

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA



PONTIFICIA
UNIVERSIDAD
CATÓLICA
DEL PERÚ

Análisis, Diseño e Implementación de un DataWarehouse de Soporte de Decisiones para un Hospital del Sistema de Salud Público

Tesis para optar por el Título de Ingeniero Informático, que presenta el
bachiller

Álvaro Villanueva Ojeda

Asesor: Ing. Carla Basurto

LIMA – PERÚ

2008

Resumen

Las entidades de salud del sector público deben de tomar decisiones orientadas a satisfacer la demanda de servicios de los pacientes que acuden a los centros de salud y es por ello muy importante buscar mejorar los sistemas de información ligados a estos procesos de decisión. El presente tema de tesis propone la construcción de un Data Warehouse que servirá de apoyo en el proceso de toma de decisiones del directorio del hospital, el cual, decidirá en base a datos históricos y cuadros generados en línea.

Un sistema de este tipo permitirá reducir carga de pabellones, optimizar el uso del personal, mejorar la atención al paciente, mejorar la calidad de servicio otorgada, brindar un servicio especializado a los pacientes, gestionar recursos, conocer el estado actual de los pacientes, identificar fallas en los procesos, realizar auditorías y realizar notificaciones en tiempo real, entre otras cosas.

Los datos almacenados por la entidad no tienen utilidad si es que no se transforman en información que sirva como base para tomar decisiones. Es por ello que es necesario que todos los datos históricos sean sometidos a un proceso de limpieza para poder garantizar su confiabilidad. Este sistema se encargará de hacer una limpieza de los datos almacenados para poder generar con ellos reportes que ayuden al directorio a la toma de decisiones.

Para la realización del actual tema de tesis, se está optando por utilizar la *suite* de Inteligencia de Negocios proporcionada por Pentaho, la cual es una herramienta libre y completa. Con el uso de esta herramienta se garantiza que la entidad de salud pública no tendrá que destinar costos adicionales por licencias de software. Sin embargo, la dificultad en implementar con esta herramienta viene dada por su misma naturaleza libre (*open source*) y su poco tiempo en producción. Por esta razón, el presente proyecto dará pautas para la utilización e instalación de esta *suite*, lo cual servirá de base para proyectos similares que deseen implementar proyectos con ella.

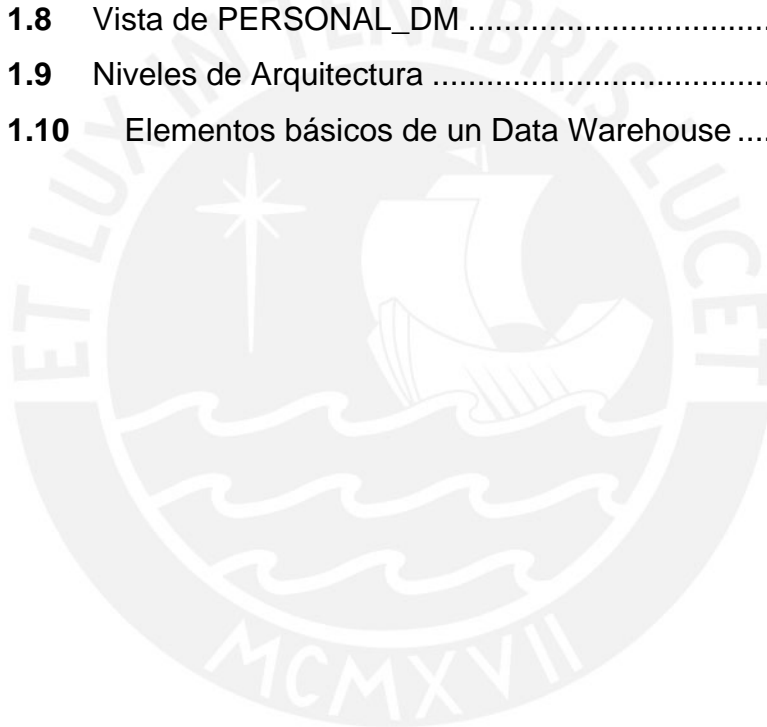
Para implementar este proyecto de tesis se realizarán todos los pasos de un proyecto de Inteligencia de Negocios: diseño y construcción del Data Warehouse y los Data Marts, creación y programación de los procesos ETL, creación de los cubos, creación de los informes, y finalmente implementación de la plataforma BI (Web).

INDICE GENERAL	
Resumen	2
INDICE DE FIGURAS	6
INDICE DE CUADROS	7
1. Marco Conceptual	8
1.1. Definición del problema	8
1.2. Conceptos Relacionados	12
Business Intelligence	12
Data Warehouse	13
Data Mart	13
DSS	14
OLAP	14
MOLAP, ROLAP, HOLAP	15
Otras definiciones	16
1.3. Estado del Arte	16
2. Análisis	24
2.1. Metodología	24
I. Justificación	27
II. Planeamiento	27
III. Análisis del negocio	27
IV. Diseño	28
V. Construcción	29
VI. Instalación	29
2.2. Requerimientos Funcionales	30
2.3. Requerimientos No Funcionales	32
2.4. Plan de Pruebas	33
Objetivos	33
Características a ser probadas	33
Recursos	34
Responsables y Tareas	34
Riesgos	35
Aprobación	35
Casos de Prueba	35
2.5. Software a Utilizar	36

3.	Diseño	39
3.1.	Modelamiento de tablas	39
3.2.	Análisis Dimensional	42
3.3.	Arquitectura utilizada	49
3.4.	Estándares de Reportes:	51
	Configuración del Reporte:	51
	Esquema de Publicación.....	59
3.5.	Diseño de Extracción	59
4.	Construcción y Pruebas.....	61
4.1.	Puesta en producción	61
	Configuración del software.....	61
	Construcción de la Metadata	61
	Configuración de la herramienta de ETL	62
	Instalación de la plataforma del Pentaho	62
	Instalación del Pentaho Design Studio	63
	Pentaho Report Designer.....	64
	Pentaho Cube Designer.....	64
	Pentaho Workbench	64
4.2.	Desarrollo de las Pruebas.....	64
	Pruebas de Proceso ETL.....	64
5.	Observaciones, Conclusiones y Recomendaciones	71
5.1.	Observaciones	71
5.2.	Conclusiones	72
5.3.	Recomendaciones	74
6.	Bibliografía.....	75

INDICE DE FIGURAS

Figura 1.1	Ciclo de vida del de desarrollo de un sistema de toma de decisiones.....	26
Figura 1.2	Data Mart de Admisión	42
Figura 1.3	Data Mart de Caja	43
Figura 1.4	Data Mart de Cita	44
Figura 1.5	Data Mart de Cuentas Corrientes	45
Figura 1.6	Vista de GENERAL_DM.....	46
Figura 1.7	Data Mart de Historia Clínica.....	47
Figura 1.8	Vista de PERSONAL_DM	48
Figura 1.9	Niveles de Arquitectura	49
Figura 1.10	Elementos básicos de un Data Warehouse	50



INDICE DE CUADROS

Cuadro 1.1	Comparativo de herramientas de Inteligencia de Negocios	23
Cuadro 1.2	Requerimientos Funcionales	32
Cuadro 1.3	Requerimientos No Funcionales	33
Cuadro 1.4	Casos de prueba para los reportes del Data Mart de Caja_DM	36
Cuadro 1.5	Comparación entre modelo Estrella y Copo de Nieve	40
Cuadro 1.6	Reportes a elaborar	55
Cuadro 1.7	Esquema de publicación de los reportes	59



1. Marco Conceptual

En este capítulo se brinda una descripción del problema por el cual se ven afectados los hospitales públicos del país en la actualidad, y se da una visión de lo que se desea obtener con el proyecto de fin de carrera. Asimismo, se brinda una descripción de las investigaciones que se han hecho respecto al tema.

1.1. Definición del problema

Los hospitales en el Perú reciben una gran demanda de servicios, muchas veces no satisfecha en su totalidad, por parte de los sectores menos favorecidos de la población peruana. Los sectores pobres del país acuden a los centros hospitalarios de atención pública como única alternativa, ya que no cuentan con recursos suficientes para acceder a un seguro privado. Por otra parte, los seguros públicos, como ESSALUD, tienen una tasa baja de cobertura, lo que limita las posibilidades de acceso a los servicios de salud del seguro social. La falta de hospitales en el país y la centralización de los mismos en la capital peruana, sumado al crecimiento

urbano, hacen necesario la optimización de los procesos administrativos de los hospitales. Esta optimización de los procesos administrativos debe considerarse como parte del proceso de perfeccionamiento de los servicios que ofrecen los centros de atención médica, orientado a satisfacer la demanda y mejorar la calidad de los servicios hospitalarios. El proceso de perfeccionamiento es iniciado por cada hospital, ya que son conscientes de que para lograr la equidad y eficiencia en la atención al público deben mejorar los procesos administrativos.

Decisiones médicas oportunas y acertadas requieren de un soporte administrativo de calidad. Los sistemas de información creados pensando en las necesidades y características particulares del usuario contribuyen a que este proceso de toma de decisiones, muchas veces ligado a salvar vidas humanas, utilice la mejor y mayor información disponible. Es, por esto, importante resaltar la necesidad de contar con sistemas de información creados especialmente para los sistemas administrativos de salud pública, considerando todos los detalles que servirán de apoyo durante el proceso de toma de decisiones a distintos niveles de la administración hospitalaria. En contraste, parece imposible crear un único sistema de información que resuelva la problemática entera de un hospital, es decir, que integre cada sub-proceso, por lo que debe considerarse que existen diversas fuentes de información distintas, de cada uno de los diferentes sistemas, con diferentes estándares o formas de representar los datos.

Actualmente las actividades de internamiento hospitalario absorben una gran proporción de recursos, y el primer nivel de atención en salud que recibe el paciente, es débil y requiere fortalecerse. Es importante una adecuada gestión de recursos en la entidad de salud. Hay varias direcciones hacia las que se pueden dirigir los recursos, como tratamientos, equipos, personal, infraestructuras, investigación y capacitación. Es por esto importante encontrar un balance y una proporción adecuada entre los recursos dirigidos a cada uno de estos puntos. Esto beneficiaría tanto a pacientes como al personal de la entidad de salud, y crearía una mejor equidad, lo que terminaría en una mejora en la calidad general del servicio brindado por el hospital.

Según el Estudio financiero-actuarial y de la gestión de ESSALUD [DUR 2005], se identificó la elevada estancia hospitalaria promedio, como uno de los grandes problemas de gestión de servicios hospitalarios. El problema genera elevados costos, y se explica por una serie de factores: problemas de programación interna

dentro de los hospitales, infección hospitalaria, lentitud en los exámenes de laboratorio, falta de conocimiento hacia la gestión de los costos en el nivel clínico, cancelación intempestiva de cirugías y complicación en los pacientes, entre otros. La introducción de conceptos de gestión clínica en los hospitales, podrían impactar no sólo en la duración de la estancia hospitalaria, sino también en el costo directo de las intervenciones hospitalarias. Asimismo, la seguridad social podría verse beneficiada considerablemente por la adopción de nuevos enfoques de calidad y eficiencia en la gestión hospitalaria.

La seguridad social es un instrumento para el progreso y el desarrollo de los pueblos, y el cumplimiento de los principios de equidad y solidaridad de este servicio está condicionado por la existencia de recursos suficientes para aplicar políticas redistributivas a favor de los grupos menos privilegiados. Sin embargo, según las estadísticas de salud del año 2001 [BEN 2005], sólo el 35.9% de las personas contaban con seguro médico en Lima Metropolitana y el 6.2% en las zonas rurales en 1997. De las personas aseguradas, 86.5% estaban afiliadas al sistema de seguro social de salud del Perú (ESSALUD), mientras que el resto, 13.5%, estaban afiliados a seguros privados. Esta última cifra disminuiría a 3% en el 2001 [IPE 2004]. Se puede ver que la gran mayoría de la población no tiene acceso a servicios de salud, debido a que los precios no son accesibles para ellos. Dada la situación del país, es necesario reducir aún más los precios o prestar servicios gratuitamente para los sectores menos privilegiados. Sin embargo, para lograr esto, las entidades de salud necesitan reducir costos y mejorar sus procesos internos, objetivos que se logran por medio de una buena toma de decisiones.

En el año 2004 existían en el Perú 441 hospitales pertenecientes al sector salud MINSA (Ministerio de Salud) y ESSALUD. La distribución de estos establecimientos no es homogénea en el país, ya que se observa un mayor número de estos en Lima, Ancash, Puno, San Martín, Junín y La Libertad [INEI 2006]. Es por todo lo antes mencionado, que la gestión administrativa en los hospitales públicos necesita tomar decisiones oportunas y acertadas, ya que la toma de decisiones en los hospitales de la capital es vital para una mejora significativa en la atención a las personas que acuden a estos centros de salud, que muchas veces son personas provenientes de provincia o de lugares alejados al distrito donde se encuentra el hospital, que esperan una mejor atención a comparación de su centro de salud asignado, es decir, de su distrito o provincia de residencia. Las decisiones tomadas por el directorio de cada hospital se ven reflejadas en: la atención que recibe el

paciente; el tiempo que demora en ser atendido un paciente; adquisición de tecnología de última generación; políticas de crecimiento y expansión de servicios.

Actualmente uno de los principales problemas de los hospitales públicos es que el directorio de los hospitales está conformado, en su mayoría, por médicos no especializados en labores administrativas. Es por esta razón que se debe de aprovechar las herramientas informáticas para que el directorio pueda tomar decisiones correctas y pertinentes basándose en información acertada y veraz de la situación dentro de cada institución. Actualmente, generar un reporte demora días e incluso semanas, ya que se tiene que cumplir obligatoriamente una serie de pasos administrativos para elaborarlo. En cambio con herramientas tecnológicas, estos reportes son generados, en forma personalizada, en minutos o incluso segundos.

Los Sistemas de Soporte de Decisiones (DSS) proveen información adecuada para la toma de decisiones, ya que adquieren, manipulan, transforman, validan y presentan datos almacenados en una forma resumida y bastante visual. Muchas veces los hospitales tienen los datos almacenados, pero no son capaces de aprovecharlos, ya que no generan ninguna información con ellos; por esto, es necesario crear un repositorio central de datos, que integre todos estos datos de las distintas áreas del hospital, les aplique un control de calidad y los almacene con un estándar. Con los DSS se puede utilizar los datos para transformarlos en información, y así conseguir que el directorio pueda planificar sus metas y cumplir mejor con su gestión administrativa, por medio de reportes, como por ejemplo, Balance Scorecard o reportes presupuestales. Cabe resaltar que los Sistemas de Soporte de Decisiones usan tecnologías informáticas, documentos de datos, conocimiento y modelos analíticos para identificar y solucionar problemas, ya que se brinda un sistema de fácil manejo en el que se puede hacer comparaciones cuantitativas con años, meses, semanas y días anteriores, para que el usuario pueda ser capaz de optimizar el proceso de toma de decisiones.

En conclusión, el directorio del hospital será capaz de: planificar sus metas, tomar decisiones para prevenir eventos adversos, responder a situaciones imprevistas, cambios en la demanda de servicios; mejorar la calidad de atención a los pacientes teniendo en cuenta comparaciones con cifras anteriores, siendo posible medir los cambios en los indicadores de calidad y eficiencia de gestión del hospital y analizando el impacto de sus decisiones de forma directa en los pacientes y en el personal del hospital. Mientras que los pacientes se verán beneficiados, ya que se

observarán cambios en la atención, y las capacidades del hospital se verán mejoradas y aprovechadas. También se mejorará la productividad de los médicos y de todo el personal del hospital, ya que se evitarán largos periodos de espera hasta la toma de decisiones que beneficien el trabajo médico, tal como la adquisición de aparatos médicos de alta inversión que exigen estudios minuciosos y exactos. El directorio podrá analizar la manera en que sus decisiones benefician a sus trabajadores. Finalmente, el hospital no sólo acumulará datos constantemente, sino que también acumulará y analizará consistentemente la información.

1.2. Conceptos Relacionados

Para comprender el contexto en que se desarrollará el presente trabajo de tesis, es importante entender lo que es la Inteligencia de Negocios (*Business Intelligence-BI*) y todo lo que ésta implica, puesto que dichos conceptos van de la mano con el trabajo a desarrollarse.

Business Intelligence

Según Vitt [VITT 2002], el término de BI (*Business Intelligence*) es usado por diferentes expertos y fabricantes de software para distinguir un amplio rango de tecnologías, plataformas de software, aplicaciones específicas y procesos. Se utiliza este término desde tres diferentes perspectivas:

- Tomar mejores decisiones rápidamente
- Convertir los datos en información
- Utilizar un método razonable para la gestión empresarial

El objetivo primario de la Inteligencia de Negocios es ayudar a las personas a tomar decisiones que mejoren el rendimiento de la compañía e impulsen su ventaja competitiva en el mercado. Es decir, faculta a las organizaciones a tomar las mejores decisiones rápidamente. Para tomar mejores decisiones más rápidamente, los directivos y gerentes necesitan de información relevante y útil al alcance de la mano. Pero es común una larga brecha entre la información que los responsables en la toma de decisiones requieren, y las grandes cantidades de datos que las organizaciones recopilan cada día. Para saltar esta brecha, las organizaciones hacen significativas inversiones en desarrollar sistemas de BI para convertir los

datos originales en información de utilidad. Los sistemas de BI más efectivos tienen acceso a inmensas cantidades de datos para posteriormente entregar a los responsables en la toma de decisiones, información expresada de una forma que ellos pueden asimilar fácilmente. La inteligencia de Negocios puede ser definida como un método para la gestión empresarial, una forma de pensamiento organizacional y una filosofía de gestión. Tanto las personas como las organizaciones se interesan en la Inteligencia de Negocios, porque creen que el uso de un enfoque racional y basado en hechos a la hora de tomar decisiones resulta positivo en la medida que sea posible. El interés por adoptar el BI tiene las siguientes características:

- Buscar hechos (datos) que se puedan medir cuantitativamente acerca del negocio.
- Usar métodos organizados y tecnologías para analizar los hechos.
- Inventar o compartir modelos que expliquen las relaciones de causa y efecto entre las decisiones operativas y los efectos que éstas tienen en alcanzar los objetivos de negocio.
- Experimentar con métodos alternos y supervisar con retroalimentación sobre los resultados.
- Gestión de la empresa (decisiones e iniciativas) basadas en todas estas características.

Data Warehouse

Un Data Warehouse, según [IBM 1999] es un repositorio no volátil de datos, transacciones, y eventos. Incluye data corporativa, operacional y externa. Según [VITT 2002], los datos en el Data Warehouse deben estar integrados, consolidados, seguros, y limpios para que sea una fuente segura de soporte de decisiones y aplicaciones de información.

Data Mart

Otro concepto de mucha importancia en lo que es BI es el de Data Mart. Un Data Mart, según [IBM 1999], es un subconjunto del Data Warehouse, con un alcance de contenido limitado. Éste se usa para un solo departamento de una organización y/o un problema particular de análisis dentro de la organización. Un Data Mart por si

solo, no es un Data Warehouse, ya que un Data Warehouse tiene más usuarios y más temas que un Data Mart, y provee una vista completa de las áreas funcionales de la organización. Un Data Mart, al igual que un Data Warehouse, consiste en una base de datos. Así mismo, Vitt [VITT 2002] define el Data Warehouse como un repositorio colectivo y centralizado que nutre o alimenta una serie de almacenes que tienen una orientación específica o dominio específico, o tema específico, llamados Data Marts.

DSS

Según [IBM 1999], un Sistema de Soporte de Decisiones (*DSS- Decision Support System*) contiene todos los servicios y procesos, para seleccionar, manipular, y analizar información y presentar resultados. Debe de permitir acceso transparente a la data en varias partes del Data Warehouse y proveer una interfaz común para los diferentes grupos de usuarios. Un DSS también puede ser definido como un sistema computacional diseñado para apoyar en los procesos de la toma de decisiones en una organización. Un DSS es la ventana del usuario a los datos almacenados en el ambiente del Data Warehouse.

OLAP

En lo que es consultas para la presentación de los datos, es importante explicar el concepto de OLAP (*Online Analytical Processing*). Según [VITT 2002], OLAP proporciona un modelo de datos intuitivo y conceptual, para que los usuarios que no tengan experiencia como analistas puedan comprender y relacionar los datos mostrados. Este modelo es llamado análisis multidimensional, siendo habilitado para ver los datos a través de múltiples filtros, o dimensiones. Los sistemas OLAP organizan los datos directamente como estructuras multidimensionales, incluyendo herramientas fáciles de usar por usuarios para conseguir la información en múltiples y simultáneas vistas dimensionales. OLAP es también rápido para el usuario. Rápidos tiempos de respuesta permiten que los gerentes y analistas puedan preguntar y resolver más situaciones en un corto período de tiempo. Una dimensión es una vista de los datos categóricamente consistente. Una característica de las dimensiones es la habilidad de hacer *slice-and-dice*. *Slice* (rebanada) y *dice* (cubo) hacen particiones de los datos en una base de datos multidimensional de acuerdo a los valores de ciertas dimensiones. Otra capacidad inherente en el diseño de OLAP

es la rotación y anidamiento (*Pivoting-and-Nesting*) de las dimensiones. El *pivoted* permite rotar los datos desde las columnas hasta las filas. También es importante mencionar el concepto de *drill*, que en los sistemas OLAP tiene un significado muy específico. *Drill down* es la acción de seleccionar un miembro para ver el siguiente nivel inferior de detalle en la jerarquía. *Drill up* es seleccionar un miembro para ver el siguiente nivel superior, esto es, una acción de *bottom-up*. La mecánica o funcionamiento de las interfaces OLAP, especialmente *pointing-and-clicking* (apuntar y seleccionar) para hacer *drill-down* dentro de las capas de interés se hace posible por la velocidad con que las consultas son resueltas. Esta funcionalidad permite por completo a los gerentes y analistas un nuevo proceso para tratar con grandes cantidades de datos, un proceso conocido con el nombre de análisis ad hoc. En resumen, los sistemas OLAP organizan los datos por intersecciones multidimensionales. Esta organización, acompañada por una herramienta de interface para rotar y anidar dimensiones, permite a los usuarios visualizar rápidamente valores en detalle, patrones, variaciones y anomalías en los datos que estarían de otra manera ocultos por un análisis dimensional simple. A mayor número de dimensiones (dentro de los límites razonables), mayor es la profundidad del análisis.

MOLAP, ROLAP, HOLAP

Existen variaciones de OLAP según la cantidad de datos y la eficiencia requerida. OLAP, no se recomienda para consultas complejas y que recorran muchas tablas. Una de estas variaciones es MOLAP (*Multidimensional online analytical processing*), según [VITT 2002], los datos son colocados en estructuras especiales que se encuentran en un servidor central. MOLAP ofrece el mayor rendimiento de recuperación de información. Por otra parte, existe la solución ROLAP (*Relational online analytical processing*), según [VITT 2002], permite tomar ventaja de uno de sus más grandes beneficios, el almacenamiento de inmensas cantidades de datos. El rendimiento de recuperación de la información para ROLAP frecuentemente no es tan rápido como otras opciones de almacenamiento. ROLAP es recomendado para consultas pesadas que no se usan muy a menudo. Finalmente existe HOLAP (*Hybrid online analytical processing*), que es un híbrido entre MOLAP y ROLAP, y según [VITT 2002], HOLAP no es realmente un modo diferente de almacenamiento de datos. Más bien es la habilidad para diseminar los datos a través de bases de datos relacionales y multidimensionales con la finalidad de obtener lo mejor de

ambos sistemas.

Otras definiciones

- **Dimensión:** es un grupo de miembros consistentes categóricamente representados como una arista específica de un cubo OLAP, por ejemplo, el tiempo, clientes, productos.
- **Jerarquía:** es la organización de niveles dentro de una dimensión que refleje: cómo los datos añadidos están agregados nivel a nivel, y el camino que permita hacer drill-down de arriba abajo dentro de la dimensión. Por ejemplo: año, trimestre y mes.
- **Miembro:** es el nombre o etiqueta para cualquier miembro de cualquier nivel en una jerarquía. Los miembros inferiores son llamados algunas veces miembros hoja o miembros de nivel cero.
- **Generación jerárquica:** este término se utiliza para describir las relaciones entre miembros de una jerarquía. Lo más común es usar nombres de familia, como los siguientes:
 - **Hijo:** es un miembro directamente subordinado o por debajo de otro miembro en una jerarquía.
 - **Padre:** es un miembro que está directamente encima de otro miembro en una jerarquía.
 - **Hermano (*Sibling*):** es un miembro que está al mismo nivel de uno o más miembros compartiendo el mismo padre.
 - **Descendiente:** cualquier miembro en cualquier nivel en relación a otro miembro específico.
 - **Ancastro:** cualquier miembro de cualquier nivel superior en relación a otro miembro.

1.3. Estado del Arte

Instituciones y empresas de manera rutinaria, acumulan información que los sistemas de manejo de datos generan, relacionados a los procesos de decisión propios de cada organización. Esta información es la base de decisiones futuras, puesto que decisiones estratégicas requieren de datos históricos que presenten los

resultados de decisiones similares; de ahí la importancia de contar con sistemas automatizados de manejo de la información que permita un acceso rápido a estas fuentes históricas, sin interferencias con los sistemas administrativos presentes en la organización [GUT 2001].

Un Sistema de Soporte de Decisiones es un sistema computarizado diseñado para apoyar la toma de decisiones de la gerencia. El sistema de soporte de decisiones tiene 5 componentes principales: la base de datos; el modelo de la base de datos; el hardware y el software de la computadora; el administrador (usuario) y la red de comunicación. En conjunto, provee un modo interactivo que permite el diálogo en línea entre el usuario del sistema, el directorio del hospital para el presente proyecto, y la computadora. El sistema de soporte de decisiones permite mostrar los resultados de manera gráfica, a través de herramientas computacionales que se detallarán adelante.

Por muchos años, los hospitales han estado monitoreando sus operaciones mediante el análisis de sus reportes financieros y operacionales provenientes de distintas áreas, pero debido al carácter de rápido cambio de la industria médica, los datos estadísticos en el papel no son suficientes para la toma de decisiones. El objetivo de esta tesis es construir un sistema computarizado de manejo de información orientado a asistir a los directores de los hospitales en el análisis de sus operaciones en forma efectiva y precisa.

A continuación se detallarán trabajos y herramientas existentes en el área de los DSS para hospitales.

I) Trabajos relacionados

1) Construcción de un sistema de apoyo a la toma de decisiones para el área gerencial del Hospital de Clínicas

Gutiérrez [GUT 2001] ha realizado en Uruguay, un proyecto de fin de carrera titulado “Construcción de un sistema de apoyo a la toma de decisiones para el área gerencial del Hospital de Clínicas”. Gutiérrez analiza distintos sistemas de manejo de información, resaltando las ventajas de los más avanzados frente a los primitivos, que ofrecen mínima interacción entre el usuario y la máquina, son de respuesta lenta y no permiten respuestas a cuestiones complejas de interés para el usuario. Sistemas más modernos ofrecen mejoras, en cuanto a herramientas de

análisis, mayor capacidad de manejo de información, interacciones complejas, y uso de sistemas operacionales presentes en la empresa.

La metodología usada por Gutiérrez (2001) en este proyecto fue:

Etapas 1. Adquirir conocimientos de base: datos históricos, entrevistas.

Etapas 2. Análisis de información y construcción del sistema legado (en base a sistemas operacionales ya existentes).

Etapas 3. Diseño conceptual multidimensional, en esta etapa se elige la herramienta de diseño a usar para representar el diseño conceptual del modelo de datos del Data Warehouse, y el enfoque elegido para el diseño lógico del modelo de datos del Data Warehouse. Se crea también una tabla de correspondencias entre el diseño conceptual del sistema legado y el diseño conceptual del Data Warehouse.

Etapas 4. Diseño lógico multidimensional, en esta etapa se selecciona el enfoque a seguir en el diseño. Se diseña la arquitectura en estrella para cada relación dimensional del Data Warehouse. Se crea una tabla de correspondencias entre el diseño lógico del sistema legado y el diseño lógico de la data. Se diseñan los procesos de carga y refresco.

Etapas 5. Construcción del prototipo de carga y refresco del Data Warehouse.

Gutiérrez resalta que el proceso de implementación del sistema automatizado permitió a los usuarios conocer de manera detallada las actividades y tiempos necesarios para la completa puesta en marcha de tal sistema, identificándose los cuellos de botella, que necesitan ser atendidos previamente para una exitosa implementación.

Más allá de la metodología en sí, este trabajo permite mostrar las actividades y tiempos que involucra la construcción de un sistema de información con características de apoyo a la toma de decisiones. El sistema construido se basa en el uso de tecnología *data warehousing* y OLAP. Para el Hospital de Clínicas este trabajo tuvo los siguientes aportes:

- Documentación completa del sistema legado de partida.
- Documentación de diseño del sistema de Data Warehouse e implementación de un prototipo del cubo de Compras.
- Conocimiento a nivel gerencial de la situación actual de la institución respecto de los sistemas informáticos que necesitan finalización para poder ser integrados a un Data Warehouse corporativo.
- Conocimientos a nivel gerencial de las posibilidades de apoyo que un Data Warehouse puede brindar a su gestión.

Si bien se cuenta con mejor tecnología para resolver un sistema de información que ayude a la toma de decisiones, es importante resaltar el tiempo que consume la actividad de preparación para el uso de la misma, en particular el relevamiento de sistemas legados con documentación escasa o inexistente. También se pudo experimentar un alto costo en tiempo debido a la brecha que puede existir entre los requerimientos a nivel gerencial y los sistemas realmente existentes en una institución de gran porte.

2) *Improving Outcomes With Community-wide Distribution of Health Care Data*

Un estudio realizado en EE.UU. [RON 2004], investigó las mejoras que resultan de compartir información hospitalaria entre comunidades que sirven una misma área. Este estudio describió una serie de esfuerzos de todos los hospitales en un área de EEUU para mejorar los resultados y la eficiencia en el área de salud a través del compartimiento general de los datos del área de salud. Se estudió el impacto del intercambio diario y semanal de la información entre diferentes administradores de diferentes centros médicos. La distribución general de los resultados y la utilización de los datos en la comunidad estaban dirigidas a generar resultados positivos a través del intercambio de información entre los proveedores de salud de la comunidad. Gracias al intercambio de la información, los administradores hospitalarios pudieron aprender de las experiencias de los otros, y así mejorar sus resultados y la eficiencia en los servicios de cada centro de salud. Este sistema se centró en el compartir datos relacionados de los hospitales, servicios de emergencia, cuidados a largo plazo y salud mental. Se basó en la distribución diaria de reportes entre todos los centros de salud que brindaban este tipo de servicios. Como resultado, se obtuvo una reducción del tiempo de espera de los pacientes, mejoramiento en los servicios de emergencia, y en los servicios de salud mental. Este estudio demostró asimismo, que la utilización del resumen electrónico de los datos diarios podía proveer información que podría ayudar al mejoramiento de los resultados en la atención en los hospitales y la eficiencia de los mismos.

Ma, Yanqiang Allen (2003), realizó en EE.UU una tesis sobre los factores determinantes para la integridad y la productividad de los sistemas de información [YAN 2003]. Indicó que los dos temas dominantes en la industria de la salud eran el costo y la calidad. El costo de la salud ha ido aumentando en los últimos veinte años, por lo que los hospitales han tenido gran presión para reducir estos costos.

Debido a que los Sistemas de Información de los Hospitales (HIS) fueron introducidos en la administración de los hospitales, se esperaba que éstos mejoraran la calidad del cuidado de los pacientes, así como también, la dirección de los hospitales. Sin embargo, la relación entre la integridad de los HIS y el comportamiento del hospital no ha sido nada claro. La integridad de los HIS es medida a través de la estructura de la funcionabilidad, conectividad y funcionabilidad de decisiones administrativas. La estructura de la funcionabilidad se manifiesta en la habilidad del sistema para automatizar las tareas organizacionales; conectividad se manifiesta en la habilidad del sistema de información para facilitar las comunicaciones; funcionabilidad de decisiones administrativas significa la habilidad del sistema para recuperar, manipular y mostrar información desde una base de datos integrada para tomar decisiones administrativas específicas. Este estudio examinó los tres factores que los administradores de los hospitales consideran cuando tienen que decidir acerca de un sistema de información para el hospital. Estos factores son: el tamaño del hospital, la integridad del sistema, y el número de servicios tecnológicos que brinda. Este estudio concluyó que los hospitales tienen niveles relativamente altos de funcionabilidad estructural y conectividad; sin embargo, la funcionabilidad de soporte de decisiones es baja. Así mismo, concluyó que es importante para los ejecutivos el saber que existen variantes en la integridad de los HIS, especialmente en la funcionabilidad del soporte de decisiones.

3) *Medical data warehousing as a generator of system component for decision support in health care*

Catibusic S, Hadzagic-Catibusic F, Zubcevic S. (2004) [CAT 2004] realizó en Bosnia una investigación sobre el estudio del Data Warehouse como generador de un componente del sistema para soporte de decisiones en la salud.. Esta investigación tenía como objetivo estudiar el crecimiento del rol del Data Warehouse como información estratégica para la toma de decisiones, y mejorar el conocimiento general sobre los requerimientos del Data Warehouse desde el punto de vista de los usuarios finales, como del sistema de información. El documento final presenta las ventajas y los argumentos a favor para la implementación y la exploración de la información como producto final del proceso del Data Warehouse.

II) Herramientas para la implementación

1) Data Stage

DataStage [DAT 2007] es una herramienta que permite soportar la información que necesita la compañía, y construir un Data Warehouse en “tiempo real”. El DataStage es una herramienta ETL (*Extract/Transform/Load* - Extracción, Transformación y Carga) que utiliza notación gráfica para construir integración de datos para dar soluciones, y está disponible en varias versiones, como Server Edition y Enterprise Edition. Es una de las herramientas ETL más rápidas y potentes del mercado.

2) SSIS

El software SQL Server Integration Services (SSIS) [SQL 2007], permite la integración de los datos de cualquier fuente. SSIS provee una plataforma escalable y extendible que capacita al equipo desarrollador a construir, mantener, y desplegar soluciones de integración para alcanzar soluciones de integración únicas de acuerdo a las necesidades. Destacan sus herramientas de minería de datos y administración de objetos.

3) Sunopsis

También existe en el mercado, Sunopsis [SUN 2007], que ofrece un alto desempeño y una integración efectiva, cubriendo las necesidades de integración. Esta herramienta permite el desarrollo y el mantenimiento simple, que permite que los proyectos de integración se realicen a tiempo y en presupuesto. Sunopsis trabaja con una arquitectura ELT (Extracción, Load, Transform) en lugar de la tradicional ETL.

4) Microstrategy

Existen soluciones como MicroStrategy Business Intelligence Solutions [MIC 2007] que permite mejorar y predecir el comportamiento del negocio, poniendo información en las manos de toda persona de negocios en la empresa. Esta tecnología ofrece capacidades de monitoreo, de reportes y de análisis, que permiten tomar mejores decisiones cada día, y lograr las metas planteadas en cada organización. Esta herramienta permite la generación de scorecards y dashboards, reportes, análisis OLAP, análisis avanzado y predictivo, alertas y notificaciones.

5) Cognos

Cognos 8 Business Intelligence [COG 2007] es una plataforma del grupo IBM que

permite la generación y visualización de reportes, cubos, dashboards y Balance scorecards, además de la gestión de permisos y usuarios necesaria para la implementación de la plataforma.

6) Business Objects

BusinessObjects [BUS 2007] es otra plataforma BI que se caracteriza por ofrecer distintas funcionalidades según el tamaño de la empresa que la adquiere y la licencia.

7) Pentaho

Existen herramientas orientadas a la Inteligencia de Negocios, de código abierto (*Open Source*) y de uso libre. Entre estas herramientas se encuentra Pentaho [PEN 2007], la cual es una herramienta muy completa, pues incluye elaboración de reportes, cubos, dashboards, data mining, ETL y una plataforma BI (lugar desde donde se puede acceder a los datos).

8) Octopus

Octopus [OCT 2007] es, al igual que pentaho, una herramienta libre pero sólo se centra en los procesos ETL. Está basada en Java y por lo tanto se puede conectar a cualquier fuente JDBC.

A continuación se presenta un cuadro comparativo, con las herramientas mencionadas anteriormente, que muestra las características trascendentales para un trabajo como el que se desarrollará en el presente proyecto de tesis.

Herramienta	Permite ETL	Elaboración Reportes	Uso Libre (OpenSource)
DataStage	Si	No	No
SQL Server Integration Services	Si	Si	No
Sunopsis	Si	No	No
MicroStrategy Business Intelligence Solutions	No	Si	No
Bussiness Objects	No	Si	No
Cognos Business Intelligence	No	Si	No

Pentaho	Si	Si	Si
Octopus	Si	No	Si

Cuadro 1.1 Comparativo de herramientas de Inteligencia de Negocios



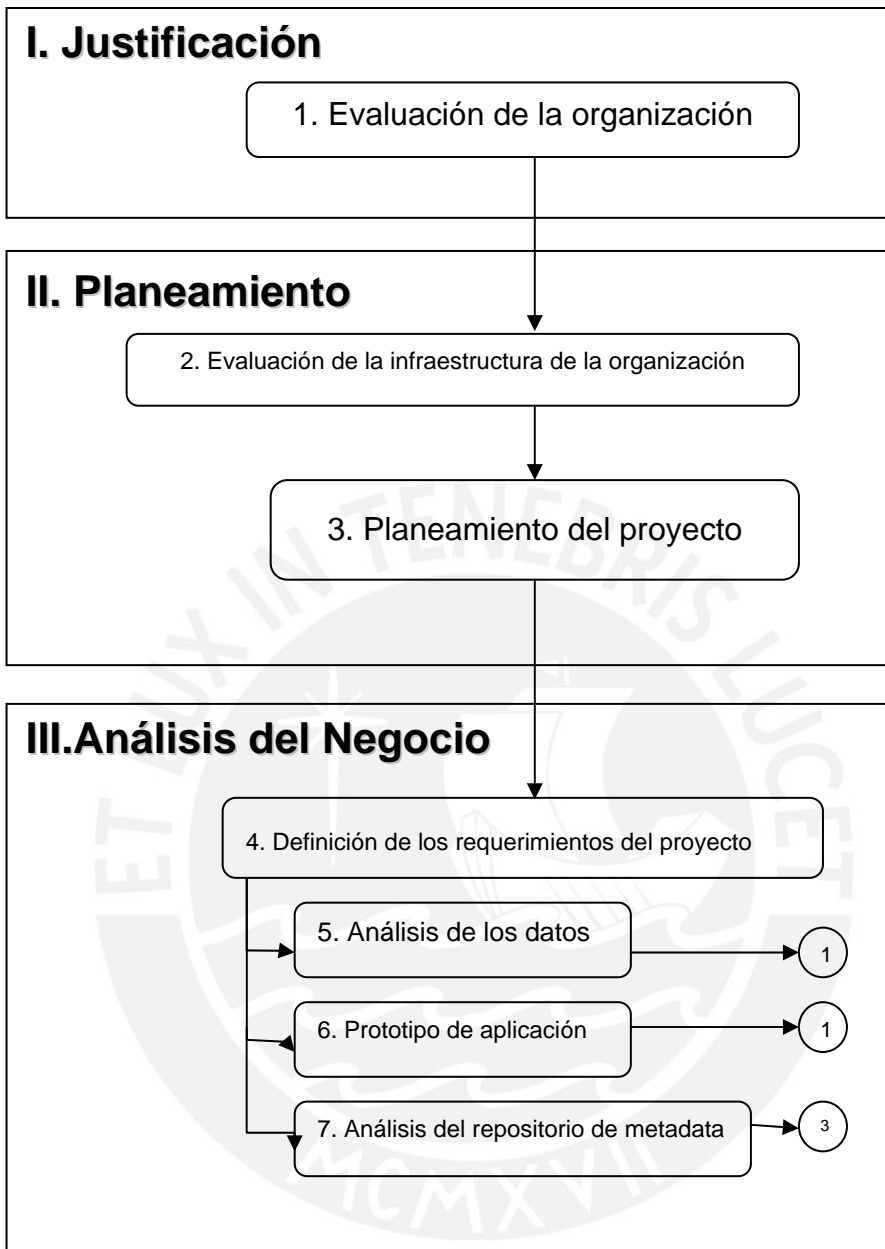
2. Análisis

En este capítulo se brinda la metodología a usar en este proyecto, así como los requerimientos funcionales y no funcionales que deberá cumplir el proyecto. Asimismo, se brinda una descripción de por qué se han elegido las herramientas a utilizar, y el plan de pruebas que se contemplará en el presente proyecto.

2.1. Metodología

Según Moss [MOS 2003], casi todo proyecto de ingeniería, estructural o de software, pasa a través de seis etapas desde la concepción hasta la implementación. También se indica que los procesos de ingeniería son iterativos, ya que una vez puestos en producción, los proyectos son continuamente mejorados dadas las sugerencias de la comunidad que usa el producto. Cada iteración produce una nueva versión del producto, y el producto final va madurando y mejorando.

El siguiente gráfico ilustra el ciclo de vida para desarrollar un sistema de soporte de decisiones según [MOS 2003]. Este ciclo se usó como base para desarrollar el presente proyecto de tesis.



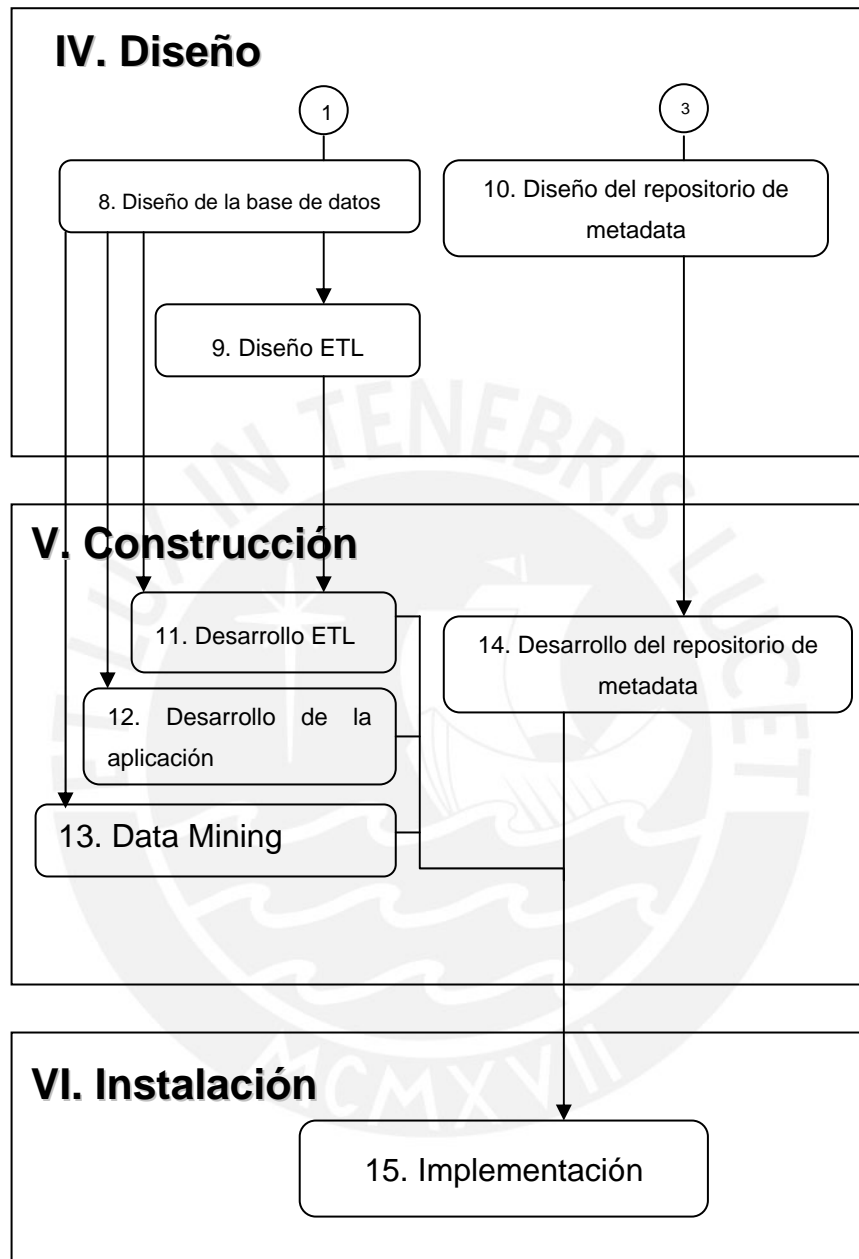


Figura 1.1 Ciclo de vida del de desarrollo de un sistema de toma de decisiones

I. Justificación

Evalúa la necesidad de la organización de construir un nuevo sistema de soporte de decisiones.

2.1.1.1 Evaluación de la organización

En esta etapa, el problema o la oportunidad del negocio es definida, y una solución de inteligencia de negocio (BI) es propuesta. Cada aplicación BI debe ser justificada monetariamente, debe definir claramente los beneficios, debe plantear la solución a los problemas de la organización y definir las ventajas que le darán a la empresa.

II. Planeamiento

Desarrolla estrategias y planificación, que permiten saber cómo se logrará y desarrollará el proyecto.

2.1.1.2 Evaluación de la infraestructura de la organización

Debido a que las aplicaciones BI son iniciativas que afectan a toda la organización, debe ser creada una infraestructura para soportarlas. Algunos componentes de esta infraestructura pueden ya existir, pero otros deberán ser adquiridos para este proyecto. La infraestructura tiene dos componentes:

- Infraestructura técnica: incluye hardware, software, sistemas de manejo de base de datos, sistemas operativos, sistema de red, repositorios de metadata, sistemas utilitarios.
- Infraestructura no técnica: estándares de meta data, modelo lógico del negocio, metodologías, procedimiento de pruebas, procesos del control de cambio.

2.1.1.3 Planeamiento del proyecto

Los sistemas de soporte de decisiones son muy dinámicos. Los cambios en el alcance, el personal, el presupuesto, la tecnología, y los sponsors pueden afectar en gran medida el éxito del proyecto, por lo cual el planeamiento del proyecto debe ser detallado, y el progreso y cumplimiento de éste debe ser seguido y reportado.

III. Análisis del negocio

Desarrolla un análisis detallado del problema o la oportunidad en la organización, para entender en forma completa los requerimientos para una posible solución.

2.1.1.4 Definición de los Requerimientos del proyecto

Decidir el alcance del proyecto es una de las tareas más difíciles en un sistema de soporte de decisiones. El deseo de tener todo instantáneamente entusiasma a todos, pero los requerimientos se deben definir de acuerdo a las posibilidades del cumplimiento en cada entregable. El desarrollador debe esperar que estos requerimientos cambien a lo largo de proyecto.

2.1.1.5 Análisis de los datos

El gran reto de un sistema BI es la calidad de los datos fuente, por lo que en esta etapa se evalúan los estándares en los datos.

2.1.1.6 Prototipo de aplicación

Esta etapa permite a los desarrolladores y a los involucrados ver el potencial y las limitaciones de la tecnología, y también brinda la oportunidad de ajustar los requerimientos del proyecto, y las expectativas del mismo.

2.1.1.7 Análisis del repositorio de metadata

Toda la metadata del negocio debe ser guardada en un repositorio, y éstos pueden ser comprados o construidos. En cualquier caso, los requerimientos para el tipo de metadata deben ser documentados en un modelo lógico. En el presente proyecto de tesis sólo se ha construido y guardado la metadata en un repositorio.

IV. Diseño

Concibe un producto que soluciona el problema de la organización.

2.1.1.8 Diseño de la base de datos

El diseño de la base de datos debe estar acorde con los requerimientos para acceder a la información de la organización.

2.1.1.9 Diseño ETL (Extracción, Transformación y Carga)

Los datos fuente para la aplicación BI vendrán de varias plataformas. El propósito de esta etapa es fusionar los datos de las plataformas en un formato para el Data Warehouse.

2.1.1.10 Diseño del repositorio de Metadata

Si se compra el repositorio de metadata, éste debe cumplir los requerimientos del modelo lógico de la metadata. Si se construye el repositorio, se debe tomar la decisión si éste estará basado en entidad-relación u orientado a objetos. En cualquier caso, debe de cumplir con los requerimientos del modelo lógico. Esta etapa no se ha considerado en el presente trabajo de tesis.

V. Construcción

Construye el producto en un marco de tiempo pre-determinado.

2.1.1.11 Desarrollo ETL

Existen muchas herramientas disponibles para el proceso ETL, algunas de ellas son sofisticadas, y otras simples. Se debe tomar en cuenta la limpieza de los datos, la transformación requerida para los datos, el análisis de los datos, y el diseño ETL para tomar la decisión acerca de cuál herramienta conviene más.

2.1.1.12 Desarrollo de la aplicación

Una vez que el prototipo se ha ajustado a los requerimientos de la empresa u organización, se dará inicio al análisis y el desarrollo de cómo se accederá a los datos. El desarrollo de la aplicación que se mostrará al usuario, se hará en forma paralela con el desarrollo del ETL y del repositorio de meta data.

2.1.1.13 Data Mining

Muchas aplicaciones de BI son limitadas por reportes preestablecidos que reemplazan a antiguos reportes generados manualmente. El aprovechamiento de un sistema de soporte de decisiones viene de la información escondida en los datos de la organización que sólo puede ser descubierta gracias a las herramientas de Data Mining. Esta etapa se encuentra fuera del alcance del proyecto de tesis.

2.1.1.14 Desarrollo del repositorio de metadatos

Si se decidió desarrollar el repositorio de datos, en vez de adquirirlo, se debe considerar que este subproyecto consume gran parte del tiempo. En el caso de este proyecto, se adquirió un repositorio de metadatos libre proporcionado por la plataforma Pentaho.

VI. Instalación

Implementa el producto final, y luego mide la efectividad para determinar si la solución alcanza, excede o falla en alcanzar los requerimientos.

2.1.1.15 Implementación

Una vez que se ha probado cada componente del proyecto, se comienza a instalar el motor de la base de datos y la plataforma Web. Se programa entrenamiento para los usuarios. En esta etapa comienza las funciones de soporte, que incluye mantener la base de datos, programar y correr los jobs ETL, monitorear el comportamiento del sistema y afinar la base de datos.

2.2. Requerimientos Funcionales

A continuación se presenta un cuadro con los requerimientos funcionales del DSS a desarrollar. Estos requerimientos se han desarrollado tomando en cuenta las consideraciones planteadas en la metodología y serán validados con usuarios del ambiente de salud y con el asesor del proyecto de tesis. Estas validaciones se llevarán a cabo por medio de la presentación del sistema ya implementado y analizando si se cumplen con los requerimientos uno a uno.

Número	Requerimiento	Nivel Prioridad	Exigible / Deseable
1	Se extraerá de forma adecuada la información de los sistemas fuentes, según la naturaleza de estos (bases de datos, archivos Excel, archivos de texto).	2	D
2	Se creará un Data Warehouse que cubra las áreas en una entidad de salud pública de: admisión, caja, cita, general, cuentas corrientes, historia clínica, hospitalización,, personal.	1	E
3	Se cargará el Data Warehouse con todos los datos extraídos de las distintas fuentes, manteniendo la información relevante para los propósitos planteados.	1	E
4	Se estandarizarán los formatos de los datos, obtenidos de las fuentes transaccionales, dentro del Data Warehouse.	1	E
5	Se crearán rutinas de limpieza de datos, que permitan verificar la validez y calidad de los datos, según los criterios de la entidad de salud. Estas rutinas incluyen el formato correcto de una cadena, los filtros de los valores nulos o no válidos y la verificación de data consistente por medio de constraints.	2	E
6	Se crearán Data Marts a partir del data Warehouse, según las áreas temáticas de	1	E

	Admisión, General, Historia Clínica, Personal, Caja, Cuentas corrientes y Citas, que se ajusten a las necesidades de una entidad de salud, sirviendo como base para aquellas entidades que deseen implementar el DSS.		
7	Se automatizará el proceso ETL para cada Data Mart, según la necesidad de actualización de cada Data Mart.	2	E
8	Se creará procesos ETL totales para las cargas absolutas y procesos ETL incrementales para cargas periódicas.	1	E
9	Se creará un repositorio en Base de Datos para almacenar la metadata de los procesos ETL.	1	E
10	Se elaborarán 8 reportes que presenten los datos cargados en los Data Marts y cuyo diseño ayude a una correcta toma de decisiones en la entidad de salud. Estos reportes se describen en las secciones 3.4.1.5, 3.4.1.6 y en el anexo A: Reportes.	1	E
11	Se proporcionará una plataforma de BI Web para la presentación de los reportes.	1	E
12	Se permitirá que el usuario elija los criterios que desea para sus reportes. Podrá realizar las operaciones de <i>Drill Down</i> , <i>Drill Up</i> , <i>Slicing</i> , <i>Dicing</i> y <i>Pivoting</i> , descritas en la sección de marco conceptual, en el capítulo 1.	1	E
13	El sistema permitirá comparar la información en los reportes, en instantes de tiempo distintos, por medio de filtros luego de la ejecución del informe según seleccione el usuario.	2	E
14	El sistema presentará selecciones dinámicas antes de ejecutarse el informe para que los resultados dependan de la selección del usuario.	2	D

15	El sistema permitirá presentar los informes de manera gráfica, para una mejor visualización de la información a través del tiempo o según criterios elegidos por el usuario.	2	E
16	El sistema notificará al usuario cuando se haya cargado satisfactoriamente nueva información en el Data Mart, y por lo tanto los informes estén actualizados para un nuevo periodo de tiempo. Esto se realizará vía mail.	2	D
17	Se diseñarán indicadores pertinentes, que vayan de acorde al rubro de salud y según los requisitos de los usuarios, y que permitan analizar la organización, sin desviarla hacia áreas que no son de su interés, para la correcta gestión de la entidad de salud. Los indicadores están descritos en las secciones 3.4.1.5, 3.4.1.6 y en el anexo A: Reportes.	1	E
18	El sistema presentará Dashboards para visualizar el estado de los indicadores clave del la entidad de salud.	2	D

Cuadro 1.2 Requerimientos Funcionales

E= Exigible

D= Deseable

Nivel de Prioridad de mayor a menor: 1, 2

2.3. Requerimientos No Funcionales

Número	Requerimiento	Nivel Prioridad	Exigible / Deseable
1	El sistema deberá estar desarrollado en una herramienta de software libre.	1	E
2	La interfaz de la plataforma BI debe ser fácil de usar, para que el personal de la entidad de salud, no especializados en herramientas informáticas, pueda hacer uso de los reportes	1	E

	fácilmente.		
3	El sistema deberá permitir la presentación de los reportes en línea, con disponibilidad las 24 horas del día, los 7 días de la semana.	1	E
4	El sistema correrá bajo la plataforma de Internet Explorer y Mozilla Firefox	1	E
5	El sistema utilizará un gestor de base de datos de licencia libre.	1	E

Cuadro 1.3 Requerimientos No Funcionales

E= Exigible

D= Deseable

Nivel de Prioridad de mayor a menor: 1, 2

2.4. Plan de Pruebas

Objetivos

Este plan de pruebas tiene como finalidad dictar los pasos a seguir para realizar un conjunto de pruebas para verificar la consistencia del producto final desarrollado.

Las pruebas a realizar serán del tipo de caja negra. Esto es, se tendrá un conjunto de datos de entrada y se analizará si la salida es la correcta respecto a los datos de entrada. Para esto, se determinará los datos de salida que se deben obtener a partir de los datos de entrada. Luego se ejecutará el proceso y se dará a la prueba como satisfactoria sólo si el resultado coincide con los datos de salida predichos.

Características a ser probadas

- Se probará que cada proceso de extracción de datos para las diferentes fuentes se realice en forma correcta.
- Se probará que cada proceso de transformación se realice en forma correcta para todos los procesos.
- Se probará que cada proceso de carga se realice de forma correcta sin violar la integridad referencial de los datos.
- Se probará en forma conjunta el proceso de ETL para cada uno de los procesos.
- Se probará que el proceso de ETL se ejecute según un horario programado.

- Se probará que todos los reportes mostrados muestren los datos actualizados a la última carga, y los gráficos según lo planeado. Por ejemplo, si en caso se mostrara la opción de poder realizar *drill-down* en un reporte, entonces se verificará que se cumpla ese requerimiento.
- Se verificará que los filtros mostrados para ejecutar cada reporte, sean los indicados para cada uno de éstos, y que los datos mostrados en los reportes cumplan con lo mostrado en los filtros.

Recursos

Se dispondrá de los recursos para poder realizar las pruebas y de todo el software necesario para cumplir con cada una de las mismas. Se deberá contar con los siguientes programas instalados en la máquina de prueba:

- Pentaho Data Integration: Kettle Project
- Pentaho Report Designer
- Pentaho Análisis Services: Mondrian
- Erwin Data Modeler
- PostgreSQL
- Microsoft Excel

Así mismo, se deberá de contar con los archivos fuentes necesarios para el proceso de extracción de datos, y con las rutinas de procesamiento y conversión.

Responsables y Tareas

Es responsabilidad del tesista el realizar las pruebas para cada reporte que se elabore. Asimismo, será responsable de que todos los procesos del ETL para la explotación de los reportes sean congruentes con los resultados deseados.

Se deberá realizar la siguiente documentación:

- Casos de prueba: contendrá cada uno de los casos a ser probados en los procesos, así como el método a usar.
- Resultado de la prueba: El desarrollador será responsable de generar un documento en el cual se muestren los casos de prueba realizados y los resultados obtenidos para cada prueba.

Luego de finalizadas las pruebas, se verificará la consistencia de estas y si existiesen problemas, se realizará la revisión del proceso afectado.

Riesgos

Es importante recalcar que existen posibles riesgos asociados al desarrollo del plan de pruebas, entre los cuales se nombrarán los más importantes, y los planes de contingencia asociados a cada uno:

- Plan de pruebas: es posible se realice en forma incorrecta el plan de pruebas, por lo que las pruebas realizadas serán defectuosas o incorrectas. Si en caso hubiera dicho problema se volverá a hacer el plan de pruebas, y se verificará nuevamente la consistencia.
- Atrasos en la corrección de errores: se puede presentar el caso en el cual se descubran muchos errores, y no haya tiempo de corregirlos. Para evitar dicho problema, se tomará la medida verificar la consistencia de lo que vaya avanzando, y verificar los datos de salida en los reportes, y procesos ETL.

Aprobación

Las pruebas realizadas serán verificadas por el asesor asignado para darle la aprobación final.

Casos de Prueba

En esta sección se detallará el formato de caso de prueba de un Data Mart. Para consultar los casos de prueba de todos los Data Marts revise el anexo L: Desarrollo de las Pruebas.

Caso de prueba para los reportes del Data Mart de Caja_DM:

Datos de entrada Tablas del Data Warehouse	Resultados Intermedios. Tablas del Data Mart	Resultados Finales Informe
Caja	D_Tipo_Documento	Operaciones e ingresos de caja por tipo documento, horario y día

Dia	D_Caja	
Tipo_documento	F_Carga_Caja	
Documento_venta		
Linea_doc_venta		
Zona_hospital		

Cuadro 1.4 Casos de prueba para los reportes del Data Mart de Caja_DM

2.5. Software a Utilizar

Se especificó, como uno de los requerimientos no funcionales, que las herramientas a utilizar en el presente proyecto de tesis fueran de uso libre.

A continuación se identificarán las necesidades de software que se tienen para el proyecto:

1) Modelador de Datos

Se necesita un modelador de base de datos, que permita mostrar, de una manera gráfica, la interacción entre las distintas tablas que se relacionan con la solución, facilitando el diseño de estas, el análisis y el diseño del Data Warehouse y los Data Marts. Estas herramientas facilitan además la creación de los scripts necesarios para inicializar la base de datos, según las tablas en el modelador. Se ha elegido utilizar la herramienta Erwin, debido a las siguientes razones:

- Se tiene una licencia educativa proporcionada por la universidad.
- Contiene todas las características básicas de los modeladores de base de datos (generar tablas, manejo de llaves, manejo de tipos de datos, vistas, comentarios en tablas y columnas) además de características avanzadas tales como sincronización de base de datos con modelo.
- Facilita la documentación, autogenera reportes.
- Está optimizado para trabajar con cualquier gestor de base de datos.
- Generador SQL.
- Se posee experiencia con el uso de esta herramienta, por lo que se reduce la curva de aprendizaje.

2) Gestor de Base de Datos

En cuanto al gestor de base de datos, se ha optado por utilizar PostgreSQL [POS 2007]. Se realizó esta elección debido a que, según [PEC 2007], PostgreSQL es un gestor de licencia libre, soporta la integridad referencial, soporta el uso de transacciones, posee una facilidad de configuración e instalación y posee una gran escalabilidad, pudiéndose aplicar a grandes bases de datos. La desventaja es que consume una gran cantidad de recursos (ya que verifica integridad referencial), pudiendo resultar muy lento. Cabe mencionar que se opta por este motor de base de datos sólo para el desarrollo, siendo posible utilizar otros gestores al momento de implementar el proyecto en un sistema en producción, permitiendo escoger el motor que más se acomode a las necesidades de la organización. La presencia de integridad referencial es importante para el desarrollo ya que facilita la depuración de errores en el proceso de ETL.

3) Plataforma BI

Se optó por utilizar la plataforma Pentaho Open Source Business Intelligence como plataforma BI. Esto es debido a que se han creado muchos módulos que se adaptan a esta plataforma. Además es una alternativa de licencia libre. Pentaho funciona sobre cualquier navegador, soporta la ejecución de Dashboards y reportes en tiempo real, permite gestionar usuarios, colgar documentación, monitorear la ejecución de “jobs” y finalmente, posee una versión de demostración que permite aprender rápido el manejo de la plataforma, reduciendo la curva de aprendizaje, y posee ejemplos de los cuales partir. Además posee mucha documentación para consultar, tiene una interfaz muy amigable para el usuario, e incluso, es posible editar esta interfaz para ajustarse más a las necesidades de cada usuario.

4) ETL

Para el proceso de ETL, se utilizará Kettle Data Integration. Kettle es una herramienta libre que se encarga de este proceso y que ahora forma parte del proyecto Pentaho. Kettle posee los módulos Pan y Spoon, para diseñar el proceso de ETL, y Kitchen y Chef, para la programación de los Jobs de la ETL. En general, Kettle es una herramienta gráfica, lo que facilita su aprendizaje y su uso y puede trabajar con diversas fuentes de datos y conectarse a muchos motores de base de

datos, tanto como fuente, como para destino. Otra de las ventajas de Kettle, es que posee rutinas que facilitan el proceso de limpieza de datos.

5) Diseñador de Cubos

Otro de los módulos que forman parte del proyecto Pentaho, y que se utilizará en el presente proyecto de tesis, es el Cube Designer (Mondrian). Mondrian es un servidor OLAP escrito en Java, que permite el análisis de grandes volúmenes de datos almacenados en SQL, sin necesidad de escribir SQL. Posee un rendimiento muy bueno, pues hace uso del lenguaje MDX para modelos multidimensionales, y es eficiente al realizar exploración dimensional de los datos (*drills, slicing, dicing, etc*). El Cube Designer permite la definición de cubos, jerarquías, conexiones a base de datos y Facts, en archivos XML que posteriormente pueden ser leídos por el Report Designer del Pentaho.

6) Diseñador de informes

Finalmente, para la elaboración de reportes, se utilizará Pentaho Reporting (o Report Designer), el cual también pertenece al proyecto Pentaho. Esta herramienta permite realizar reportes simples, cuadros, gráficos de diversos estilos y Dashboards. Estos reportes se pueden diseñar para que antes de ejecutarse pidan datos que influyan en la ejecución de los informes. Es posible publicar los reportes desarrollados por esta herramienta en el servidor Pentaho (plataforma BI), para la visualización por parte de los usuarios. Además, es posible incluir hipervínculos en los reportes, posee un editor de agregaciones complejas y permite un fácil alineamiento de los elementos de los reportes.

En general, en cuanto a las herramientas relacionadas con los procesos de Inteligencia de Negocios, se han elegido todos los que pertenecen al proyecto Pentaho porque son herramientas muy completas, de uso libre, muy bien documentadas y que han demostrado excelentes resultados en otros proyectos. Cada una de las características mencionadas de estas herramientas es una razón más de su utilización.

3. Diseño

En este capítulo se brinda una descripción del diseño a utilizar. Se describirán las características, estándares y modelamiento que se usará.

3.1. Modelamiento de tablas

Como se mencionó antes, se utilizará la herramienta de modelamiento Erwin para la estructura del Data Warehouse y de los Data Marts.

Cada vista contiene un área temática y/o un Data Mart. En una vista pueden aparecer tablas de otra vista. Para facilitar su diferenciación se ha colocado colores iguales a las tablas de cada vista, pero colores distintos entre vistas. Las vistas presentes en el modelo se presentan en la sección 3.2.

Se ha optado por utilizar un modelo híbrido de los enfoques Estrella y Copo de nieve. Se tendrá la característica del esquema Estrella de poseer una alta granularidad en las dimensiones al final de una jerarquía (heredando los atributos

del padre), pero también se tendrá la característica de normalización del Copo de nieve, al poseer más de una tabla en las jerarquías.

A continuación se presenta un cuadro con las características de ambos modelos.

	Estrella	Copo de Nieve
Número de Tablas	Menor	Mayor
Complejidad del Query	Baja	Alta
Complejidad del Modelo	Alta	Baja
Desempeño del cubo	Rápido	Lento

Cuadro 1.5 Comparación entre modelo Estrella y Copo de Nieve

La razón de la elección de este enfoque híbrido fue la construcción de los informes (cubos). Al tener una alta granularidad en la dimensión inmediata a la Fact se acelera el tiempo de respuesta de las consultas, lo cual es importante puesto que existe una gran cantidad de datos en el caso del hospital. Sin embargo, al poseer también la normalización del enfoque Copo de Nieve es más sencillo de entender el modelo, pero aún más importante es que este enfoque permita la ejecución de queries complejos (de ser necesarios) como consultas de Data Mining. En general, el esquema híbrido proporciona las ventajas de ambos enfoques y se tendría una solución robusta que pueda cumplir con las necesidades de una organización que necesita obtener todo tipo de consultas a tiempo, tal como son los hospitales.

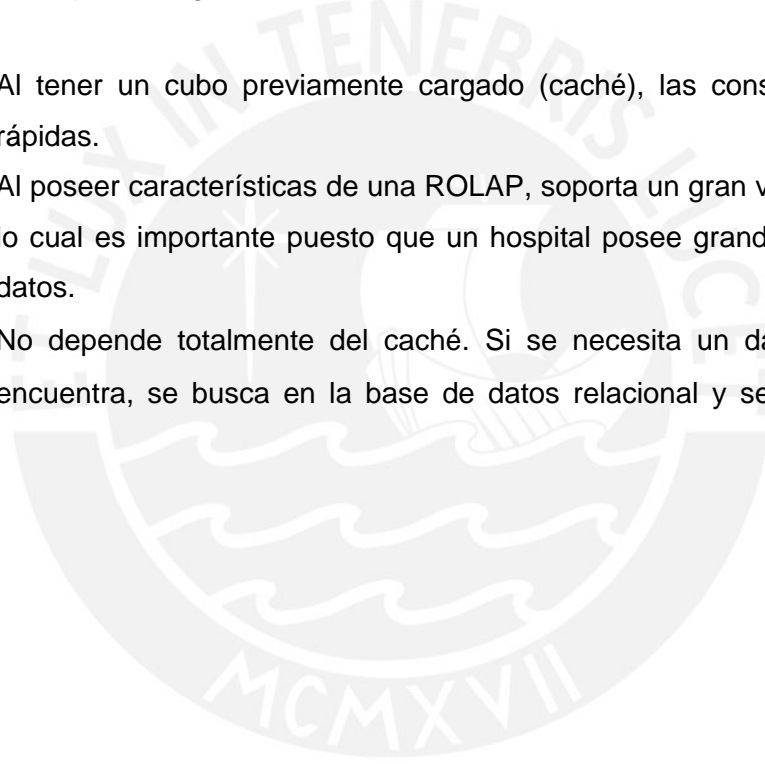
En cuanto a los estándares del modelo, se ha optado por lo siguiente:

- Cada vista que pertenezca al Data Warehouse, poseerá el sufijo DW al final.
- Las vistas correspondientes a algún Data Mart, tendrán el sufijo DM al final.
- Cuando un campo sea una fecha, se utilizará el prefijo "F" en el nombre, seguido de un guión bajo.
- Cuando un campo sea un número que represente la cantidad de algo, se usará el prefijo "N" seguido de un guión bajo.
- Todas las tablas pertenecientes a una vista, poseen un mismo color, para facilitar la identificación. Los Data Marts que provienen de un Warehouse, tendrán el mismo color que las tablas del Warehouse.
- Se crearán llaves artificiales para el Data Warehouse. Las llaves transaccionales se guardarán en un campo llamado "CODIGO", en el Warehouse, si es que se posee una llave en el transaccional.

- Para el nombre de los constraints de Foreign Key se está utilizando la siguiente forma: *FK_DescriptorTablaPadre_DescriptorTablaHijo*, donde con descriptor se refiere a cualquier nombre que pueda identificar a una tabla.
- En los Data Marts, las dimensiones empezarán con el prefijo “D”, seguido por un guión bajo, mientras que las Facts tendrán el prefijo “F” seguido de un guión bajo.

En cuanto a la arquitectura que hay que elegir, se ha barajado las opciones de ROLAP, MOLAP y HOLAP (descritas en el capítulo 1). Se ha decidido finalmente usar HOLAP por las siguientes razones:

- Al tener un cubo previamente cargado (caché), las consultas se vuelven rápidas.
- Al poseer características de una ROLAP, soporta un gran volumen de datos, lo cual es importante puesto que un hospital posee grandes volúmenes de datos.
- No depende totalmente del caché. Si se necesita un dato y este no se encuentra, se busca en la base de datos relacional y se obtiene el dato.



3.2. Análisis Dimensional

En esta sección se presentarán las diversas vistas del modelo según el modelador de Base de Datos elegido para el proyecto

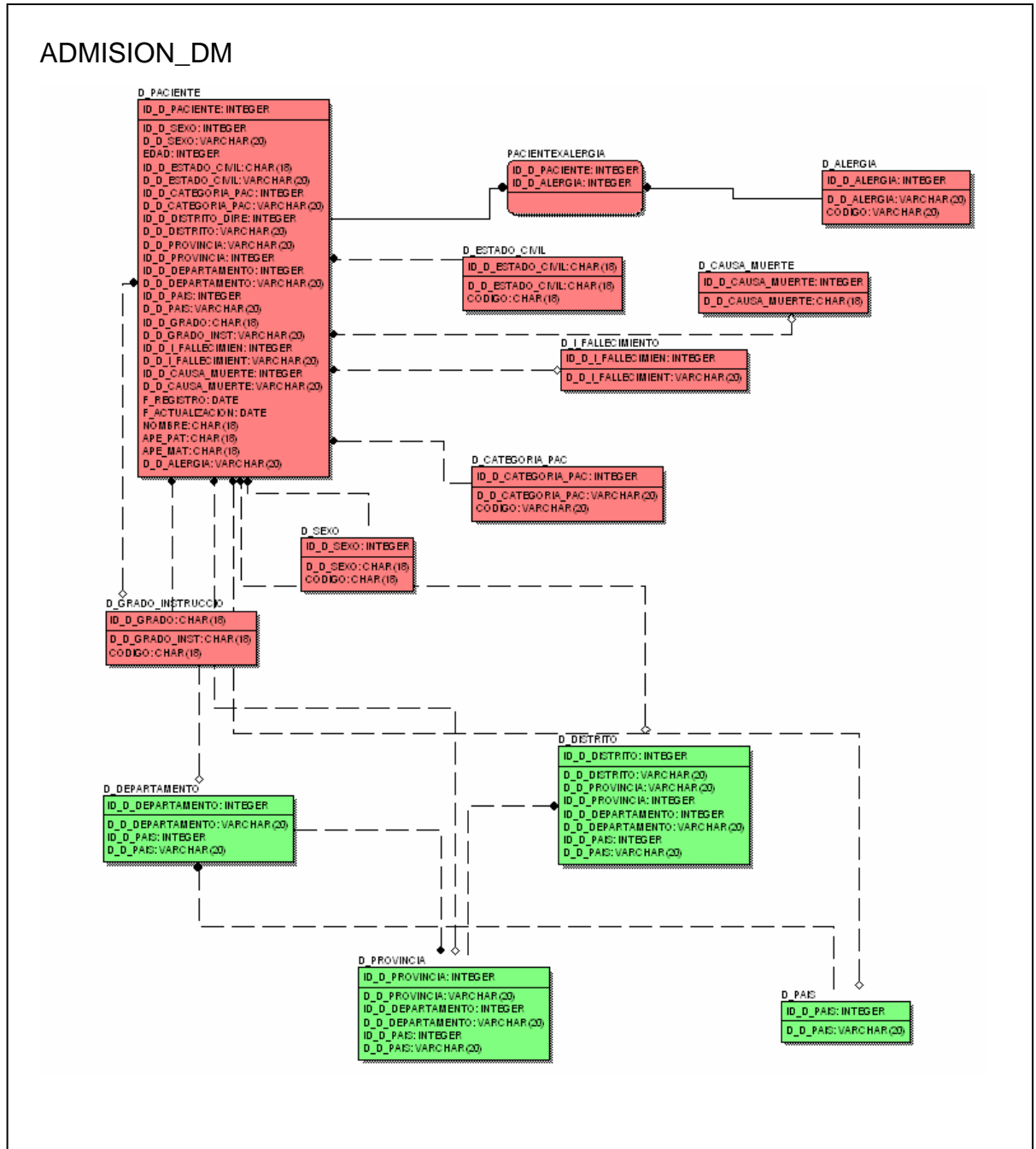


Figura 1.2 Data Mart de Admisión

Esta vista contiene las tablas correspondientes al Data Mart de Admisión, el cual ofrece datos sobre los pacientes del hospital, tales como nombre, fecha de nacimiento, lugar de residencia, datos clínicos, categoría del paciente e indicador de fallecimiento.

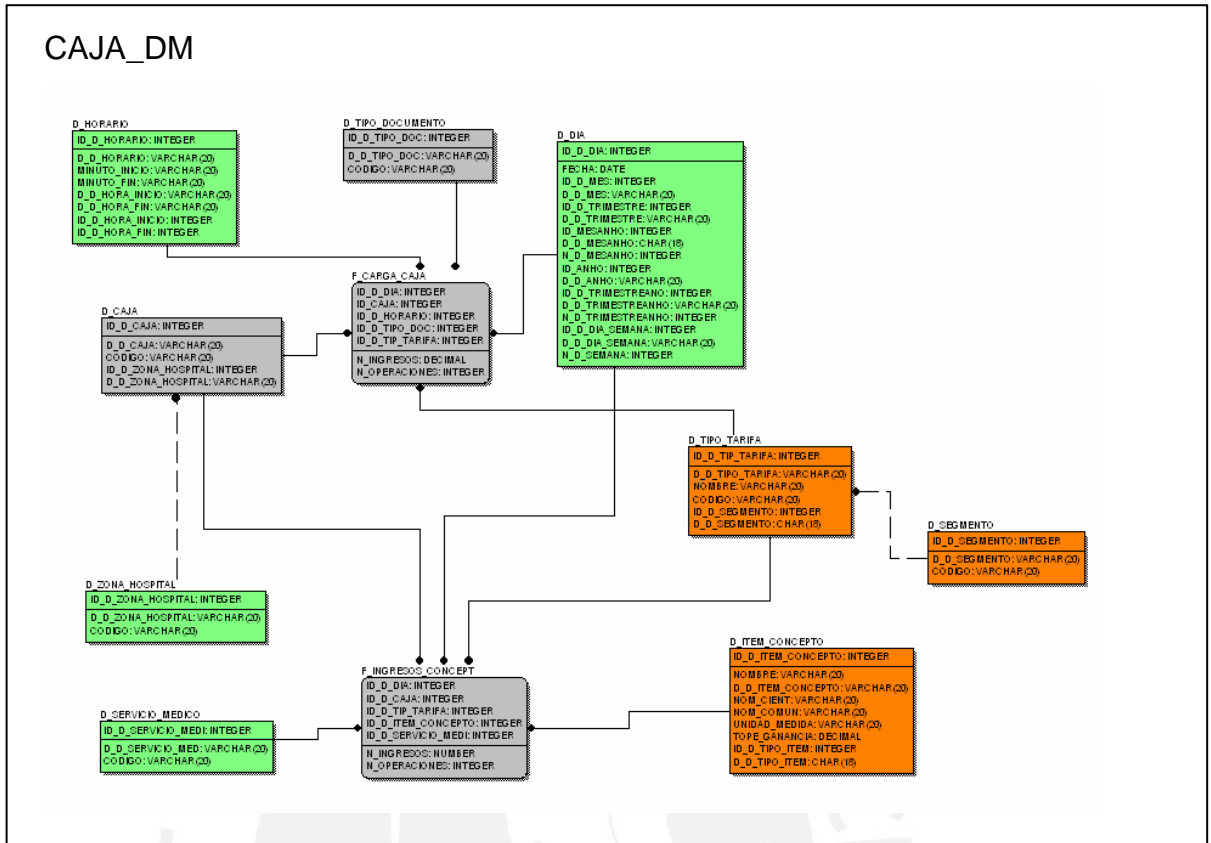


Figura 1.3 Data Mart de Caja

El Data Mart de Caja ofrece información acerca de las operaciones realizadas en los diversos puntos de facturación y los ingresos, según día, caja, horario, documento de venta y tipo de tarifa aplicada a la venta.

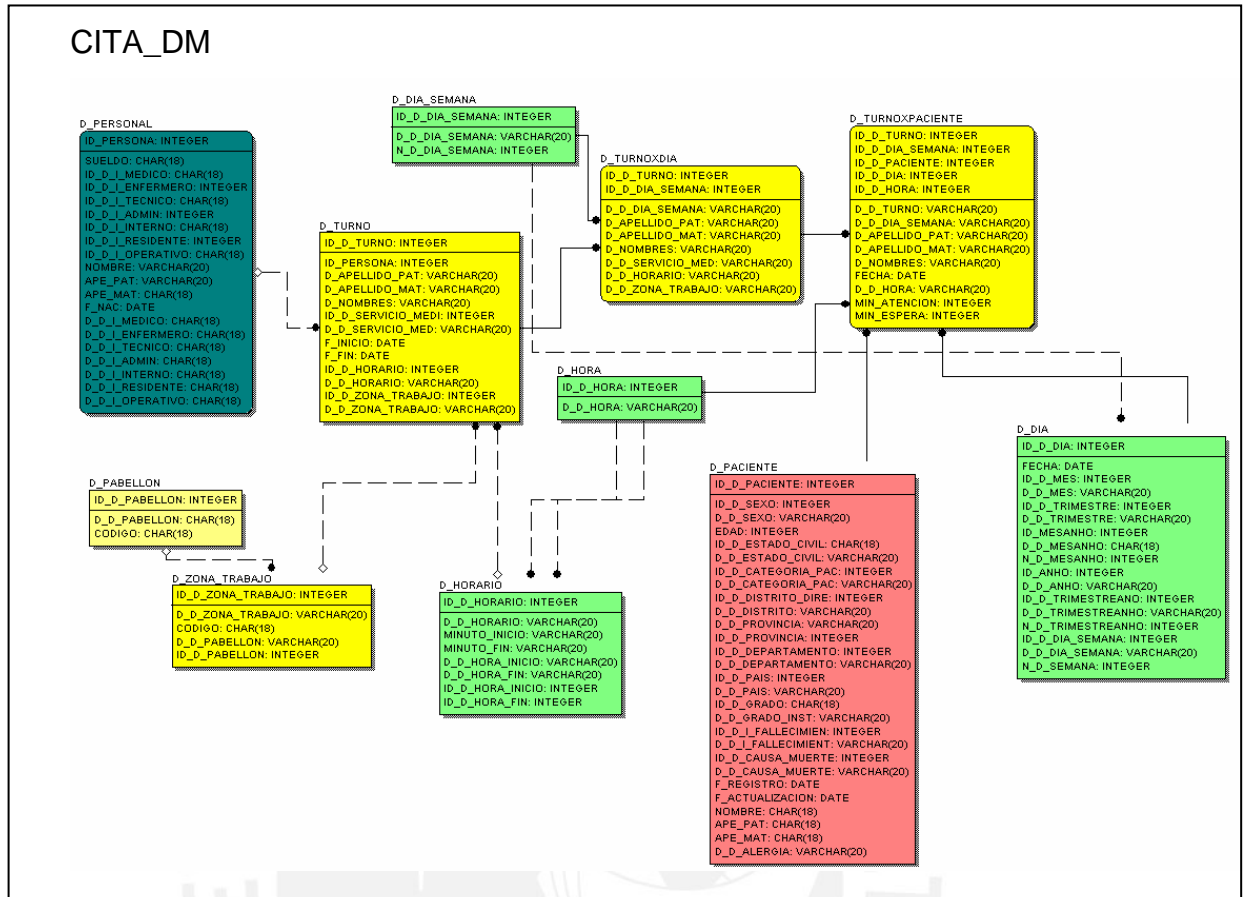


Figura 1.4 Data Mart de Cita

El Data Mart de Cita ofrece información acerca de las consultas que reciben los pacientes, guardando la información relevante tal como hora de la cita, médico que atiende, tiempos de espera y atención, especialidad de la cita y fecha de la cita.

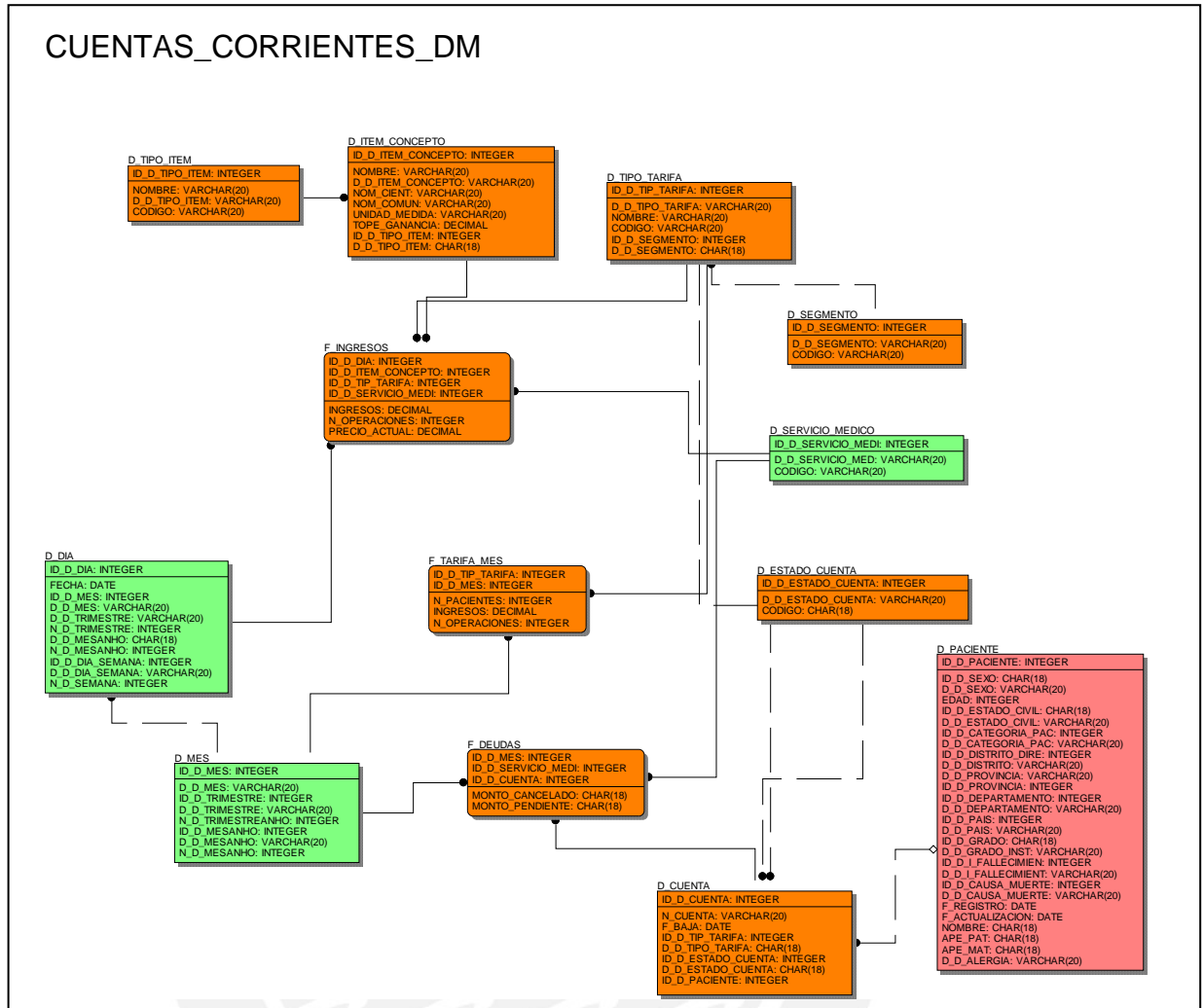


Figura 1.5 Data Mart de Cuentas Corrientes

En el Data Mart de cuentas corrientes, se dispone de los datos económicos de los pacientes y del hospital mismo. Las cuentas corrientes contienen deudas que los pacientes tienen con el hospital y esta vista contiene los diferentes pagos en cada cuenta.

GENERAL_DM

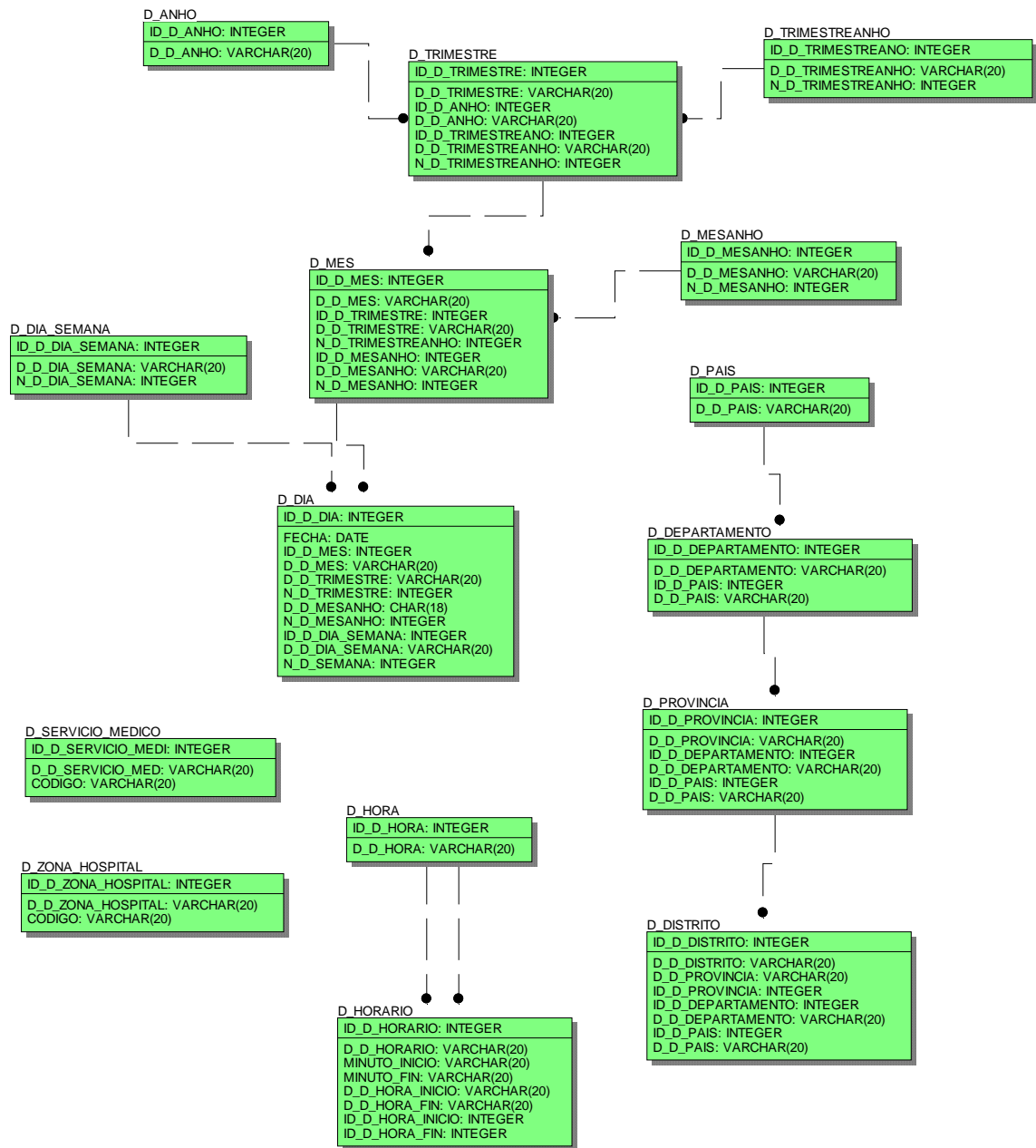


Figura 1.6 Vista de GENERAL_DM

La vista General_DM contiene las dimensiones de tiempo, lugares (ubigeo), horarios, servicios médicos y zonas de hospital, las cuales son utilizados por los demás Data Marts.

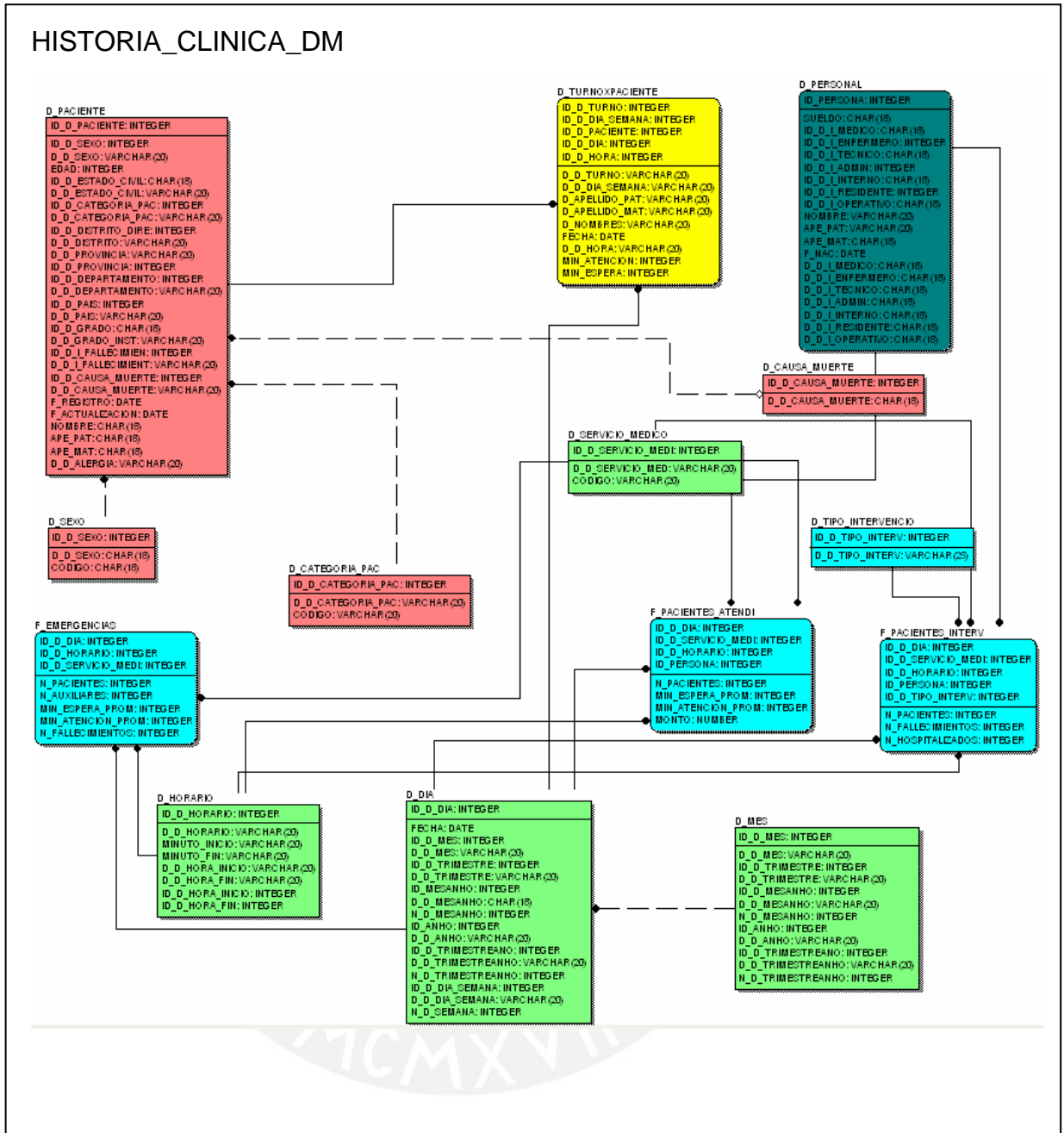


Figura 1.7 Data Mart de Historia Clínica

La vista de Historia Clínica trata exclusivamente con los pacientes. Incluye información acerca de las intervenciones a los pacientes, las atenciones, tiempos de espera, atención, ingresos por consultas y finalmente contiene indicadores para el área de emergencia.

PERSONAL_DM



Figura 1.8 Vista de PERSONAL_DM

Finalmente, la vista de Personal contiene el Data Mart con las dimensiones necesarias para modelar al personal del hospital.

Las vistas relacionadas con el Data Warehouse se encuentran en el Anexo C: Vistas del Warehouse.

3.3. Arquitectura utilizada

Existen dos filosofías conocidas acerca de diseño del Datawarehouse. Por un lado, Inmon [INM 2000], plantea el siguiente diseño:

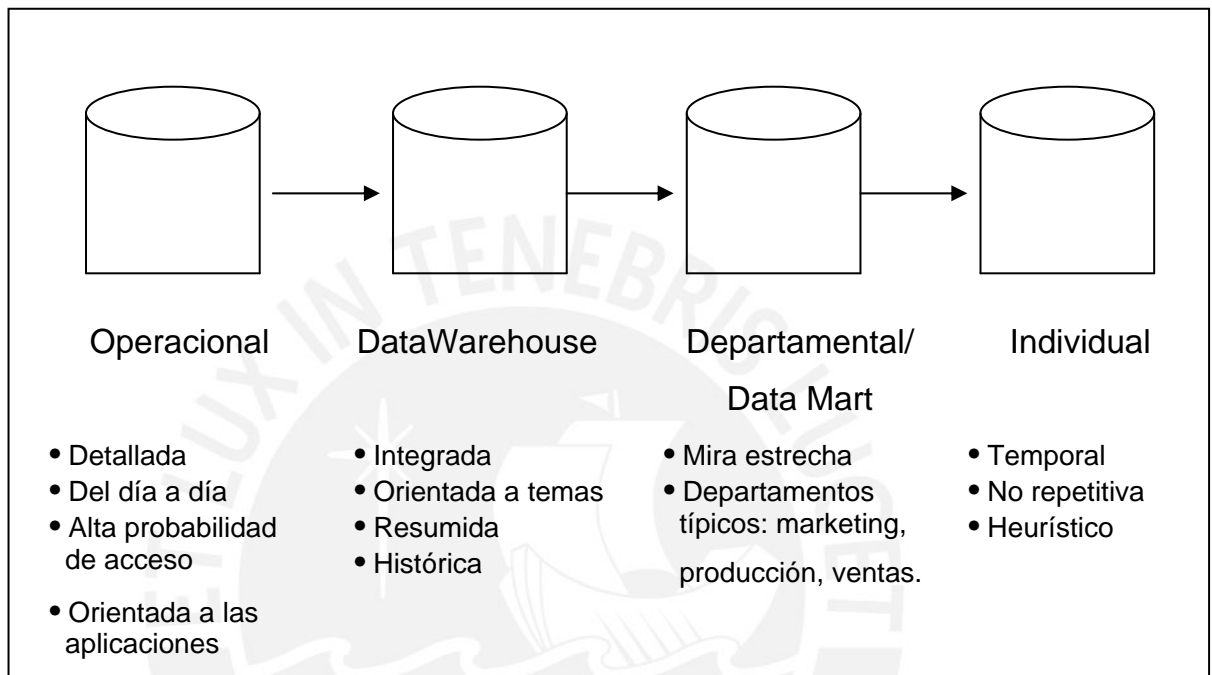


Figura 1.9 Niveles de Arquitectura

Según Inmon, existen cuatro niveles de datos en la arquitectura: el nivel operacional, el atómico o Datawarehouse, el departamental o nivel de Data Mart y el nivel individual. El nivel operacional guarda los datos de las aplicaciones y sirve para los usuarios que necesitan un buen tiempo de respuesta en sus transacciones del día a día. El nivel de Datawarehouse guarda datos integrados e históricos que no pueden ser modificados. El nivel departamental contiene datos moldeados según los requerimientos del usuario final en una forma que satisfaga las necesidades del departamento. En el nivel individual es donde son realizados los procesos heurísticos.

Algunas personas piensan que esta arquitectura genera mucha redundancia de datos. En el nivel operacional se tienen los datos actualizados para los registros, en cambio en el nivel del Datawarehouse se guardan los datos históricos, quedando una relación del tiempo con cada uno de los registros del Datawarehouse.

El nivel departamental o de Data Mart contiene información útil para los diferentes departamentos de la compañía. El Datawarehouse es la fuente para los datos departamentales. Finalmente, el nivel individual es usualmente temporal y pequeño, realizándose en este nivel análisis heurístico.

Inmon menciona que un aspecto importante es el de la integración de datos que ocurre a través de toda la arquitectura. Cuando los datos pasan del nivel operacional al nivel del Datawarehouse son integrados, ya que no tiene sentido el pasarlos sin integrarlos. Si los datos llegaran al Datawarehouse sin ser integrados no podrían ser usados para ser mostrados en reportes.

Por otro lado, Ralph Kimball [KIM 2001] plantea el siguiente diseño:

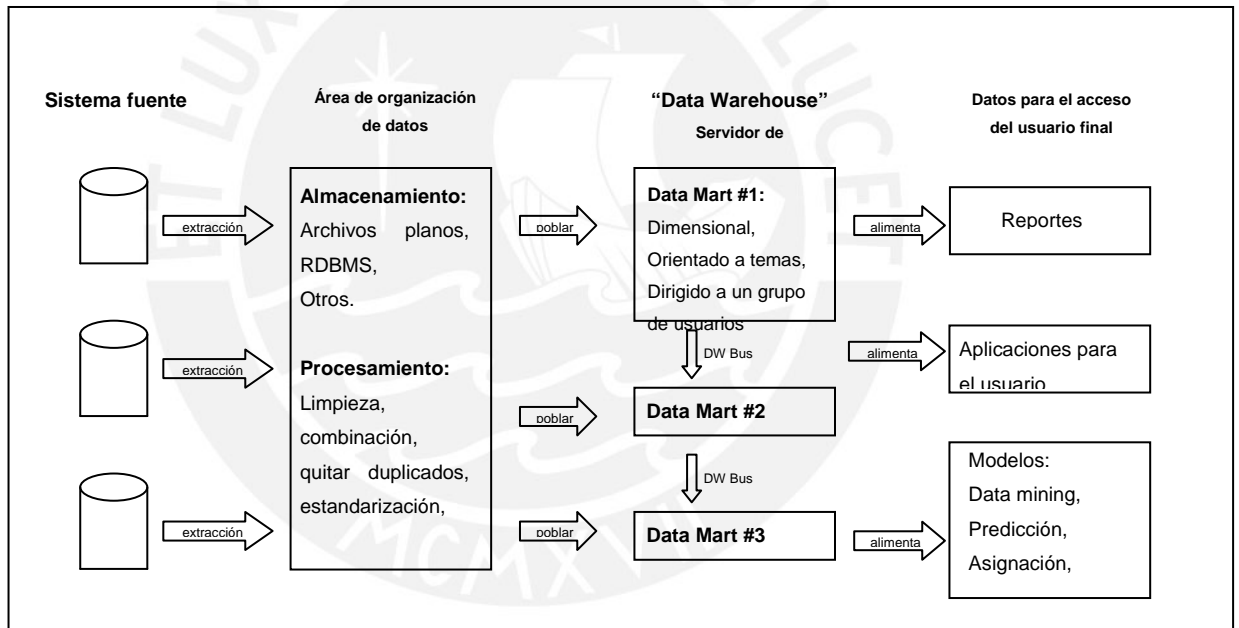


Figura 1.10 Elementos básicos de un Data Warehouse

Según Kimball, el sistema fuente es un sistema operacional cuya función es capturar las transacciones del negocio. Es usualmente llamado “sistema legado” cuya prioridad es su disponibilidad. Las consultas a estos sistemas son limitadas, y son parte del flujo de transacciones normales del día a día. Se asume que los sistemas fuente mantienen pocos datos históricos, y que manejar reportes desde sistemas fuentes es una carga para estos sistemas.

El área de organización de datos es un área para almacenar y preparar procesos que limpian, transforman, combinan, eliminan duplicaciones, archivan y preparan

una fuente de datos para el uso en el Servidor de Presentaciones. Esta área es dominada por actividades de clasificación y procesamiento secuencial y, en algunos casos, no necesita estar basada en una tecnología relacional. Después de verificar los datos con todas las reglas del negocio que se hayan definido, no tendría sentido construir una base de datos física basada en entidad-relación.

El Servidor de Aplicaciones es donde los datos son organizados y almacenados para las consultas directas por los usuarios finales, reportes y otras aplicaciones. Según Kimball, tres diferentes sistemas son requeridos para la función del Data Warehouse: el sistema fuente, el área de organización de datos y el servidor de presentación. El sistema fuente debe ser pensado fuera del Data Warehouse, ya que se asume que no se tiene control sobre el contenido y el formato de los datos en el sistema legado.

El área de organización de datos es un área de almacenamiento inicial y un sistema de limpieza para los datos que se mueven hacia el servidor de presentaciones, y se recalca que el área de organización de datos puede consistir también en un sistema de archivos planos. Es en el servidor de presentaciones donde los datos deben ser presentados y almacenados en un marco dimensional. En tal sentido un Data Mart es parte del Data Warehouse. El Data Warehouse es formado a partir de la unión de todos los Data Marts, y es alimentado por el área de organización de datos.

Para el presente proyecto se ha decidido utilizar la arquitectura planteada por Inmon, ya que se necesita tener los datos, provenientes de las distintas áreas, integrados para asegurar la consistencia de los reportes mostrados. Asimismo, se necesita tener un histórico de los datos del hospital, por lo que este modelo se adapta mejor a este proyecto.

3.4. Estándares de Reportes:

Configuración del Reporte:

Formato: Web

Orientación : Vertical u Horizontal (Dependiendo del tipo de reporte)

Márgenes : Se ocupará toda la ventana, dejando un margen del 10% a cada extremo.

Imágenes : PNG

3.4.1.1 Cabecera

Dato	Posición	Tamaño/ Tipo Letra	Color	Formato	Observación
Fecha – hora reporte	Izquierda superior	Arial 10	Negro	dd/mm/yyyy hh:mm	Fecha y hora de generación del reporte
Fecha actualización	Derecha Superior	Arial 8	Azul	dd/mm/yyyy hh:mm	Fecha y hora de actualización del reporte. Aplicable para reportes que han sido guardados previamente y sobre los que se realiza una nueva consulta.
Logo	Derecha Superior				A la derecha de la fecha de actualización, de haber alguna

3.4.1.2 Pie de Página

El pie de página contendrá los siguientes datos

- Número de página
- Nombre físico del reporte.
- Nombre del hospital.

3.4.1.3 Cuerpo del Reporte

A continuación se detallará el formato estándar de los reportes. Los colores no están incluidos.

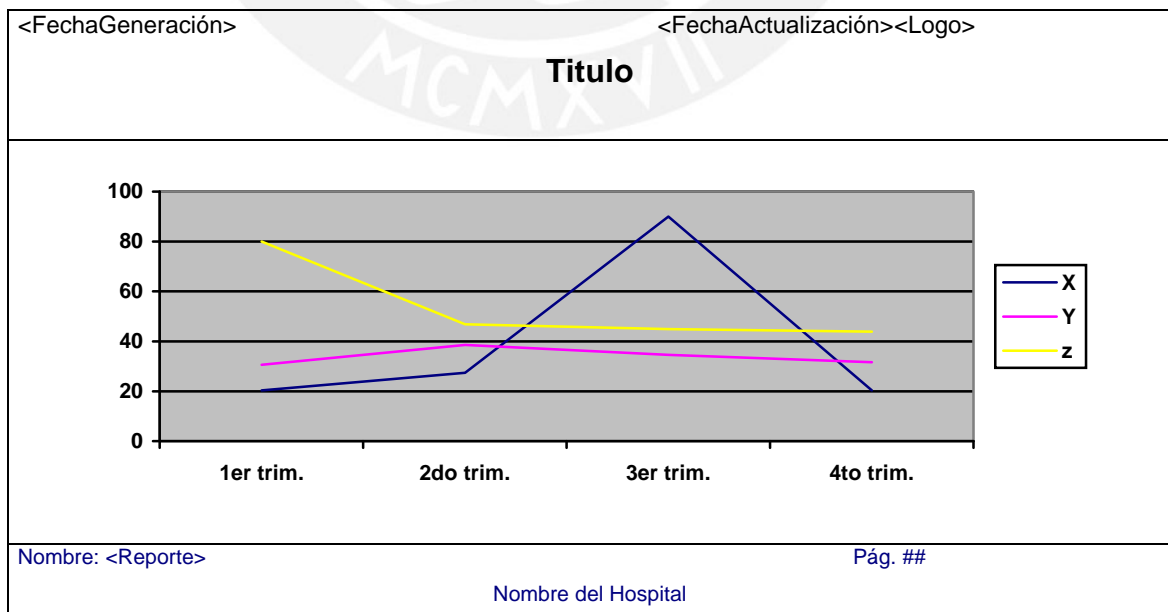
Reporte tipo lista:

<FechaGeneración>		<FechaActualización><Logo>		
Titulo				
XXXXX	0.00	0.00	0.00	0.00
XXXXX	0.00	0.00	0.00	0.00
XXXXX	0.00	0.00	0.00	0.00
XXXXX	0.00	0.00	0.00	0.00
XXXXX	0.00	0.00	0.00	0.00
XXXXX	0.00	0.00	0.00	0.00
TOTAL	0.00	0.00	0.00	0.00

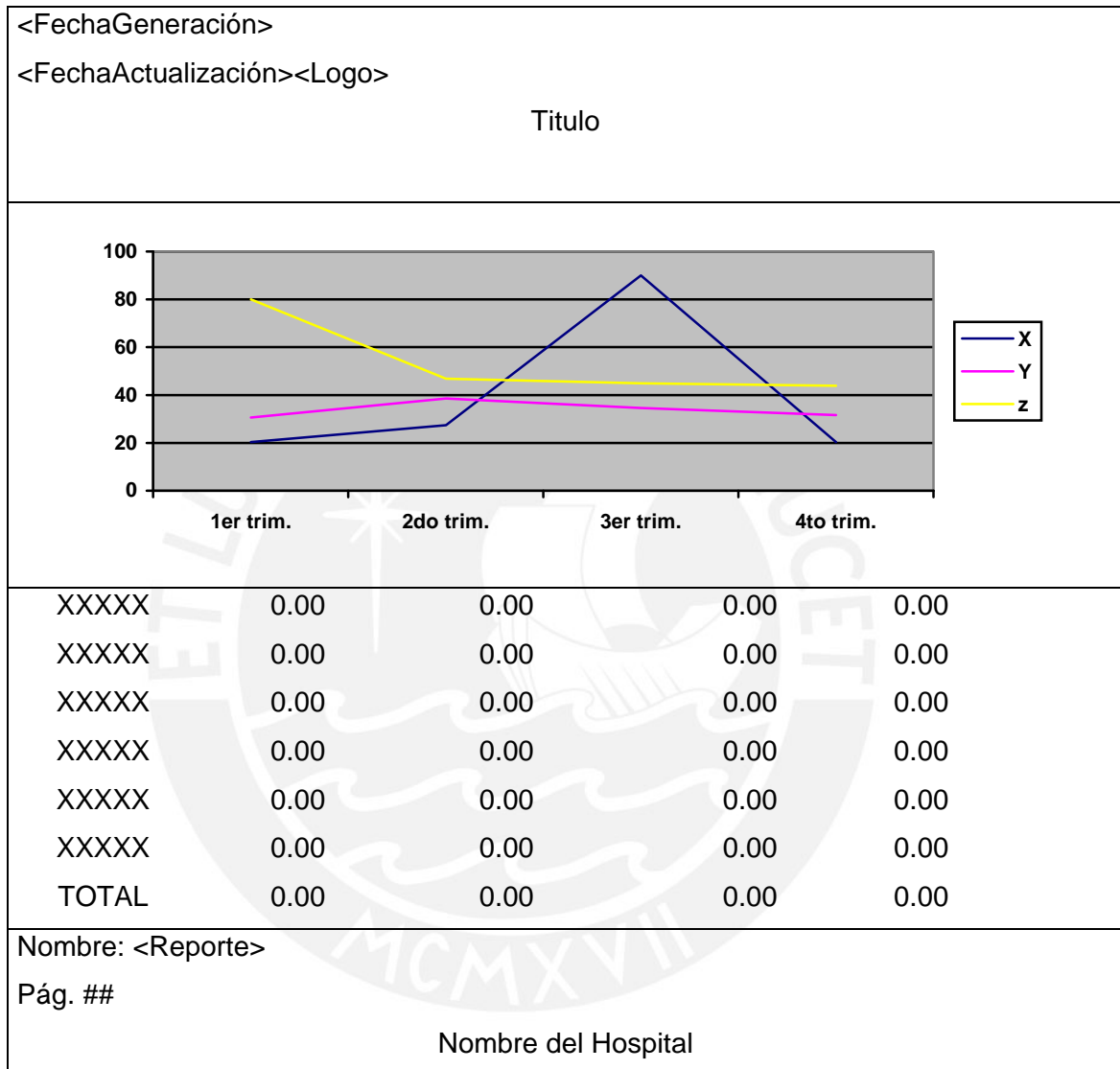
Nombre: <Reporte> Pág. ##

Nombre del Hospital

Reporte tipo gráfico: considere que habrán distintos tipos de gráfico. A continuación sólo se muestra un tipo de gráfico para demostrar los estándares de una ventana con un reporte gráfico.



Reporte Híbrido: Considerar que el reporte lista, no necesariamente aparecerá debajo del gráfico, pudiendo aparecer encima o a al lado derecho o izquierdo. También puede aparecer un listado pero varios gráficos en distintas posiciones.



Es importante mencionar que en los reportes lista, algunas de las celdas poseerán umbrales, es decir, la coloración de la celda dependerá del valor de dicha celda, según un rango que se le especifica al informe.

3.4.1.4 Vistas y Reportes

A continuación se detallan los temas y los reportes a elaborar por tema

Tema	Reporte
Historia clínica	Indicadores en Emergencias.
	Atenciones a pacientes.
	Pacientes intervenidos por servicio médico
Cuentas	Ingresos por cuentas corrientes.
	Deudas e ingresos por cuenta, pacientes y servicio médico.
	Operaciones e ingresos por tipos de tarifas y mes.
Caja	Operaciones de Caja.
	Ingresos por Concepto

Cuadro 1.6 Reportes a elaborar

A continuación se presenta la estructura que seguirán los reportes para cada uno de los temas mencionados. Se mencionará los datos a mostrar, y los filtros y dimensiones a utilizar. Las estructuras son referenciales ya que la forma de presentación de los reportes cambia según las selecciones del usuario.

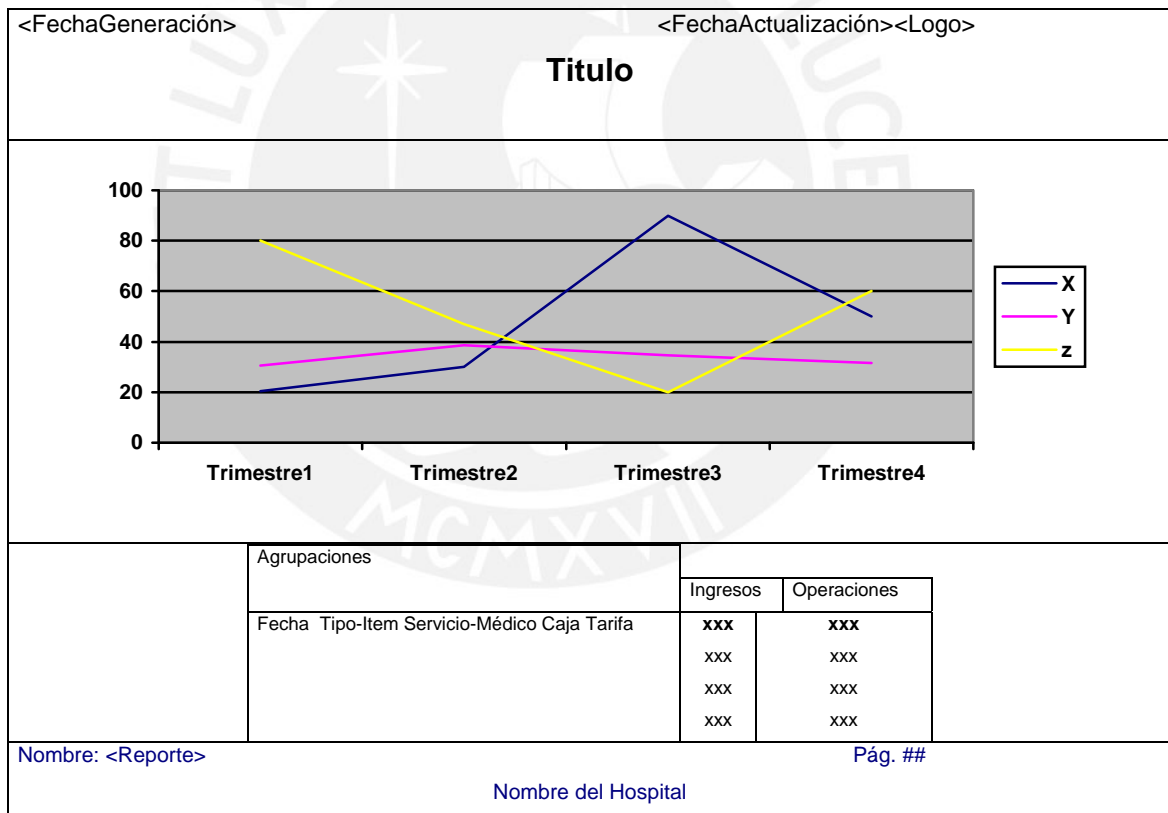
3.4.1.5 Ingresos por Concepto

Muestra los ingresos y operaciones, agrupados por ítem concepto de venta, caja, servicio médico, tipo ítem y fecha.

Este informe permite evaluar los siguientes aspectos:

- Carga operativa de cajas.
- Tarifas que presentan mayor movimiento económico.
- Servicios médicos más rentables.
- Ítems que generan la mayor cantidad de ingresos.
- Evolución de ingresos a través de trimestres.

Diseño:



Filas:

No.	Dimensión	Nivel / Categoría
1	Fecha	Día

Columnas:

No.	Dimensión	Nivel / Categoría
1	Tipo Item	Tipo Item
2	Servicio Medico	Servicio Medico
3	Caja	Caja
4	Tarifa	Tarifa

Medida:

No.	Medida	Formato
1	Ingresos	En soles
2	Operaciones	Número de operaciones

Filtro:

No.	Operación
1	Día = <Días a ingresar>

3.4.1.6 Deudas e ingresos por cuenta, pacientes y servicio médico.

Muestra las deudas que tienen cada cuenta con el hospital, el monto cancelado y el monto pendiente, a un mes y según un servicio médico.

Este informe permite evaluar los siguientes aspectos:

- Servicios médicos que generan más ingresos por cuentas corrientes.
- Servicios médicos donde se adeuda más.
- Cuentas que más adeudan.
- Seguimiento de pago de las cuentas.

Diseño:

<FechaGeneración>		<FechaActualización><Logo>		
Título				
		Monto cancelado	SMonto Pendiente	
	Fecha Cuenta Servicio-Médico	xxx xxx xxx xxx	xxx xxx xxx xxx	
Nombre: <Reporte>		Pág. ##		
Nombre del Hospital				

Filas:

No.	Dimensión	Nivel / Categoría
1	Fecha	Mes

Columnas:

No.	Dimensión	Nivel / Categoría
1	Cuenta	Número de Cuenta
2	Servicio Médico	Servicio Médico

Medida:

No.	Medida	Formato
1	Monto Cancelado	En soles
2	Monto pendiente	En soles

Filtro:

No.	Operación
1	Mes= <Mes a ingresar>

Para ver el diseño y definición de los demás reportes, revise el anexo A: Reportes.

Esquema de Publicación

Nivel	Nombre Carpeta	Nombre Reporte
1	Historia Clínica	Indicadores en Emergencias
		Atenciones a pacientes.
		Pacientes intervenidos por servicio médico
1	Cuentas	Ingresos por cuentas corrientes.
		Deudas e ingresos por cuenta, pacientes y servicio médico.
		Operaciones e ingresos por tipos de tarifas y mes.
1	Caja	Operaciones de Caja
		Ingresos por Concepto

Cuadro 1.7 Esquema de publicación de los reportes

3.5. Diseño de Extracción

A continuación, se presentará el mapeo de cómo se cargarán las tablas destino y de sus tipos de datos, indicando las diversas fuentes.

CAJA_DM

D_CAJA

Columna	Llave	Tipo	Mapeo
ID_D_CAJA	Sí	Numérico	CAJA.ID_CAJA
D_D_CAJA	No	Cadena	CAJA.D_CAJA
CODIGO	No	Cadena	CAJA.CODIGO
ID_D_ZONA_HOSPITAL	No	Numérico	CAJA.ID_ZONA_HOSPITAL

D_TIPO_DOCUMENTO

Columna	Llave	Tipo	Mapeo
ID_D_TIPO_DOCUMENTO	Sí	Numérico	TIPO_DOCUMENTO.ID_TIPO_DOCUMENTO
D_D_TIPO_DOCUMENTO	No	Cadena	TIPO_DOCUMENTO.D_TIPO_DOCUMENTO
CODIGO	No	Cadena	TIPO_DOCUMENTO.CODIGO

F_CARGA_CAJA

Columna	Llave	Tipo	Mapeo
ID_D_DIA	Sí	Numérico	DOCUMENTO_VENTA.FECHA
ID_D_CAJA	Sí	Numérico	D_CAJA.ID_CAJA
ID_D_HORARIO	Sí	Numérico	HORARIOXCAJA.ID_HORARIO
ID_D_TIPO_DOCUMENTO	Sí	Numérico	TIPO_DOCUMENTO.ID_TIPO_DOCUMENTO
N_INGRESOS	No	Numérico	SUM(DOCUMENTO_VENTA.TOTAL)
N_OPERACIONES	No	Numérico	COUNT(DOCUMENTO_VENTA.ID_DOCUMENTO_VENTA)

Para el mapeo de las tablas de las demás vistas, consultar el anexo B: Diseño de la Extracción.

4. Construcción y Pruebas

En este capítulo se muestra la configuración que se tiene que realizar para que las herramientas utilizadas funcionen en forma correcta. Así como también se brinda una descripción gráfica de las pruebas que se han realizado y las salidas esperadas y obtenidas.

4.1. Puesta en producción

Configuración del software

Para empezar la instalación del software necesario, es necesario instalar el PostgreSQL 8.2. Para esto se tiene que crear un nuevo usuario con permisos de administrador y comenzar la instalación en una nueva sesión. La configuración del PostgreSQL se encuentra en el Anexo D: Configuración PostgreSQL.

Construcción de la Metadata

Luego de configurar el PostgreSQL, se procede exportar la metadata. El detalle de la exportación de la metadata se encuentra en el Anexo E: Construcción de la Metadata

Configuración de la herramienta de ETL

Se accede al Spoon mediante el ícono Spoon. Para poder obtener el Pentaho es necesario descargar el paquete completo de la página web: www.pentaho.org.

El Spoon muestra la siguiente pantalla de bienvenida:



Se selecciona la opción de *New* para conectarse al repositorio de datos.

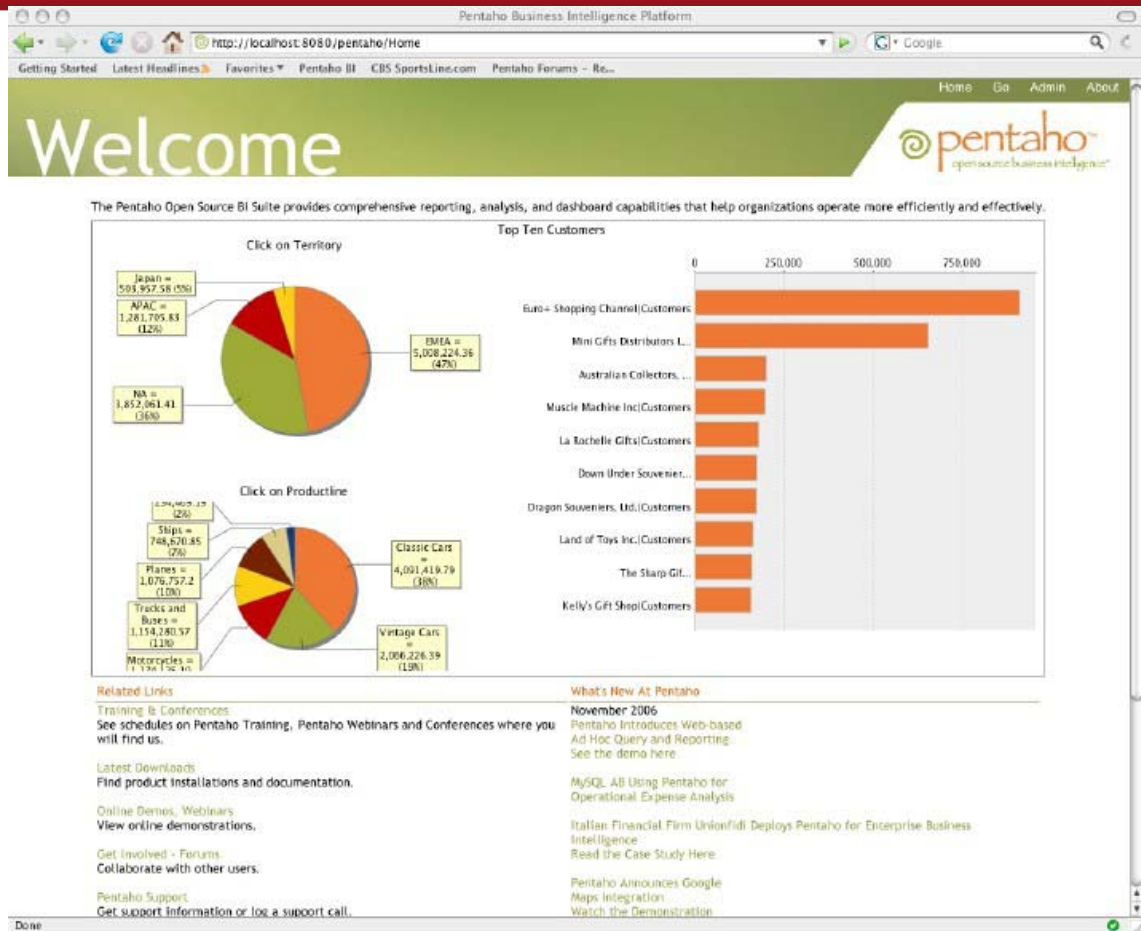
El detalle de la configuración de la herramienta de ETL se encuentra en el Anexo F: Configuración de la herramienta de ETL

Instalación de la plataforma del Pentaho

Pentaho es un proyecto de BI que recopila otros proyectos por separado y los integra para crear una solución BI completa. Así como se instaló el Spoon por separado, a continuación se explica los pasos necesarios para instalar la plataforma Pentaho pre configurada, la cual presenta un buen punto de partida para empezar a trabajar.

De la página *web* de Pentaho, es necesario descargar el archivo `pentaho_demo-x.x.x.x`.

Para inicializar el servidor local simplemente se ejecuta el archivo `start-pentaho.bat` el cual correrá el script necesario para inicializar el servidor. Luego, con la dirección <http://localhost:8080/> desde un *browser*, podremos acceder al servidor:



En esta dirección podremos ver los distintos ejemplos, funcionalidades y características que nos brinda el Pentaho. Sin embargo, los ejemplos que se presentan son de una base de datos HyperSonic preconfigurada, la cual se inicia al iniciar el servidor.

Para configurar la plataforma según nuestras necesidades refiérase al anexo K: Configuración de la plataforma BI.

Instalación del Pentaho Design Studio

El Design Studio facilita la tarea de modificar manualmente los archivos XML o de texto que requiere el Pentaho para su configuración. Además, permite crear Action Scripts, los cuales son la fuente de las acciones que realizará la plataforma Pentaho, como por ejemplo ejecutar informes, mandar e-mails, pedir filtros, entre otros.

El detalle de la instalación del Pentaho Design Studio se encuentra en el Anexo G: Pentaho Design Studio.

Pentaho Report Designer

Este módulo facilita la creación de reportes (JFreeReports) para posteriormente publicarlos en la plataforma Pentaho. Su uso es bastante intuitivo.

El detalle de la instalación del Pentaho Report Designer se encuentra en el Anexo H: Pentaho Report Designer

Pentaho Cube Designer

Este módulo, sirve para crear un fichero de esquema Mondrian (archivo XML), el cual contendrá la definición de las dimensiones, jerarquías, hechos y conexiones a la base de datos. Sin embargo, el Cube Designer aún no posee todas las características avanzadas, como agregados, dimensiones compartidas o cubos múltiples. Para esos casos, se recomienda crear un archivo simple con el Cube Designer, y posteriormente editar el archivo XML a mano para agregar las características deseadas.

El detalle de la instalación del Pentaho Report Designer se encuentra en el Anexo I: Pentaho Cube Designer

Pentaho Workbench

El módulo Workbench sirve como un editor de los cubos en formato XML creados por el cube designer. Permite generar cubos más complejo que los que permite actualmente el cube designer. En el anexo J se encuentra el detalle de la configuración del Workbench.

4.2. Desarrollo de las Pruebas

En esta sección se describe la manera en que se llevan a cabo las distintas pruebas y se presentan los resultados.

Pruebas de Proceso ETL

En esta sección se mostrará la ejecución de los casos de prueba descritos en la sección 2.4. Además se presentará las pruebas para los procesos ETL.

Caso de prueba del informe “Operaciones de caja”: Sólo se muestran los campos y tablas más relevantes, debido a la gran extensión de algunas tablas.

Esta prueba tiene como objetivo verificar que las tablas para cargar el informe mencionado sean correctamente cargadas por medio del ETL, tomando como fuente la vista del Data Warehouse CAJA_DW, y que el informe “Operaciones de

caja” muestre la data correcta.

Datos de entrada:

- Tabla Dia: Sólo se muestra un registro de esta tabla para las pruebas dado a la gran cantidad de datos de la misma.

Id_dia	Fecha	Id_mes	Id_dia_semana
38730	2007-01-14	1273	2

- Tabla Documento_venta: Por motivos de prueba sólo se llenó esta tabla con un total, sin entrar en detalle a la tabla Linea_doc_venta. Sin embargo, esta última tabla estará incluida en el proceso ETL para realizar validación de datos y podría extraerse información adicional a futuro. Sólo se listan las columnas pertinentes.

Id_documento_venta	Id_tipo_documento	total	fecha	Id_caja
1	1	300	2007-01-14	1
2	1	500	2007-01-14	1
3	1	250	2007-01-14	2
4	2	800	2007-01-14	1

Se incluyeron más datos, pero debido a su extensión no se presentarán.

Resultados Intermedios:

A continuación se presenta la data como debería aparecer cargada en las tablas de la vista CAJA_DM, luego de la ejecución del ETL tomando como fuente el Data Warehouse.

- Tabla F_carga_caja:

Id_caja	N_ingresos	N_operaciones	Id_d_tipo_doc	Id_d_dia
1	8000	50	1	38730
1	7000	40	2	38730
1	5000	30	1	38730
1	2000	10	2	38730

2	5000	30	1	38730
2	3000	20	1	38730
3	5000	40	2	38730
3	4000	30	2	38730
3	10000	110	1	38730

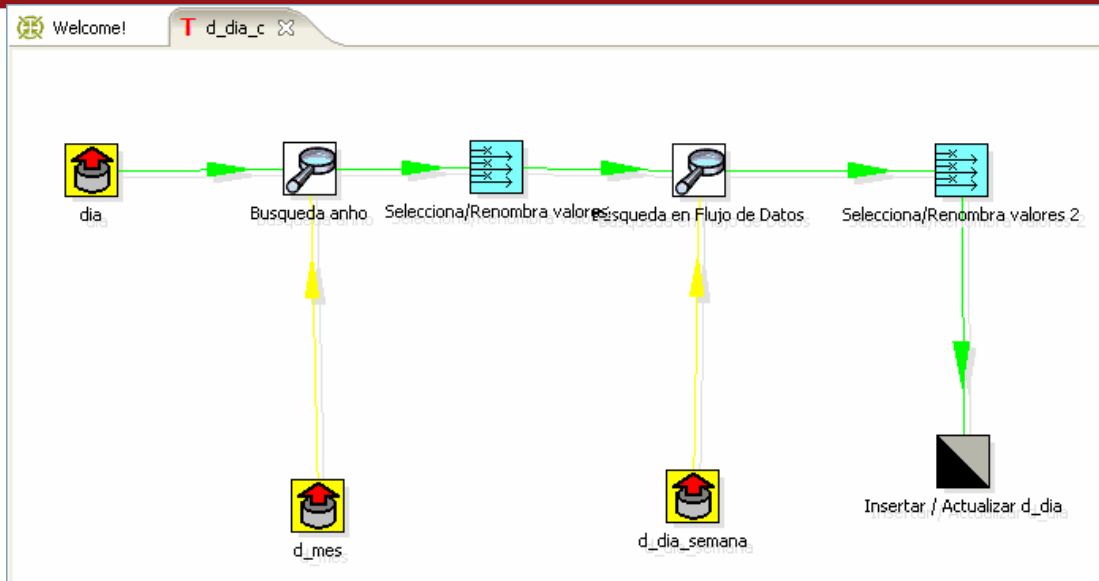
- Tabla D_Caja

Id_caja	D_caja	Id_zona_hospital	D_d_zona_hospital
1	Caja 1	1	Piso11
2	Caja 2	1	Piso11
3	Caja 3	2	Piso12

Resultado Final: A continuación se presenta el reporte que muestra a los ingresos agregados correctamente por cada categoría, al nivel de día:

			Medidas
d_dia	d_caja.d_d_zona_hospital	d_tipo_documento.d_d_tipo_doc	• SUM of f_carga_caja.n_ingresos
-All d_dia.tiempos	+All d_caja.d_d_zona_hospital	+All d_tipo_documento.d_d_tipo_doc	49000,0
-ENERO 2007	+All d_caja.d_d_zona_hospital	+All d_tipo_documento.d_d_tipo_doc	49000,0
2007-01-14	-All d_caja.d_d_zona_hospital	+All d_tipo_documento.d_d_tipo_doc	49000,0
	-Piso11	+All d_tipo_documento.d_d_tipo_doc	30000,0
	Caja 1	-All d_tipo_documento.d_d_tipo_doc	22000,0
		Boleta	13000,0
	Factura	9000,0	
	Caja 2	-All d_tipo_documento.d_d_tipo_doc	8000,0
		Boleta	8000,0
	-Piso12	+All d_tipo_documento.d_d_tipo_doc	19000,0
	Caja 3	-All d_tipo_documento.d_d_tipo_doc	19000,0
		Boleta	10000,0
Factura		9000,0	

Ahora se presenta el proceso de Extracción, Transformación y Carga para la dimensión día, en un solo *Job*, por medio de la herramienta Spoon (Kettle).



En este proceso se extrae Data de la tabla DIA y por medio de lookups se determina el mes y el día de la semana que le corresponde. Se seleccionan los valores deseados y finalmente se carga la tabla D_DIA.

Entradas: No se poseen parámetros como input. Sin embargo, se parte de los contenidos de las tablas día (del Data Warehouse) y d_mes (de la vista GENERAL_DM).

Salida Esperada: La dimensión día totalmente cargada, cumpliendo integridad referencial y con los valores correctos.

Ejecución y pruebas: Al ejecutar el Job, se presenta el siguiente diagrama:

▲	Nombre paso	Numero Copia	Leído	Escrito	Entrada	Salida	Actualizado	Rejected	Errores	Activo
1	d_mes	0	0	1490	1490	0	0	0	0	Finalizado
2	Busqueda anho	0	46783	45293	0	0	0	0	0	Finalizado
3	dia	0	0	45293	45293	0	0	0	0	Finalizado
4	Selecciona/Renombra valores	0	45293	45293	0	0	0	0	0	Finalizado
5	d_dia_semana	0	0	9	9	0	0	0	0	Finalizado
6	Búsqueda en Flujo de Datos	0	45302	45293	0	0	0	0	0	Finalizado
7	Selecciona/Renombra valores 2	0	45293	45293	0	0	0	0	0	Finalizado
8	Insertar / Actualizar d_dia	0	45293	0	45293	0	5074	0	0	Finalizado

```

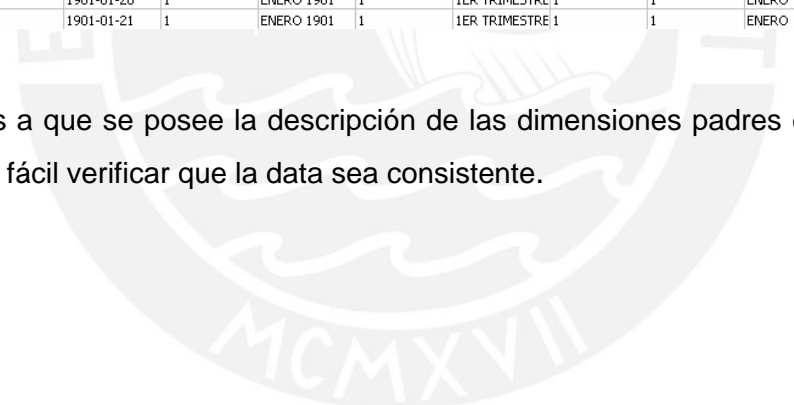
2007/10/28 12:36:11 - dia.0 - linetr 40000
2007/10/28 12:36:11 - Insertar / Actualizar d_dia.0 - NumLinea 35000
2007/10/28 12:36:14 - Selecciona/Renombra valores.0 - NumLinea 40000
2007/10/28 12:36:15 - Búsqueda en Flujo de Datos.0 - Número de línea 40000
2007/10/28 12:36:16 - Selecciona/Renombra valores 2.0 - NumLinea 40000
2007/10/28 12:36:16 - Busqueda anho.0 - Número de línea 45000
2007/10/28 12:36:17 - Insertar / Actualizar d_dia.0 - NumLinea 40000
2007/10/28 12:36:17 - dia.0 - linetr 45000
2007/10/28 12:36:17 - dia.0 - Finished reading query, closing connection.
2007/10/28 12:36:17 - dia.0 - Procesamiento finalizado (I=45293, O=0, R=0, W=45293, U=0, E=0
2007/10/28 12:36:19 - Busqueda anho.0 - Procesamiento finalizado (I=0, O=0, R=46783, W=45293, U=0, E=0
2007/10/28 12:36:21 - Selecciona/Renombra valores.0 - NumLinea 45000
2007/10/28 12:36:21 - Selecciona/Renombra valores.0 - Procesamiento finalizado (I=0, O=0, R=45293, W=45293, U=0, E=0
2007/10/28 12:36:22 - Búsqueda en Flujo de Datos.0 - Número de línea 45000
2007/10/28 12:36:23 - Búsqueda en Flujo de Datos.0 - Procesamiento finalizado (I=0, O=0, R=45302, W=45293, U=0, E=0
    
```

Notar que la columna Errores contiene todas sus entradas con el valor de 0. Esto indica que la ejecución del proceso no tuvo errores. Si se visualiza el *log* presentado en la sección inferior, se notará que no existe ninguna entrada que indique la presencia de algún error.

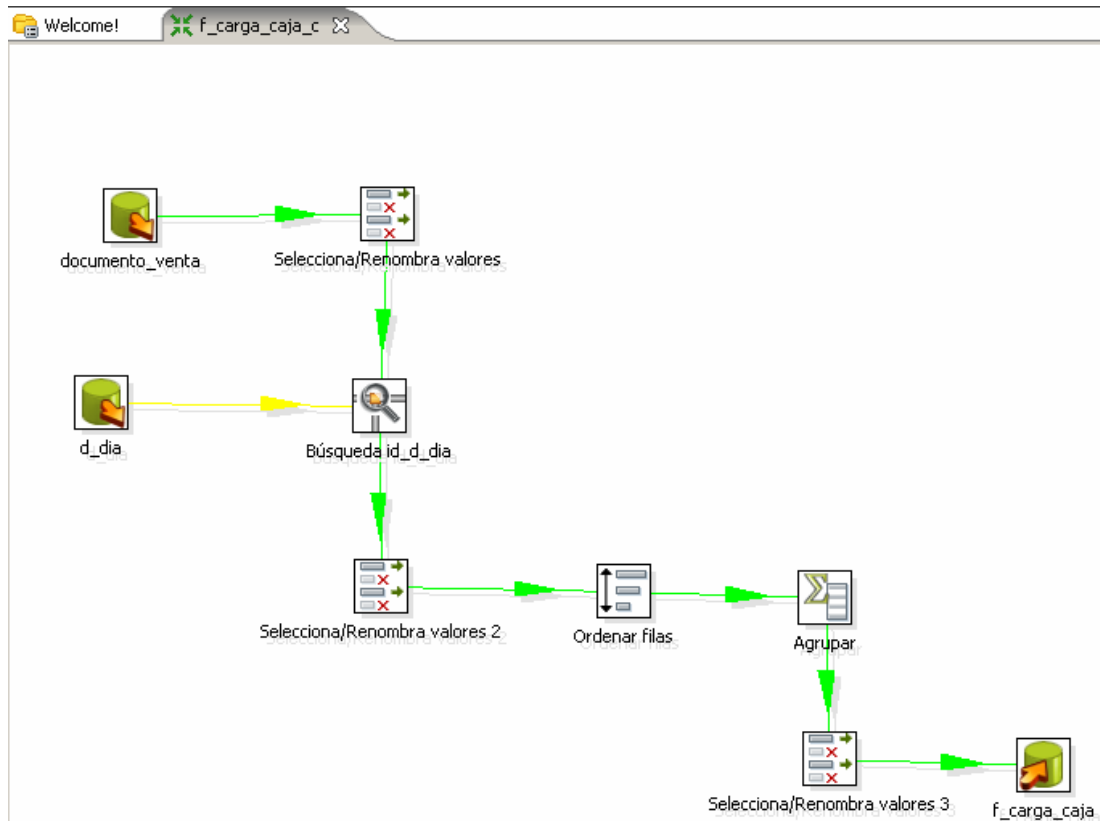
Para verificar si los datos han sido cargados/actualizados correctamente, se verifica por la base de datos que los datos estén presentes y los *constraints* habilitados (lo que significa que se respetó la integridad referencial).

	id_d_dia [PK] integer	fecha date	id_d_mes integer	d_d_mes character var	id_d_trimestr integer	d_d_trimestr character var	n_d_trimestr integer	id_mesanho integer	d_d_mesanho character(18)	n_d_mesanho integer	id_d_dia integer
1	-2		-2	NO APLICA	-2	NO APLICA		-2	NO APLICA	-2	-2
2	-1		-1	SIN DATO	-1	SIN DATO		-1	SIN DATO	-1	-1
3	1	1901-01-01	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	4
4	2	1901-01-02	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	5
5	3	1901-01-03	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	6
6	4	1901-01-04	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	7
7	5	1901-01-05	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	1
8	6	1901-01-06	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	2
9	7	1901-01-07	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	3
10	8	1901-01-08	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	4
11	9	1901-01-09	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	5
12	10	1901-01-10	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	6
13	11	1901-01-11	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	7
14	12	1901-01-12	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	1
15	13	1901-01-13	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	2
16	14	1901-01-14	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	3
17	15	1901-01-15	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	4
18	16	1901-01-16	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	5
19	17	1901-01-17	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	6
20	18	1901-01-18	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	7
21	19	1901-01-19	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	1
22	20	1901-01-20	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	2
23	21	1901-01-21	1	ENERO 1901	1	1ER TRIMESTRE	1	1	ENERO	1	3

Gracias a que se posee la descripción de las dimensiones padres en la dimensión hija, es fácil verificar que la data sea consistente.



El siguiente proceso ETL muestra la carga de la Fact f_carga_caja:



En este proceso se extrae de las tablas documento de venta y día, para posteriormente agrupar los datos según las llaves de la Fact f_carga_caja y finalmente cargar los datos a esta Fact.

Entradas: No se poseen parámetros como input. Sin embargo, se parte de los contenidos de las tablas día (del Data Data Mart) y documento_venta (de la vista CAJA_DW).

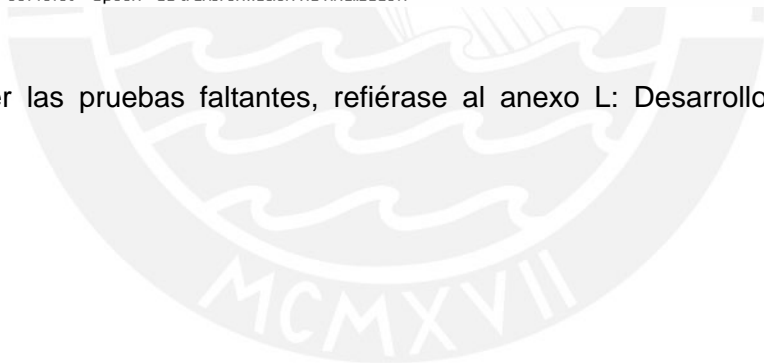
Salida Esperada: La fact f_carga_caja totalmente cargada, cumpliendo integridad referencial y con los valores correctos.

Ejecución y pruebas: Al ejecutar el Job, se presenta el siguiente diagrama, que muestra que la ejecución no presentó errores. Se puede verificar si fue cargada correctamente por sentencias SQL, similar al job anterior.

#	Nombre paso	Numero Copia	Leído	Escrito	Entrada	Salida	Actualizado
1	d_dia	0	0	45293	45293	0	0
2	Búsqueda id_d_dia	0	45305	12	0	0	0
3	documento_venta	0	0	12	12	0	0
4	Selecciona/Renombrar valores	0	12	12	0	0	0
5	Selecciona/Renombrar valores 2	0	12	12	0	0	0
6	Ordenar filas	0	12	12	0	0	0
7	Agrupar	0	12	11	0	0	0
8	Selecciona/Renombrar valores 3	0	11	11	0	0	0
9	f_carga_caja	0	11	11	0	11	0

2008/09/27 10:40:34 - documento_venta.0 - Iniciando ejecución...
 2008/09/27 10:40:34 - Selecciona/Renombrar valores.0 - Iniciando ejecución...
 2008/09/27 10:40:34 - Selecciona/Renombrar valores 2.0 - Iniciando ejecución...
 2008/09/27 10:40:34 - f_carga_caja.0 - Iniciando ejecución...
 2008/09/27 10:40:34 - Ordenar filas.0 - Iniciando ejecución...
 2008/09/27 10:40:34 - Selecciona/Renombrar valores 3.0 - Iniciando ejecución...
 2008/09/27 10:40:34 - d_dia.0 - Iniciando ejecución...
 2008/09/27 10:40:34 - Búsqueda id_d_dia.0 - Leyendo valores de búsqueda del paso [d_dia]
 2008/09/27 10:40:34 - documento_venta.0 - Finished reading query, closing connection.
 2008/09/27 10:40:34 - documento_venta.0 - Procesamiento finalizado (I=12, O=0, R=0, W=12, U=0, E=0
 2008/09/27 10:40:34 - Selecciona/Renombrar valores.0 - Procesamiento finalizado (I=0, O=0, R=12, W=12, U=0, E=0
 2008/09/27 10:40:38 - d_dia.0 - Finished reading query, closing connection.
 2008/09/27 10:40:38 - d_dia.0 - Procesamiento finalizado (I=45293, O=0, R=0, W=45293, U=0, E=0
 2008/09/27 10:40:38 - Búsqueda id_d_dia.0 - Procesamiento finalizado (I=0, O=0, R=45305, W=12, U=0, E=0
 2008/09/27 10:40:38 - Selecciona/Renombrar valores 2.0 - Procesamiento finalizado (I=0, O=0, R=12, W=12, U=0, E=0
 2008/09/27 10:40:38 - Ordenar filas.0 - Procesamiento finalizado (I=0, O=0, R=12, W=12, U=0, E=0
 2008/09/27 10:40:39 - Agrupar.0 - Procesamiento finalizado (I=0, O=0, R=12, W=11, U=0, E=0
 2008/09/27 10:40:39 - Selecciona/Renombrar valores 3.0 - Procesamiento finalizado (I=0, O=0, R=11, W=11, U=0, E=0
 2008/09/27 10:40:39 - f_carga_caja.0 - Procesamiento finalizado (I=0, O=11, R=11, W=11, U=0, E=0
 2008/09/27 10:40:39 - Spoon - La transformación ha finalizado!!

Para ver las pruebas faltantes, refiérase al anexo L: Desarrollo de las Pruebas



5. Observaciones, Conclusiones y Recomendaciones

La siguiente sección presenta los puntos considerados como observaciones, conclusiones y recomendaciones recolectados a lo largo del desarrollo de la tesis. Las conclusiones se refieren al desarrollo mismo de la tesis, mientras que las recomendaciones y observaciones están orientadas para aquellas personas que deseen implementar la tesis desarrollada o utilizar las herramientas que se usaron durante el desarrollo de la tesis.

5.1. Observaciones

- La utilización de *software* libre para los procesos de ETL y de explotación de los cubos ha hecho posible que el presente trabajo de tesis sea viable, ya que las herramientas para el ETL y para la explotación de los reportes son muy costosas actualmente, reduciendo los costos de implementación, y siendo una alternativa real para los hospitales del sector público.
- Las desventajas de la utilización de herramientas libres están en la falta de soporte, documentación y pruebas a las que han sido sometidas. A pesar de esto, dichas herramientas están en un proceso continuo de crecimiento, ofreciendo futuras posibilidades de mejora.
- El DataWarehouse diseñado puede ser usado como base para la construcción de otro para un hospital o una clínica, pero debe de ser adaptado a las necesidades de cada entidad. Asimismo, los Data Marts han

sido diseñados pensando en los requerimientos del Hospital Santa Rosa.

- El DataWarehouse puede ser ampliado en un futuro de acuerdo de nuevos requerimientos, habiéndose incluido áreas que no se han explotado en este trabajo de tesis, pero dando la oportunidad para explotarse en un futuro.

5.2. Conclusiones

- El trabajo de tesis presenta una solución que los hospitales pueden implementar para satisfacer sus necesidades de gestión, análisis y toma de decisiones. Otorga un panorama de lo que está sucediendo en el hospital y presenta esta información en línea.
- Los reportes finales no están limitados a presentar la información calculada en el trabajo de tesis. Si un hospital posee información adicional que desearía presentar en su plataforma de Inteligencia de Negocios, es posible agregar esta información al Data Warehouse para satisfacer esta necesidad. Probablemente sería necesario crear nuevas tablas y agregar nuevos procesos al ETL. El presente trabajo de tesis deja abierta esta posibilidad. Sólo se presentan los procesos básicos que posee un hospital, no se trabaja con procesos adicionales que pudiera contener un hospital en particular.
- La creación de un Data Warehouse previa a el desarrollo de los Data Marts, según la arquitectura planteada por Inmon, ayuda a que el hospital tenga toda su información consolidada y ordenada en un solo lugar, lo cual es muy importante en este tipo de organizaciones debido a la sensibilidad e importancia de la información, y brinda coherencia entre todos los Data Marts, pues estos partirían desde una misma fuente de información.
- Tener todos los datos consistentes y ordenados en el Data Warehouse brinda una fuente confiable y estandarizada para el desarrollo de futuros Data Marts o para la ampliación del alcance de los existentes, facilitando el desarrollo de estos.
- Es muy importante desarrollar una buena fase de análisis para evitar que a lo largo del proyecto surjan problemas que ameriten una reestructuración de los procesos, mapeos o de los reportes mismos. Algunos inconvenientes no saltan a la vista hasta que se tiene el reporte terminado, puesto que saltan incongruencias en los datos del informe o se identifica que los datos no eran agregables y por lo tanto se está presentando información incorrecta. En estos casos, se debe regresar a los procesos anteriores para resolver el problema.

- Un proceso puede estar muy bien definido y el reporte bien estructurado, sin embargo, en algunos casos intervienen otros factores que pueden interferir en el proceso, como: inmenso volumen de datos, datos “sucios” e incoherentes, procesos lentos cuando la información presenta una característica en particular, entre otros. Estos casos no fueron identificados sino hasta una vez terminado el proceso y durante la fase de pruebas.
- El software libre utilizado en este proyecto permite implementar todo lo necesario para llevar a producción una plataforma de Inteligencia de negocios y el desarrollo de los reportes y procesos ETL. Sin embargo, los softwares libres carecen algunas veces de facilidades de uso y funcionalidades, las cuales se deben reemplazar con mayores tiempos de desarrollo, investigación y un mayor uso de ingenio. El software libre no termina siendo completamente superior al licenciado. En este trabajo de tesis, se usó el software libre debido a su principal ventaja: el costo económico. Las principales ventajas del software licenciado sobre el libre son: facilidad de uso y soporte técnico.
- En este tema de tesis se ha planteado la elaboración de un Data Warehouse cuyas características se ajustan a las necesidades básicas de un hospital. Se ha implementado una plataforma BI que explote el Data Warehouse en múltiples cubos y muestre su información a las autoridades pertinentes. Se ha desarrollado todos los pasos para llevar a los informes finales que ayudaran a los usuarios en la toma de decisiones. Teniendo todos estos pasos descritos y probados, el trabajo de tesis se puede presentar e implantar directamente en cualquier hospital, aún cuando parte del levantamiento de información fue con el hospital Santa Rosa. Este trabajo de tesis presenta una propuesta sobre los informes básicos que debería tener todo hospital y la herramienta libre permite que este al alcance de cualquier entidad de salud. Sin embargo, como cada entidad de salud presenta particularidades que las hacen distintas de otras, puede que los procesos de ETL deban ser cambiados según la entidad para la que se implementa. Pero el Data Warehouse no debería ser modificado pues alberga los datos básicos que una

entidad de salud debería manejar. Se podría modificar el Data Warehouse para agregar datos particulares que las entidades quieran mostrar en sus informes finales, pero se sigue cumpliendo el objetivo de la tesis al presentar un proyecto completo de Inteligencia de Negocios totalmente implementable y que presenta información relevante para cualquier hospital.

5.3. Recomendaciones

- Se recomienda documentarse bien en el uso de las herramientas y realizar pruebas antes de iniciar el uso de “producción” de estas. Se puede conocer muy bien el proceso a desarrollar, pero si las herramientas no son utilizadas de la manera correcta entonces llevará al fracaso del producto final.
- Identificar los datos no agregables, datos “sucios”, procesos lentos, optimizaciones posibles y tiempos aceptables de ejecución para los procesos. Si estos son identificados durante la fase de análisis entonces esto conllevará a un gran ahorro de tiempo y recursos en el proyecto.
- Investigar acerca de todas las posibles optimizaciones para los distintos casos en particular que se presentan, antes que se inicie la fase de implementación. Un ejemplo de esto puede ser utilizar el comando “copy” del sistema operativo para concatenar dos archivos muy grandes, en vez de utilizar el Stage Unión del Spoon, pues para archivos grandes este proceso puede tardar mucho y consumir demasiados recursos. Identificar las optimizaciones con anterioridad ahorra tiempo de desarrollo.
- Revisar varios reportes que hagan referencia al mismo Data Mart. Los datos deben ser congruentes entre ellos e idénticos para los casos en que muestran lo mismo, dado los filtros.
- Algunos procesos, tanto en el ETL como la construcción de reportes, presentan casos particulares que son muy difíciles de solucionar y en los cuales la documentación no presenta una solución. Para tratar estos problemas, resulta muy útil consultar los foros de las herramientas mismas u otras páginas de Internet dedicadas a resolver consultas de este tipo. Resulta frecuente estas situaciones, y al ser una tecnología relativamente nueva, se crean foros de ayuda comunitaria en donde los usuarios comparten sus diversas experiencias. Se puede considerar a estos lugares como una fuente muy útil de información.

6. Bibliografía

[BEN 2005] Huynen MMTE, Vollebregt L, Martens P, Benavides BM. The epidemiologic transition in Peru. Rev Panam Salud Publica. 2005. Pan American Health Organization. http://journal.paho.org/index.php?a_ID=252
Último acceso: septiembre 2008

[IPE 2004] Perú, Instituto Peruano de Economía. Lima: IPE; 2004. www.ipe.org.pe/publicaciones. Último acceso: septiembre 2008.

[INEI 2006] Perú, Instituto Nacional de Estadística e Informática. Lima: INEI; 2006. <http://www.inei.gob.pe/> Último acceso: diciembre 2006

[DUR 2005] Durán Valverde Fabio. Estudio financiero-actuarial y de la gestión de EsSalud: análisis y recomendaciones técnicas. Lima: 2005. http://www.oitandina.org.pe/documentos/informe_actuarial_essalud_oit_25_may_05_final.pdf Último acceso: agosto 2007

[GUT 2001] Alejandro Gutierrez, Regina Motz, Beatriz revello, Lydia Silvia. Construcción de un sistema de apoyo a la toma de decisiones para el área gerencial del Hospital de Clínicas. Instituto de Computación, Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay. 2001. http://www.fing.edu.uy/inco/grupos/csi/esp/Proyectos/csic2000/pub/pub2_sist_dwhc.pdf

[RON 2004] Ronald J Lagoe. Improving Outcomes With Community-wide Distribution of Health Care Data. Health Care Management Review. 2004. <http://www.hcmrjournal.com>. Ultimo acceso: agosto 2007

[YAN 2003] Ma, Yanqiang Allen, Ph.D . Determinants of hospital information system (HIS) integrity and hospital performance. Virginia Commonwealth University, 2003. <http://had.vcu.edu>. Ultimo acceso: diciembre 2006

[CAT 2004] Catibusic S, Hadzagic-Catibusic F, Zubcevic S. Medical data warehousing as a generator of system component for decision support in health care – 2004.

http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?cmd=Retrieve&db=PubMed&list_uids=15137228&dopt=Abstract. Ultimo acceso: agosto 2007

[DAT 2007] DataStage Web Page

<http://www-304.ibm.com/jct03002c/software/data/integration/datastage/>

Ultimo acceso: agosto 2007

[SQL 2007] SQL Server Integration Services (SSIS) Web Page

<http://www.microsoft.com/sql/technologies/integration/default.aspx>

Ultimo acceso: agosto 2007

[SUN 2007] Sunopsis Web Page

<http://www.sunopsis.com/corporate/index.htm>

Ultimo acceso: agosto 2007

[MIC 2007] MicroStrategy Business Intelligence Solutions

<http://www.microstrategy.com>

Ultimo acceso: agosto 2007

[COG 2007] Cognos 8 Business Intelligence

<http://www.cognos.com>

Ultimo acceso: agosto 2007

[BUS 2007] BussinessObjects

<http://www.businessobjects.com>

Ultimo acceso: agosto 2007

[PEN 2007] Pentaho

<http://www.pentaho.com>

Ultimo acceso: agosto 2007

[OCT 2007] Enhydra Octopus

<http://www.enhydra.org/tech/octopus/index.html>

Ultimo acceso: agosto 2007

[IBM 1999] IBM - Fundamentals of Data Warehouse and Business Intelligence for Knowledge Management – Instructor Guide. Diciembre 1999. Página 48

[VITT 2002] Vitt Elizabeth, Luckevich Michael, Misner Stacia. Business Intelligence: Técnicas de análisis para la toma de decisiones estratégicas. 2002.

[MOS 2003] Moss Larissa, Atre Shaku. Business Intelligence Roadmap. The Complete Project Lifecycle for Decisión-Support Applications. 2003 EE.UU.

[POS 2007] PostGreSQL. Global Development Group 2007. <http://www.postgresql.org/> Ultimo acceso: agosto 2007

[PEC 2007] Daniel Pecos. <http://www.netpecos.org/docs/> Ultimo acceso:
agosto 2007



PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA



PONTIFICIA
**UNIVERSIDAD
CATÓLICA**
DEL PERÚ

**Análisis, Diseño e Implementación de un
DataWarehouse de Soporte de Decisiones para un
Hospital del Sistema de Salud Público**

Tesis para optar por el Título de Ingeniero Informático, que presenta el
bachiller

Anexo

Álvaro Villanueva Ojeda

Asesor: Ing. Carla Basurto

LIMA – PERÚ

2008

ANEXO A: Reportes	3
1.1. Operaciones e ingresos por tipos de tarifas y mes	3
1.2. Operaciones de Caja	4
1.3. Pacientes intervenidos por servicio médico	6
1.4. Ingresos por cuentas corrientes	7
1.5. Atenciones a pacientes	8
1.6. Indicadores en Emergencias	10
ANEXO B: Diseño de Extracción	12
ANEXO C: Vistas del Data Warehouse	28
Anexo D: Configuración PostGresSQL	36
Anexo E: Construcción de la Metadata	40
Anexo F: Configuración de la herramienta de ETL	42
Anexo G: Pentaho Design Studio	44
Anexo H: Pentaho Report Designer	45
Anexo I: Pentaho Cube Designer	46
Anexo J: Pentaho Workbench	48
Anexo K: Configuración de la plataforma BI	49
Anexo L: Desarrollo de las Pruebas	51

ANEXO A: Reportes

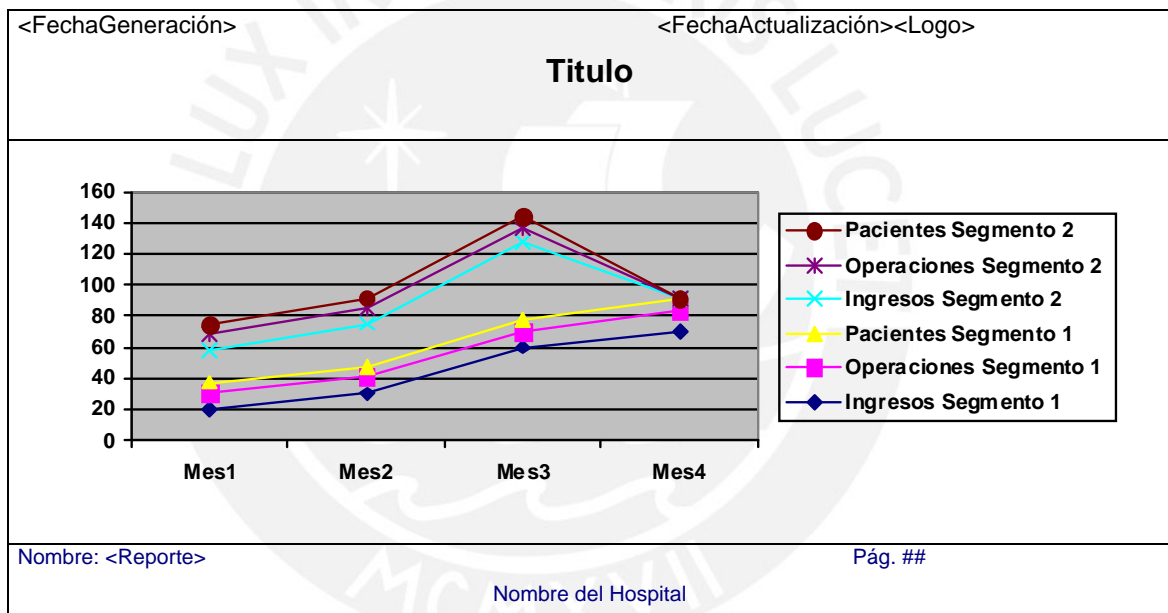
1.1. Operaciones e ingresos por tipos de tarifas y mes

Muestra los ingresos por tarifa, las operaciones por tarifa y los pacientes pertenecientes a cada tarifa, agrupados por segmento de la tarifa.

Este informe permite evaluar los siguientes aspectos:

- Número de pacientes por segmento y tarifa.
- Ingresos por segmento.
- Operaciones por segmento.

Diseño:



Filas:

No.	Dimensión	Nivel / Categoría
1	Segmento	Segmento

Columnas:

No.	Dimensión	Nivel / Categoría
1	Precio	Precio

Medida:

No.	Medida	Formato
1	Precio	En soles

Filtro:

No.	Operación
1	MEs= <Mes a ingresar>

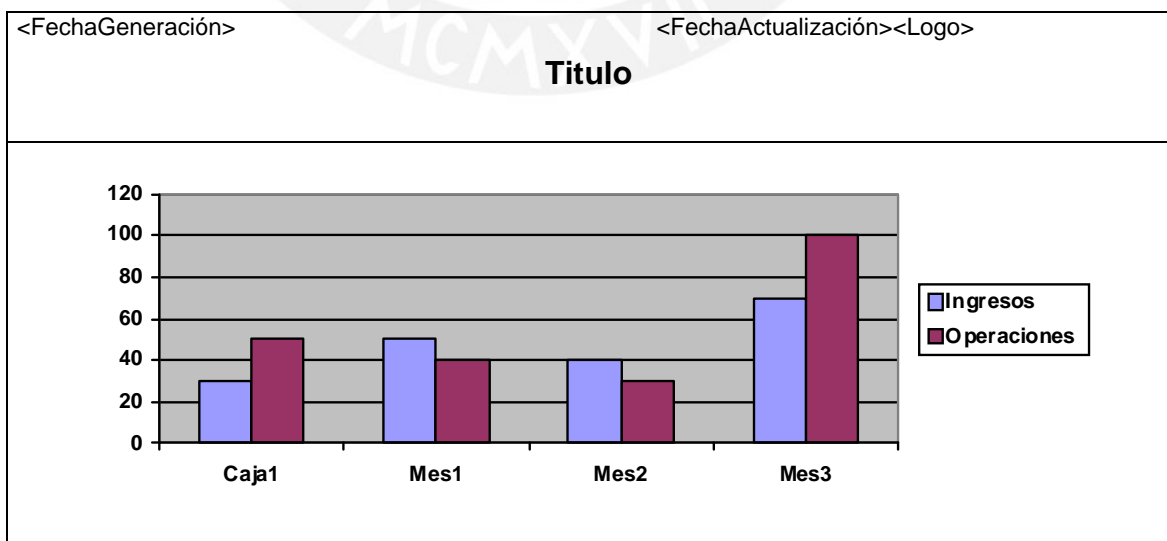
1.2. Operaciones de Caja

Muestra el número de operaciones y los ingresos generados por dichas operaciones por zona del hospital, tipo de documento, mes, horario y tipo tarifa.

Este informe permite evaluar los siguientes aspectos:

- Tipo documentos, Zonas del hospital y Horarios con más ingresos y movimientos.
- Evolución de estos aspectos por mes.

Diseño:



		Ingresos	Operaciones	
	Zona Hospital Tipo-Doc Fecha Horario Tipo-Tarifa	Xxx	xxx	
		Xxx	xxx	
		Xxx	xxx	
		xxx	xxx	
Nombre: <Reporte>		Pág. ##		
Nombre del Hospital				

Filas:

No.	Dimensión	Nivel / Categoría
2	Zona Hospital	Zona Hospital

Columnas:

No.	Dimensión	Nivel / Categoría
1	Tipo Documento	Tipo Documento
2	Fecha	Día
3	Horario	Horario
4	Tipo Tarifa	Tipo Tarifa

Medida:

No.	Medida	Formato
1	Ingresos	En miles de soles
2	Operaciones	Numérico Natural

Filtro:

No.	Operación
1	Día= <Día a ingresar>

1.3. Pacientes intervenidos por servicio médico

Muestra la cantidad de fallecimientos, la cantidad de intervenciones resultadas en hospitalización y la cantidad de pacientes intervenidos, por día, servicio medico, horario, personal y tipo de intervención.

Este informe permite evaluar los siguientes aspectos:

- Cantidad de pacientes intervenidos por servicio médico, horario, personal y tipo de intervención.
- Personal que presenta mayor ratio de fallecimientos en sus intervenciones.
- Número de intervenciones que resultaron en hospitalizaciones y fallecimientos.

Diseño:

<FechaGeneración>		<FechaActualización><Logo>	
Título			
		Pacientes hospitalizaciones Fallecimientos Ratio-Fallecimiento	
Fecha	Servicio-Médico	Horario	
Tipo-Intervencion		Personal	
			Xxxxxx xxxxxxxx xxxxxxxx xxxxxxxxxxxxxxxx
			Xxxxxx xxxxxxxx xxxxxxxx xxxxxxxxxxxxxxxx
			Xxxxxx xxxxxxxx xxxxxxxx xxxxxxxxxxxxxxxx
Nombre: <Reporte>		Pág. ##	
Nombre del Hospital			

Filas:

No.	Dimensión	Nivel / Categoría
1	Fecha	Día

Columnas:

No.	Dimensión	Nivel / Categoría
1	Servicio médico	Servicio médico
2	Horario	Horario
3	Personal	Personal
4	Tipo Intervención	Tipo Intervención

Medida:

No.	Medida	Formato
1	Pacientes Atendidos	Número de pacientes
2	Pacientes fallecido	Número de fallecimientos
3	Pacientes hospitalizados	Número de hospitalizaciones

Filtro:

No.	Operación
1	Mes= <Mes a ingresar>

1.4. Ingresos por cuentas corrientes

Muestra los ingresos y operaciones por cuentas, agrupados por ítem concepto de venta.

Este informe permite evaluar los siguientes aspectos:

- Tipo tarifas, ítems y servicios médicos que presentan mayor número de ingresos y operaciones dentro de las cuentas corrientes.

Diseño:

<FechaGeneración>		<FechaActualización><Logo>	
Titulo			
		Ingresos	Operaciones
	Fecha Tipo-Tarifa Item-Concepto Servicio-médico	Xxx	xxx
		Xxx	xxx
		Xxx	xxx
		xxx	xxx
Nombre: <Reporte>		Pág. ##	
Nombre del Hospital			

Filas:

No.	Dimensión	Nivel / Categoría
1	Tiempo	Mes

Columnas:

No.	Dimensión	Nivel / Categoría
1	Tipo tarifa	Tipo Tarifa
2	Item concepto	Item concepto
3	Servicio Médico	Servicio Médico

Medida:

No.	Medida	Formato
1	Ingresos	En miles de soles
2	Operaciones	Numérico Natural

Filtro:

No.	Operación
1	Mes= <Mes a ingresar>

1.5. Atenciones a pacientes

Muestra la cantidad de pacientes, el tiempo de espera promedio por paciente, el tiempo de atención promedio por paciente y el monto generado por las consultas medicas, por mes, servicio medico, horario y personal que atiende la consulta.

Este informe permite evaluar los siguientes aspectos:

- Personal con mayor número de pacientes y tiempos de espera y atención.
- Servicios médicos que reciben la mayor cantidad de pacientes e ingresos.
- Servicios médicos con mayores tiempos de espera y atención.
- Horarios con mayores tiempos de espera y atención.

Diseño:

<FechaGeneración>		<FechaActualización><Logo>			
Título					
		Atencion y Espera Promedio	Monto	Pacientes	
	Fecha Horario Personal Servicio-Médico	Xxx	xxx	Xxx	
		Xxx	xxx	Xxx	
		Xxx	xxx	Xxx	
		xxx	xxx	xxx	
Nombre: <Reporte>			Pág. ##		
Nombre del Hospital					

Filas:

No.	Dimensión	Nivel / Categoría
1	Tiempo	Día

Columnas:

No.	Dimensión	Nivel / Categoría
1	Horario	Horario
2	Personal	Personal
3	Servicio Médico	Servicio Médico

Medida:

No.	Medida	Formato
1	Pacientes Atendidos	Número de pacientes
2	Minutos espera promedio	Minutos
3	Minutos atención promedio	Minutos
4	Ingresos	Soles

Filtro:

No.	Operación
1	dia= <día a ingresar>

1.6. Indicadores en Emergencias

Muestra los minutos de atención promedio de espera y atención para los pacientes recibidos en el área de emergencias, la cantidad de pacientes recibidos, la cantidad de fallecimientos y la cantidad de personal disponible, por mes, horario y servicio médico.

Este informe permite evaluar los siguientes aspectos:

- Tiempos de espera y atención en el área de emergencias.
- Personal disponible por paciente en el área de emergencias.
- Horarios y servicios médicos que presentan más pacientes, fallecimientos y tiempos de espera y atención en el área de emergencias.

Diseño:

<FechaGeneración>		<FechaActualización><Logo>	
Titulo			
		Atención-Prom	Espera-Prom
		Fallecimientos	Personal
		Pacientes	PersonalxPaciente
Tiempo	Horario	Servicio-médico	
			Xxxxxx xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx
			Xxxxxx xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx
			Xxxxxx xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx
			Xxxxxx xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx
Nombre: <Reporte>		Pág. ##	
Nombre del Hospital			

Filas:

No.	Dimensión	Nivel / Categoría
1	Tiempo	Día

Columnas:

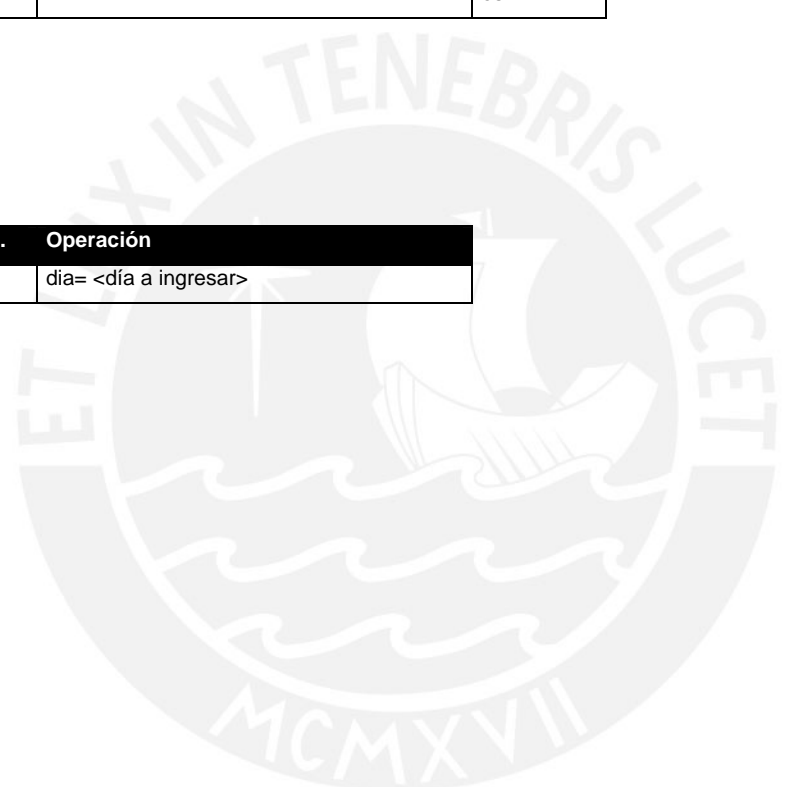
No.	Dimensión	Nivel / Categoría
1	Horario	Horario
2	Servicio Médico	Servicio Médico

Medida:

No.	Medida	Formato
1	Pacientes Atendidos	Número de pacientes
2	Médicos Atendiendo	Número de personal disponible
3	Minutos espera promedio	Minutos
4	Minutos Atención promedio	Minutos
5	Fallecimientos	Cantidad de fallecimientos

Filtro:

No.	Operación
1	dia= <día a ingresar>



ANEXO B: Diseño de Extracción

CUENTAS_CORRIENTES_DM

D_TIPO_ITEM

Columna	Llave	Tipo	Mapeo
ID_D_TIPO_ITEM	Sí	Numérico	TIPO_ITEM.ID_TIPO_ITEM
NOMBRE	No	Cadena	TIPO_ITEM.NOMBRE
D_D_TIPO_ITEM	No	Cadena	TIPO_ITEM.D_TIPO_ITEM
CODIGO	No	Cadena	TIPO_ITEM.CODIGO

D_ITEM_CONCEPTO

Columna	Llave	Tipo	Mapeo
ID_D_ITEM_CONCEPTO	Sí	Numérico	ITEM_CONCEPTO.ID_ITEM_CONCEPTO
NOMBRE	No	Cadena	ITEM_CONCEPTO.NOMBRE
D_D_ITEM_CONCEPTO	No	Cadena	TIPO_ITEM.D_TIPO_ITEM
NOM_CIENT	No	Cadena	ITEM_CONCEPTO.NOMBRE_CIENT
NOM_COMUN	No	Cadena	ITEM_CONCEPTO.NOMBRE_COMUN
UNIDAD_MEDIDA	No	Cadena	ITEM_CONCEPTO.UNIDAD_MEDIDA
TOPE_GANANCIA	No	Cadena	ITEM_CONCEPTO.TOPE_GANANCIA
ID_D_TIPO_ITEM	No	Numérico	ITEM_CONCEPTO.ID_TIPO_ITEM

D_D_TIPO_ITE M	No	Cadena	TIPO_ITEM.D_TIPO_ITEM
-------------------	----	--------	-----------------------

D_SEGMENTO

Columna	Llave	Tipo	Mapeo
ID_D_SEGME NTO	Sí	Numérico	SEGMENTO.ID_SEGMENTO
D_D_SEGMEN TO	No	Cadena	SEGMENTO.D_SEGMENTO
CODIGO	No	Cadena	SEGMENTO.CODIGO

D_TIPO_TARIFA

Columna	Llave	Tipo	Mapeo
ID_D_TIP_TAR IFA	Sí	Numérico	TIPO_TARIFA.ID_TIPO_TARIFA
D_D_TIPO_TA RIFA	No	Cadena	TIPO_TARIFA.D_TIPO_TARIFA
NOMBRE	No	Cadena	TIPO_TARIFA.NOMBRE
CODIGO	No	Cadena	TIPO_TARIFA.CODIGO
ID_D_SEGME NTO	No	Numérico	TIPO_TARIFA.ID_SEGMENTO
D_D_SEGMET NO	No	Cadena	SEGMENTO.D_SEGMENTO

D_ESTADO_CUENTA

Columna	Llave	Tipo	Mapeo
ID_D_ESTADO _CUENTA	Sí	Numérico	ESTADO_CUENTA.ID_ ESTADO_CUENTA
D_D_ESTADO _CUENTA	No	Cadena	ESTADO_CUENTA.D_ ESTADO_CUENTA
CODIGO	No	Cadena	ESTADO_CUENTA.CODIGO

D_CUENTA

Columna	Llave	Tipo	Mapeo
ID_D_CUENTA	Sí	Numérico	CUENTA.ID_CUENTA
N_CUENTA	No	Cadena	ESTADO_CUENTA.CODIGO
F_BAJA	No	Fecha	CUENTA.F_BAJA
ID_D_TIP_TARIFA	No	Numérico	CUENTA.ID_TIPO_TARIFA
D_D_TIPO_TARIFA	No	Cadena	TIPO_TARIFA.D_TIPO_TARIFA
ID_D_ESTADO_CUENTA	No	Numérico	CUENTA.ID_ESTADO_CUENTA
D_D_ESTADO_CUENTA	No	Cadena	ESTADO_CUENTA.D_ESTADO_CUENTA
ID_D_PACIENTE	No	Numérico	CUENTA.ID_PACIENTE

F_INGRESOS

Columna	Llave	Tipo	Mapeo
ID_D_DIA	Sí	Numérico	CUENTA_ITEM.F_PAGO
ID_D_ITEM_CONCEPTO	Sí	Numérico	CUENTA_ITEM.ID_ITEM_CONCEPTO
ID_D_TIP_TARIFA	Sí	Numérico	CUENTA_ITEM.ID_TIPO_TARIFA
ID_D_SERVICIO_MEDICO	Sí	Numérico	DOCUMENTO_VENTA.ID_SERVICIO_MEDICO
INGRESOS	No	Numérico	DOCUMENTO_VENTA.TOTAL
N_OPERACIONES	No	Numérico	COUNT(DOCUMENTO_VENTA.ID_DOCUMENTO_VENTA)
PRECIO_ACTUAL	No	Numérico	TARIFA_ITEM.PRECIO

F_TARIFA_MES

Columna	Llave	Tipo	Mapeo
ID_D_TIP_TARIFA	Sí	Numérico	TIPO_TARIFA.TARIFA
ID_D_MES	Sí	Numérico	DOCUMENTO_VENTA.FECHA_CREACION
N_PACIENTES	No	Numérico	COUNT(CUENTA.ID_CUENTA DONDE (F_BAJA=NULO O F_BAJA>MES) ESTADO_CUENTA=ACTIVA) Y
INGRESOS	No	Numérico	SUM(DOCUMENTO_VENTA.TOTAL)
N_OPERACIONES	No	Numérico	COUNT(DOCUMENTO_VENTA.ID_DOCUMENTO_VENTA)

F_DEUDAS

Columna	Llave	Tipo	Mapeo
ID_D_SERVICIO_MEDICO	Sí	Numérico	DOCUMENTO_VENTA.ID_SERVICIO_MEDICO
ID_D_MES	Sí	Numérico	CUENTA_ITEM.FECHA_CREACION
ID_D_CUENTA	Sí	Numérico	CUENTA_ITEM.ID_CUENTA
MONTO_CANCELADO	No	Numérico	SUM(CUENTA_ITEM.TOTAL_PAGADO)
MONTO_PENDIENTE	No	Numérico	SUM(DOCUMENTO_VENTA.TOTAL-CUENTA_ITEM.TOTAL_PAGADO)

PERSONAL_DM

D_TIPO_PERSONA

Columna	Llave	Tipo	Mapeo
ID_D_TIPO_PERSONA	Sí	Número	TIPO_PERSONA.ID_D_TIPO_PERSONA
D_D_TPO_PERSONA	No	Cadena	TIPO_PERSONA.D_TIPO_PERSONA

D_I_OPERATIVO

Columna	Llave	Tipo	Mapeo
ID_D_I_OPERATIVO	Sí	Número	I_OPERATIVO.ID_I_OPERATIVO
D_D_I_OPERATIVO	No	Cadena	I_OPERATIVO.D_I_OPERATIVO

D_I_RESIDENTE

Columna	Llave	Tipo	Mapeo
ID_D_I_RESIDENTE	Sí	Número	I_RESIDENTE.ID_I_RESIDENTE
D_D_I_RESIDENTE	No	Cadena	I_RESIDENTE.D_I_RESIDENTE

D_I_ADMIN

Columna	Llave	Tipo	Mapeo
ID_D_I_INTERNO	Sí	Número	I_INTERNO.ID_I_INTERNO
D_D_I_INTERNO	No	Cadena	I_INTERNO.D_I_INTERNO

D_I_ADMIN

Columna	Llave	Tipo	Mapeo
ID_D_I_ADMIN	Sí	Número	I_ADMIN.ID_I_ADMIN
D_D_I_ADMIN	No	Cadena	I_ADMIN.D_I_ADMIN

D_I_TECNICO

Columna	Llave	Tipo	Mapeo
ID_D_I_TECNICO	Sí	Número co	I_TECNICO.ID_I_TECNICO
D_D_I_TECNICO	No	Cadena	I_TECNICO.D_I_TECNICO

D_I_ENFERMERO

Columna	Llave	Tipo	Mapeo
ID_D_I_ENFERMERO	Sí	Número co	I_ENFERMERO.ID_I_ENFERMERO
D_D_I_ENFERMERO	No	Cadena	I_ENFERMERO.D_I_ENFERMERO

D_I_MEDICO

Columna	Llave	Tipo	Mapeo
ID_D_I_MEDICO	Sí	Número co	I_MEDICO.ID_I_MEDICO
D_D_I_MEDICO	No	Cadena	I_MEDICO.D_I_MEDICO

D_PERSONA

Columna	Llave	Tipo	Mapeo
ID_PERSONA	Sí	Número	PERSONA.ID_PERSONA
NOMBRE	No	Cadena	PERSONA.NOMBRE
ID_D_TIPO_PERSONA	No	Número	PERSONA.ID_TIPO_PERSONA
D_D_TIPO_PERSONA	No	Cadena	TIPO_PERSONA.D_TIPO_PERSONA

D_PERSONA_JURIDICA

Columna	Llave	Tipo	Mapeo
ID_PERSONA	Sí	Número	PERSONA.ID_PERSONA
RAZON_SOCIAL	No	Cadena	PERSONA_JURIDICA.RAZON_SOCIAL

AL			AL
RUC	No	Cadena	PERSONA_JURIDICA.RUC
D_D_TIPO_PERSONA	No	Cadena	TIPO_PERSONA.D_TIPO_PERSONA
ID_D_TIPO_PERSONA	No	Numérico	PERSONA.ID_TIPO_PERSONA

D_PERSONA_NATURAL

Columna	Llave	Tipo	Mapeo
ID_PERSONA	Sí	Numérico	PERSONA.ID_PERSONA
NOMBRE	No	Cadena	PERSONA_NATURAL.NOMBRE
APE_PAT	No	Cadena	PERSONA_NATURAL.APE_PAT
APE_MAT	No	Cadena	PERSONA_NATURAL.APE_MAT
F_NAC	No	Fecha	PERSONA_NATURAL.F_NAC
ID_D_DISTRITO	No	Numérico	PERSONA_NATURAL.ID_DISTRITO
D_D_DISTRITO	No	Cadena	DISTRITO.D_DISTRITO
D_D_TIPO_PERSONA	No	Cadena	TIPO_PERSONA.D_TIPO_PERSONA
ID_D_TIPO_PERSONA	No	Numérico	PERSONA.ID_TIPO_PERSONA

D_PERSONAL

Columna	Llave	Tipo	Mapeo
ID_PERSONA	Sí	Numérico	PERSONA.ID_PERSONA
SUELDO	Sí	Numérico	PERSONAL.SUELDO
ID_D_I_MEDICO	No	Numérico	PERSONAL.ID_I_MEDICO

ID_D_I_ENFERMERO	No	Numérico	PERSONAL.ID_I_ENFERMERO
ID_D_I_TECNICO	No	Numérico	PERSONAL.ID_I_TECNICO
ID_D_I_ADMIN	No	Numérico	PERSONAL.ID_I_ADMIN
ID_D_I_INTERNO	No	Numérico	PERSONAL.ID_I_INTERNO
ID_D_I_RESIDENTE	No	Numérico	PERSONAL.ID_I_RESIDENTE
ID_D_I_OPERATIVO	No	Numérico	PERSONAL.ID_I_OPERATIVO
NOMBRE	No	Cadena	PERSONA_NATURAL.NOMBRE
APE_PAT	No	Cadena	PERSONA_NATURAL.APE_PAT
APE_MAT	No	Cadena	PERSONA_NATURAL.APE_MAT
F_NAC	No	Fecha	PERSONA_NATURAL.F_NAC
D_I_MEDICO	No	Cadena	I_MEDICO.D_I_MEDICO
D_I_ENFERMERO	No	Cadena	I_ENFERMERO.D_I_ENFERMERO
D_I_TECNICO	No	Cadena	I_TECNICO.D_I_TECNICO
D_I_ADMIN	No	Cadena	I_ADMIN.D_I_ADMIN
D_I_INTERNO	No	Cadena	I_INTERNO.D_I_INTERNO
D_I_RESIDENTE	No	Cadena	I_RESIDENTE.D_I_RESIDENTE
D_I_OPERATIVO	No	Cadena	I_OPERATIVO.D_I_OPERATIVO

ADMISION_DM

D_PACIENTE

Columna	Llave	Tipo	Mapeo
ID_D_PACIENTE	Sí	Numérico	PACIENTE.ID_PACIENTE
ID_D_SEXO	No	Numérico	PACIENTE.ID_SEXO
D_D_SEXO	No	Cadena	SEXO.D_SEXO
EDAD	No		AÑO_ACTUAL-F_NACIMIENTO.AÑO
ID_D_ESTADO_CIVIL	No	Numérico	PACIENTE.ID_ESTADO_CIVIL
D_D_ESTADO_CIVIL	No	Cadena	ESTADO_CIVIL.D_ESTADO_CIVIL
ID_D_CATEGORIA_PAC	No	Numérico	PACIENTE.ID_CATEGORIA_PAC
D_D_CATEGORIA_PAC	No	Cadena	CATEGORIA_PACIENTE.D_CATEGORIA_PAC
ID_D_DISTRITO_DIR	No	Numérico	PACIENTE.ID_DISTRITO
D_D_DISTRITO	No	Cadena	DISTRITO.D_DISTRITO
D_D_PROVINCIA	No	Cadena	PROVINCIA.ID_PROVINCIA
ID_D_PROVINCIA	No	Numérico	PROVINCIA.D_D_PROVINCIA
ID_D_DEPARTAMENTO	No	Numérico	DEPARTAMENTO.ID_DEPARTAMENTO
D_D_DEPARTAMENTO	No	Cadena	DEPARTAMENTO.D_DEPARTAMENTO
ID_D_PAIS	No	Numérico	PAIS.ID_PAIS
D_D_PAIS	No	Cadena	PAIS.D_PAIS
ID_D_GRADO	No	Numérico	PACIENTE.ID_GRADO

		co	
D_D_GRADO_INST	No	Cadena	GRADO_INSTRUCCION.D_GRADO_INST
ID_D_I_FALLECIMIENTON	No	Número co	PACIENTE.ID_I_FALLECIMIENTO
D_D_I_FALLECIMIENTO	No	Cadena	I_FALLECIMIENTO.D_I_FALLECIMIENTO
ID_D_CAUSA_MUERTE	No	Número co	PACIENTE.ID_CAUSA_MUERTE
D_D_CAUSA_MUERTE	No	Cadena	CAUSA_MUERTE.D_CAUSA_MUERTE
F_REGISTRO	No		PACIENTE.F_REGISTRO
F_ACTUALIZACION	No		PACIENTE.F_ACTUALIZACION
NOMBRE	No	Cadena	PACIENTE.NOMBRE
APE_PAT	No	Cadena	PACIENTE.APE_PAT
APE_MAT	No	Cadena	PACIENTE.APE_MAT

D_GRADO_INSTRUCCION

Columna	Llave	Tipo	Mapeo
ID_D_GRADO	Sí	Número co	GRADO_INSTRUCCION.ID_GRADO_INS
D_D_GRADO	No	Cadena	GRADO_INSTRUCCION.D_GRADO_INST
CODIGO	No	Cadena	GRADO_INSTRUCCION.CODIGO

D_SEXO

Columna	Llave	Tipo	Mapeo
ID_D_SEXO	Sí	Número co	SEXO.ID_SEXO
D_D_SEXO	No	Cadena	SEXO.D_SEXO
CODIGO	No	Cadena	SEXO.CODIGO

D_I_FALLECIMIENTO

Columna	Llave	Tipo	Mapeo
ID_D_I_FALLECIMIENTO	Sí	Número	I_FALLECIMIENTO.ID_I_FALLECIMIENTO
D_D_I_FALLECIMIENTO	No	Cadena	I_FALLECIMIENTO.D_I_FALLECIMIENTO

D_ESTADO_CIVIL

Columna	Llave	Tipo	Mapeo
ID_D_ESTADO_CIVIL	Sí	Número	ESTADO_CIVIL.ID_ESTADO_CIVIL
D_D_ESTADO_CIVIL	No	Cadena	ESTADO_CIVIL.D_ESTADO_CIVIL
CODIGO	No	Cadena	ESTADO_CIVIL.CODIGO

D_CATEGORIA_PAC

Columna	Llave	Tipo	Mapeo
ID_D_CATEGORIA_PAC	Sí	Número	CATEGORIA_PACIENTE.ID_CATEGORIA_PAC
D_D_CATEGORIA_PAC	No	Cadena	CATEGORIA_PACIENTE.D_CATEGORIA_PAC
CODIGO	No	Cadena	CATEGORIA_PACIENTE.CODIGO

D_ALERGIA

Columna	Llave	Tipo	Mapeo
ID_D_ALERGIA	Sí	Número	ALERGIA.ID_ALERGIA
D_D_ALERGIA	No	Cadena	ALERGIA.D_ALERGIA
CODIGO	No	Cadena	CODIGO

D_CAUSA_MUERTE

Columna	Llave	Tipo	Mapeo
ID_D_CAUSA_MUERTE	Sí	Número	CAUSA_MUERTE.ID_CAUSA_MUERTE

D_D_CAUSA_MUERTE	No	Cadena	CAUSA_MUERTE.D_CAUSA_MUERTE
------------------	----	--------	-----------------------------

CITA_DM**D_ZONA_TRABAJO**

Columna	Llave	Tipo	Mapeo
ID_D_ZONA_TRABAJO	Sí	Numérico	ZONA_TRABAJO.ID_ZONA_TRABAJO
D_D_ZONA_TRABAJO	No	Cadena	ZONA_TRABAJO.D_ZONA_TRABAJO
CODIGO	No	Cadena	ZONA_TRABAJO.CODIGO
D_D_PABELLON	No	Cadena	PABELLON.D_PABELLON
ID_D_PABELLON	No	Numérico	ZONA_TRABAJO.ID_PABELLON

D_TURNO

Columna	Llave	Tipo	Mapeo
ID_D_TURNO	Sí	Numérico	TURNO.ID_TURNO
ID_PERSONA	No	Numérico	TURNO.ID_PERSONAL
D_APELLIDO_PAT	No	Cadena	PERSONA_NATURAL.APE_PAT
D_APELLIDO_MAT	No	Cadena	PERSONA_NATURAL.APE_MAT
D_NOMBRES	No	Cadena	PERSONA_NATURAL.NOMBRE
ID_D_SERVICIO_MEDICO	No	Numérico	TURNO.ID_SERVICIO_MEDICO
F_INICIO	No	Fecha	TURNO.F_INICIO

F_FIN	No	Fecha	TURNO.F_FIN
ID_D_HORARIO	No	Numérico	TURNO.ID_HORARIO
D_D_HORARIO	No	Cadena	HORARIO.D_HORARIO
ID_D_ZONA_TRABAJO	No	Numérico	TURNO.ID_ZONA_TRABAJO
D_D_ZONA_TRABAJO	No	Cadena	ZONA_TRABAJO.D_ZONA_TRABAJO

D_TURNOXDIA

Columna	Llave	Tipo	Mapeo
ID_D_TURNO	Sí	Numérico	TURNOXDIA.ID_TURNO
ID_D_DIA_SEMANA	Sí	Numérico	TURNOXDIA.ID_D_DIA_SEMANA
D_D_DIA_SEMANA	No	Cadena	DIA_SEMANA.D_DIA_SEMANA
D_APELLIDO_PAT	No	Cadena	PERSONA_NATURAL.APE_PAT
D_APELLIDO_MAT	No	Cadena	PERSONA_NATURAL.APE_MAT
D_NOMBRES	No	Cadena	PERSONA_NATURAL.NOMBRE
D_D_SERVICIO_MEDICO	No	Cadena	SERVICIO_MEDICO.D_SERVICIO_MEDICO
D_D_HORARIO	No	Cadena	HORARIO.D_HORARIO
D_D_ZONA_TRABAJO	No	Cadena	ZONA_TRABAJO.D_ZONA_TRABAJO

D_TURNXPACIENTE

Columna	Llave	Tipo	Mapeo
ID_D_TURNO	Sí	Numérico	TURNOXPACIENTE.ID_TURNO
ID_D_DIA_SEMANA	Sí	Numérico	TURNOXPACIENTE.ID_D_DIA_SEMANA
ID_D_PACIENTE	Sí	Numérico	TURNOXPACIENTE.ID_PACIENTE
ID_D_DIA	Sí	Numérico	TURNOXPACIENTE.ID_DIA
ID_D_HORA	Sí	Numérico	TURNOXPACIENTE.ID_HORA
D_D_TURNO	No	Cadena	HORARIO.D_HORARIO + PERSONAL.CODIGO
D_D_DIA_SEMANA	No	Cadena	DIA_SEMANA.D_DIA_SEMANA
D_APELLIDO_PAT	No	Cadena	PERSONA_NATURAL.APE_PAT
D_APELLIDO_MAT	No	Cadena	PERSONA_NATURAL.APE_MAT
D_NOMBRES	No	Cadena	PERSONA_NATURAL.NOMBRE
FECHA	No	Fecha	ID_DIA.FECHA
D_D_HORA	No	Cadena	HORA.D_HORA
MIN_ATENCION	No	Numérico	TURNOXPACIENTE.MIN_ATENCION
MIN_ESPERA	No	Numérico	TURNOXPACIENTE.MIN_ESPERA

CITA_DM

F_EMERGENCIAS

Columna	Llave	Tipo	Mapeo
ID_D_DIA	Sí	Numérico	INTERVENCION.ID_DIA
ID_D_HORARIO	Sí	Numérico	TURNO.ID_HORARIO
ID_D_SERVICIO_MEDICO	Sí	Numérico	EPISODIO.ID_SERVICIO_MEDICO
N_PACIENTES	No	Numérico	COUNT(ID_INTERVENCION), TIPO_INTERVENCION.NOMBRE=E MERGENCIA
N_AUXILIARES	No	Numérico	COUNT(PERSONALXINTERVENCIO N.ID_PERSONA)
MIN_ESPERA_PROM	No	Numérico	AVG(TURNOXPACIENTE.MIN_ESP ERA)
N_FALLECIMIENTOS	No	Numérico	SUM(INTERVENCION.ID_I_CAUSO_ MUERTE),

F_PACIENTES_ATENDI

Columna	Llave	Tipo	Mapeo
ID_D_DIA	Sí	Numérico	TURNOXPACIENTE.ID_DIA
ID_D_SERVICIO_MEDICO	Sí	Numérico	EPISODIO.ID_SERVICIO_MEDICO
ID_D_HORARIO	Sí	Numérico	TURNO.ID_HORARIO
ID_PERSONA	Sí	Numérico	TURNO.ID_PERSONAL
N_PACIENTES	No	Numérico	COUNT(TURNOXPACIENTE.ID_PE RSONA)
MIN_ESPERA_PROM	No	Numérico	AVG(TURNOXPACIENTE.MIN_ESP

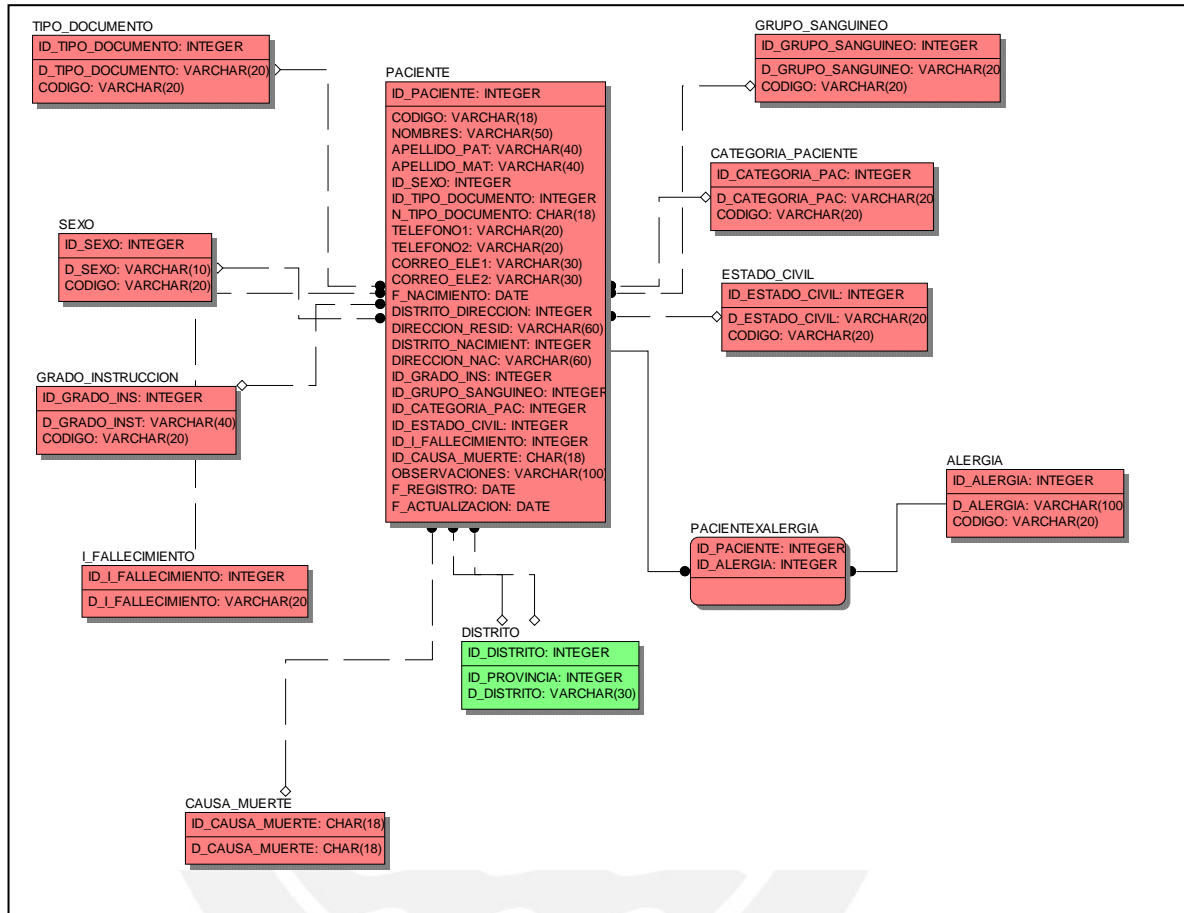
PROM			ERA)
MIN_ATENCIO N_PROM	No	Numérico	AVG(TURNOXPACIENTE.MIN_ATE NCION)
MONTO	No	Numérico	DOCUMENTO_VENTA.MONTO

F_PACIENTES_INTERV

Columna	Llave	Tipo	Mapeo
ID_D_DIA	Sí	Numérico	INTERVENCION.ID_DIA
ID_D_SERVICI O_MEDI	Sí	Numérico	EPISODIO.ID_SERVICIO_MEDI
ID_D_HORARI O	Sí	Numérico	TURNO.ID_HORARIO
ID_PERSONA	Sí	Numérico	PERSONALXINTERVENC.ID_PERS ONA, PERSONALXINTERVEC.I_PRINCIP AL=1
ID_D_TIPO_IN TERVENCION	Sí	Numérico	INTERVENCION.ID_TIPO_INTERVE NCION
N_PACIENTES	No	Numérico	COUNT(INTERVENCION.ID_PACIE NTE)
N_FALLECIMIE NTOS	No	Numérico	SUM(INTERVENCION.ID_I_CAUSO_ MUERTE)
N_HOSPITALIZ ACIONES	No	Numérico	SUM(INTERVENCION.ID_I_GENER A_HOSP)

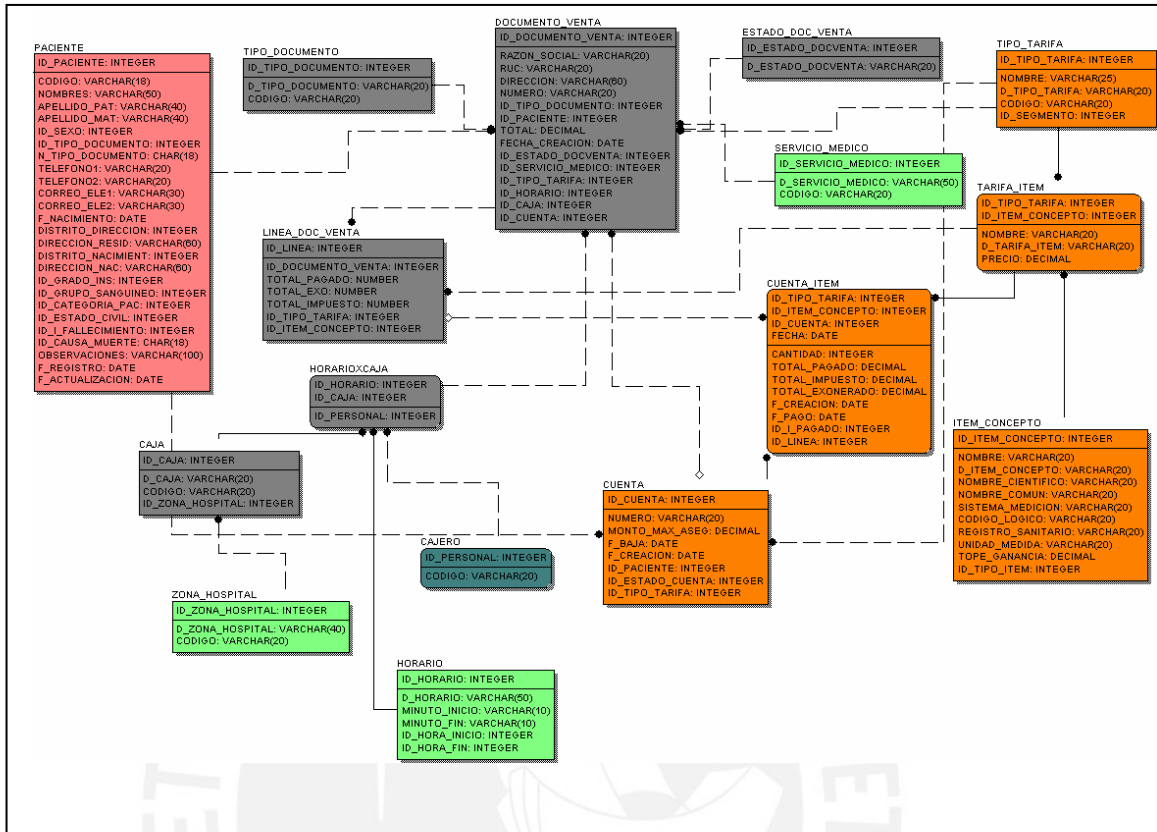
ANEXO C: Vistas del Data Warehouse

ADMISION_DW



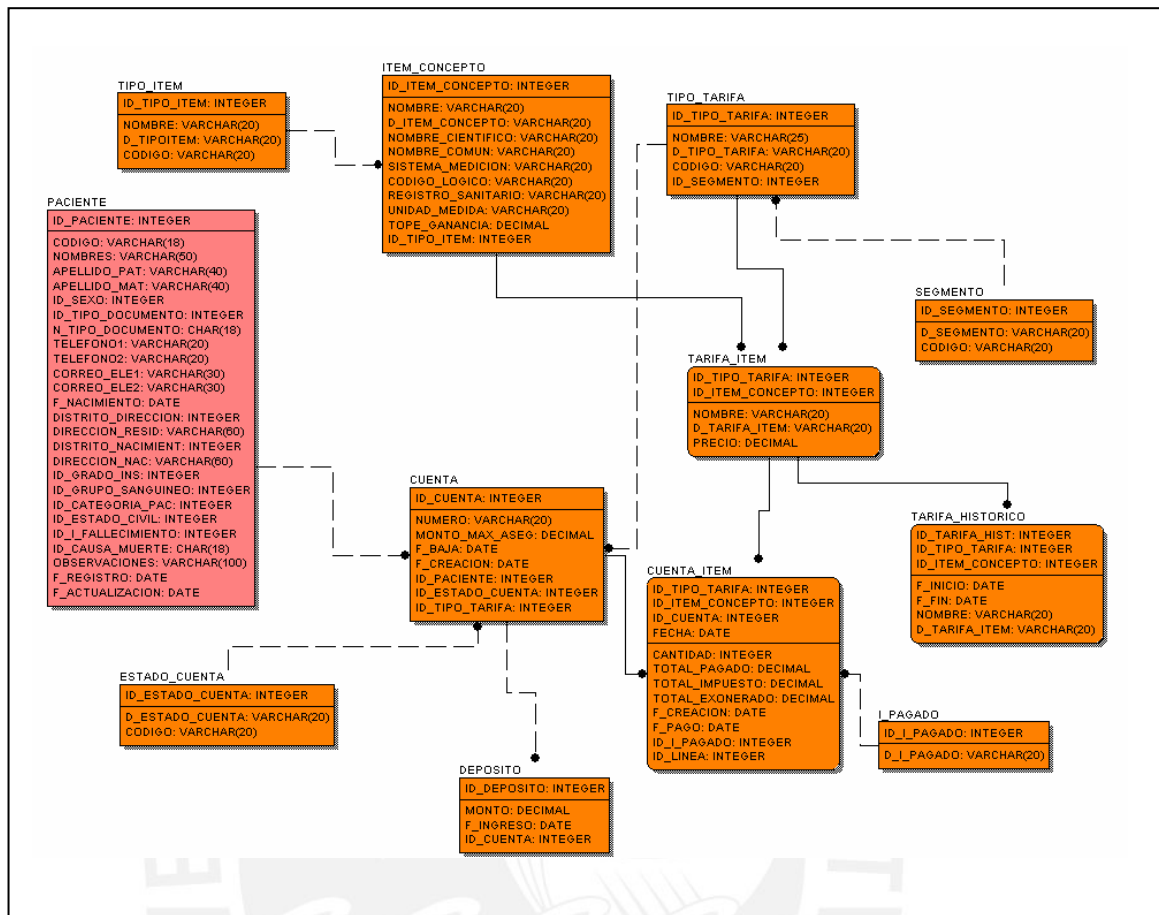
Esta vista contiene todos los datos acerca de los pacientes, tales como sus datos personales, lugar de residencia y algunos datos propios del hospital acerca del paciente.

CAJA_DW



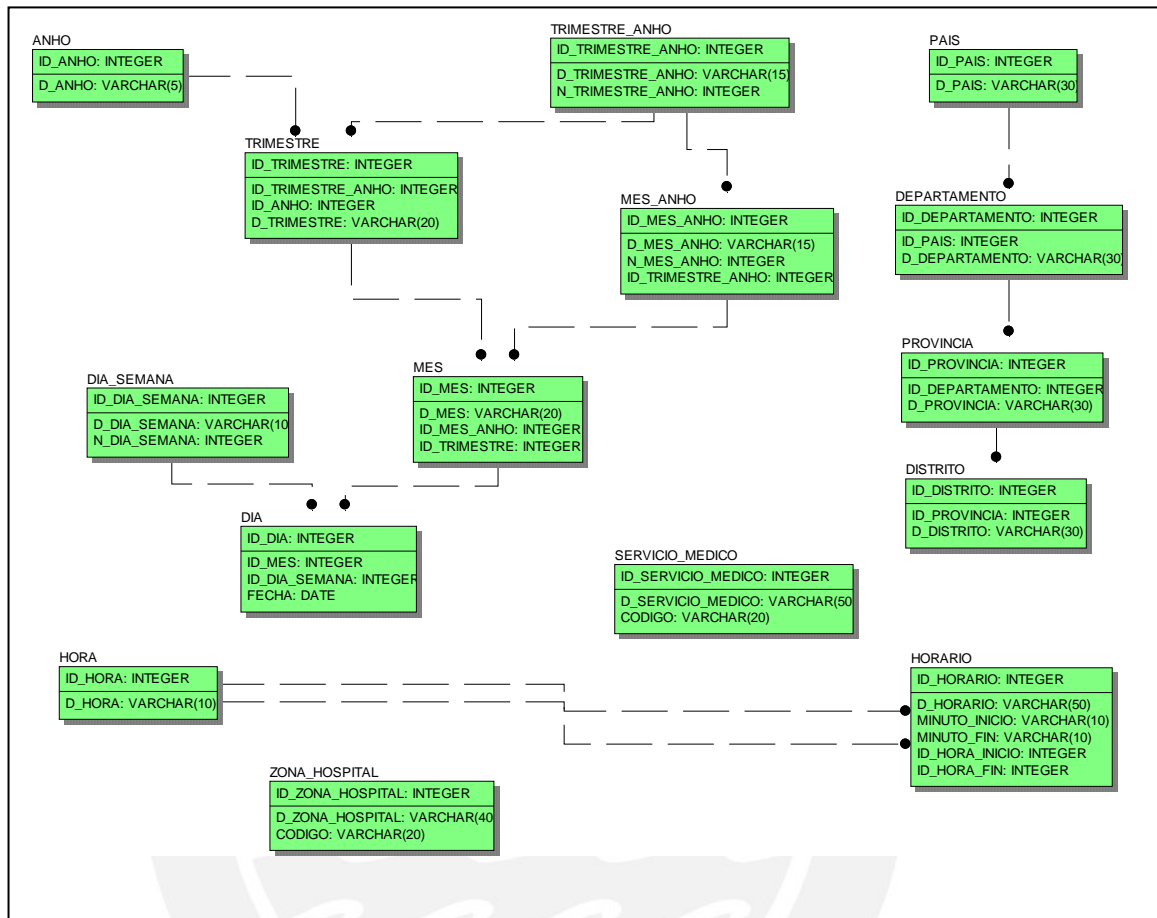
Esta vista contiene todas las operaciones que se realizan en las distintas cajas del hospital. Se registra el concepto de venta, el tipo de documento de venta y si el documento de venta está asociado al pago de alguna cuenta corriente.

CUENTAS_CORRIENTES_DW



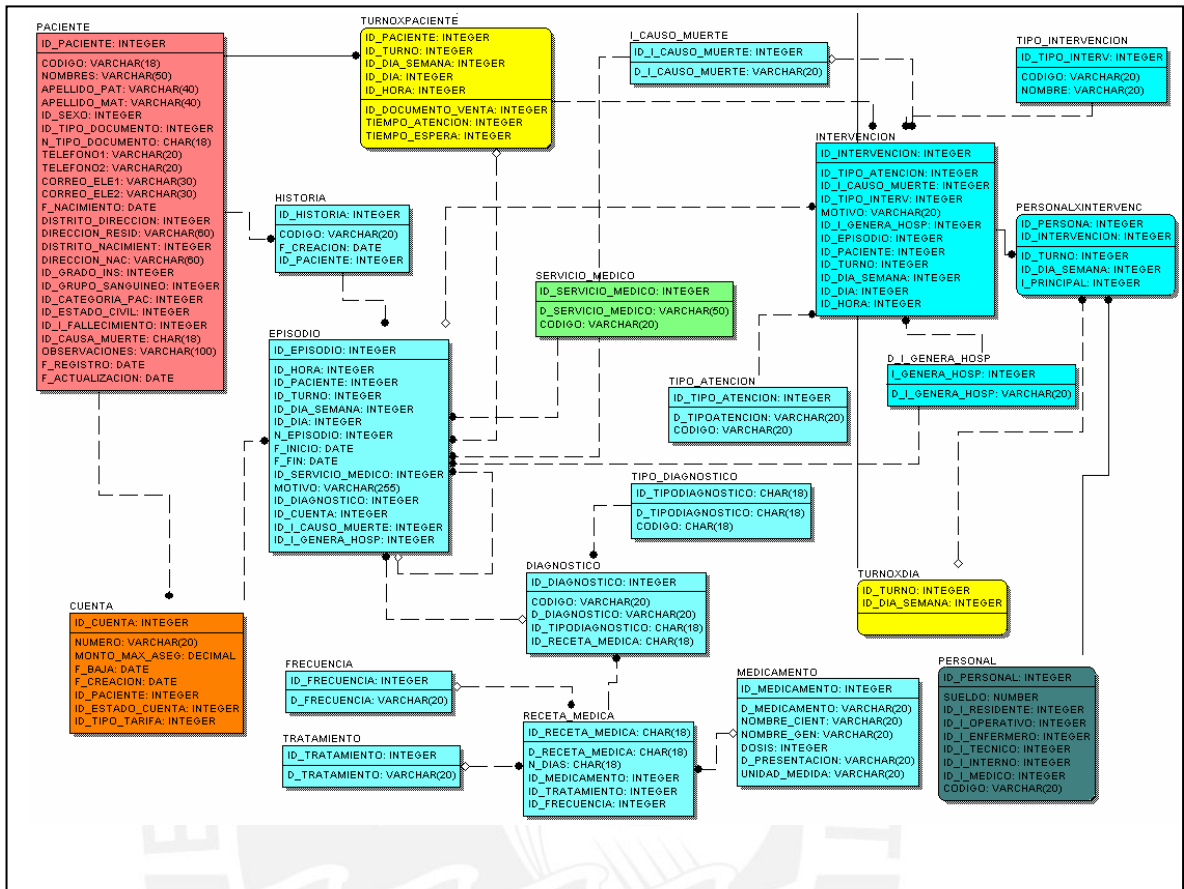
Esta vista contiene todas las cuentas que algún individuo tiene dentro del hospital. Cada cuenta tiene un monto a pagar y un monto cancelado. Además esta vista contiene las distintas tarifas que se manejan para los distintos segmentos de pacientes.

GENERAL_DW



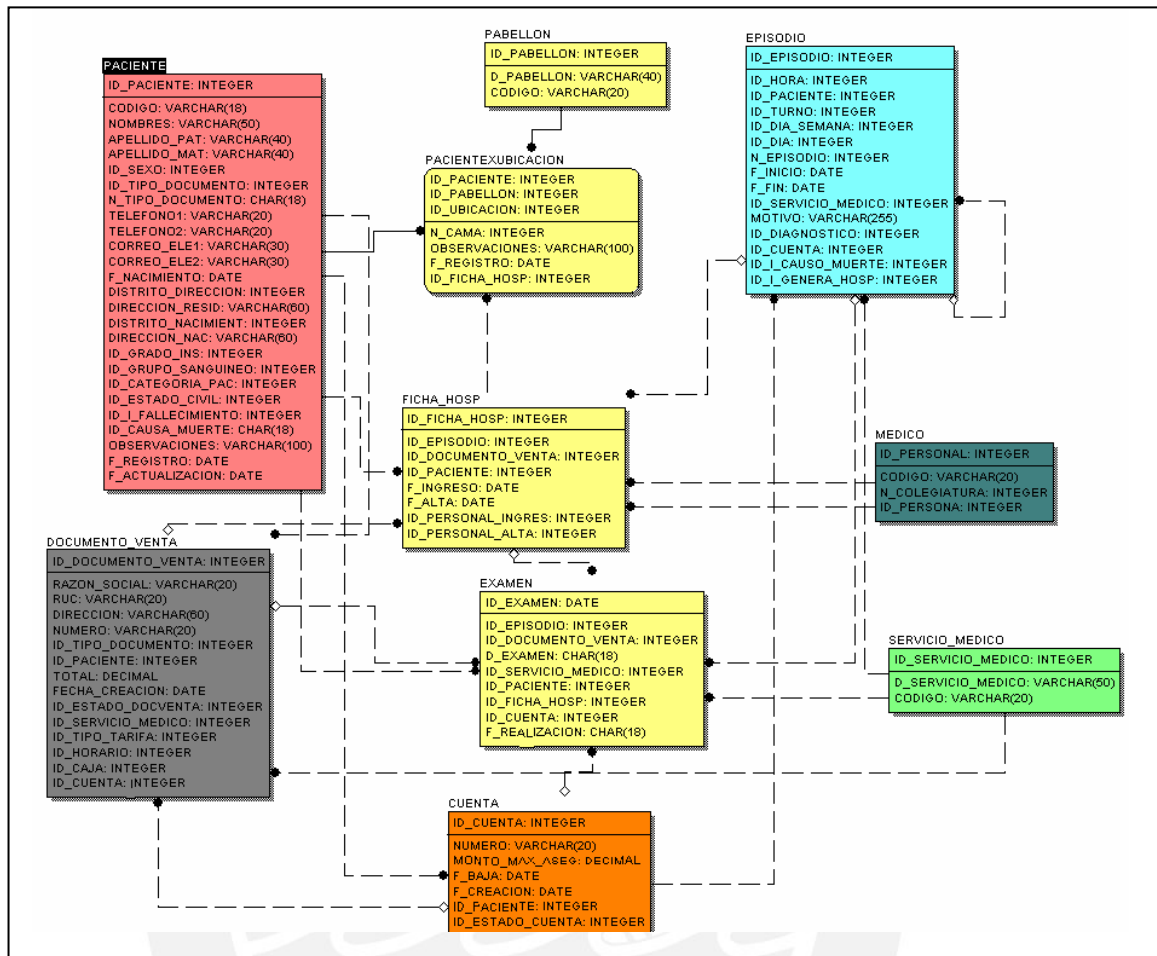
Esta vista contiene las tablas generales, tales como tiempo y ubigeo.

HISTORIA_CLINICA_DW



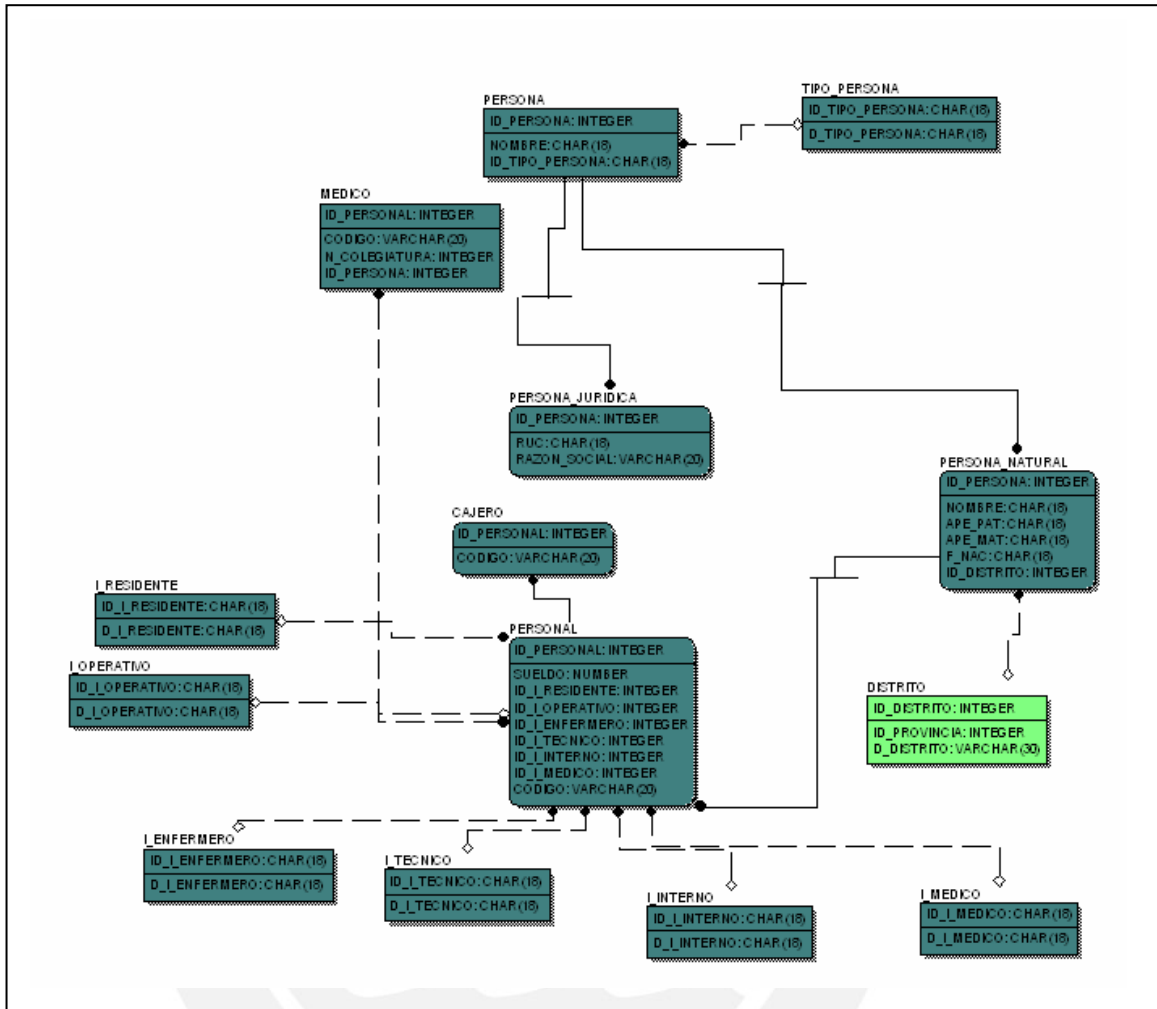
Esta vista contiene la historia clínica de los pacientes, lo cual incluye los distintos episodios médicos que pudo haber tenido el paciente, las intervenciones médicas y los diagnósticos recibidos.

HOSPITALIZACION_DW



Esta vista contiene las hospitalizaciones que reciben los pacientes, indicando el pabellón en que se ubican, el personal encargado, el tiempo de hospitalización y los distintos exámenes médicos recibidos.

PERSONAL_DW



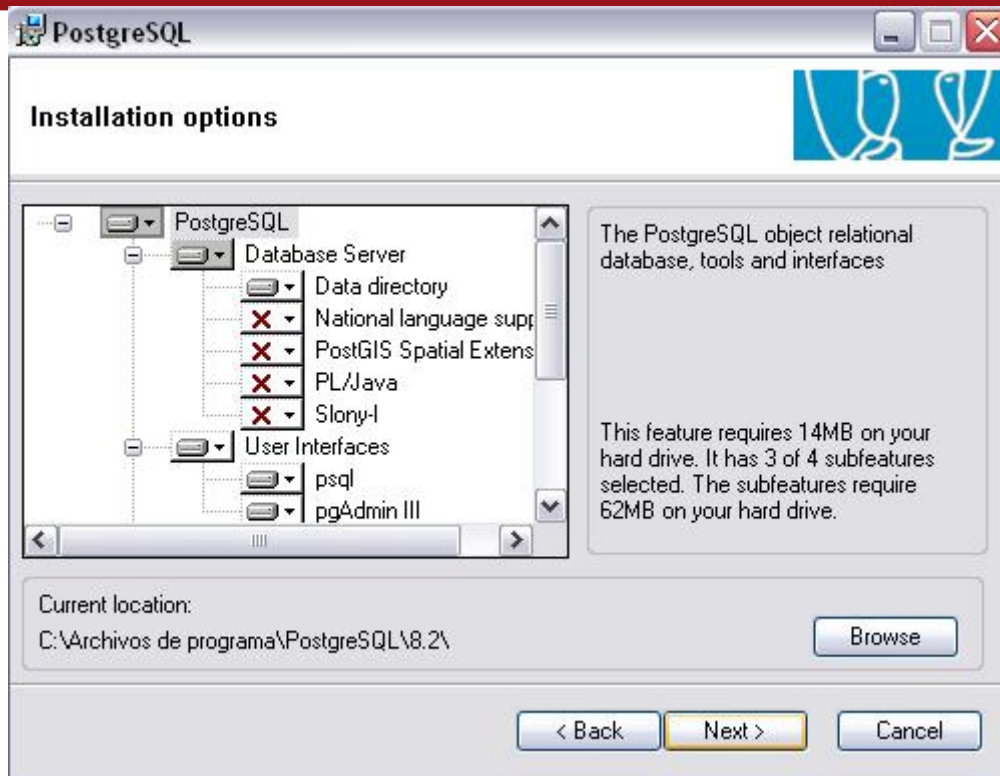
Finalmente, esta vista contiene los datos del personal del hospital.

Anexo D: Configuración PostGresSQL

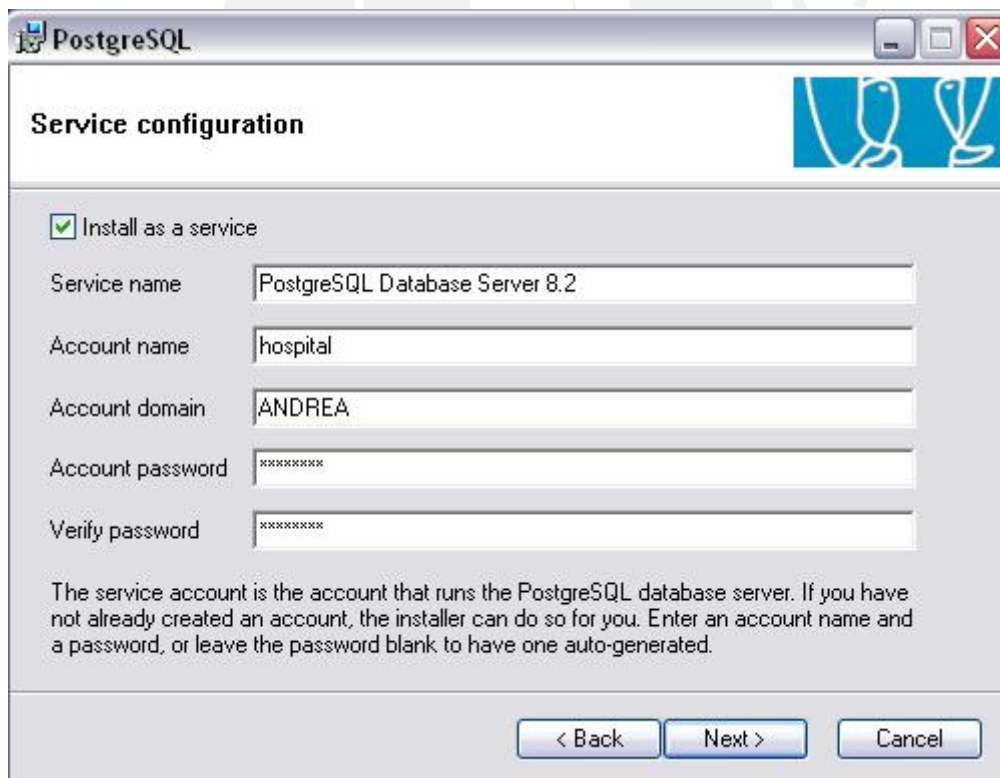
La primera pantalla que nos aparece será la siguiente:



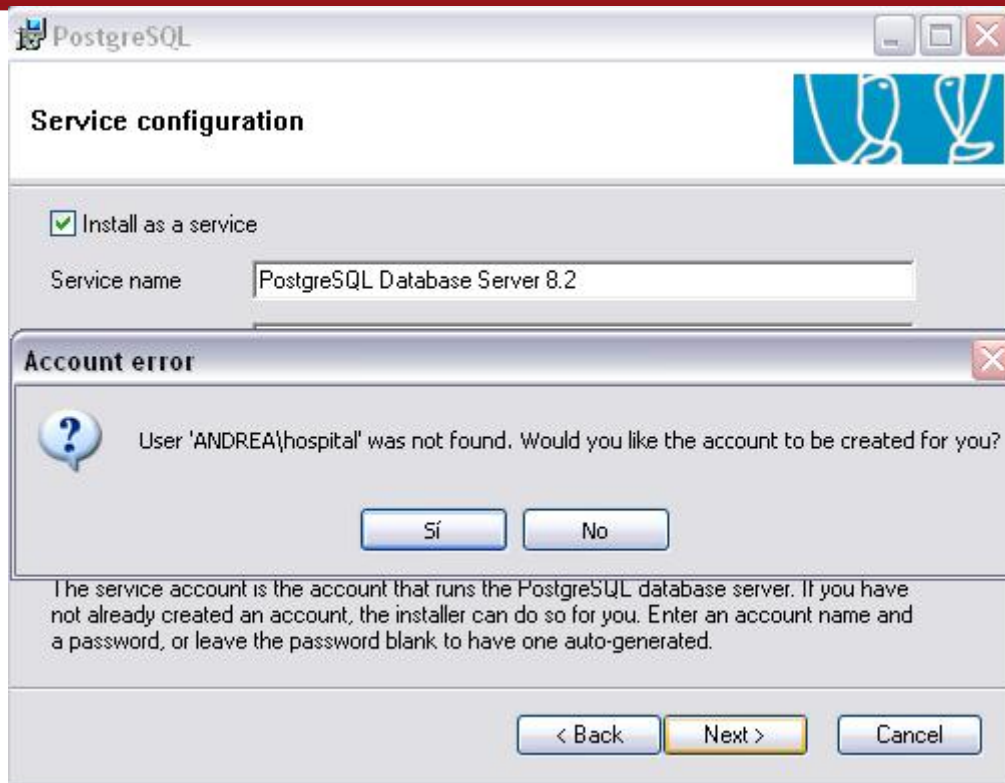
Siguiendo con el wizard de instalación, escogemos las opciones para instalar por defecto:



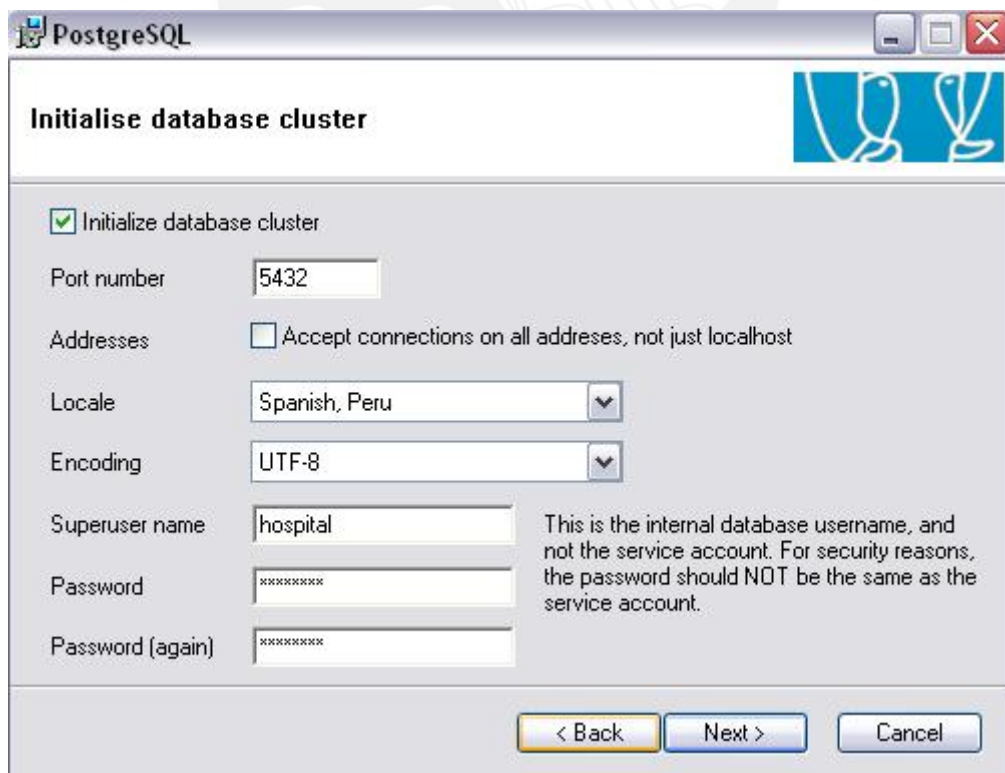
En la configuración del servicio ponemos los siguientes datos:



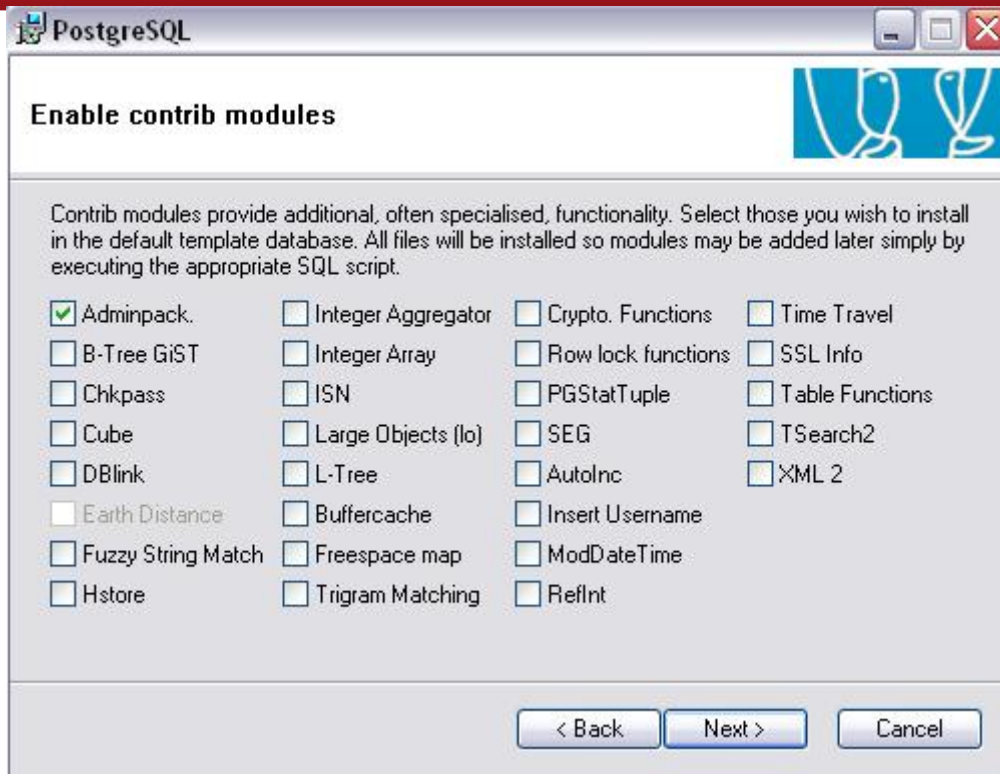
Nos saldrá un mensaje de error como el siguiente y aceptamos el mensaje:



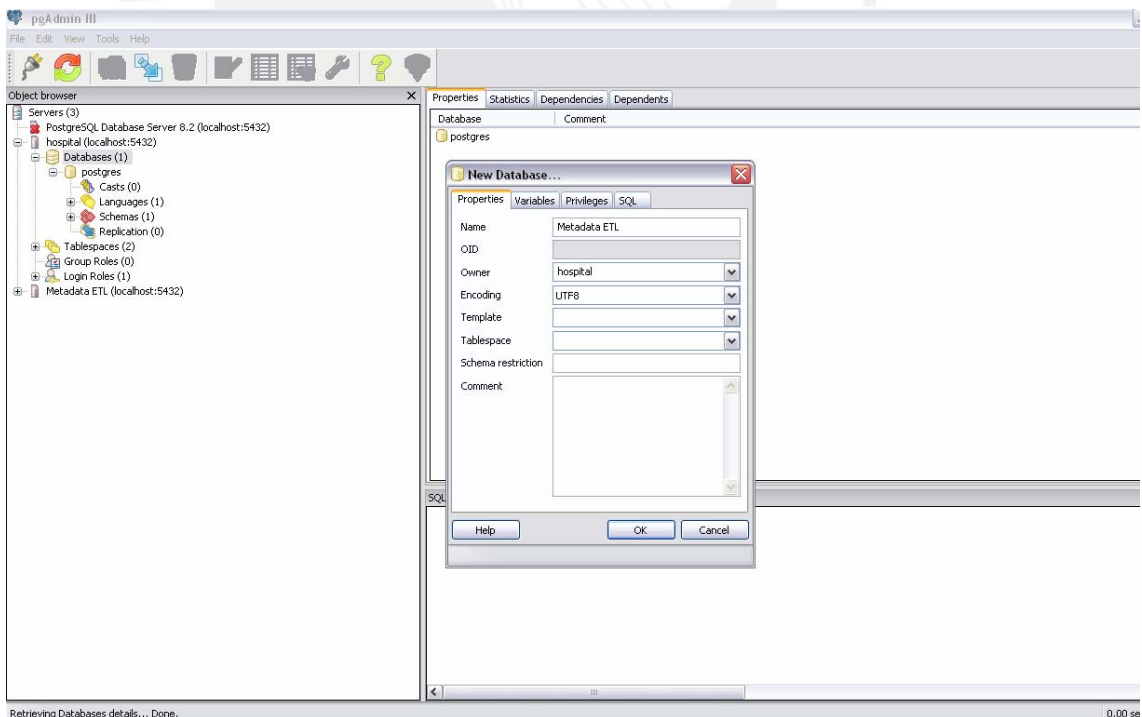
Para la configuración de la base de datos ponemos los siguientes datos, donde password= hospital:



Nos aparece la siguiente ventana y ponemos las siguientes opciones:

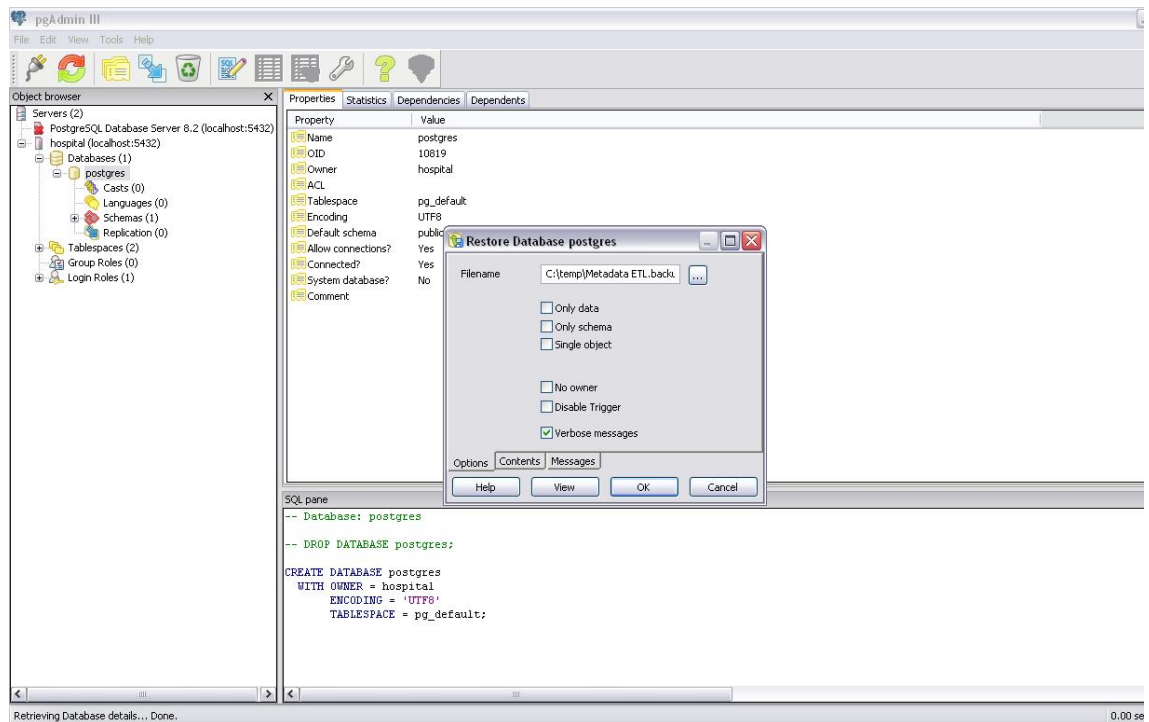


Finalmente, accedemos a la base de datos por medio del ícono del pgAdminIII. Aquí debemos de crear una nueva base de datos. Para esto seguimos lo siguiente:

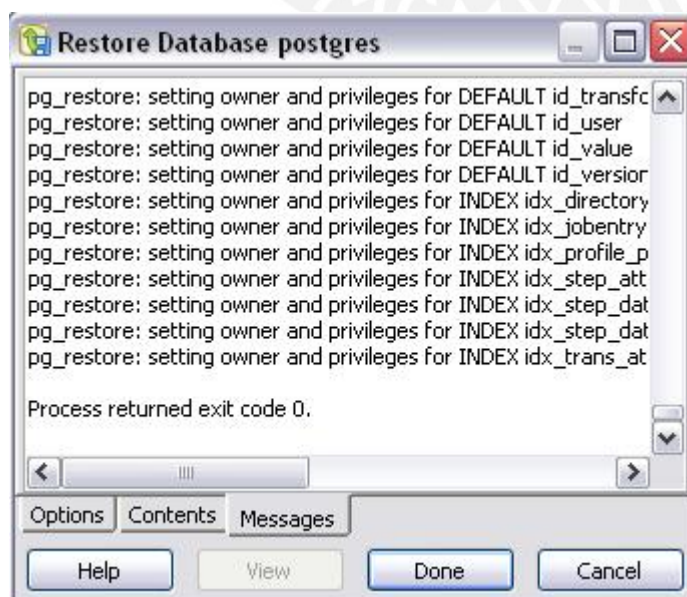


Anexo E: Construcción de la Metadata

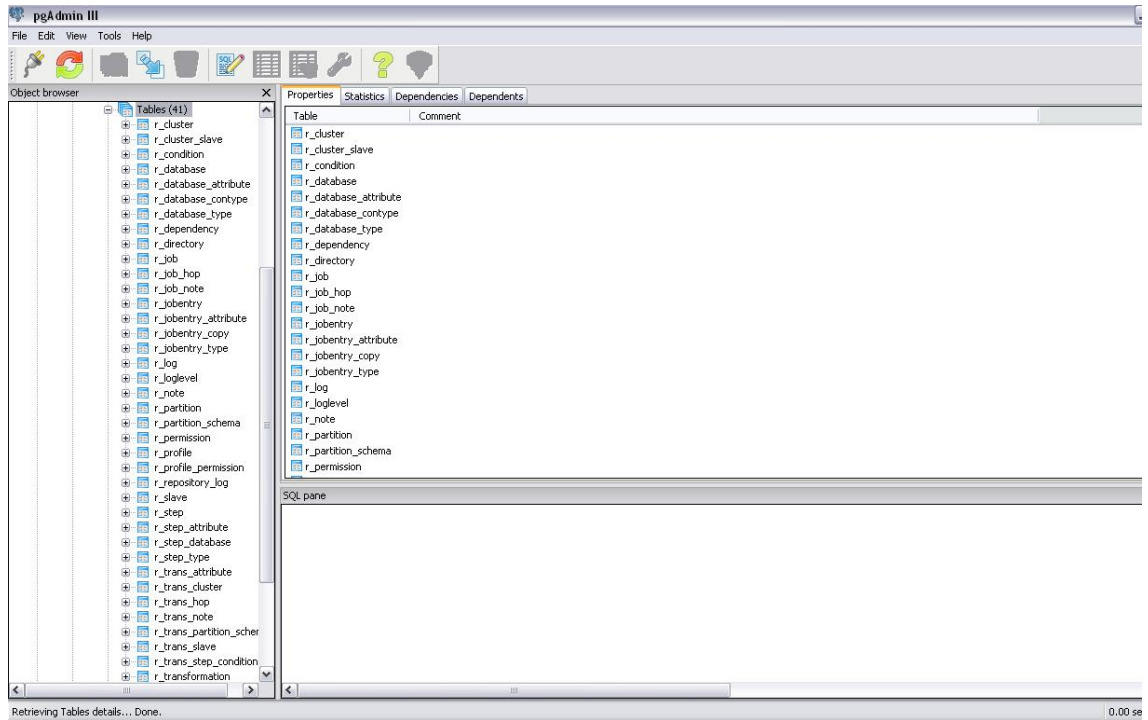
Luego de situarse en la base de datos creada, se selecciona la opción *Restore*. Aparece la siguiente pantalla:



El archivo de *backup* de la metadata debe estar en una ruta específica para poder extraerlo de esa ruta. Si la importación de la metadata se realiza en forma correcta, aparecerá un log como el siguiente, y se selecciona la opción *DONE* para finalizar:

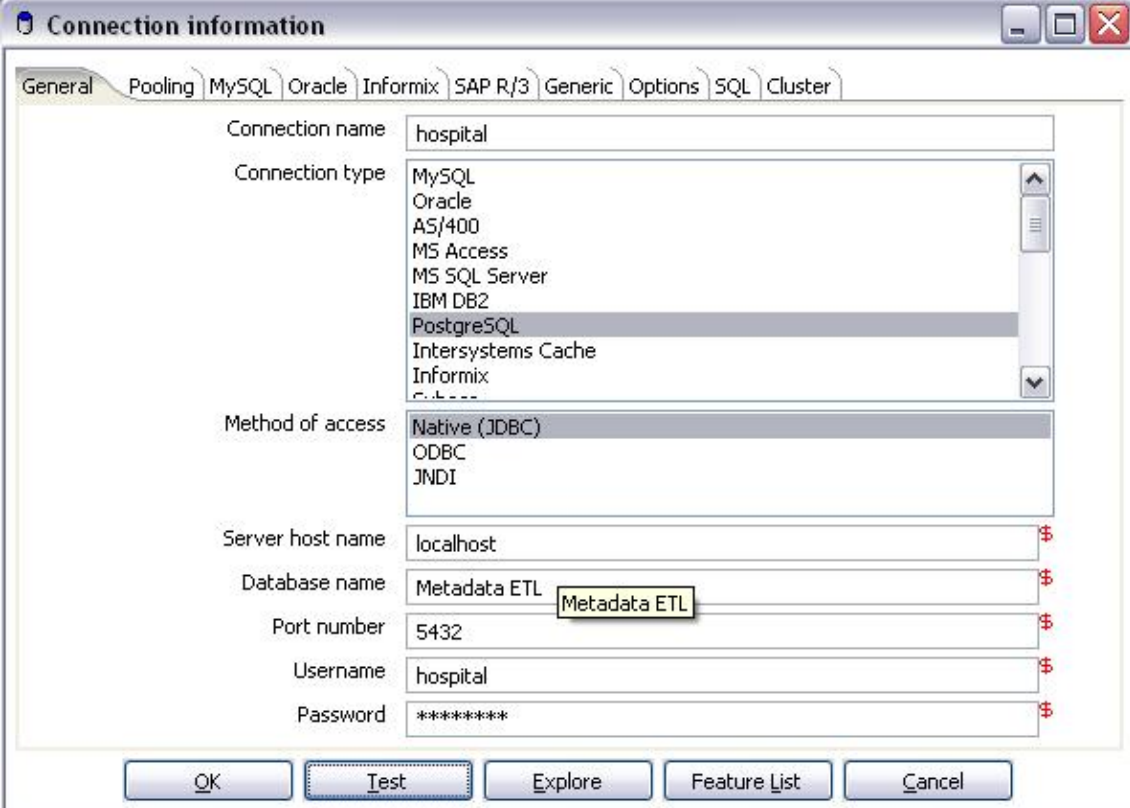


Finalmente se puede verificar que las tablas de la metadata se encuentren en la base de datos:



Anexo F: Configuración de la herramienta de ETL

Se ingresan los siguientes datos en la pantalla de configuración:



Connection information

General Pooling MySQL Oracle Informix SAP R/3 Generic Options SQL Cluster

Connection name: hospital

Connection type: PostgreSQL

Method of access: Native (JDBC)

Server host name: localhost

Database name: Metadata ETL

Port number: 5432

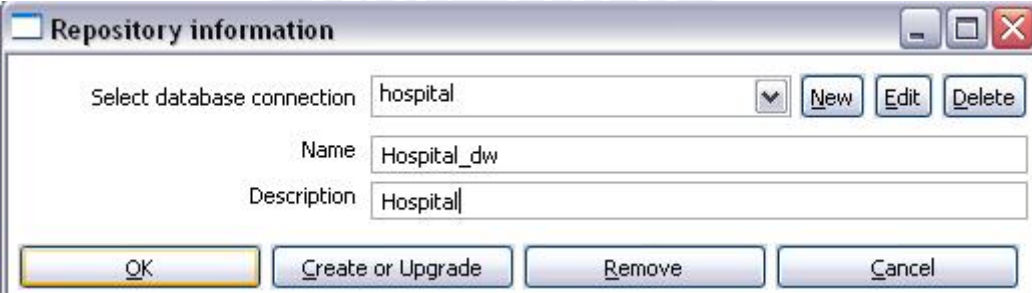
Username: hospital

Password: *****

OK Test Explore Feature List Cancel

Donde *password* es: hospital.

Se selecciona la opción *Test* para comprobar si la conexión es correcta. Luego, se deben llenar los siguientes datos y seleccionar la opción *Create* or *Upgrade*, para finalmente seleccionar la opción *OK*:



Repository information

Select database connection: hospital

Name: Hospital_dw

Description: Hospital

OK Create or Upgrade Remove Cancel

Por último es necesario registrarse por medio de la siguiente pantalla:

Select a repository



pentaho™
open source business intelligence™

Pentaho Data Integration
Previously Kettle

Welcome to Spoon version 2.5.0

Repository: Hospital_dw

Login: admin

Password: *****



Anexo G: Pentaho Design Studio

Antes de tener instalar el Pentaho Design Studio, es necesario instalar la última versión del JRE. Si no se tiene instalada, la versión de la plataforma BI, antes descrita, posee ya una versión del JRE. En ambos casos, verificar que la ruta "jre/bin" se encuentra en la variable del sistema PATH.

Para instalar el Design Studio, es necesario descargar el archivo pentaho-design-studio_X.X.X.X.zip de la web del Pentaho. Se ejecuta el archivo PentahoWorkBench.exe.



Dentro del Design Studio, se selecciona Archivo -> Nuevo -> Proyecto. Se selecciona proyecto simple, se nombra el proyecto (Hospital para este caso) y deseccionamos el *checkbox* de uso *default*, para especificar en la ruta que aparece, el lugar donde se encuentra instalada nuestra plataforma (en este caso .../pentaho-demo/pentaho-solutions).

Anexo H: Pentaho Report Designer

Este módulo se debe descargar por separado de la página Web del Pentaho: pentaho-report-design-wizard-x.x.xx.zip. Al descomprimirlo, se debe ejecutar el archivo startdesigner.bat



Es importante señalar que este módulo no viene con todos los drivers necesarios para conectarse a todas las bases de datos. Como en nuestro caso estamos usando postgresql, es necesario primero descargar de la Web el JDBC de postgresql. Luego, este driver se debe colocar en la carpeta lib/jdbc del report designer, para que así este lo reconozca y podamos crear reportes a partir de esa base de datos.

Un uso detallado de este módulo se encuentra en la Web: <http://wiki.pentaho.org/display/studio/Step-by-Step+Guide>.

Sin embargo, el Report Designer posee algunas limitaciones en lo que son reportes avanzados. En esos casos, es necesario manipular manualmente los archivos de los reportes, lo cual sólo es recomendado para usuarios avanzados.

Finalmente, es importante saber que el Report Designer es capaz de leer las definiciones de los cubos, creados por el Cube Designer, a través del lenguaje MDX (Multidimensional Expressions).

Anexo I: Pentaho Cube Designer

Descargar de la Web de Pentaho el archivo: CubeDesigner-0.7.2.0_Win32.zip y descomprimirlo en el lugar de su elección. Al igual que el Report Designer, es necesario colocar el jdbc a usar en la carpeta lib/jdbc. Para ejecutarlo, sólo es necesario correr el archivo CubeDesigner.exe



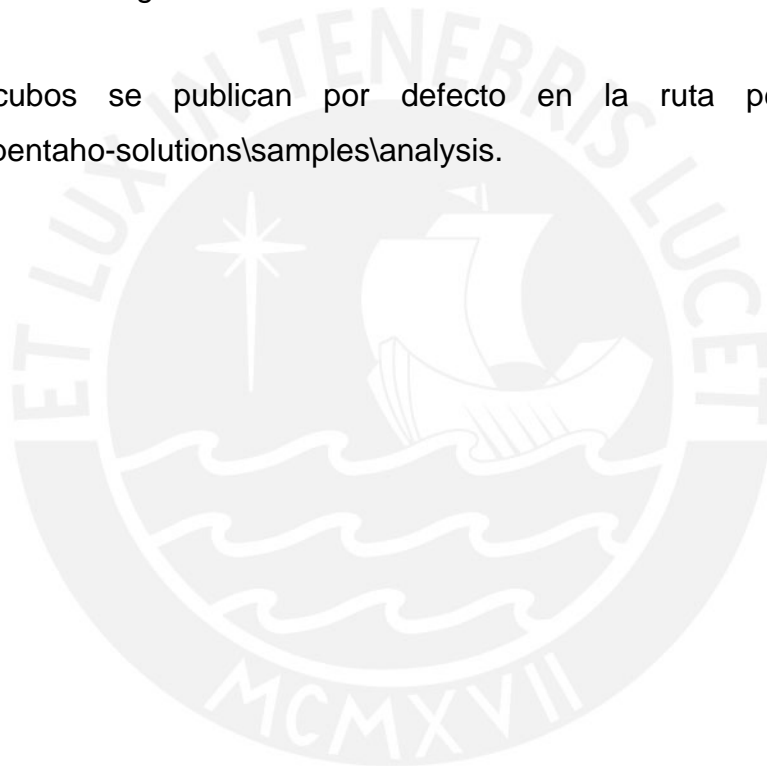
Existe una documentación muy detallada de cómo usar este módulo en la Web del Pentaho. En general, este módulo consiste en un Wizard de seis pasos, que permitirá seleccionar la base de datos a conectarse, seleccionar las tablas a usar, definir las dimensiones, mapear las tablas y dimensiones, crear atributos, jerarquías y medidas, y finalmente publicar un cubo en la plataforma del Pentaho.

No se entrará en detalle en el uso de esta herramienta, pues se puede aprender a través de un tutorial, pero es importante mencionar que al momento de publicar un cubo al servidor, este debe estar inactivo y se debe

usar las credenciales suzy/password o joe/password (usuarios predeterminados que vienen con la plataforma) para publicar el cubo, al menos que se haya creado usuarios propios. Además, el password de publicación debe ser según el password definido en el archivo: pentaho\pentaho-demo\pentaho-solutions\system\publisher-config.xml. De no existir este archivo, deberá crearlo con la siguiente estructura:

```
<publisher-config>  
<publisher-password>[password]</publisher-password>  
</publisher-config>
```

Los cubos se publican por defecto en la ruta pentaho\pentaho-demo\pentaho-solutions\samples\analysis.



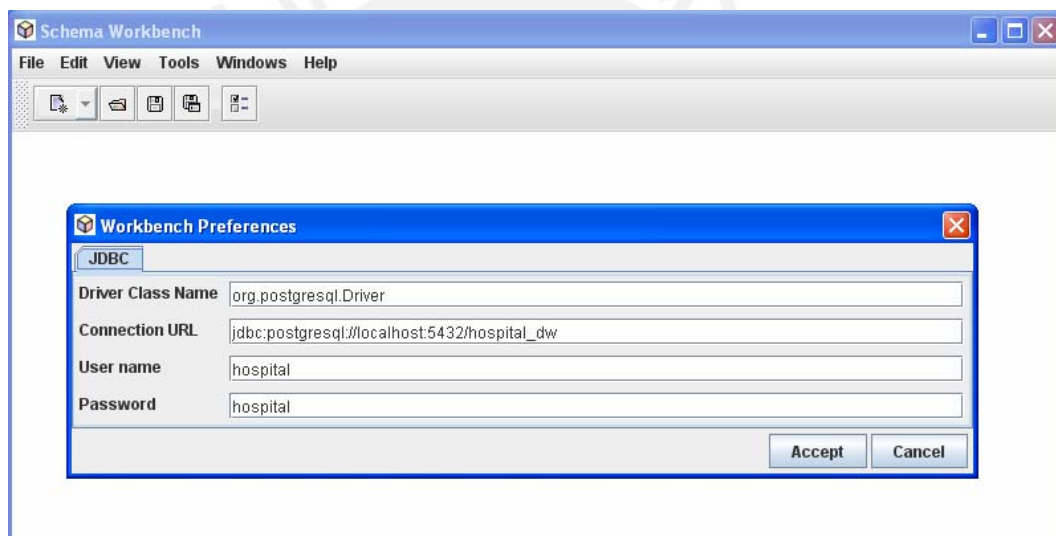
Anexo J: Pentaho Workbench

Descargar de la Web del Pentaho el archivo workbench-2.3.2.9247.zip (o la versión más actual de este) y descomprimirlo en el lugar de su elección. Al igual que el Report Designer, es necesario colocar el jdbc a usar en la carpeta lib. Antes de ejecutar el workbench, es necesario editar el archivo workbench.bat para agregar las siguientes líneas:

```
set CP=%CP%;lib/commons-vfs.jar;lib/commons-logging.jar
```

```
set CP=%CP%;lib/postgresql-8.2-506.jdbc4.jar
```

Luego ejecute el archivo workbench.bat:



La imagen muestra la ventana de preferencias del workbench, la cual es necesaria configurar para poder lograr la conexión. Driver Class Name es un nombre único según el driver de base de datos que se utilice. Es importante distinguir entre mayúsculas y minúsculas. Connection URL es la cadena de conexión a la base de datos. Finalmente, User Name y Password son el usuario y la contraseña para conectarse a la base de datos.

Luego de crear o modificar un cubo con el workbench, se debe ingresar al servidor y sobrescribir el archivo que se desea cambiar en la ruta donde desea que aparezca (esto es debido a que el workbench no cuenta con la opción de publicar los informes las servidor). Por ejemplo: pentaho\pentaho-demo\pentaho-solutions\samples\analysis

Anexo K: Configuración de la plataforma BI.

Para que uno pueda usar su propia base de datos con sus datos y no la base de datos HyperSonic que viene con la plataforma, es necesario seguir los siguientes pasos:

Crear un archivo que contenga la información de conexión a la base de datos. Por ejemplo, llamaremos a este archivo ds.xml y lo configuraremos para adaptarse a la base de datos que se utiliza en esta tesis:

```
<?xml version="1.0" encoding="UTF-8"?>
<datasources>
  <local-tx-datasource>
    <jndi-name>postgresql</jndi-name>
    <connection-
url>jdbc:postgresql://localhost:5432/hospital_dw/</connection-url>
    <driver-class>org.postgresql.Driver</driver-class>
    <user-name>hospital</user-name>
    <password>hospital</password>
  </local-tx-datasource>
</datasources>
```

Este archivo lo debemos colocar en la ruta:
pentaho-demo\jboss\server\default\deploy

Luego, es necesario editar el archivo pentaho-demo\jboss\server\default\deploy\pentaho.war\WEB-INF\web.xml para agregar la siguiente entrada:

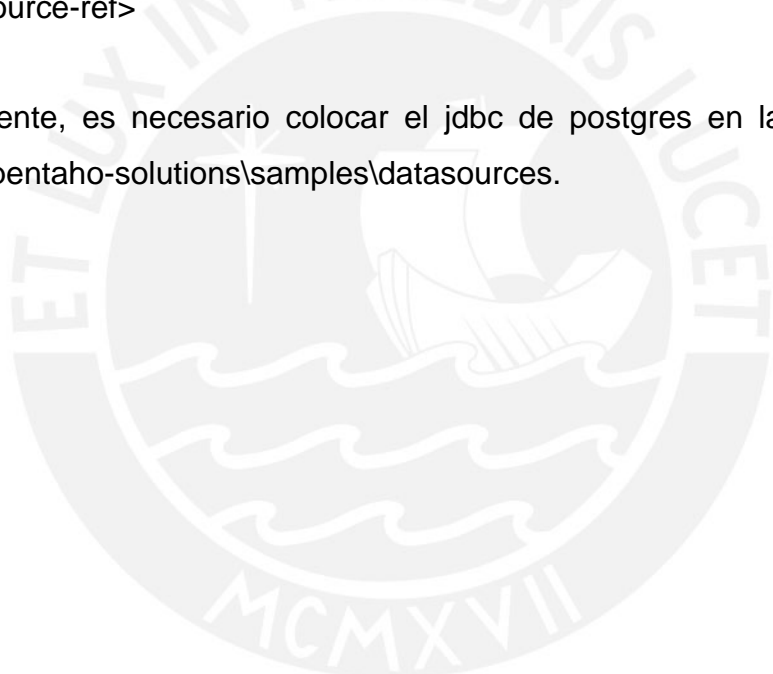
```
<resource-ref>
  <description>postgresql</description>
  <res-ref-name>jdbc/postgresql</res-ref-name>
  <res-type>javax.sql.DataSource</res-type>
  <res-auth>Container</res-auth>
</resource-ref>
```

Esto debe colocarse dentro del tag `<web-app>` `</web-app>`

Editar igualmente el archivo `pentaho-demo\jboss\server\default\deploy\pentaho.war\WEB-INF\jbiss-web.xml` para agregar la entrada:

```
<resource-ref>  
  <res-ref-name>jdbc/postgresql</res-ref-name>  
  <res-type>javax.sql.DataSource</res-type>  
  <jndi-name>java:/postgresql</jndi-name>  
</resource-ref>
```

Finalmente, es necesario colocar el jdbc de postgres en la ruta `pentaho-demo\pentaho-solutions\samples\datasources`.



Anexo L: Desarrollo de las Pruebas

A continuación se presentan los casos de pruebas de los procesos ETL. Debido a la extensión de las tablas, al presentar los datos de entrada se ha limitado la data con sólo aquellos registros necesarios.

Caso de prueba del informe “Ingresos por concepto”: Esta prueba tiene como objetivo verificar que las tablas para cargar el informe mencionado sean correctamente cargadas por medio del ETL, tomando como fuente la vista del Data Warehouse CAJA_DW, y que el informe “Ingresos por concepto” muestre la data correcta.

Datos de Entrada:

- Tabla Dia: Esta tabla posee una gran extensión por lo que sólo se mostrará un registro ejemplo

Id_dia	Fecha	Id_mes	Id_dia_semana
38730	2007-01-14	1273	2

- Tabla Documento_venta: Se listan las columnas significativas para el proceso y una porción del número de registros.

Id_documento_venta	Id_tipo_documento	total	fecha	Id_caja
1	1	300	2007-01-14	1
2	1	500	2007-01-14	1
3	1	250	2007-01-14	2
4	2	800	2007-01-14	1

- Tabla Linea_doc_venta: Se listan las columnas significativas para el proceso y una porción del número de registros.

Id_linea	Id_documento_venta	Total_pagado	Id_tipo_tarifa	Id_item_con
----------	--------------------	--------------	----------------	-------------

1	1	119	1	2
2	2	238	1	2
3	3	238	1	2
4	4	500	1	3
5	5	350	2	3

- Tabla Caja

Id_caja	D_caja	Codigo	Id_zona_hospital
1	Caja 1	CJ001	1
2	Caja 2	CJ002	1
3	Caja 3	CJ003	2

- Tabla Tipo_Documento:

Id_tipo_documento	D_tipo_documento	Codigo
1	Boleta	BOL
2	Factura	FAC

- Tabla Zona_hospital:

Id_zona_hospital	D_zona_hospital	Codigo
1	Piso11	Z001
2	Piso12	Z002

- Tabla Item_concepto: Se listan las columnas significativas para el proceso y una porción del número de registros

Id_item_concepto	Nombre	Id_tipo_item
1	Med1	1
2	Consuta	2

- Tabla D_Tipo_Documento:

Id_tipo_documento	D_tipo_documento	Codigo
-------------------	------------------	--------

1	Boleta	BOL
2	Factura	FAC

Resultados intermedios:

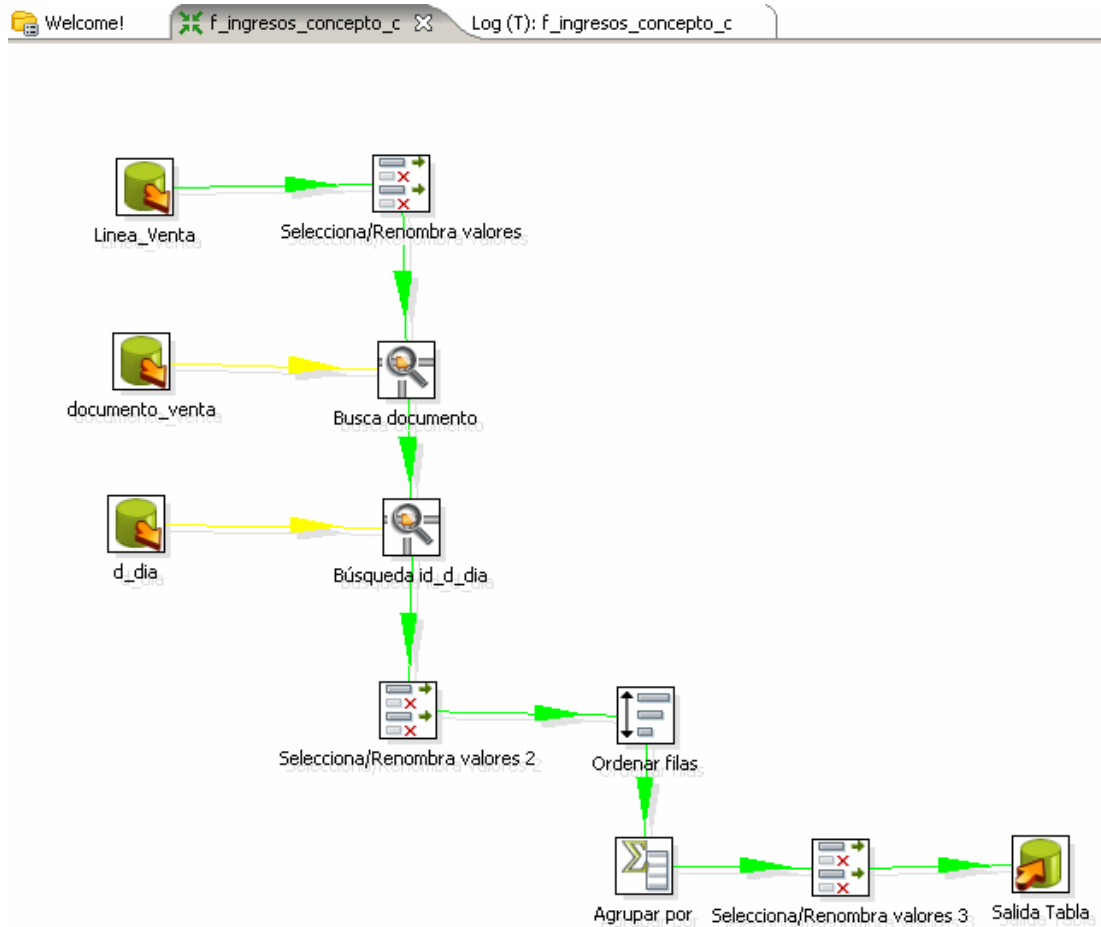
- Tabla f_ingresos_concepto

Id_d_dia	Id_caja	Id_d_tarif	Id_item	Id_serv	N_ingres	N_operac
38717	1	1	1	1	333	1
38717	1	1	3	1	800	1
38717	1	2	1	1	250	1
38717	2	1	2	1	238	1
38717	2	2	2	1	300	1
38718	1	1	3	1	500	1
38718	2	1	2	1	350	1
38718	2	2	3	1	350	1
38718	2	2	4	1	1600	2
39143	1	1	2	1	238	1
39146	1	1	2	1	119	1

Resultado Final:

Fecha	Servicio_medico	Caja	Tarifa	Tipo_item	Measures
*All_d_dia.fecha	*All_d_servicio_medico.d_d_servicio_med	*All_d_caja.d_d_caja	*All_d_tipo_tarifa.d_d_tipo_tarifa	*All_d_item_concepto.d_d_item_concepto	*SUM of f_ingresos_concepto.n_ingresos
				-All d_item_concepto.d_d_item_concepto	4978.0
				-Medicamentos	583.0
				Med1	583.0
				-Consultas	1145.0
				Consulta	1145.0
				-Inyectables	3250.0
				Med3	1650.0
				Med4	1600.0
			Tarifa1-Seg1	*All d_item_concepto.d_d_item_concepto	2478.0
			Tarifa2-Seg1	*All d_item_concepto.d_d_item_concepto	2500.0
*2007	*All_d_servicio_medico.d_d_servicio_med	*All_d_caja.d_d_caja	*All_d_tipo_tarifa.d_d_tipo_tarifa	*All d_item_concepto.d_d_item_concepto	4621.0
				-Medicamentos	583.0
				Med1	583.0
				-Consultas	788.0
				-Inyectables	3250.0
			Tarifa1-Seg1	*All d_item_concepto.d_d_item_concepto	2121.0
			Tarifa2-Seg1	*All d_item_concepto.d_d_item_concepto	2500.0
		Caja 1	*All d_tipo_tarifa.d_d_tipo_tarifa	*All d_item_concepto.d_d_item_concepto	1883.0
			Tarifa1-Seg1	*All d_item_concepto.d_d_item_concepto	1633.0
			Tarifa2-Seg1	*All d_item_concepto.d_d_item_concepto	250.0
		Caja2	*All d_tipo_tarifa.d_d_tipo_tarifa	*All d_item_concepto.d_d_item_concepto	2738.0
			Tarifa1-Seg1	*All d_item_concepto.d_d_item_concepto	488.0
			Tarifa2-Seg1	*All d_item_concepto.d_d_item_concepto	2250.0
	ServicioMedico1	*All_d_caja.d_d_caja	*All_d_tipo_tarifa.d_d_tipo_tarifa	*All d_item_concepto.d_d_item_concepto	4621.0
			Tarifa1-Seg1	*All d_item_concepto.d_d_item_concepto	2121.0
			Tarifa2-Seg1	*All d_item_concepto.d_d_item_concepto	2500.0

ETL:



En el proceso mostrado se llena la tabla `f_ingresos_concepto`, a partir de datos de las tablas `documento_venta`, `d_dia` y `linea_venta`. Los datos se agrupan para obtener la información de la forma adecuada según las llaves de la Fact a cargar.

Entradas: Se parte de los contenidos de las tablas `linea_venta` y `documento_venta` (del Data Warehouse) y `d_dia` (de la vista `GENERAL_DM`).

Salida Esperada: La Fact `f_ingresos_concepto` totalmente cargada, cumpliendo integridad referencial y con los valores correctos.

Ejecución y pruebas: Al ejecutar el Job, se presenta el siguiente diagrama:

#	Nombre paso	Numero Copia	Leído	Escrito	Entrada	Salida	Actualizado
1	Linea_Venta	0	0	12	12	0	0
2	Selecciona/Renombra valores	0	12	12	0	0	0
3	Busca documento	0	24	12	0	0	0
4	documento_venta	0	0	12	12	0	0
5	d_dia	0	0	45293	45293	0	0
6	Búsqueda id_d_dia	0	45305	12	0	0	0
7	Selecciona/Renombra valores 2	0	12	12	0	0	0
8	Ordenar filas	0	12	12	0	0	0
9	Agrupar por	0	12	11	0	0	0
10	Selecciona/Renombra valores 3	0	11	11	0	0	0
11	Salida Tabla	0	11	11	0	11	0

2008/09/27 11:15:46 - Selecciona/Renombra valores 2.0 - Iniciando ejecución...
 2008/09/27 11:15:46 - Ordenar filas.0 - Iniciando ejecución...
 2008/09/27 11:15:46 - Selecciona/Renombra valores 3.0 - Iniciando ejecución...
 2008/09/27 11:15:46 - Salida Tabla.0 - Iniciando ejecución...
 2008/09/27 11:15:46 - Agrupar por.0 - Iniciando ejecución...
 2008/09/27 11:15:46 - documento_venta.0 - Finished reading query, closing connection.
 2008/09/27 11:15:46 - documento_venta.0 - Procesamiento finalizado (I=12, O=0, R=0, W=12, U=0, E=0
 2008/09/27 11:15:46 - Linea_Venta.0 - Finished reading query, closing connection.
 2008/09/27 11:15:47 - Linea_Venta.0 - Procesamiento finalizado (I=12, O=0, R=0, W=12, U=0, E=0
 2008/09/27 11:15:47 - Selecciona/Renombra valores.0 - Procesamiento finalizado (I=0, O=0, R=12, W=12, U=0, E=0
 2008/09/27 11:15:47 - Busca documento.0 - Procesamiento finalizado (I=0, O=0, R=24, W=12, U=0, E=0
 2008/09/27 11:15:50 - d_dia.0 - Finished reading query, closing connection.
 2008/09/27 11:15:50 - d_dia.0 - Procesamiento finalizado (I=45293, O=0, R=0, W=45293, U=0, E=0
 2008/09/27 11:15:50 - Búsqueda id_d_dia.0 - Procesamiento finalizado (I=0, O=0, R=45305, W=12, U=0, E=0
 2008/09/27 11:15:50 - Selecciona/Renombra valores 2.0 - Procesamiento finalizado (I=0, O=0, R=12, W=12, U=0, E=0

Y al revisar los contenidos de la Fact encontramos lo siguiente:

	id_d_dia [PK] integer	id_caja [PK] integer	id_d_tip_tarifa [PK] integer	id_d_item_co [PK] integer	id_d_servicio [PK] integer	n_ingresos numeric	n_operacione integer
1	38717	1	1	1	1	333	1
2	38717	1	1	3	1	800	1
3	38717	1	2	1	1	250	1
4	38717	2	1	2	1	238	1
5	38717	2	2	2	1	300	1
6	38718	1	1	3	1	500	1
7	38718	2	1	2	1	250	1
8	38718	2	2	3	1	350	1
9	38718	2	2	4	1	1600	2
10	39143	1	1	2	1	238	1
11	39146	1	1	2	1	119	1

Se comprueba que la data es consistente.

Caso de prueba del informe “Operaciones e ingresos por tipos de tarifas y mes”: Esta prueba tiene como objetivo verificar que las tablas para cargar el informe mencionado sean correctamente cargadas por medio del ETL, tomando como fuente la vista del Data Warehouse CUENTAS_CORRIENTES_DW, y que el informe “Operaciones e ingresos por tipos de tarifas y mes” muestre la data correcta.

Datos de Entrada:

- Tabla Dia: Esta tabla posee una gran extensión por lo que sólo se mostrará un registro ejemplo

Id_dia	Fecha	Id_mes	Id_dia_semana
38730	2007-01-14	1273	2

- Tabla Cuenta: Se listan las columnas significativas para el proceso.

Id_cuenta	Id_paciente	Id_Estado	Id_tipo_tarifa
1	1	1	1
2	1	2	2
3	1	3	1
4	1	1	1

- Tabla Cuenta_item: Se listan las columnas significativas para el proceso y una porción del número de registros.

Cant	Id_tip_tarifa	Id_item_con	Id_cuenta	I_pagado	Id_linea	pagado
1	1	1	1	1	1	119
1	1	1	1	1	11	333
2	1	2	1	1	2	476
1	1	2	1	0	12	250
1	1	2	1	1	3	238

- Tabla Tipo Tarifa

Id_tipo_tarifa	nombre	D_tipo_tarifa	Codigo	Id_segmento
1	Tarifa1-1	Tarifa1-Segmento1	TT001	1
2	Tarifa1-2	Tarifa1-Segmento2	TT002	1

- Tabla Item_concepto:

Id_item_concepto	Nombre	Id_tipo_item
1	Med1	1
2	Consuta	2

Resultados intermedios:

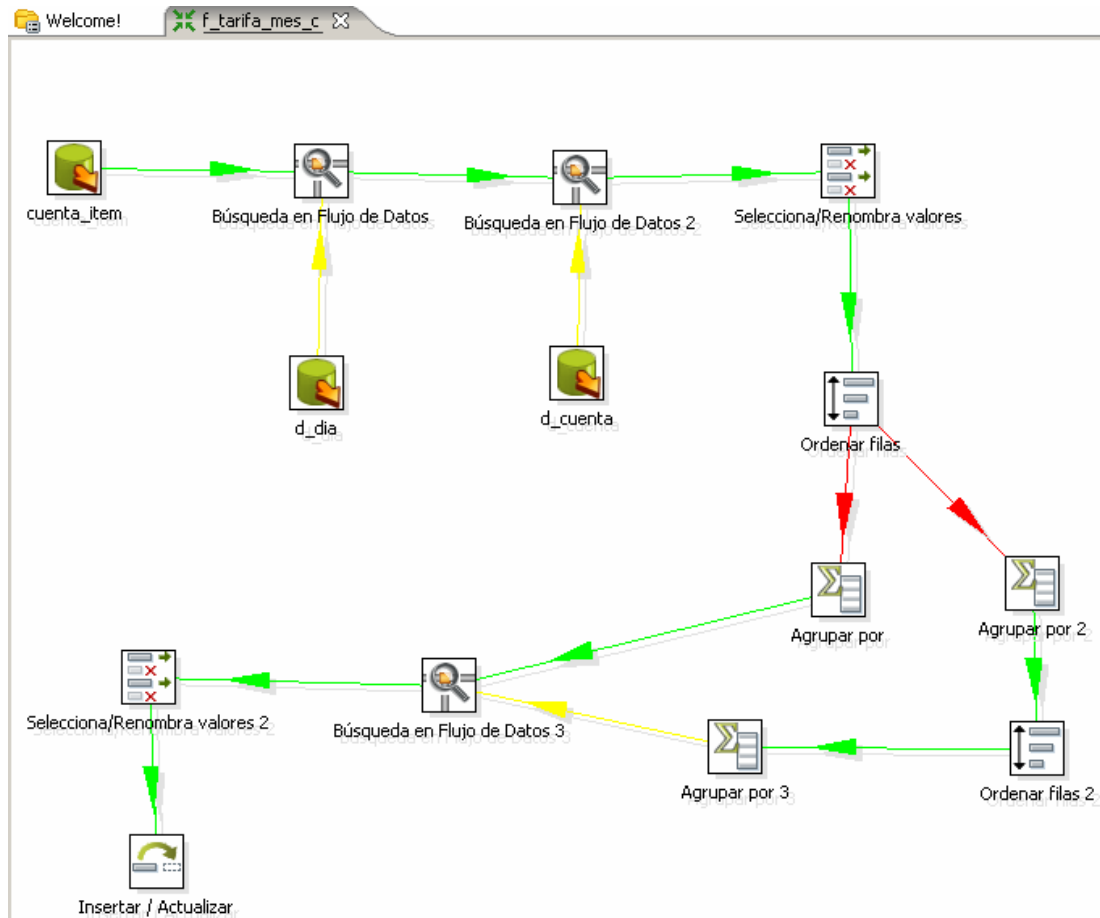
- Tabla f_ingresos_concepto

Ingresos	N_operaciones	Id_d_tarif	Id_d_mes	N_pacientes
2895	8	1	1273	1
2226	4	2	1273	1

Resultado Final:

tiempo	segmento	Measures		
		SUM of f_tarifa_mes.ingresos	SUM of f_tarifa_mes.n_operaciones	SUM of f_tarifa_mes.n_pacientes
-All tiempo	+All segmento.d_segmento.d_d_segmentos	5121.0	12.0	2.0
-2007	-All segmento.d_segmento.d_d_segmentos	5121.0	12.0	2.0
	+Segmento1	5121.0	12.0	2.0
ENERO 2007	-All segmento.d_segmento.d_d_segmentos	5121.0	12.0	2.0
	-Segmento1	5121.0	12.0	2.0
	Tarifa1-Seg1	2895.0	8.0	1.0
	Tarifa2-Seg1	2226.0	4.0	1.0

ETL



Se carga la tabla `f_tarifa_mes` a partir de las tablas `cuenta_item`, `cuenta` y `d_dia`. Es necesario realizar una agrupación adicional para hallar el número de pacientes según la tarifa y el mes.

Entradas: Se parte de los contenidos de las tablas `cuenta` y `cuenta_item` (del Data Warehouse) y `d_dia` (de la vista `GENERAL_DM`).

Salida Esperada: La Fact `f_tarifa_mes` totalmente cargada, cumpliendo integridad referencial y con los valores correctos.

Ejecución y pruebas: Al ejecutar el Job, se presenta el siguiente diagrama:

Welcome! f_tarifa_mes_c Log (T): f_tarifa_mes_c

#	Nombre paso	Numero Copia	Leído	Escrito	Entrada	Salida
1	cuenta_item	0	0	12	12	0
2	Búsqueda en Flujo de Datos	0	45305	12	0	0
3	d_dia	0	0	45293	45293	0
4	Búsqueda en Flujo de Datos 2	0	16	12	0	0
5	d_cuenta	0	0	4	4	0
6	Selecciona/Renombrar valores	0	12	12	0	0
7	Ordenar filas	0	12	24	0	0
8	Agrupar por	0	12	2	0	0
9	Agrupar por 2	0	12	2	0	0
10	Ordenar filas 2	0	2	2	0	0
11	Agrupar por 3	0	2	2	0	0
12	Búsqueda en Flujo de Datos 3	0	4	2	0	0
13	Selecciona/Renombrar valores 2	0	2	2	0	0
14	Insertar / Actualizar	0	2	2	2	0

2008/09/27 12:31:33 - Insertar / Actualizar.0 - Iniciando ejecución...
 2008/09/27 12:31:34 - d_cuenta.0 - Finished reading query, closing connection.
 2008/09/27 12:31:34 - cuenta_item.0 - Finished reading query, closing connection.
 2008/09/27 12:31:34 - cuenta_item.0 - Procesamiento finalizado (I=12, O=0, R=0, W=12, U=0, E=0
 2008/09/27 12:31:34 - d_cuenta.0 - Procesamiento finalizado (I=4, O=0, R=0, W=4, U=0, E=0
 2008/09/27 12:31:52 - d_dia.0 - Finished reading query, closing connection.
 2008/09/27 12:31:52 - d_dia.0 - Procesamiento finalizado (I=45293, O=0, R=0, W=45293, U=0, E=0
 2008/09/27 12:31:52 - Búsqueda en Flujo de Datos.0 - Procesamiento finalizado (I=0, O=0, R=45305, W=12, U=0, E=0
 2008/09/27 12:31:52 - Búsqueda en Flujo de Datos 2.0 - Procesamiento finalizado (I=0, O=0, R=16, W=12, U=0, E=0
 2008/09/27 12:31:52 - Selecciona/Renombrar valores.0 - Procesamiento finalizado (I=0, O=0, R=12, W=12, U=0, E=0
 2008/09/27 12:31:52 - Ordenar filas.0 - Procesamiento finalizado (I=0, O=0, R=12, W=24, U=0, E=0
 2008/09/27 12:31:52 - Agrupar por 2.0 - Procesamiento finalizado (I=0, O=0, R=12, W=2, U=0, E=0
 2008/09/27 12:31:52 - Agrupar por.0 - Procesamiento finalizado (I=0, O=0, R=12, W=2, U=0, E=0
 2008/09/27 12:31:52 - Ordenar filas 2.0 - Procesamiento finalizado (I=0, O=0, R=2, W=2, U=0, E=0
 2008/09/27 12:31:53 - Agrupar por 3.0 - Procesamiento finalizado (I=0, O=0, R=2, W=2, U=0, E=0
 2008/09/27 12:31:53 - Búsqueda en Flujo de Datos 3.0 - Procesamiento finalizado (I=0, O=0, R=4, W=2, U=0, E=0
 2008/09/27 12:31:53 - Selecciona/Renombrar valores 2.0 - Procesamiento finalizado (I=0, O=0, R=2, W=2, U=0, E=0
 2008/09/27 12:31:53 - Insertar / Actualizar.0 - Procesamiento finalizado (I=2, O=0, R=2, W=2, U=0, E=0
 2008/09/27 12:31:53 - Spoon - La transformación ha finalizado!!

Y finalmente, al revisar los contenidos de la Fact se comprueba que los datos son consistentes.

	ingresos numerico	n_operacione integer	id_d_tip_tarif [PK] integer	id_d_mes [PK] integer	n_pacientes integer
1	2895	8	1	1273	1
2	2226	4	2	1273	1
*					

Caso de prueba del informe “Deudas e ingresos por cuenta, pacientes y servicio médico”: Esta prueba tiene como objetivo verificar que las tablas para cargar el informe mencionado sean correctamente cargadas por medio del ETL, tomando como fuente la vista del Data Warehouse CUENTAS_CORRIENTES_DW, y que el informe “Deudas e ingresos por cuenta, pacientes y servicio médico” muestre la data correcta.

Datos de Entrada:

- Tabla Dia: Esta tabla posee una gran extensión por lo que sólo se mostrará un registro ejemplo

Id_dia	Fecha	Id_mes	Id_dia_semana
38730	2007-01-14	1273	2

- Tabla Cuenta: Se listan las columnas significativas para el proceso.

Id_cuenta	Id_paciente	Id_estado	Id_tipo_tarifa
1	1	1	1
2	1	2	2
3	1	3	1
4	1	1	1

- Tabla Cuenta_item: Se listan las columnas significativas para el proceso y una porción del número de registros.

Cant	Id_tip_tarifa	Id_item_con	Id_cuenta	I_pagado	Id_linea	pagado
1	1	1	1	1	1	119
1	1	1	1	1	11	333
2	1	2	1	1	2	476
1	1	2	1	0	12	250
1	1	2	1	1	3	238

- Tabla Tipo Tarifa

Id_tipo_tarifa	nombre	D_tipo_tarifa	Codigo	Id_segmento
1	Tarifa1-1	Tarifa1- Segmento1	TT001	1
2	Tarifa1-2	Tarifa1- Segmento2	TT002	1

- Tabla Item_concepto:

Id_item_concepto	Nombre	Id_tipo_item
1	Med1	1
2	Consulta	2

- Tabla Documento_venta: Se listan las columnas significativas para el proceso y una porción del número de registros.

Id_documento_venta	Id_tipo_documento	total	fecha	Id_caja
1	1	300	2007-01-14	1
2	1	500	2007-01-14	1
3	1	250	2007-01-14	2
4	2	800	2007-01-14	1

- Tabla Linea_doc_venta: Se listan las columnas significativas para el proceso y una porción del número de registros.

Id_linea	Id_documento_venta	Total_pagado	Id_tipo_tarifa	Id_item_con
1	1	119	1	2
2	2	238	1	2
3	3	238	1	2
4	4	500	1	3
5	5	350	2	3

Resultados intermedios:

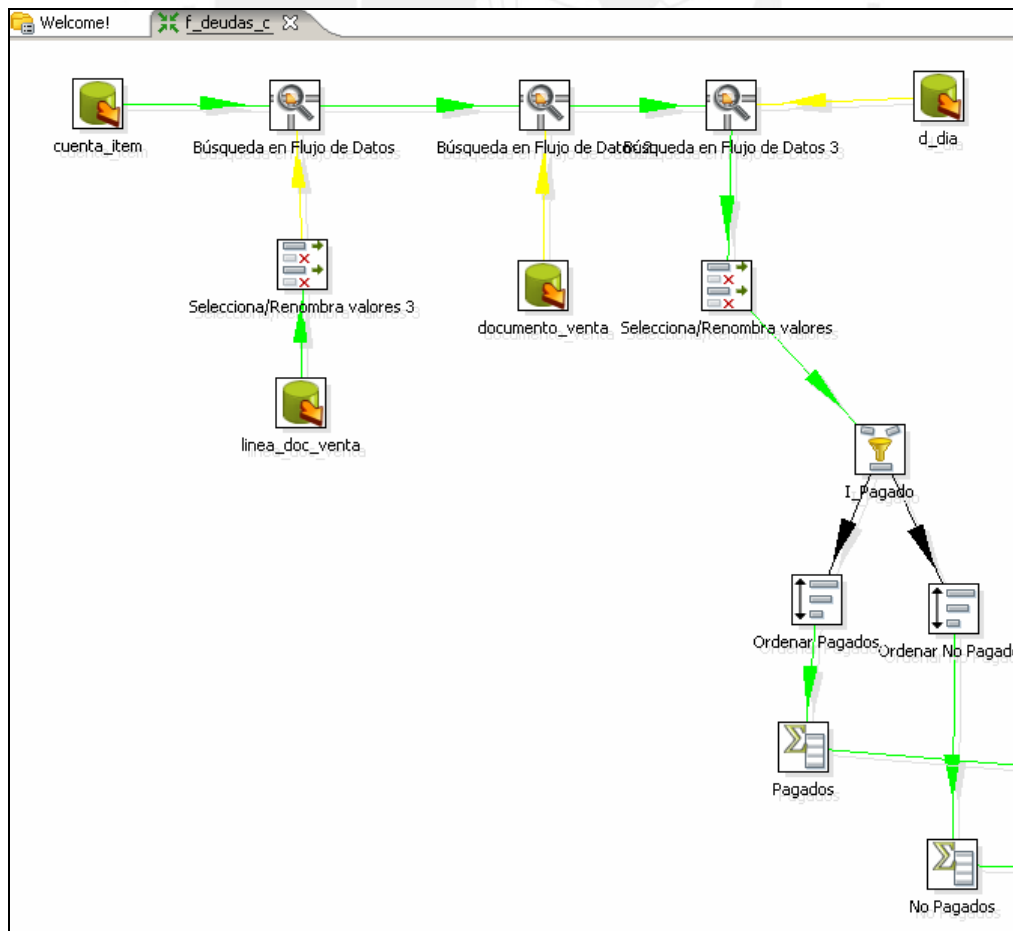
- Tabla f_deudas

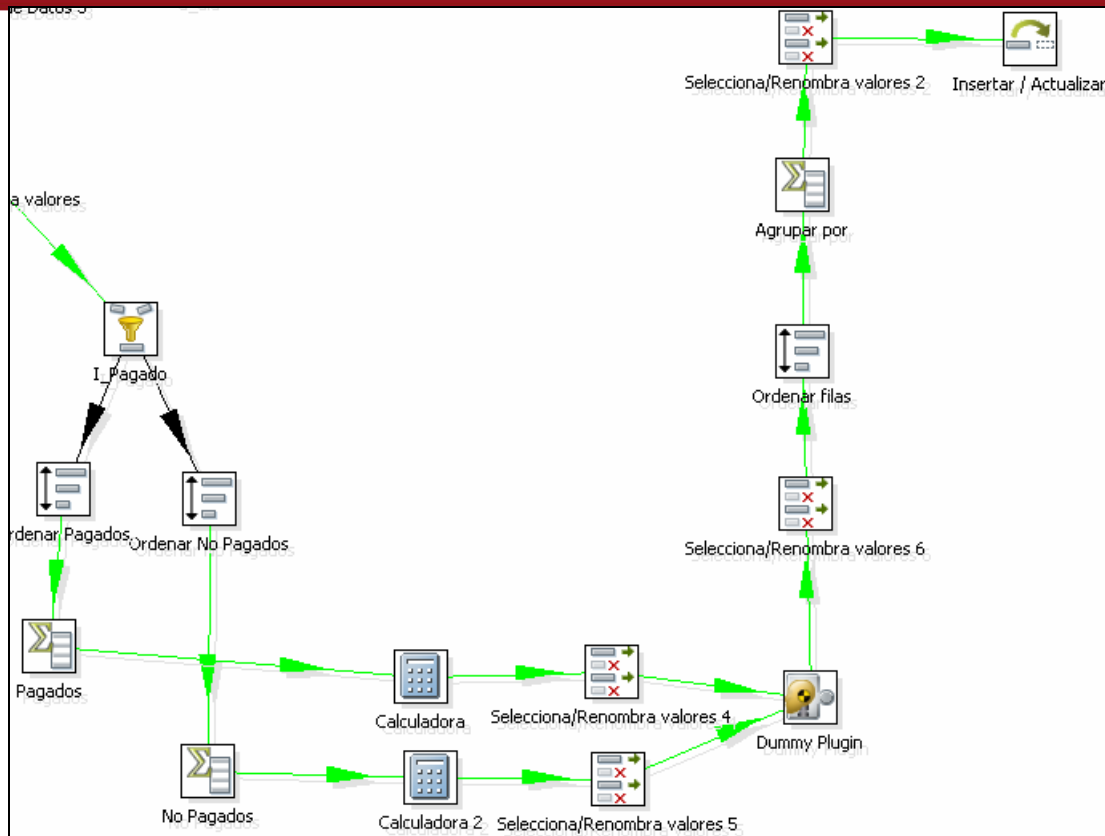
Id_d_mes	Monto_cancelado	Monto_pendiente	Id_serv	Id_cuenta
1273	3278	1700	1	1

Resultado Final:

			Measures
tiempo	cuenta	servicio_med	• SUM of f_deudas.monto_cancelado
-All mes	-All cuenta	-All servicio_med	3278.0
		ServicioMedico1	3278.0
	0000001	-All servicio_med	3278.0
		ServicioMedico1	3278.0
+2007	+All cuenta	+All servicio_med	3278.0

ETL:





Este proceso ETL extrae de las tablas documento_venta, linea_doc_venta, cuenta, cuenta_item y d_dia para cargar la Fact. Las filas pagadas y no pagadas se procesan de una forma distinta y es por eso que se separan en el ETL. Cuando se llega al final del proceso, todas estas filas se agrupan por las llaves de la tabla f_deudas.

Entradas: Se parte de los contenidos de las tablas documento_venta, linea_doc_venta, cuenta y cuenta_item (del Data Warehouse) y d_dia (de la vista GENERAL_DM).

Salida Esperada: La Fact f_deudas totalmente cargada, cumpliendo integridad referencial y con los valores correctos.

Ejecución y pruebas: Al ejecutar el Job, se presenta el siguiente diagrama:

Welcome! F_deudas_c Log (T): f_deudas_c

#	Nombre paso	Numero Copia	Leído	Escrito	Entrada	Salida	Actualizado
1	Búsqueda en Flujo de Datos	0	24	12	0	0	0
2	Búsqueda en Flujo de Datos 2	0	24	12	0	0	0
3	I_Pagado	0	12	12	0	0	0
4	Ordenar Pagados	0	8	8	0	0	0
5	Ordenar No Pagados	0	4	4	0	0	0
6	Pagados	0	8	1	0	0	0
7	No Pagados	0	4	1	0	0	0
8	Selecciona/Renombra valores	0	12	12	0	0	0
9	Calculadora	0	1	1	0	0	0
10	Selecciona/Renombra valores 4	0	1	1	0	0	0
11	Calculadora 2	0	1	1	0	0	0
12	Selecciona/Renombra valores 5	0	1	1	0	0	0
13	Dummy Plugin	0	2	2	0	0	0
14	Agrupar por	0	2	1	0	0	0
15	Selecciona/Renombra valores 2	0	1	1	0	0	0
16	Ordenar filas	0	2	2	0	0	0
17	Selecciona/Renombra valores 6	0	2	2	0	0	0
18	Insertar / Actualizar	0	1	1	1	0	0

2008/09/27 12:45:52 - d_dia.0 - Procesamiento finalizado (I=45293, O=0, R=0, W=45293, U=0, E=0)
 2008/09/27 12:45:52 - Búsqueda en Flujo de Datos 3.0 - Procesamiento finalizado (I=0, O=0, R=45305, W=12, U=0, E=0)
 2008/09/27 12:45:52 - Selecciona/Renombra valores.0 - Procesamiento finalizado (I=0, O=0, R=12, W=12, U=0, E=0)
 2008/09/27 12:45:52 - I_Pagado.0 - Procesamiento finalizado (I=0, O=0, R=12, W=12, U=0, E=0)
 2008/09/27 12:45:52 - Ordenar Pagados.0 - Procesamiento finalizado (I=0, O=0, R=8, W=8, U=0, E=0)
 2008/09/27 12:45:52 - Ordenar No Pagados.0 - Procesamiento finalizado (I=0, O=0, R=4, W=4, U=0, E=0)
 2008/09/27 12:45:52 - Pagados.0 - Procesamiento finalizado (I=0, O=0, R=8, W=1, U=0, E=0)
 2008/09/27 12:45:52 - No Pagados.0 - Procesamiento finalizado (I=0, O=0, R=4, W=1, U=0, E=0)
 2008/09/27 12:45:52 - Calculadora.0 - Procesamiento finalizado (I=0, O=0, R=1, W=1, U=0, E=0)
 2008/09/27 12:45:52 - Calculadora 2.0 - Procesamiento finalizado (I=0, O=0, R=1, W=1, U=0, E=0)
 2008/09/27 12:45:52 - Selecciona/Renombra valores 4.0 - Procesamiento finalizado (I=0, O=0, R=1, W=1, U=0, E=0)
 2008/09/27 12:45:52 - Selecciona/Renombra valores 5.0 - Procesamiento finalizado (I=0, O=0, R=1, W=1, U=0, E=0)
 2008/09/27 12:45:52 - Dummy Plugin.0 - Finished, processing 2 rows
 2008/09/27 12:45:52 - Selecciona/Renombra valores 6.0 - Procesamiento finalizado (I=0, O=0, R=2, W=2, U=0, E=0)
 2008/09/27 12:45:52 - Ordenar filas.0 - Procesamiento finalizado (I=0, O=0, R=2, W=2, U=0, E=0)
 2008/09/27 12:45:52 - Agrupar por.0 - Procesamiento finalizado (I=0, O=0, R=2, W=1, U=0, E=0)
 2008/09/27 12:45:52 - Selecciona/Renombra valores 2.0 - Procesamiento finalizado (I=0, O=0, R=1, W=1, U=0, E=0)
 2008/09/27 12:45:52 - Insertar / Actualizar.0 - Procesamiento finalizado (I=1, O=0, R=1, W=1, U=0, E=0)
 2008/09/27 12:45:52 - Spoon - La transformación ha finalizado!!

Y finalmente, al revisar los contenidos de la Fact se comprueba que los datos son consistentes.

	id_d_mes [PK] integer	monto_cance double precis	monto_pendi double precis	id_d_servicio [PK] integer	id_d_cuenta [PK] integer
1	1273	3278	1700	1	1
*					

Caso de prueba del informe “Ingresos por cuentas corrientes”: Esta prueba tiene como objetivo verificar que las tablas para cargar el informe mencionado sean correctamente cargadas por medio del ETL, tomando como fuente la vista del Data Warehouse CUENTAS_CORRIENTES_DW, y que el informe “Ingresos por cuentas corrientes” muestre la data correcta.

Datos de Entrada:

- Tabla Dia: Esta tabla posee una gran extensión por lo que sólo se mostrará un registro ejemplo

Id_dia	Fecha	Id_mes	Id_dia_semana
38730	2007-01-14	1273	2

- Tabla Cuenta: Se listan las columnas significativas para el proceso.

Id_cuenta	Id_paciente	Id_Estado	Id_tipo_tarifa
1	1	1	1
2	1	2	2
3	1	3	1
4	1	1	1

- Tabla Cuenta_item: Se listan las columnas significativas para el proceso y una porción del número de registros.

Cant	Id_tip_tarifa	Id_item_con	Id_cuenta	l_pagado	Id_linea	pagado
1	1	1	1	1	1	119
1	1	1	1	1	11	333
2	1	2	1	1	2	476
1	1	2	1	0	12	250
1	1	2	1	1	3	238

- Tabla Tipo Tarifa

Id_tipo_tarifa	nombre	D_tipo_tarifa	Codigo	Id_segmento
1	Tarifa1-1	Tarifa1- Segmento1	TT001	1
2	Tarifa1-2	Tarifa1- Segmento2	TT002	1

- Tabla Item_concepto:

Id_item_concepto	Nombre	Id_tipo_item
1	Med1	1
2	Consulta	2

- Tabla Documento_venta: Se listan las columnas significativas para el proceso y una porción del número de registros.

Id_documento_venta	Id_tipo_documento	total	fecha	Id_caja
1	1	300	2007-01-14	1
2	1	500	2007-01-14	1
3	1	250	2007-01-14	2
4	2	800	2007-01-14	1

- Tabla Linea_doc_venta: Se listan las columnas significativas para el proceso y una porción del número de registros.

Id_linea	Id_documento_venta	Total_pagado	Id_tipo_tarifa	Id_item_con
1	1	119	1	2
2	2	238	1	2
3	3	238	1	2
4	4	500	1	3
5	5	350	2	3

Resultados intermedios:

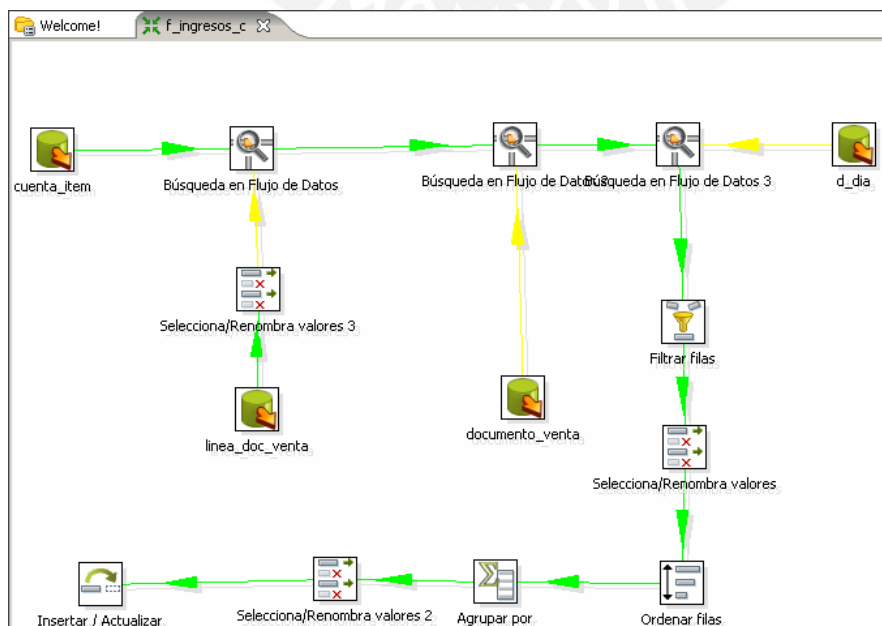
- Tabla f_deudas

Id_d_dia	Id_item_concep	Id_tip_tarif	ingresos	Id_d_Serv	operaciones
38717	1	1	119	1	1
38717	1	2	238	1	1
38717	2	1	476	1	1
38717	3	1	357	1	1
38717	4	2	646	1	1
38718	1	1	333	1	1
38718	4	2	1000	1	1
38731	2	1	238	1	1

Resultado Final:

tiempo	tipo_tarifa	item_concepto	servicio_medico	Measures	
-All tiempo	-All tipo_tarifa	-All item_concepto	-All servicio_medico	SUM of f_ingresos.ingresos	SUM of f_ingresos.n_operaciones
			ServicioMedico1	3407.0	8.0
		+Consulta	+All servicio_medico	714.0	2.0
		+Med1	+All servicio_medico	690.0	3.0
		+Med3	+All servicio_medico	357.0	1.0
		+Med4	+All servicio_medico	1646.0	2.0
	+Tarifa1-Seg1	+All item_concepto	+All servicio_medico	1523.0	5.0
	+Tarifa2-Seg1	+All item_concepto	+All servicio_medico	1884.0	3.0
+2007	+All tipo_tarifa	+All item_concepto	+All servicio_medico	3407.0	8.0

ETL:



Este proceso ETL extrae de las tablas documento_venta, linea_doc_venta, cuenta, cuenta_item y d_dia para cargar la Fact. Es importante mencionar que para este proceso sólo se trabaja con los ítem de cuentas pagados.

Entradas: Se parte de los contenidos de las tablas documento_venta, linea_doc_venta, cuenta y cuenta_item (del Data Warehouse) y d_dia (de la vista GENERAL_DM).

Salida Esperada: La Fact f_ingresos totalmente cargada, cumpliendo integridad referencial y con los valores correctos.

Ejecución y pruebas: Al ejecutar el Job, se presenta el siguiente diagrama:

#	Nombre paso	Numero Copia	Leído	Escrito	Entrada	Salida
1	Agrupar por	0	8	8	0	0
2	Selecciona/Renombra valores 2	0	8	8	0	0
3	Búsqueda en Flujo de Datos	0	24	12	0	0
4	Búsqueda en Flujo de Datos 2	0	24	12	0	0
5	Búsqueda en Flujo de Datos 3	0	45305	12	0	0
6	Filtrar filas	0	12	8	0	0
7	Selecciona/Renombra valores	0	8	8	0	0
8	Ordenar filas	0	8	8	0	0
9	Insertar / Actualizar	0	8	8	8	0
10	Selecciona/Renombra valores 3	0	12	12	0	0
11	cuenta_item	0	0	12	12	0
12	d_dia	0	0	45293	45293	0
13	documento_venta	0	0	12	12	0
14	linea_doc_venta	0	0	12	12	0

```

2008/09/27 14:08:50 - Búsqueda en Flujo de Datos.0 - Leyendo valores de búsqueda del paso [Selecciona/Renombra valores 3]
2008/09/27 14:08:50 - Selecciona/Renombra valores.0 - Iniciando ejecución...
2008/09/27 14:08:50 - Búsqueda en Flujo de Datos 3.0 - Leyendo valores de búsqueda del paso [d_dia]
2008/09/27 14:08:50 - linea_doc_venta.0 - Finished reading query, closing connection.
2008/09/27 14:08:50 - linea_doc_venta.0 - Procesamiento finalizado (I=12, O=0, R=0, W=12, U=0, E=0
2008/09/27 14:08:51 - documento_venta.0 - Finished reading query, closing connection.
2008/09/27 14:08:51 - documento_venta.0 - Procesamiento finalizado (I=12, O=0, R=0, W=12, U=0, E=0
2008/09/27 14:08:51 - Selecciona/Renombra valores 3.0 - Procesamiento finalizado (I=0, O=0, R=12, W=12, U=0, E=0
2008/09/27 14:08:51 - Búsqueda en Flujo de Datos.0 - Procesamiento finalizado (I=0, O=0, R=24, W=12, U=0, E=0
2008/09/27 14:08:51 - Búsqueda en Flujo de Datos 2.0 - Procesamiento finalizado (I=0, O=0, R=24, W=12, U=0, E=0
2008/09/27 14:08:57 - d_dia.0 - Finished reading query, closing connection.
2008/09/27 14:08:57 - d_dia.0 - Procesamiento finalizado (I=45293, O=0, R=0, W=45293, U=0, E=0
2008/09/27 14:08:57 - Búsqueda en Flujo de Datos 3.0 - Procesamiento finalizado (I=0, O=0, R=45305, W=12, U=0, E=0
2008/09/27 14:08:57 - Filtrar filas.0 - Procesamiento finalizado (I=0, O=0, R=12, W=8, U=0, E=0
2008/09/27 14:08:57 - Selecciona/Renombra valores.0 - Procesamiento finalizado (I=0, O=0, R=8, W=8, U=0, E=0
2008/09/27 14:08:57 - Ordenar filas.0 - Procesamiento finalizado (I=0, O=0, R=8, W=8, U=0, E=0
2008/09/27 14:08:57 - Agrupar por.0 - Procesamiento finalizado (I=0, O=0, R=8, W=8, U=0, E=0
2008/09/27 14:08:57 - Selecciona/Renombra valores 2.0 - Procesamiento finalizado (I=0, O=0, R=8, W=8, U=0, E=0
2008/09/27 14:08:57 - Insertar / Actualizar.0 - Procesamiento finalizado (I=8, O=0, R=8, W=8, U=0, E=0
2008/09/27 14:08:57 - Spoon - La transformación ha finalizado!!
    
```

Y finalmente, al revisar los contenidos de la Fact se comprueba que los datos son consistentes.

	id_d_dia [PK] integer	id_d_item_co [PK] integer	id_d_tip_taril [PK] integer	ingresos numeric	id_d_servicio [PK] integer	n_operacione integer
1	38717	1	1	119	1	1
2	38717	1	2	238	1	1
3	38717	2	1	476	1	1
4	38717	3	1	357	1	1
5	38717	4	2	646	1	1
6	38718	1	1	333	1	1
7	38718	4	2	1000	1	1
8	38731	2	1	238	1	1
*						



Caso de prueba del informe “Pacientes intervenidos por servicio médico”: Esta prueba tiene como objetivo verificar que las tablas para cargar el informe mencionado sean correctamente cargadas por medio del ETL, tomando como fuente la vistas del Data Warehouse CITA_DW, y HISTORIA_CLINICA_DW y que el informe “Pacientes intervenidos por servicio médico” muestre la data correcta.

Datos de Entrada:

- Tabla Dia: Esta tabla posee una gran extensión por lo que sólo se mostrará un registro ejemplo

Id_dia	Fecha	Id_mes	Id_dia_semana
38730	2007-01-14	1273	2

- Tabla Intervención: Se listan las columnas significativas para el proceso.

Id_interv	Id_tip_aten	Id_genera_hosp	Id_paciente	Id_epi	Id_turno	I_fallecim
1	1	0	1	1	4	0
2	1	1	1	2	1	0

- Tabla Episodio: Se listan las columnas significativas para el proceso.

Id_epi	Id_hora	Id_pac	Id_turno	Id_serv	id_cuenta
1	20	1	4	1	1
1	8	1	1	1	1

- Tabla Turno: Se listan las columnas significativas para el proceso.

Id_turno	Id_personal	Id_horario	Id_serv	Id_zona_trab
1	5	1	1	1
2	6	2	2	1
3	7	3	3	2

4	5	3	1	1
---	---	---	---	---

- Tabla personalxintervencion:

Id_persona	Id_interven	Id_dia_sem	Id_turno	I_principal
5	1	5	1	1
5	2	6	1	1

- Tabla Personal: Se listan las columnas significativas para el proceso y una porción del número de registros.

Id_personal	Id_i_medico	sueldo
5	1	3000

Resultados intermedios:

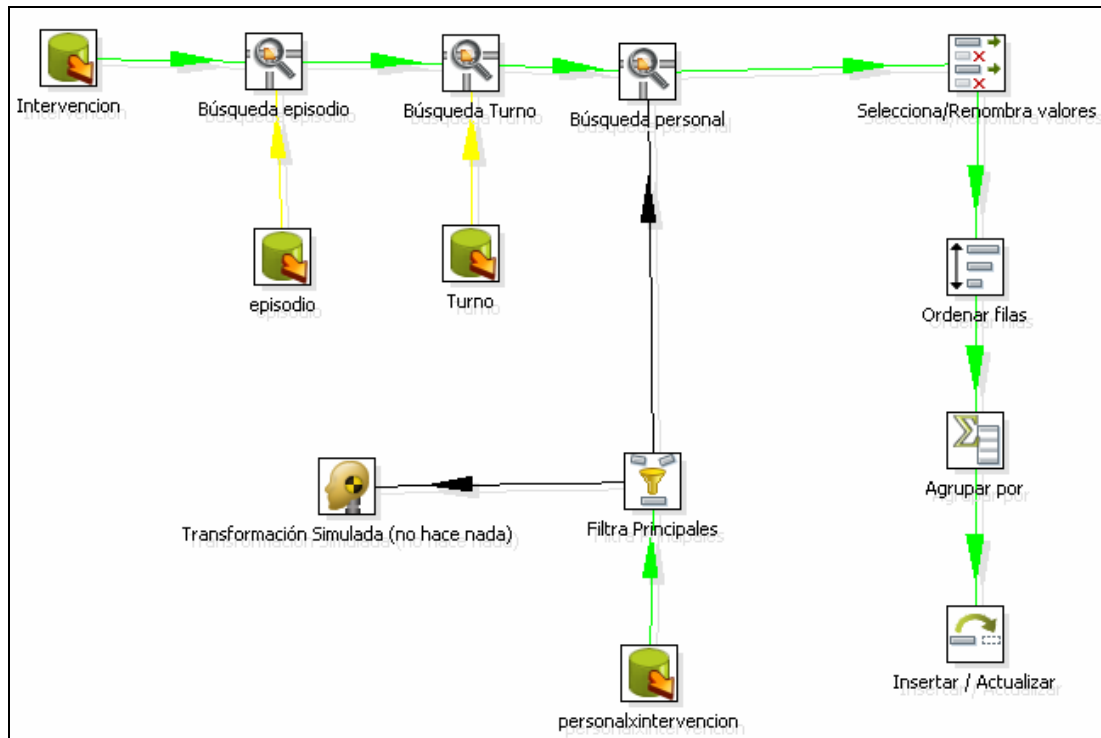
- Tabla f_pacientes_interv

Id_d_dia	Id_d_Servicio_med	Id_d_horar	Id_personal	Id_tipo_interv	N_pac	N_falle	N_hospitalizados
39143	1	3	5	1	1	0	0
39146	1	1	5	1	1	0	1

Resultado Final:

					Measures			
Dia	Serv_medico	Horario	Personal	Tipo_intervencion	N_fallecimientos	N_hosp	N_pacientes	Ratio_Fallecimientos
All dia	All serv_medico	All horario	All personal	All tipo_intervencion	.0	1.0	2.0	.00%
				Emergencia	.0	1.0	2.0	.00%
			PER00005	All tipo_intervencion	.0	1.0	2.0	.00%
		Horario Mañana	All personal	All tipo_intervencion	.0	1.0	1.0	.00%
		Horario Noche	All personal	All tipo_intervencion	.0	.0	1.0	.00%
2008	All serv_medico	All horario	All personal	All tipo_intervencion	.0	1.0	2.0	.00%
				All tipo_intervencion	.0	1.0	2.0	.00%

ETL:



Este proceso ETL extrae de las tablas intervencion, episodio, personalxintervencion, y turno para cargar la Fact.

Entradas: Se parte de los contenidos de las tablas intervencion, episodio, personalxintervencion (de la vista historia_clinica_dw) y turno (de la vista cita_dw).

Salida Esperada: La Fact f_pacientes_interv totalmente cargada, cumpliendo integridad referencial y con los valores correctos.

Ejecución y pruebas: Al ejecutar el Job, se presenta el siguiente diagrama:

#	Nombre paso	Numero Copia	Leído	Escrito	Entrada	Salida
1	episodio	0	0	2	2	
2	Búsqueda episodio	0	4	2	0	
3	Filtra Principales	0	2	2	0	
4	Búsqueda personal	0	4	2	0	
5	Transformación Simulada (no hace nada)	0	0	0	0	
6	personalxintervencion	0	0	2	2	
7	Selecciona/Renombra valores	0	2	2	0	
8	Ordenar filas	0	2	2	0	
9	Agrupar por	0	2	2	0	
10	Insertar / Actualizar	0	2	2	2	
11	Intervencion	0	0	2	2	
12	Búsqueda Turno	0	6	2	0	
13	Turno	0	0	4	4	

2008/09/27 16:57:34 - episodio.0 - Finished reading query, closing connection.
 2008/09/27 16:57:34 - Turno.0 - Iniciando ejecución...
 2008/09/27 16:57:34 - personalxintervencion.0 - Finished reading query, closing connection.
 2008/09/27 16:57:34 - episodio.0 - Procesamiento finalizado (I=2, O=0, R=0, W=2, U=0, E=0
 2008/09/27 16:57:34 - personalxintervencion.0 - Procesamiento finalizado (I=2, O=0, R=0, W=2, U=0, E=0
 2008/09/27 16:57:34 - Intervencion.0 - Finished reading query, closing connection.
 2008/09/27 16:57:34 - Turno.0 - Finished reading query, closing connection.
 2008/09/27 16:57:34 - Filtra Principales.0 - Procesamiento finalizado (I=0, O=0, R=2, W=2, U=0, E=0
 2008/09/27 16:57:34 - Intervencion.0 - Procesamiento finalizado (I=2, O=0, R=0, W=2, U=0, E=0
 2008/09/27 16:57:34 - Turno.0 - Procesamiento finalizado (I=4, O=0, R=0, W=4, U=0, E=0
 2008/09/27 16:57:34 - Transformación Simulada (no hace nada).0 - Procesamiento finalizado (I=0, O=0, R=0, W=0, U=0, E=0
 2008/09/27 16:57:34 - Búsqueda episodio.0 - Procesamiento finalizado (I=0, O=0, R=4, W=2, U=0, E=0
 2008/09/27 16:57:34 - Búsqueda Turno.0 - Procesamiento finalizado (I=0, O=0, R=6, W=2, U=0, E=0
 2008/09/27 16:57:34 - Búsqueda personal.0 - Procesamiento finalizado (I=0, O=0, R=4, W=2, U=0, E=0
 2008/09/27 16:57:34 - Selecciona/Renombra valores.0 - Procesamiento finalizado (I=0, O=0, R=2, W=2, U=0, E=0
 2008/09/27 16:57:34 - Ordenar filas.0 - Procesamiento finalizado (I=0, O=0, R=2, W=2, U=0, E=0
 2008/09/27 16:57:34 - Agrupar por.0 - Procesamiento finalizado (I=0, O=0, R=2, W=2, U=0, E=0
 2008/09/27 16:57:34 - Insertar / Actualizar.0 - Procesamiento finalizado (I=2, O=0, R=2, W=2, U=0, E=0
 2008/09/27 16:57:34 - Spoon - La transformación ha finalizado!!

Y finalmente, al revisar los contenidos de la Fact se comprueba que los datos son consistentes.

	id_d_dia [PK] integer	id_d_servicio [PK] integer	id_d_horario [PK] integer	id_persona [PK] integer	id_d_tipo_int [PK] integer	n_pacientes integer	n_fallecimien integer	n_hosp integer
1	39143	1	3	5	1	1	0	0
2	39146	1	1	5	1	1	0	1
*								

Caso de prueba del informe “Atenciones a pacientes”: Esta prueba tiene como objetivo verificar que las tablas para cargar el informe mencionado sean correctamente cargadas por medio del ETL, tomando como fuente la vistas del Data Warehouse CITA_DW, y HISTORIA_CLINICA_DW y que el informe “Atenciones a pacientes” muestre la data correcta.

Datos de Entrada:

- Tabla Dia: Esta tabla posee una gran extensión por lo que sólo se mostrará un registro ejemplo

Id_dia	Fecha	Id_mes	Id_dia_semana
38730	2007-01-14	1273	2

- Tabla Intervención: Se listan las columnas significativas para el proceso.

Id_interv	Id_tip_aten	Id_genera_hosp	Id_paciente	Id_epi	Id_turno	I_fallecim
1	1	0	1	1	4	0
2	1	1	1	2	1	0

- Tabla Episodio: Se listan las columnas significativas para el proceso.

Id_epi	Id_hora	Id_pac	Id_turno	Id_serv	id_cuenta
1	20	1	4	1	1
1	8	1	1	1	1

- Tabla Turno: Se listan las columnas significativas para el proceso.

Id_turno	Id_personal	Id_horario	Id_serv	Id_zona_trab
1	5	1	1	1
2	6	2	2	1
3	7	3	3	2
4	5	3	1	1

- Tabla personalxintervencion:

Id_persona	Id_interven	Id_dia_sem	Id_turno	I_principal
5	1	5	1	1
5	2	6	1	1

- Tabla Personal: Se listan las columnas significativas para el proceso y una porción del número de registros.

Id_personal	Id_i_medico	sueldo
5	1	3000

- Tabla Documento_venta: Se listan las columnas significativas para el proceso y una porción del número de registros.

Id_documento_venta	Id_tipo_documento	total	fecha	Id_caja
1	1	300	2007-01-14	1
2	1	500	2007-01-14	1
3	1	250	2007-01-14	2
4	2	800	2007-01-14	1

Resultados intermedios:

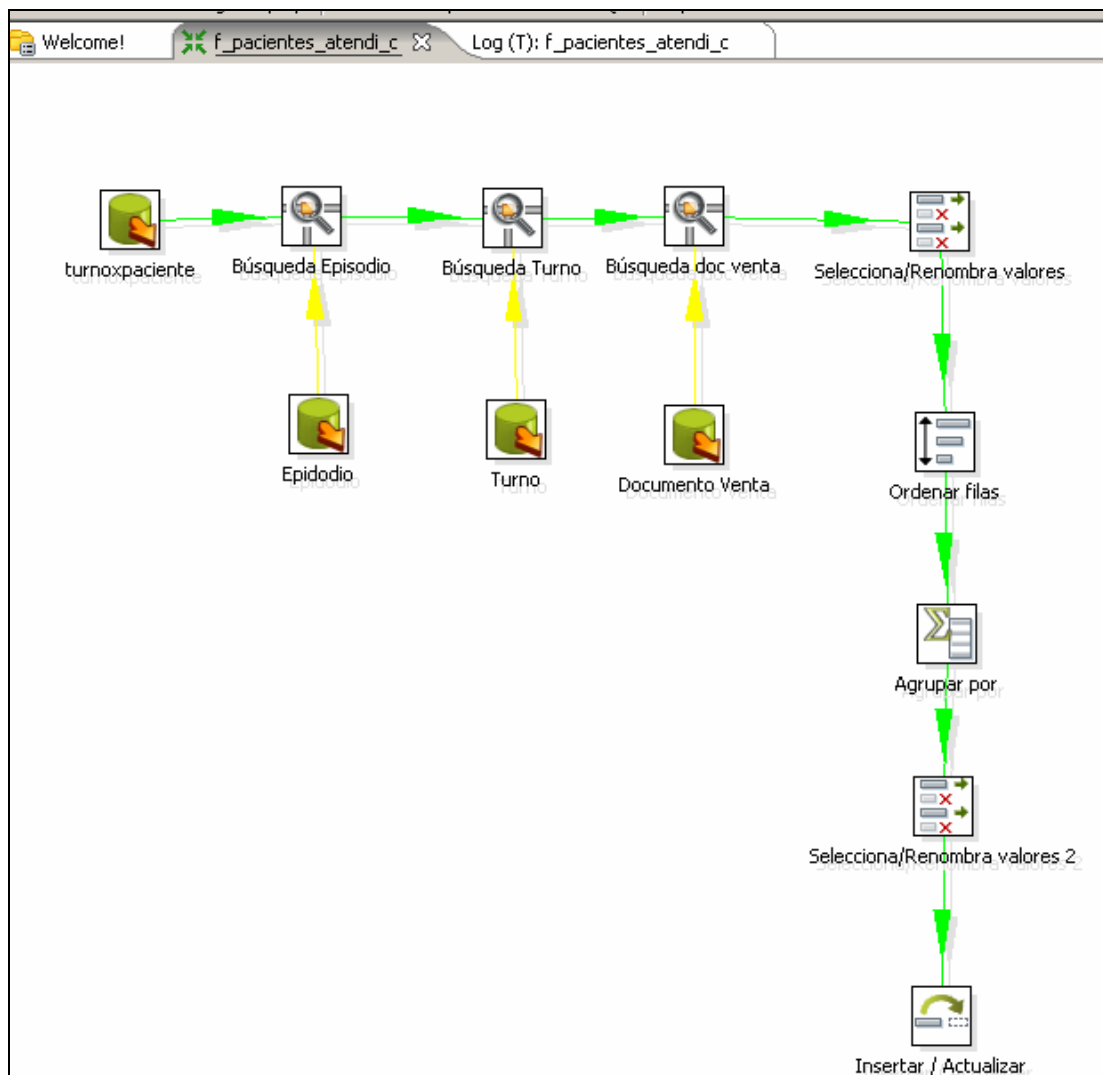
- Tabla f_pacientes_atendi

Id_d_dia	Id_d_Servicio_med	Id_d_horar	Id_pers onal	Min_es pera	N_p ac	Min_aten cion	mon to
39143	1	3	5	20	1	30	238
39146	1	1	5	10	1	30	119

Resultado Final:

				Measures			
Dia	Serv_medico	Horario	Personal	Min_atencion_prom	Min_espera_prom	Monto	N_pacientes
-All Dia.tiempos	-All serv_medico	-All horario	-All personal	30.0	15.0	357.0	2.0
			PER00005	30.0	15.0	357.0	2.0
		Horario Mañana	-All personal	30.0	10.0	119.0	1.0
			PER00005	30.0	10.0	119.0	1.0
		Horario Noche	-All personal	30.0	20.0	238.0	1.0
			PER00005	30.0	20.0	238.0	1.0
ServicioMedico 1	+All horario	+All personal	30.0	15.0	357.0	2.0	
+1ER TRIMESTRE 2008	+All serv_medico	+All horario	+All personal	30.0	15.0	357.0	2.0

ETL:



Este proceso ETL es similar al anterior, la diferencia es que no toma como fuente la tabla intervencion, sino que parte de la tabla turnoxpaciente pues ahí se encuentran las citas que han tenido los pacientes y quién los atendió.

Entradas: Se parte de los contenidos de la tabla episodio,(de la vista historia_clinica_dw), turnoypaciente y turno (de la vista cita_dw) y documento_venta (de la vista caja_dw).

Salida Esperada: La Fact f_pacientes_aten totalmente cargada, cumpliendo integridad referencial y con los valores correctos.

Ejecución y pruebas: Al ejecutar el Job, se presenta el siguiente diagrama:

#	Nombre paso	Numero Copia	Leído	Escrito	Entrada	Salida
1	turnoypaciente	0	0	2	2	0
2	Búsqueda Episodio	0	4	2	0	0
3	Epidodio	0	0	2	2	0
4	Búsqueda Turno	0	6	2	0	0
5	Turno	0	0	4	4	0
6	Búsqueda doc venta	0	14	2	0	0
7	Documento Venta	0	0	12	12	0
8	Selecciona/Renombra valores	0	2	2	0	0
9	Ordenar filas	0	2	2	0	0
10	Agrupar por	0	2	2	0	0
11	Selecciona/Renombra valores 2	0	2	2	0	0
12	Insertar / Actualizar	0	2	2	2	0

```

2008/09/27 17:16:27 - Agrupar por.0 - Iniciando ejecución...
2008/09/27 17:16:27 - Insertar / Actualizar.0 - Iniciando ejecución...
2008/09/27 17:16:27 - Turno.0 - Finished reading query, closing connection.
2008/09/27 17:16:27 - turnoypaciente.0 - Finished reading query, closing connection.
2008/09/27 17:16:27 - Documento Venta.0 - Finished reading query, closing connection.
2008/09/27 17:16:27 - Epidodio.0 - Finished reading query, closing connection.
2008/09/27 17:16:28 - Epidodio.0 - Procesamiento finalizado (I=2, O=0, R=0, W=2, U=0, E=0
2008/09/27 17:16:28 - turnoypaciente.0 - Procesamiento finalizado (I=2, O=0, R=0, W=2, U=0, E=0
2008/09/27 17:16:28 - Turno.0 - Procesamiento finalizado (I=4, O=0, R=0, W=4, U=0, E=0
2008/09/27 17:16:28 - Documento Venta.0 - Procesamiento finalizado (I=12, O=0, R=0, W=12, U=0, E=0
2008/09/27 17:16:28 - Búsqueda Episodio.0 - Procesamiento finalizado (I=0, O=0, R=4, W=2, U=0, E=0
2008/09/27 17:16:28 - Búsqueda Turno.0 - Procesamiento finalizado (I=0, O=0, R=6, W=2, U=0, E=0
2008/09/27 17:16:28 - Búsqueda doc venta.0 - Procesamiento finalizado (I=0, O=0, R=14, W=2, U=0, E=0
2008/09/27 17:16:28 - Selecciona/Renombra valores.0 - Procesamiento finalizado (I=0, O=0, R=2, W=2, U=0, E=0
2008/09/27 17:16:28 - Ordenar filas.0 - Procesamiento finalizado (I=0, O=0, R=2, W=2, U=0, E=0
2008/09/27 17:16:28 - Agrupar por.0 - Procesamiento finalizado (I=0, O=0, R=2, W=2, U=0, E=0
2008/09/27 17:16:28 - Selecciona/Renombra valores 2.0 - Procesamiento finalizado (I=0, O=0, R=2, W=2, U=0, E=0
2008/09/27 17:16:28 - Insertar / Actualizar.0 - Procesamiento finalizado (I=2, O=0, R=2, W=2, U=0, E=0
2008/09/27 17:16:28 - Spoon - La transformación ha finalizado!!
    
```

Y finalmente, al revisar los contenidos de la Fact se comprueba que los datos son consistentes.

	id_d_dia integer	id_d_servicio [PK] integer	id_d_horario [PK] integer	id_persona [PK] integer	n_pacientes integer	min_espera_ integer	min_atencion_ integer	monto numeric(38,2)
1	39143	1	3	5	1	20	30	238.00
2	39146	1	1	5	1	10	30	119.00
*								

Caso de prueba del informe “Indicadores en Emergencias”: Esta prueba tiene como objetivo verificar que las tablas para cargar el informe mencionado sean correctamente cargadas por medio del ETL, tomando como fuente la vistas del Data Warehouse CITA_DW, y HISTORIA_CLINICA_DW y que el informe “Indicadores en Emergencias” muestre la data correcta.

Datos de Entrada:

- Tabla Dia: Esta tabla posee una gran extensión por lo que sólo se mostrará un registro ejemplo

Id_dia	Fecha	Id_mes	Id_dia_semana
38730	2007-01-14	1273	2

- Tabla Intervención: Se listan las columnas significativas para el proceso.

Id_interv	Id_tip_aten	Id_genera_hosp	Id_paciente	Id_epi	Id_turno	I_fallecim
1	1	0	1	1	4	0
2	1	1	1	2	1	0

- Tabla Episodio: Se listan las columnas significativas para el proceso.

Id_epi	Id_hora	Id_pac	Id_turno	Id_serv	id_cuenta
1	20	1	4	1	1
1	8	1	1	1	1

- Tabla Turno: Se listan las columnas significativas para el proceso.

Id_turno	Id_personal	Id_horario	Id_serv	Id_zona_trab
1	5	1	1	1
2	6	2	2	1
3	7	3	3	2

4	5	3	1	1
---	---	---	---	---

- Tabla Personal: Se listan las columnas significativas para el proceso y una porción del número de registros.

Id_personal	Id_i_medico	sueldo
5	1	3000

- Tabla TurnoxPaciente: Se listan las columnas significativas para el proceso.

Id_paciente	Id_turno	Id_dia	Tiempo_aten	Tiempo_espera
1	4	39143	30	20
1	1	39146	30	10

- Tabla TurnoxDia: Se muestran sólo algunos registros debido a la extensión de la tabla

Id_turno	Id_dia_semana
1	5
1	6
1	7
4	2
4	3
4	4
4	5
4	6

Resultados intermedios:

- Tabla f_pacientes_atendi

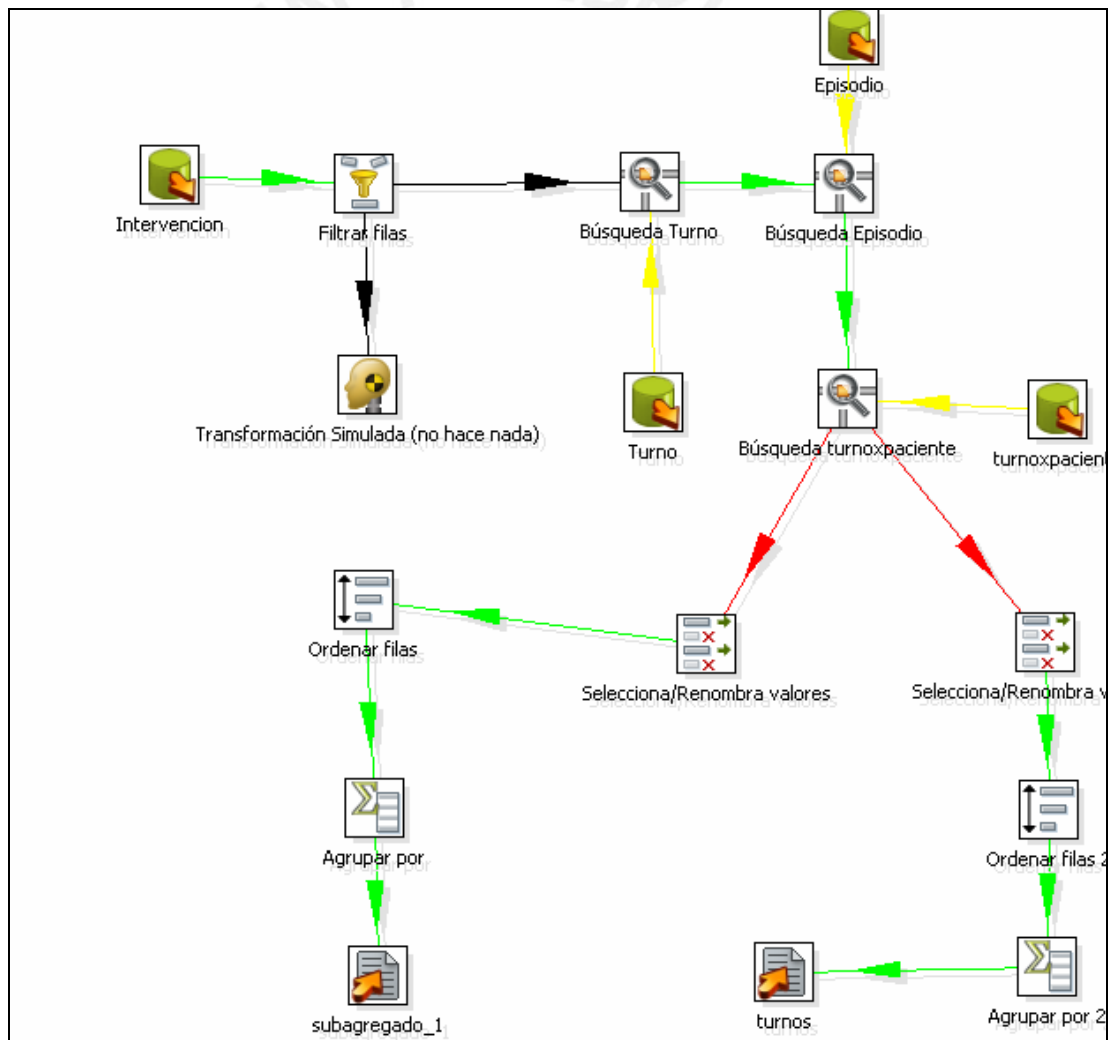
Id_d_dia	Id_d_horario	id_d_Servicio_med	N_pac	N_auxiliares	Min_espera	Min_atencion	N_fallecimientos
----------	--------------	-------------------	-------	--------------	------------	--------------	------------------

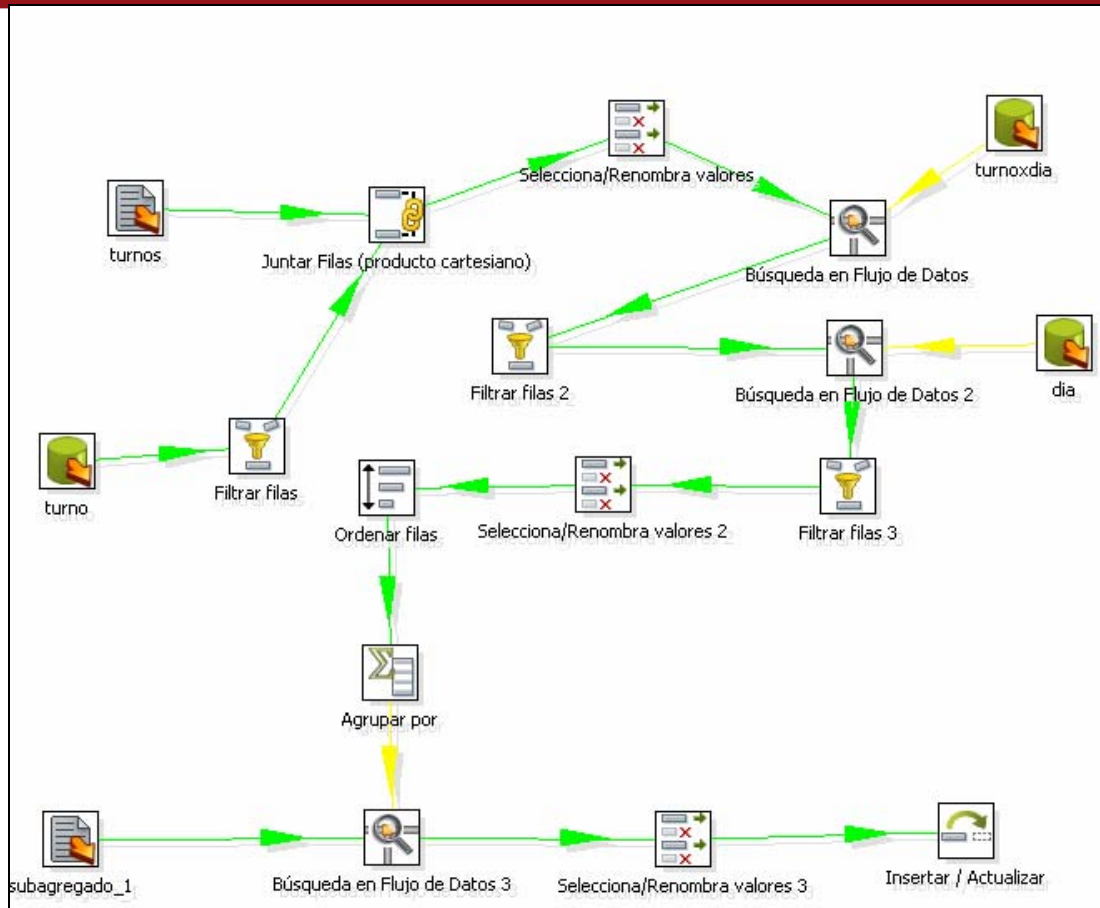
3914	3	1	1	2	20	30	0
3							
3914	1	1	1	1	10	30	0
6							

Resultado Final:

			Measures				
Tiempo	Horario	Serv_medico	Min_atencion_prom	Min_espera_prom	N_personal	N_pacientes	Personal_por_Paciente
All dia	All horario	All serv_medico	30.0	15.0	3.0	2.0	1.5
		ServicioMedico 1	30.0	15.0	3.0	2.0	1.5
	Horario Mañana	All serv_medico	30.0	10.0	1.0	1.0	1
	Horario Noche	All serv_medico	30.0	20.0	2.0	1.0	2
2008	All horario	All serv_medico	30.0	15.0	3.0	2.0	1.5

ETL:





Este proceso ETL se divide en dos jobs separados debido a que, por su complejidad, el Kettle retorna error. El job trabaja con aquellas filas de la tabla intervención que corresponden a emergencias. Para hallar al número de auxiliares, se debe realizar una multiplicación de tablas (producto cartesiano) mientras que para hallar el número de pacientes, fallecimientos y minutos de espera y atención, se debe realizar las agregaciones por separado por las llaves pertinentes.

Entradas: Se parte de los contenidos de las tablas episodio, intervencion (de la vista historia_clinica_dw), turnoypaciente, turnoxdia y turno (de la vista cita_dw) y dia (de la vista general_dw).

Salida Esperada: La Fact f_emergencias totalmente cargada, cumpliendo integridad referencial y con los valores correctos.

Ejecución y pruebas: Al ejecutar el Job, se presenta el siguiente diagrama:

#	Nombre paso	Numero Copia	Leído	Escrito	Entrada	Salida
1	turno	0	0	4	4	0
2	Filtrar filas	0	4	4	0	0
3	turnos	0	0	2	3	0
4	Juntar Filas (producto cartesiano)	0	6	3	0	0
5	turnoxdia	0	0	18	18	0
6	Búsqueda en Flujo de Datos	0	21	3	0	0
7	Filtrar filas 2	0	3	3	0	0
8	Selecciona/Renombra valores	0	3	3	0	0
9	Búsqueda en Flujo de Datos 2	0	45296	3	0	0
10	dia	0	0	45293	45293	0
11	Filtrar filas 3	0	3	3	0	0
12	Selecciona/Renombra valores 2	0	3	3	0	0
13	Ordenar filas	0	3	3	0	0
14	Agrupar por	0	3	2	0	0
15	subagregado_1	0	0	2	3	0
16	Búsqueda en Flujo de Datos 3	0	4	2	0	0
17	Selecciona/Renombra valores 3	0	2	2	0	0
18	Insertar / Actualizar	0	2	2	2	0

2008/09/27 17:37:41 - Filtrar filas.0 - Procesamiento finalizado (I=0, O=0, R=4, W=4, U=0, E=0
 2008/09/27 17:37:41 - subagregado_1.0 - Procesamiento finalizado (I=3, O=0, R=0, W=2, U=0, E=0
 2008/09/27 17:37:41 - turnos.0 - Procesamiento finalizado (I=3, O=0, R=0, W=2, U=0, E=0
 2008/09/27 17:37:42 - turno.0 - Procesamiento finalizado (I=4, O=0, R=0, W=4, U=0, E=0
 2008/09/27 17:37:42 - Selecciona/Renombra valores.0 - Procesamiento finalizado (I=0, O=0, R=3, W=3, U=0, E=0
 2008/09/27 17:37:42 - Búsqueda en Flujo de Datos.0 - Procesamiento finalizado (I=0, O=0, R=21, W=3, U=0, E=0
 2008/09/27 17:37:42 - Filtrar filas 2.0 - Procesamiento finalizado (I=0, O=0, R=3, W=3, U=0, E=0
 2008/09/27 17:37:42 - Juntar Filas (producto cartesiano).0 - Procesamiento finalizado (I=0, O=0, R=6, W=3, U=0, E=0
 2008/09/27 17:37:44 - dia.0 - Finished reading query, closing connection.
 2008/09/27 17:37:44 - dia.0 - Procesamiento finalizado (I=45293, O=0, R=0, W=45293, U=0, E=0
 2008/09/27 17:37:45 - Búsqueda en Flujo de Datos 2.0 - Procesamiento finalizado (I=0, O=0, R=45296, W=3, U=0, E=0
 2008/09/27 17:37:45 - Filtrar filas 3.0 - Procesamiento finalizado (I=0, O=0, R=3, W=3, U=0, E=0
 2008/09/27 17:37:45 - Selecciona/Renombra valores 2.0 - Procesamiento finalizado (I=0, O=0, R=3, W=3, U=0, E=0
 2008/09/27 17:37:45 - Ordenar filas.0 - Procesamiento finalizado (I=0, O=0, R=3, W=3, U=0, E=0
 2008/09/27 17:37:45 - Agrupar por.0 - Procesamiento finalizado (I=0, O=0, R=3, W=2, U=0, E=0
 2008/09/27 17:37:45 - Búsqueda en Flujo de Datos 3.0 - Procesamiento finalizado (I=0, O=0, R=4, W=2, U=0, E=0
 2008/09/27 17:37:45 - Selecciona/Renombra valores 3.0 - Procesamiento finalizado (I=0, O=0, R=2, W=2, U=0, E=0
 2008/09/27 17:37:45 - Insertar / Actualizar.0 - Procesamiento finalizado (I=2, O=0, R=2, W=2, U=0, E=0
 2008/09/27 17:37:45 - Spoon - La transformación ha finalizado!!

Y finalmente, al revisar los contenidos de la Fact se comprueba que los datos son consistentes.

	id_d_dia [PK] integer	id_d_horario [PK] integer	id_d_servicio [PK] integer	n_pacientes integer	n_auxiliares integer	min_espera_ integer	min_atencion integer	n_fallecimien integer
1	39143	3	1	1	2	20	30	0
2	39146	1	1	1	1	10	30	0
*								