764567578
7. INMON, W. *Building the Data Warehouse*. 3ra. John Wiley & Sons, Inc, 2002. 432 p. 0471081302
8. INMON, W. *Building the Data Warehouse*. 4ta. Indianapolis y Canada, Wiley Publishing, Inc., 2005. 543 p.9780764599446
9. Idem [6]. [Online]
10. Data Mart. <http://www.dataprix.com/data-warehousing-y-metodologia-hefesto/i-data-warehousing-investigacion-y-sistematizacion-concepto-13>
11. Data Warehouse. Ventajas y Desventajas. <http://www.dataprix.com/data-warehousing-y-metodologia-hefesto/i-data-warehousing-investigacion-y-sistematizacion-conceptos-5>
12. ARQUITECTURA DEL DATA WAREHOUSING. <http://www.dataprix.com/es/arquitectura-del-data-warehouse>.
13. Velázquez Vidal, Lis. Veliz Monteagudo, Mijail. Estudio Teórico-Conceptual sobre Data Warehouse.
14. PONNIAH, P. *Data Warehousing Fundamentals: A Comprehensive Guide for IT Professionals*. New York, Chichester, Weinheim, Brisbane, Singapore y Toronto, John Wiley & Sons, Inc., 2001. 518 p.0471412546.
15. JARKE, M.; M. LENZERINI, et al. *Fundamentals of Data Warehouses*. 2da. Springer-Verlag, 2003
16. VIDAL, L. V. and M. V. MONTEAGUDO. Estudio Teórico-Conceptual sobre Data Warehouse. Ciudad de La Habana, Universidad de La Habana, Facultad de Matemática y Computación, 2000. 128p.
17. SACHDEVA, S. Meta Data Architecture for Data Warehousing. DM Review Magazine, 1998.
18. KIMBALL, R. Meta Meta Data Data. DBMS Magazine, 1998. 11: 18-20 p.

19. MARCO, D. Building and Managing the Meta Data Repository: A Full Lifecycle Guide. Wiley, 2000. 416 p.9780471355236.
20. Data Warehousing: Investigación y Sistematización de Conceptos. <http://www.dataprix.com/35-query-manager>
21. Idem [20]. [Online]
22. Cueto Ramón, Ariagna. García Díaz, Joannis, Implementación de un Data Warehouse para el control del Recurso Humano de la Salud. Ciudad de La Habana, Junio de 2008
23. Idem [22]. [Online]
24. Idem [22]. [Online]
25. Idem [22]. [Online]
26. Data Warehousing: Investigación y Sistematización de Conceptos. <http://www.dataprix.com/datawarehouse-manager#x1-390003.4.1>
27. CABIBBO, L. and R. TORLONE. Advances in Database Technology. 6th International Conference on Extending Database Technology, Valencia, Spain, Springer-Verlag, 1998. 183-197 p.
28. GOLFARELLI, M. and S. RIZZI. A Methodological Framework for Data Warehouse Design. Proceedings of the ACM 1st International Workshop on Data Warehousing and OLAP, Bethesda, USA, ACM, 1998. 3-9 p.
29. BLASCHKA, M.; C. SAPIA, et al. Extending the E/R Model for the Multidimensional Paradigm. Proceedings of the 1st International Workshop on Data Warehouse and Data Mining, Singapore, Springer-Verlag, 1998. 105-116 p.
30. TRYFONA, N; F. BUSBORG, et al. starER: A Conceptual Model for Data Warehouse Design. Proceeding of ACM 2nd International Workshop on Data Warehousing and OLAP, Kansas City, USA, 1999.3-8 p
31. TRUJILLO, J.; M. PALOMAR, et al. Designing Data Warehouses with OO Conceptual Models. USA, IEEE Computer Society Press 2001. 66-75 p.
32. ABELLÓ, A.; J. SAMOS, et al. YAM² (Yet Another Multidimensional Model: An Extension of UML).Proceedings of the International Database Engineering and Application Symposium, Edmonton, Canada, IEEE Computer Society, 2002. 172-181 p.
33. OMG. OMG Unified Modeling Language Specification Version 1.5. Massachusetts, Object Management Group, Inc, 2003. 736 p.
34. Gutiérrez Kafati, Elizabeth. La plataforma Pentaho Open Source Business Intelligence. http://egkafati.bligoo.com/content/view/219538/La_plataforma_Pentaho_Open_Source_Business_Intelligence.html.

35. BUSINESS INTELLIGENCE SOLUTIONS.
http://cba365.com.ar/cms/index.php?option=com_content&view=article&id=53&Itemid=65
36. Idem [37]. [Online]
37. Herramientas y suites de la inteligencia empresarial.
38. Idem [39]. [Online]
39. Idem [39]. [Online]
40. Idem [39]. [Online]
41. [Online] <http://mondrian.pentaho.org/documentation/workbench.php>.
42. BATINI, C.; S. CERI, et al. Conceptual Database Design: An Entity-Relationship Approach. Benjamin-Cummings Publishing Company 1992.
43. TRUJILLO, J.; M. PALOMAR, et al. Extending UML for Multidimensional Modeling. Proceedings of the 5th International Conference on the Unified Modeling Language, Dresden, Germany, Springer-Verlag, 2002. 290-304 p.
44. Iznaga González Yonelbys. Diseño e Implementación de un Data Warehouse para el Sistema de Gestión Estadística en Cuba Ciudad de La Habana, Junio de 2008:
45. Idem [46]. [Online]

GLOSARIO DE TÉRMINOS

DWH: Data Warehouse

ETL: Extracción, Transformación y Carga

DM: Data Mart

SABIC: Sistema automatizado para la Banca Internacional de Comercio

SQL: Lenguaje Estructurado de Consultas

MDX: Expresiones de consultas multidimensionales

HOLAP: Procesamiento híbrido analítico en línea

MOLAP: Procesamiento multidimensional analítico en línea

ROLAP: Procesamiento relacional analítico en línea

OLAP: Procesamiento analítico en línea

SQL SERVER: Sistema para la gestión de base de datos

POSTGRESQL: Sistema para la gestión de base de datos relacional orientada a objetos y de código abierto

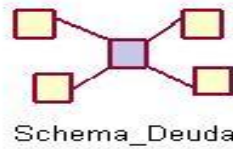
SGBD: Sistema gestor de base de datos

KETLLE: Sistema de ETL de Pentaho Data Integration

SPOON: Diseñador gráfico de las transformaciones de KETLLE

ANEXOS

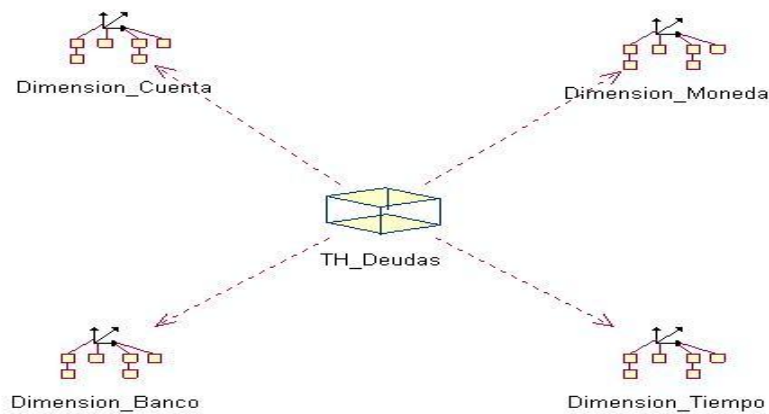
Anexo 3.1 Esquema conceptual del almacén de Datos



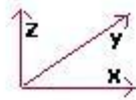
Anexo 3.2 Esquema conceptual de la fuente.

Tbl_C_Banco (from Mapeo4)	Tbl_C_Cuenta (from Mapeo1)	Tbl_C_Moneda (from Mapeo3)	Tbl_C_Pais (from Mapeo2)	Tbl_M_Diario (from Mapeo 2)
<ul style="list-style-type: none"> ↳ Cod_Banco ↳ Nom_Banco1 ↳ Sub_Instit ↳ Switch ↳ Emi_Chevia ↳ Transito ↳ Cod_Costo ↳ Nom_Server ↳ Cod_Sib ↳ Electron ↳ Tipo_Trans ↳ Cod_Reeup ↳ Dir_Tdx25 ↳ Nom_Banco2 ↳ Accionista ↳ Firma_Asoc ↳ Direccion ↳ Cod_Provin ↳ Cod_Pais ↳ Plaza ↳ Casa_Matri ↳ Cancelados ↳ Med_Comun ↳ Sist_Sabic ↳ Cod_Dpa ↳ Es_Costo ↳ Es_Centro 	<ul style="list-style-type: none"> ↳ Cue_Subcue ↳ Cta_Dispc ↳ CH_Cierre ↳ CH_Sobre ↳ Ctrl_Sobre ↳ Obl_Cueway ↳ Per_Operan ↳ Cod_Area ↳ Cue_Bancor ↳ Emis_Estado ↳ Obl_Estado ↳ Obl_Salres ↳ Cue_Fondo ↳ Contin_Db ↳ Contin_Cr ↳ Min_Dispon ↳ Es_Interna ↳ Obl_Salsob ↳ Obl_Carter ↳ Nar_Explic ↳ Aplic_Cero ↳ Obl_Tipcon ↳ Ctra_Conti ↳ Cal_Posici ↳ Obl_Asiento ↳ Canc_Cta ↳ Nom_Cuenta ↳ Nom_Subcue 	<ul style="list-style-type: none"> ↳ Cod_Moneda ↳ Cod_Monbar ↳ Canje ↳ Cod_Iso ↳ Orden ↳ Nar_Moneda ↳ Can_Decmon ↳ Nom_Moneda ↳ Sig_Moneda ↳ Cancelados 	<ul style="list-style-type: none"> ↳ Cod_Pais ↳ Nom_Pais ↳ Nacionalid ↳ Sig_Pais 	<ul style="list-style-type: none"> ↳ Cod_ccosto ↳ Fec_posteo ↳ Num_Tranor ↳ Num_asieor ↳ Tip_asidia ↳ Cod_moneda ↳ Cue_subcue ↳ Tip_contra ↳ Cod_contra ↳ Fec_valor ↳ Ref_corrie ↳ Imp_asient ↳ Cod_asient ↳ Fec_contab ↳ Cod_operad ↳ Cod_marca ↳ Fec_vencto ↳ Cod_estad ↳ Num_maquin ↳ Ref_origin ↳ Ref_extern ↳ Ext_automa ↳ Cod_transa ↳ Cod_mondcc ↳ Ide_cue32 ↳ Cod_comisi ↳ Observ ↳ Num_transa ↳ Num_asient ↳ Crc ↳ Cta_real

Anexo 3.3 Esquema conceptual del almacén de datos. Schema_Deuda



Anexo 3.4 Dimensión Moneda



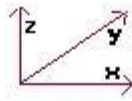
Dimen_Moneda



Datos_Moneda

D	<<Descriptor>>	id_moneda
DA	<<DimensionAttribute>>	codigo
DA	<<DimensionAttribute>>	nombre
DA	<<DimensionAttribute>>	siglas
DA	<<DimensionAttribute>>	cant_decimal
DA	<<DimensionAttribute>>	descripcion

Anexo 3.5 Dimensión Tiempo



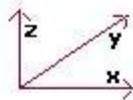
Dimen_Tiempo



Fecha

D <<Descriptor>> id_tiempo
DA <<DimensionAttribute>> anno
DA <<DimensionAttribute>> semestre
DA <<DimensionAttribute>> trimestre
DA <<DimensionAttribute>> nombre_mes
DA <<DimensionAttribute>> num_mes
DA <<DimensionAttribute>> num_dia
DA <<DimensionAttribute>> fecha

Anexo 3.6 Dimensión Cuenta



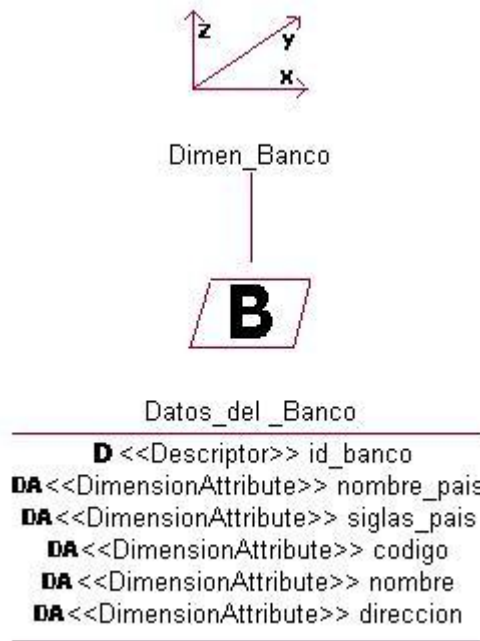
Dimen_Cuenta



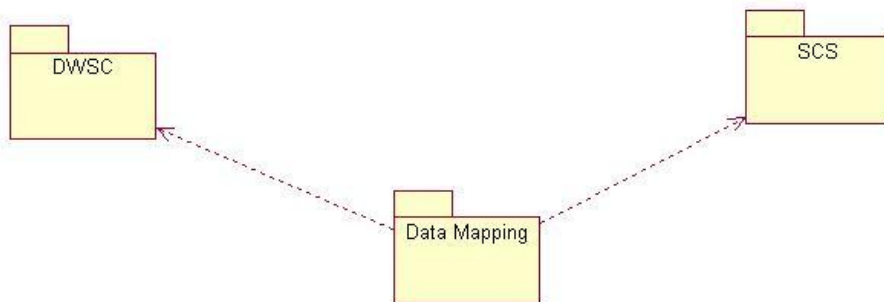
Datos_Cuenta

D <<Descriptor>> id_cuenta
DA <<DimensionAttribute>> codigo
DA <<DimensionAttribute>> nombre
DA <<DimensionAttribute>> descripcion

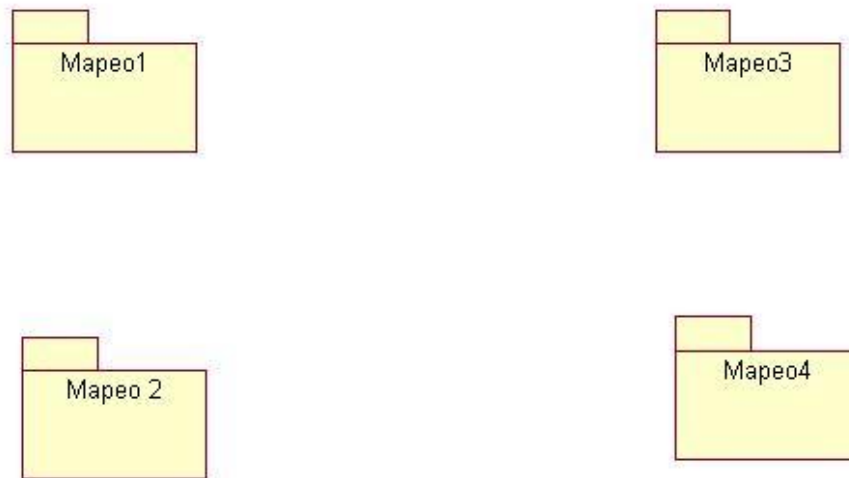
Anexo 3.7 Dimensión_Banco



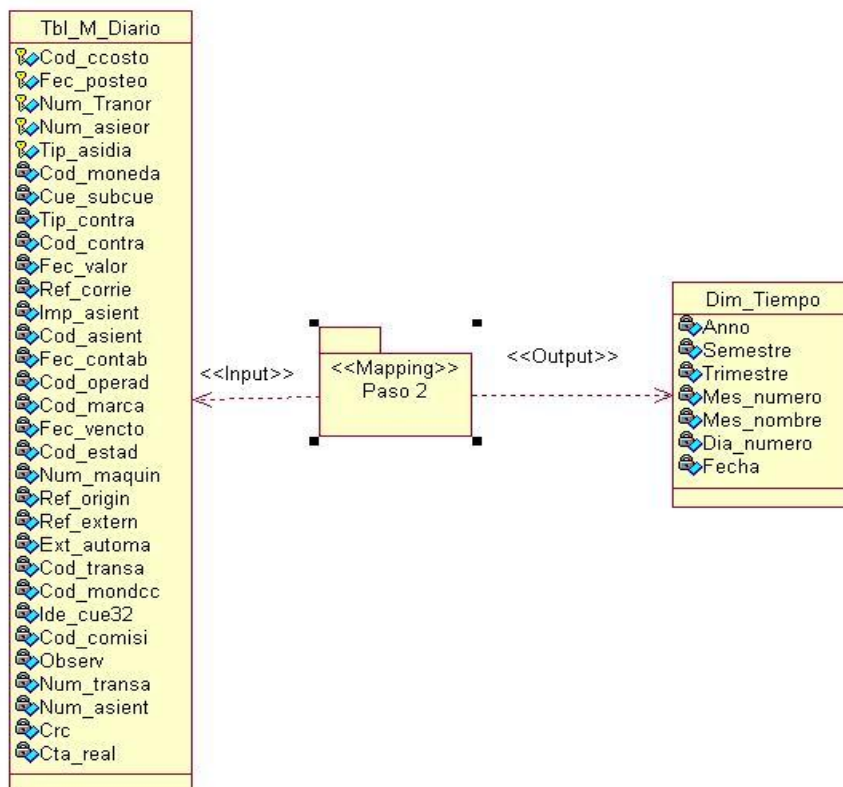
Anexo 3.8 Mapeo de Datos .nivel (0).



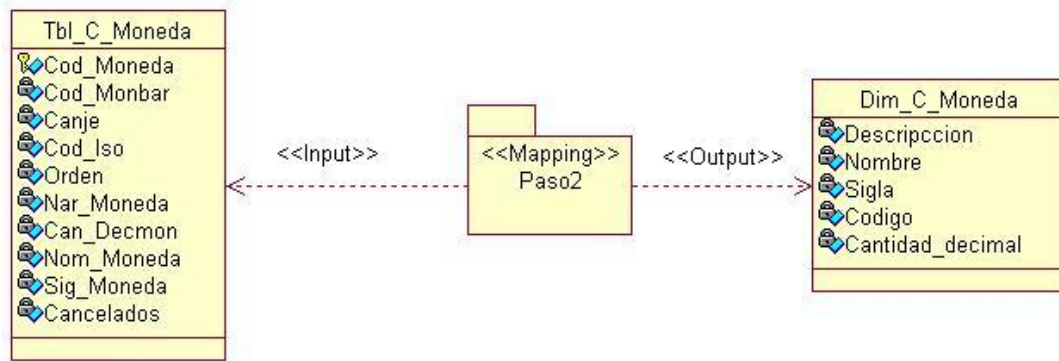
Anexo 3.9 Mapeo de Datos, nivel (1).



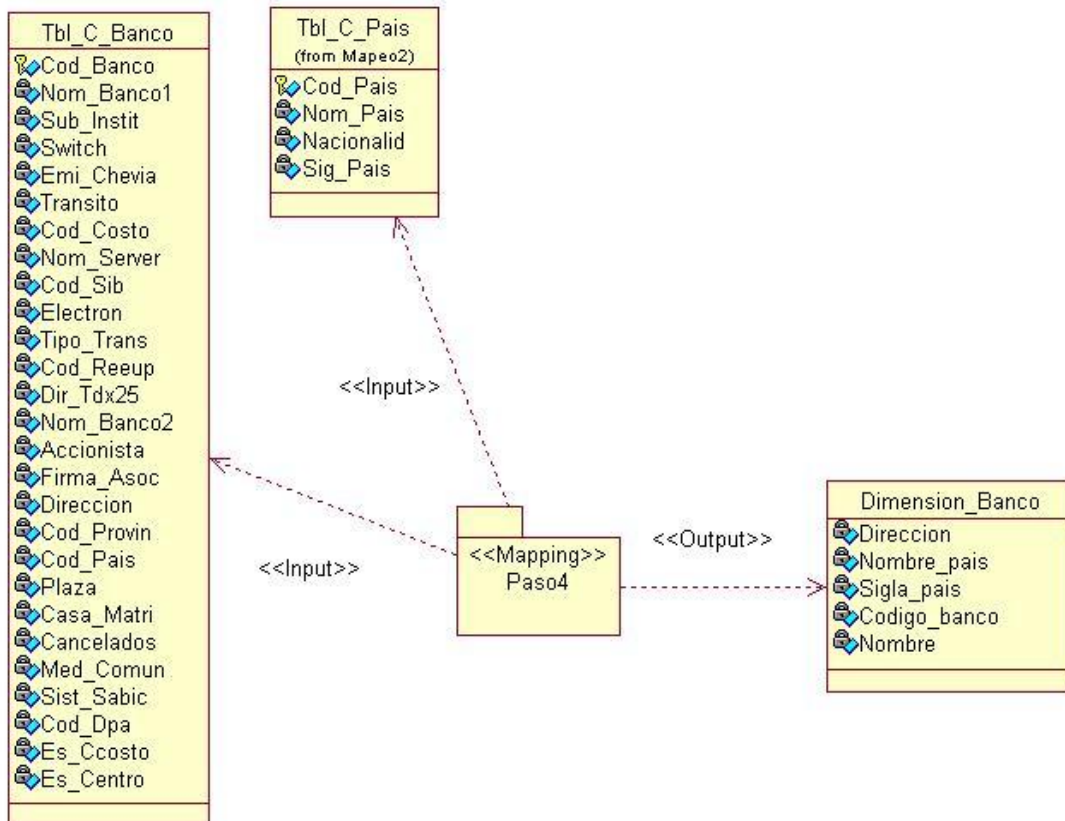
Anexo 3.10 Mapeo de Datos. Dimensión Tiempo, nivel (2)



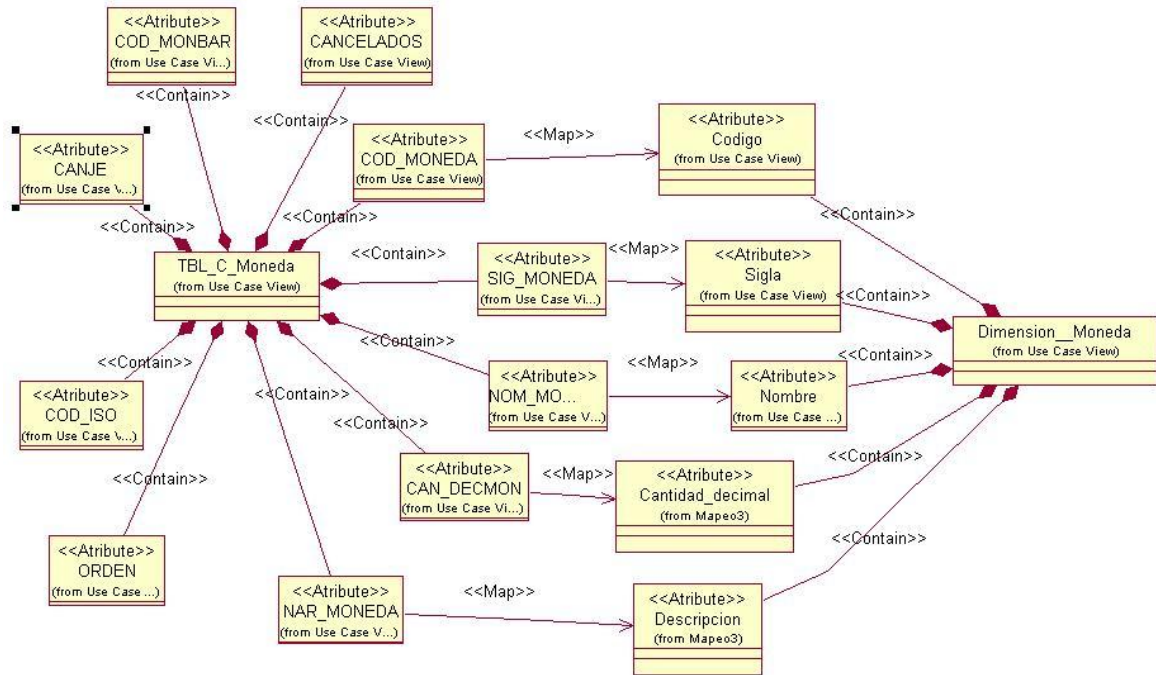
Anexo 3.11 Mapeo de Datos. Dimensión Moneda, nivel (2)



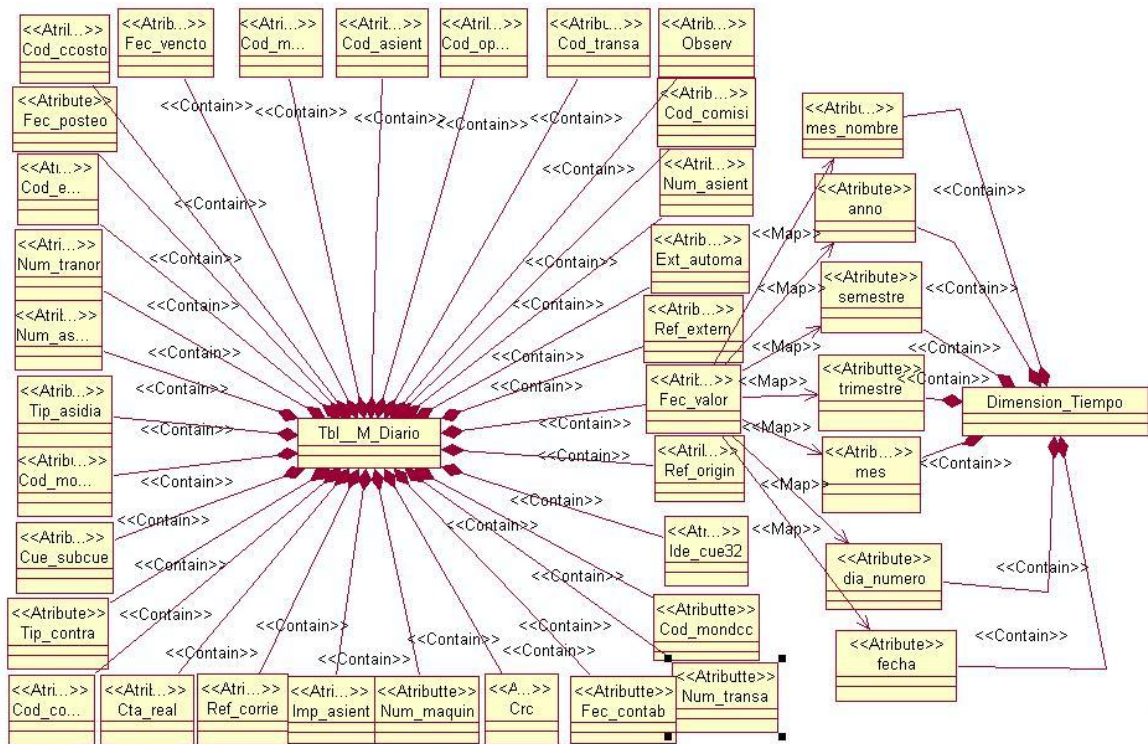
Anexo 3.12 Mapeo de Datos. Dimensión Banco, nivel (2)



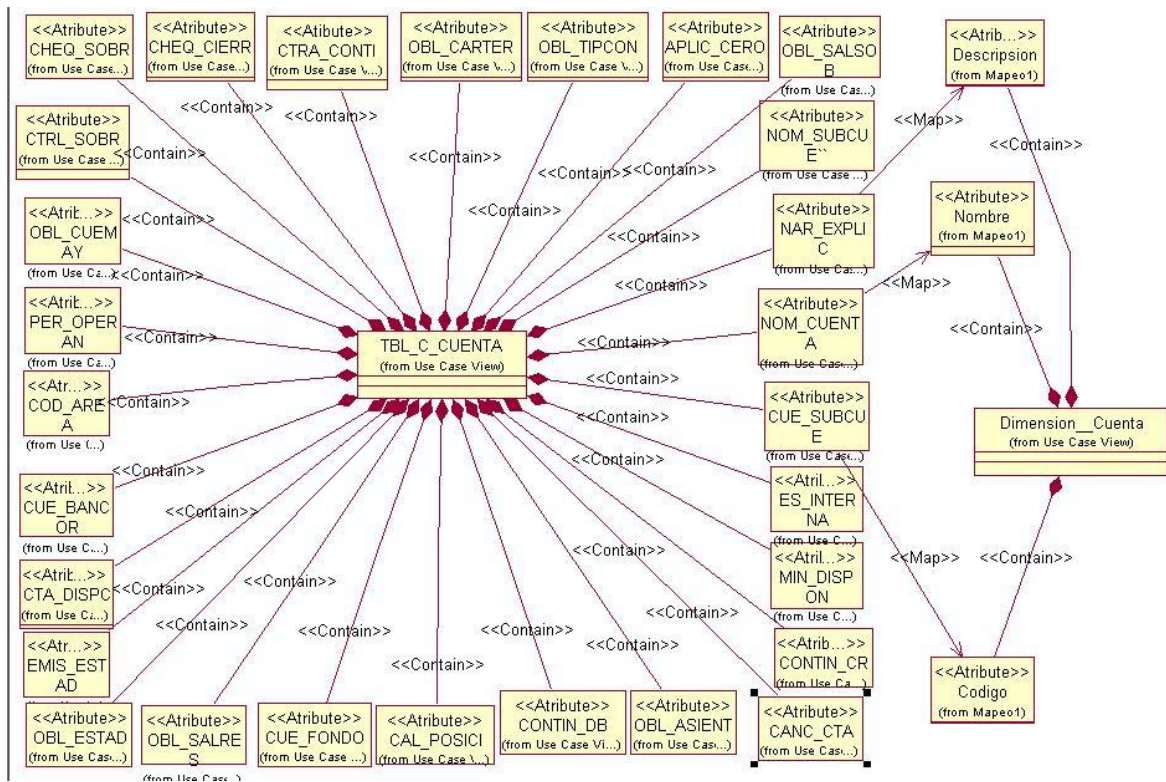
Anexo 3.15 Mapeo de Datos. Dimensión Moneda, nivel (3)



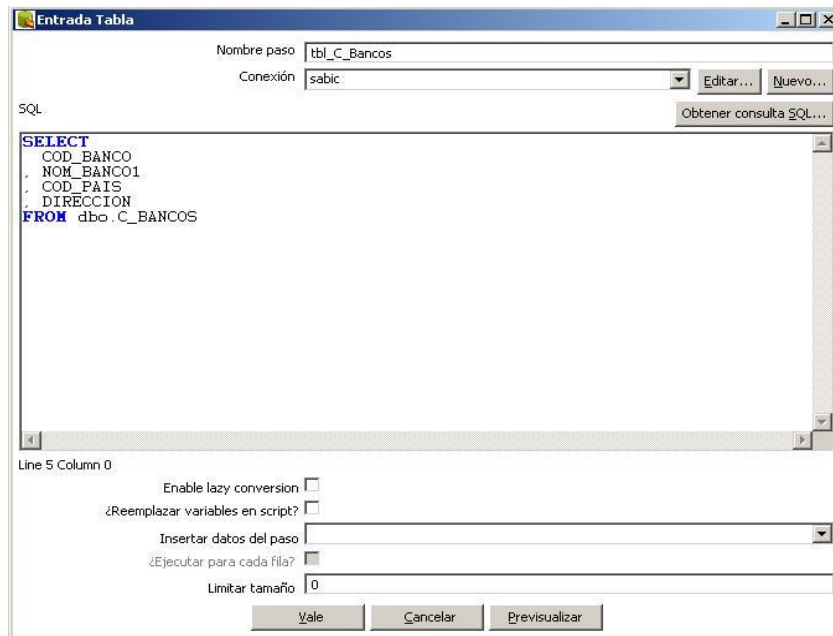
Anexo 3.16 Mapeo de Datos. Dimensión Tiempo, nivel (3)



Anexo 3.17 Mapeo de Datos. Dimensión Cuenta, nivel (3)



Anexo 3.18 Paso Entrada Tabla.



Anexo 3.19 Paso Búsqueda en Base de Datos.

Búsqueda de Valor en Base de Datos

Nombre paso: buscar_cod_pais
 Conexión: sabic
 Esquema de búsqueda: dbo
 Tabla de búsqueda: C_PAISES
 ¿Habilitar cache?:
 Tamaño de cache en filas (0=todas): 0
 Load all data from table:

La clave(s) para realizar búsqueda de valor(es):

#.	Campo de tabla	Comparador	Campo1	Campo2
1	COD_PAIS	=	COD_PAIS	

Valores a devolver de la tabla de búsqueda:

#.	Campo	Nuevo nombre	Defecto	Tipo
1	NOM_PAIS	nombre_pais	null	String
2	SIG_PAIS	siglas_pais	null	String

No procesar la fila si la búsqueda falla:
 ¿Producir error si se obtienen múltiples:
 Ordenar por:

Vale Cancelar Obtener Campos Obtener Campos Búsqueda

Anexo 3.20 Paso Filtrar Filas.

Filtrar filas

Nombre de paso: IF_Banco_Ext
 Enviar 'verdadero' a paso: break
 Enviar 'falso' a paso: Modified Java Script Value

La condición:

COD_PAIS = 247 (String)

Vale Cancelar

Anexo 3.21 Paso Añadir Secuencia.

Obtener valor de la secuencia de la base de datos

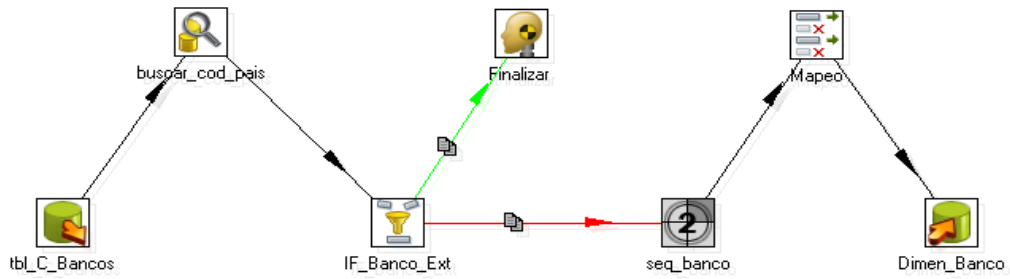
Nombre de paso: seq_banco
 Nombre de valor: valuname

Utilizar una base de datos para generar la secuencia
 ¿Utilizar base datos para obtener secuencia?:
 Conexión: destino
 Nombre de esquema:
 Nombre de secuencia: dimension_banco_id_banco

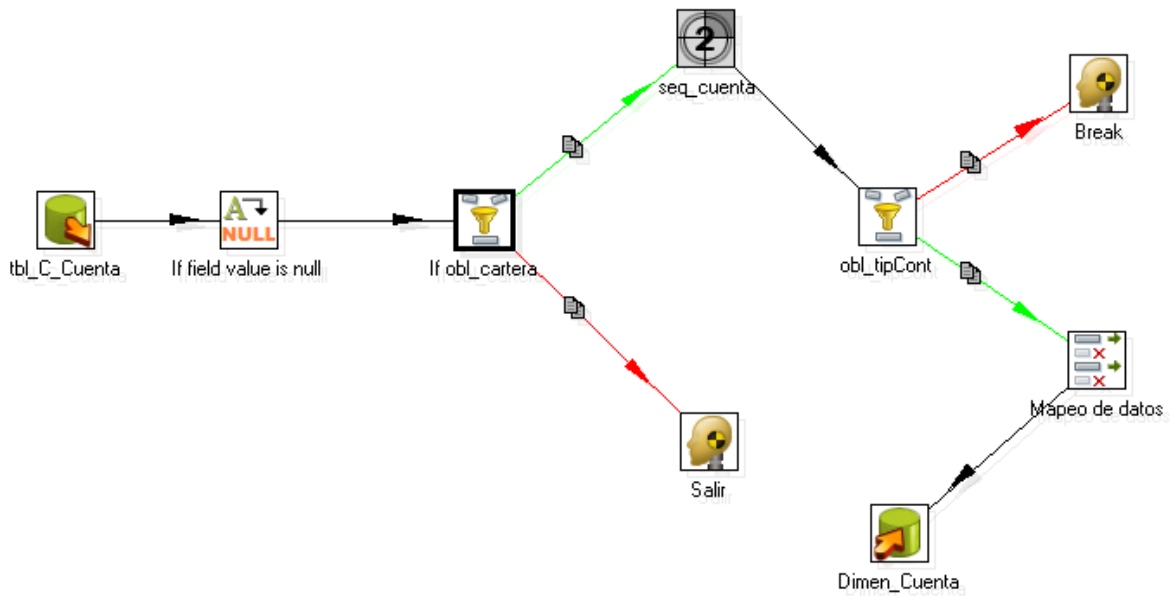
Utilizar un contador de la transformación para generar la secuencia
 ¿Utilizar contador para calcular secuencia?:
 Nombre contador (opcional):
 Valor inicial: 1
 Incremento: 1
 Valor máximo: 999999999

Vale Cancelar

Anexo 3.24 Transformación ETL_Banco



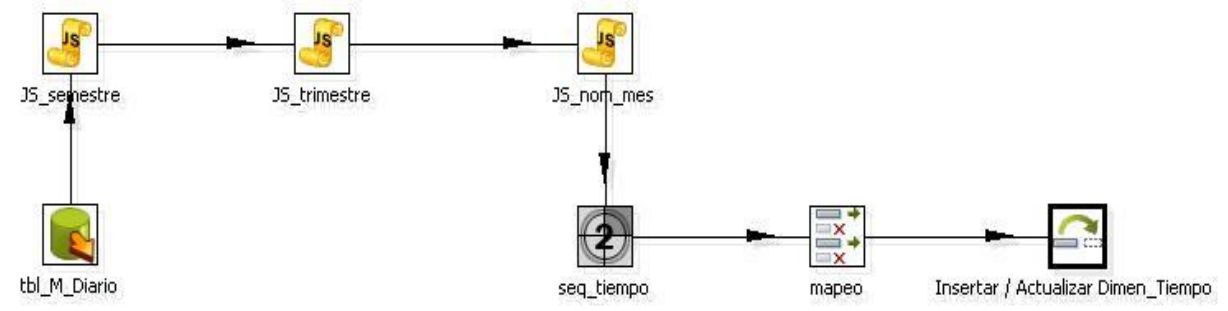
Anexo 3.25 Transformación ETL_Cuenta



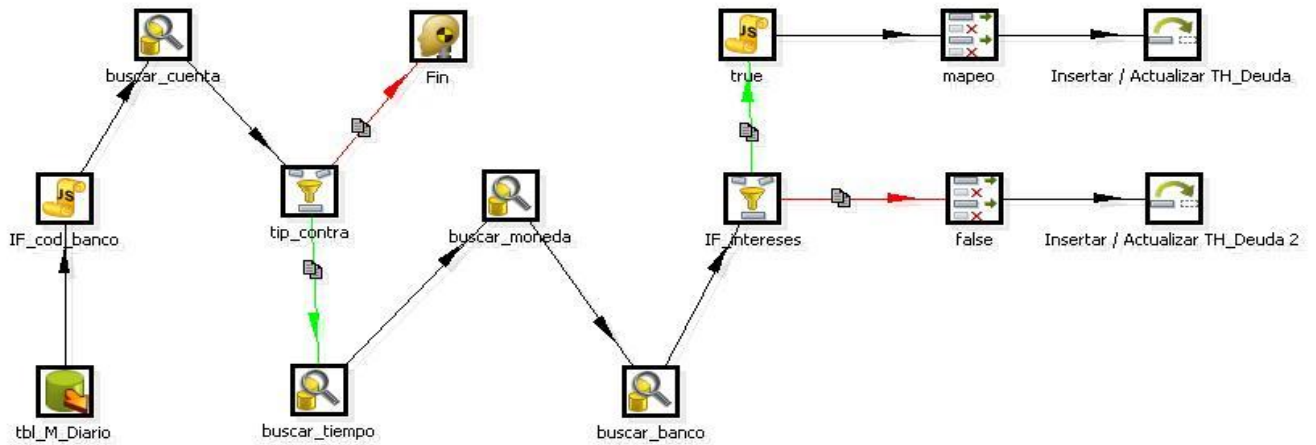
Anexo 3.26 Transformación ETL_Moneda



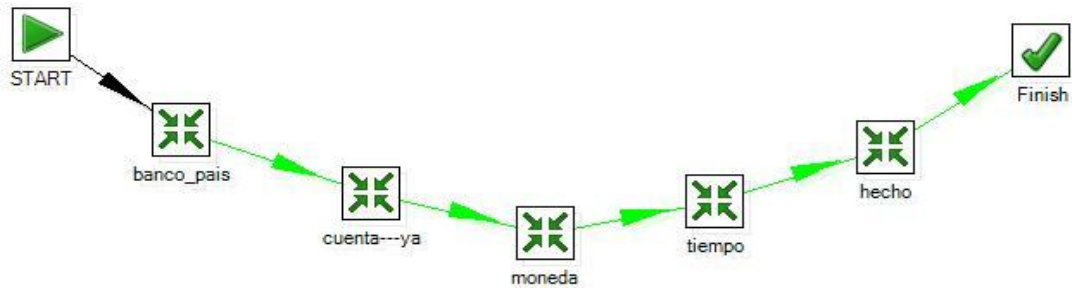
Anexo 3.27 Transformación ETL_Tiempo



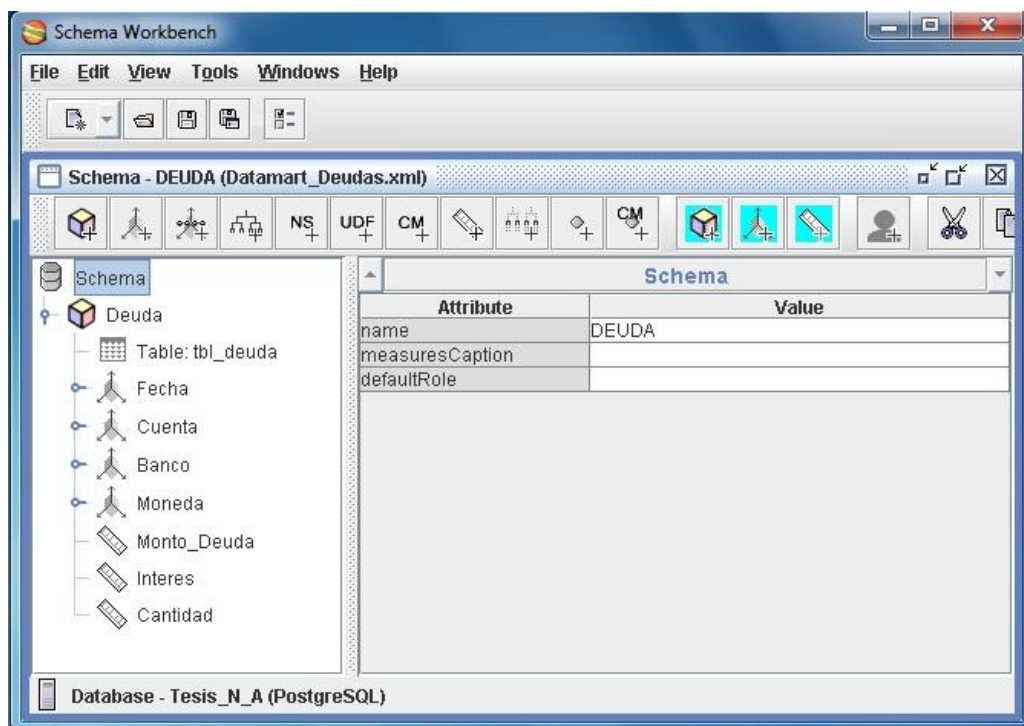
Anexo 3.28 Transformación ETL_Hecho_Deuda



Anexo 3.29 Trabajo de las transformaciones



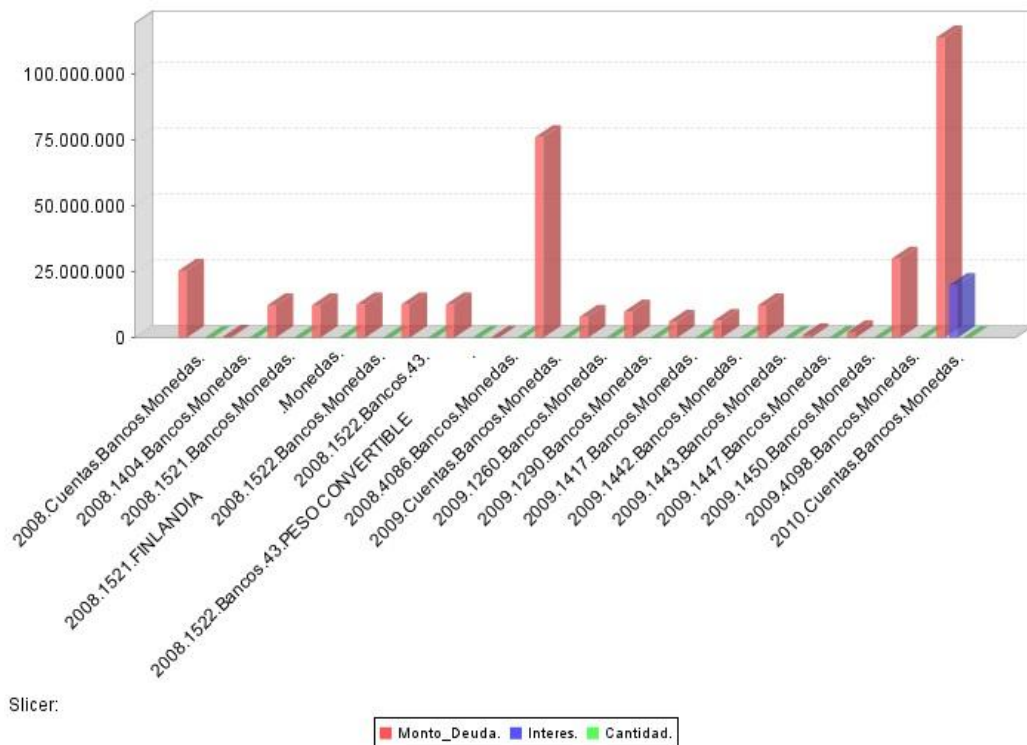
Anexo 3.30 Creación del cubo de datos, dimensiones y jerarquías



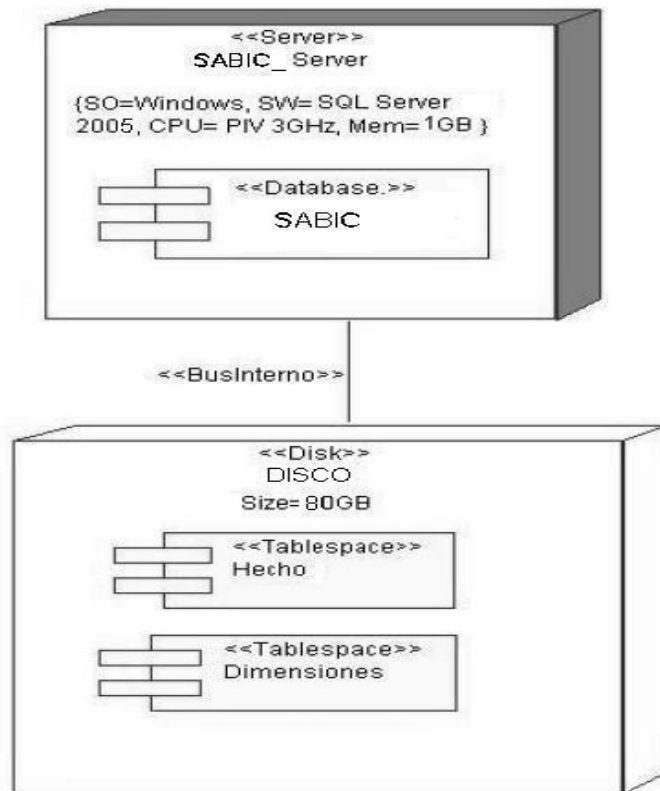
Anexo 3.31 Vista del cubo de datos con las dimensiones y jerarquías

				Medidas		
Tiempo	Cuenta	Banco	Monedas	● Monto_Deuda	● Interes	● Cantidad
+ 2008	- Cuentas	+ Bancos	+ Monedas	25.397.666		4
	+ 1404	+ Bancos	+ Monedas	400.000		1
	+ 1521	- Bancos	+ Monedas	12.356.666		1
		+ FINLANDIA	+ Monedas	12.356.666		1
	+ 1522	+ Bancos	- Monedas	12.640.000		1
			- 43	12.640.000		1
			+ PESO CONVERTIBLE	12.640.000		1
	+ 4086	+ Bancos	+ Monedas	1.000		1
+ 2009	- Cuentas	+ Bancos	+ Monedas	76.050.189		8
	+ 1260	+ Bancos	+ Monedas	8.000.000		1
	+ 1290	+ Bancos	+ Monedas	10.000.000		1
	+ 1417	+ Bancos	+ Monedas	6.123.154		1
	+ 1442	+ Bancos	+ Monedas	6.451.122		1
	+ 1443	+ Bancos	+ Monedas	12.341.400		1
	+ 1447	+ Bancos	+ Monedas	1.000.000		1
	+ 1450	+ Bancos	+ Monedas	2.134.213		1
	+ 4098	+ Bancos	+ Monedas	30.000.300		1
+ 2010	+ Cuentas	+ Bancos	+ Monedas	113.823.393	20.010.000	10

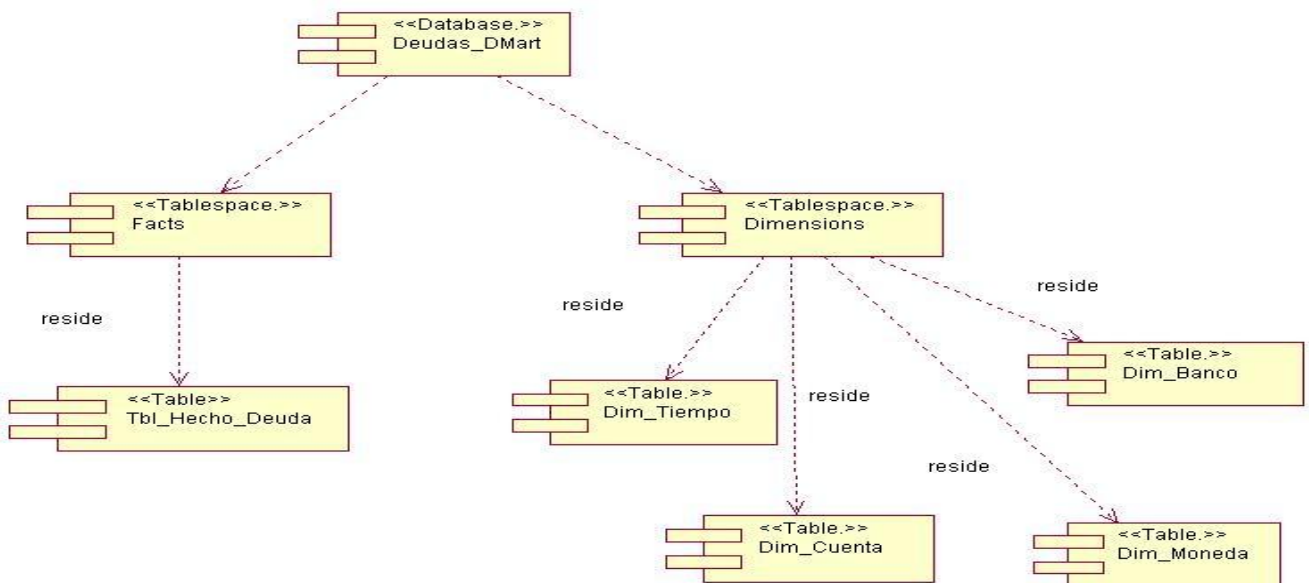
Anexo 3.32 Diseño grafico con el Pívor



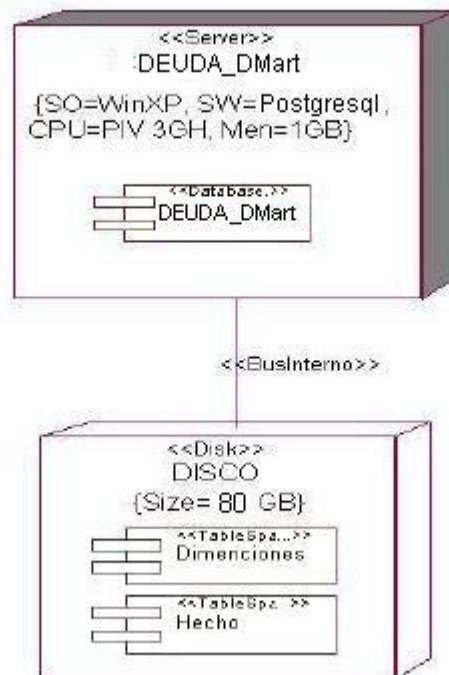
Anexo 3.33 Esquema físico de la fuente de datos



Anexo 3.34 Diagrama de componentes



Anexo 3.35 Esquema físico del Data Mart (Diagrama de despliegue)



Anexo 3.36 Diagrama de Integración de transporte

