Diseño e implementación de un almacén de datos para el estudio de pacientes en un Servicio Hospitalario de Neumología

D. Nieto

Departamento de Electrónica y Computación
Universidad de Santiago de Compostela
e–mail: dnieto[at]cesga.es

16 de enero de 2013

Resumen

En este trabajo se expone la arquitectura y principales funcionalidades de *MORFEO-Almacén*, fundamentalmente en lo que respecta al diseño e implementación de una infraestructura de gestión y almacenamiento de datos sobre actuaciones médicas, adquiridos de pacientes ingresados en la Unidad del Sueño del Servicio de Neumología del Complejo Hospitalario Universitario de Santiago. Esta plataforma garantiza una estructura básica para el análisis y explotación de dichos datos mediante técnicas computacionales con el fin de descubrir nuevo conocimiento e integrarlo en el sistema de Seguimiento que facilita el proceso asistencial a pacientes.

Palabras clave: Almacenes de datos; Minería de datos; Sistemas Biomédicos; Sistemas de información, DSS

Índice

Índice de figuras

Índice de tablas

1. Introducción

1.1. Dominio médico del problema

El presente documento expone la arquitectura y funcionalidades de un almacén de datos, diseñado para proporcionar la infraestructura necesaria para la aplicación de técnicas de minería de datos sobre la información adquirida a pacientes bajo un seguimiento en el Servicio de Neumología del Complejo Hospitalario Universitario de Santiago. Este trabajo se enmarca en el proyecto de investigación *Minería de datos en polisomnografías de pacientes con alteraciones cardiopulmonares del sueño* (PGIDIT04SIN206003PR).

Bajo este contexto, se pretende desarrollar una herramienta al servicio del personal clínico del Complejo Hospitalario Universidad de Santiago. La finalidad del proyecto no es asistencial, faceta ya resuelta por el Servicio Gallego de Salud (SERGAS), sino de investigación, complementando el anterior servicio en la labor de proporcionar la infraestructura necesaria para la aplicación de estrategias de minería de datos.

Consideramos alteraciones cardiopulmonares del sueño como el término común que nos permite referirnos a un conjunto amplio de trastornos relacionados con pacientes que presentan una obstrucción total o parcial de la vía aérea superior (APNEA):

- *Síndrome de Apnea / Hipopnea del Sueño*
- *Enfermedades Pulmonares Obstructivas Crónicas*
- *Enfermedades Restrictivas Torácicas o Neuromusculares*
- *Insuficiencia Cardíaca*

Dicho tipo de trastornos producen una disrupción de la arquitectura del sueño que condiciona somnolencia diurna excesiva, con la consiguiente disminución en la calidad de vida y la exposición a riesgos mortales por accidentes laborales o en la conducción de vehículos. En general, los sujetos afectados desconocen su enfermedad, ya que no relacionan los problemas que sufren durante el día con lo que les sucede en el período nocturno.

Este tipo de pacientes es objeto de una serie de pruebas y estudios exhaustivos en las denominadas Unidades de Sueño Neumológicas. La polisomnografía es una prueba fundamental para el diagnóstico, e indispensable para el manejo terapéutico de los enfermos, consistiendo en el registro de un conjunto muy amplio de parámetros entre los que se encuentran electroencefalogramas, electrocardiogramas, electrooculogramas, electromiogramas o datos de oximetría, entre otros.

Los estudios realizados sobre pacientes que sufren alteraciones cardiopulmonares del sueño proporcionan una gran cantidad de datos en forma de encuestas, exploraciones o pruebas de distinta naturaleza, pero se extrae información muy limitada del estudio polisomnográfico, y las herramientas software que actualmente integran los equipos de polisomnografía resultan del todo insuficientes, al detectar únicamente eventos relativamente simples de analizar como es el de la falta de flujo respiratorio asociado a la apnea del sueño.

El análisis clínico de estos datos permite al experto médico interpretar el estado fisiopatológico del paciente, realizar un diagnóstico, y decidir la terapia a administrar, tras lo que se vuelven a adquirir aquellos datos que permiten interpretar la evolución del estado fisiopatológico y decidir acciones correctoras sobre la terapia administrada, en un ciclo que forman la monitorización-diagnóstico-terapia.

El proceso de *Descubrimiento de Conocimiento* que tiene lugar en MORFEO consiste en incorporar *nuevo conocimiento* al sistema. Dicho proceso de descubrimiento, realizado mediante la aplicación de técnicas computacionales, permite obtener nueva información a partir de datos que antes no encontraban interpretación.

1.2. Descubrimiento de Conocimiento en MORFEO

El Descubrimiento de Conocimiento en Bases de Datos *(Knowledge Discovery in Databases)* se define como "el proceso no trivial de identificación de patrones válidos, novedosos, potencialmente útiles y, en última instancia, comprensibles a partir de los datos"[5]. Su objetivo es proporcionar herramientas computacionales que permitan la obtención de nuevo conocimiento mediante procesos automáticos o semiautomáticos.

El descubrimiento de conocimiento en *MORFEO* se realiza siguiendo un proceso iterativo e interactivo, que cuenta con las siguientes etapas:

- *Comprensión del dominio*, incluyendo el conocimiento previo sobre las alteraciones cardiopulmonares del sueño, así como los objetivos planteados.
- *Adquisición de datos.* Se reúnen aquí todos aquellos procesos que permiten elaborar el Cuaderno de Seguimiento de los distintos pacientes, y que proporcionan los datos recogidos en la base de datos de *MORFEO-Seguimiento.* A partir de esos datos, se hace una selección de aquellos a los que se aplicarán los distintos procesos de análisis.
- *Acondicionamiento de datos.* Los procesos de adquisición suelen ser incompletos, y proporcionan datos con abundantes errores o inconsis-

tencias. Estos datos requieren de operaciones de eliminación de ruido, y se deben proporcionar soluciones para aquellos datos que faltan. Algunas de estas operaciones se realizan antes de la incorporación de los datos al almacén. Otras tienen lugar en etapas posteriores.

- *Integración de los datos*, procedentes de fuentes múltiples y heterogéneas.
- *Reducción de los datos y proyección.* Muchos de los datos se pueden representar de un modo más útil que en su forma original, a menudo mediante la selección de un conjunto reducido de características, o mediante el uso de métodos de transformación o reducción de la dimensionalidad.
- *Minería de datos.* Constituye la etapa más importante del proceso de descubrimiento automático de conocimiento [9]. La minería de datos es la reunión de un conjunto muy heterogéneo de técnicas cuya programación permite descubrir patrones ocultos, asociaciones o anomalías en conjuntos de datos depositados en el almacén de datos de MORFEO.
- *Interpretación* de los patrones descubiertos. Resulta en último término una supervisión del proceso automático de minería de datos, que permite asegurar el interés y utilidad de los resultados obtenidos.
- *Utilización del conocimiento* obtenido, mediante su incorporación a las tareas involucradas en el seguimiento de pacientes.

El proceso de descubrimiento de conocimiento se detalla en la figura 1.

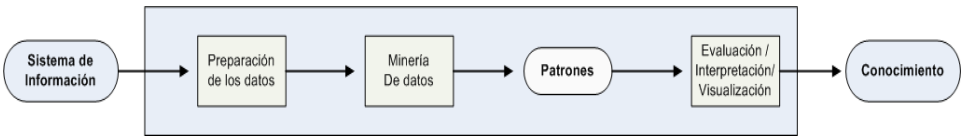

Figura 1: Proceso de descubrimiento de conocimiento

Durante los últimos años, las técnicas computacionales de minería de datos y descubrimiento de conocimiento han ido creciendo en interés e importancia. Una de las razones para ello es el incesante cambio tecnológico, que permite cada vez mayores capacidades de almacenamiento y un consiguiente aumento en el volumen de datos recogidos. De hecho, la investigación en Inteligencia Artificial en Medicina en los últimos 20 años ha sufrido un importante cambio, ya que el desarrollo de sistemas basados en conocimiento, que trataba de capturar la experiencia del médico, ha sido desplazado por los sistemas intensivos de datos [18]. Esto resulta particularmente necesario en el caso de los servicios de neumología, donde la monitorización

continua de pacientes durante la fase de sueño genera una enorme cantidad de información que incluye datos muy heterogéneos y con una componente temporal intrínseca muy importante.

El tratamiento de datos de pacientes con alteraciones cardiopulmonares del sueño permiten dirigir los procesos de minería hacia el descubrimiento de relaciones temporales, relaciones de proximidad contextual y relaciones causa/efecto.

1.3. Arquitectura de MORFEO

La cantidad ingente de datos generados por las actuaciones médicas hace que sea necesaria una estrategia de almacenamiento y gestión de los mismos para su posterior estudio.

Así, el sistema *MORFEO* cumple una doble finalidad:

- Proporcionar una plataforma de almacenamiento estructurado con los datos generados en el proceso de monitorización de pacientes ingresados en la Unidad del Sueño.
- Proporcionar un sistema de información con una serie de servicios básicos orientados al Descubrimiento de Conocimiento utilizando técnicas de análisis avanzadas.

Esta sección, presenta la arquitectura y funcionalidades del sistema de información MORFEO, que plantea dos subsistemas bien diferenciados: *MORFEO-Seguimiento* y *MORFEO-Almacén*. La figura 2 representa sus diferentes componentes y las relaciones existentes entre ellos.

1.3.1. MORFEO-Seguimiento

La utilización de pautas clínicas en la rutina médica cuenta con un amplio respaldo por parte de organizaciones y administraciones sanitarias [4]. Su realización computacional facilita al especialista médico su labor asistencial: por un lado, en lo que respecta a la organización de tareas y de la información; por otro, mediante su inclusión en sistemas de ayuda a la decisión. Para dar solución a la realización electrónica de pautas clínicas, *MORFEO-Seguimiento* asigna a cada paciente un Cuaderno de Seguimiento, que facilita al personal médico su gestión individualizada, la toma de decisiones en cuanto a las actuaciones médicas más aconsejables y la recogida de datos procedentes de las mismas. Entendemos por actuación médica toda aquella acción que involucra procesos de medida o evaluación del paciente y que tiene por fin la caracterización de su estado y evolución fisiopatológica.

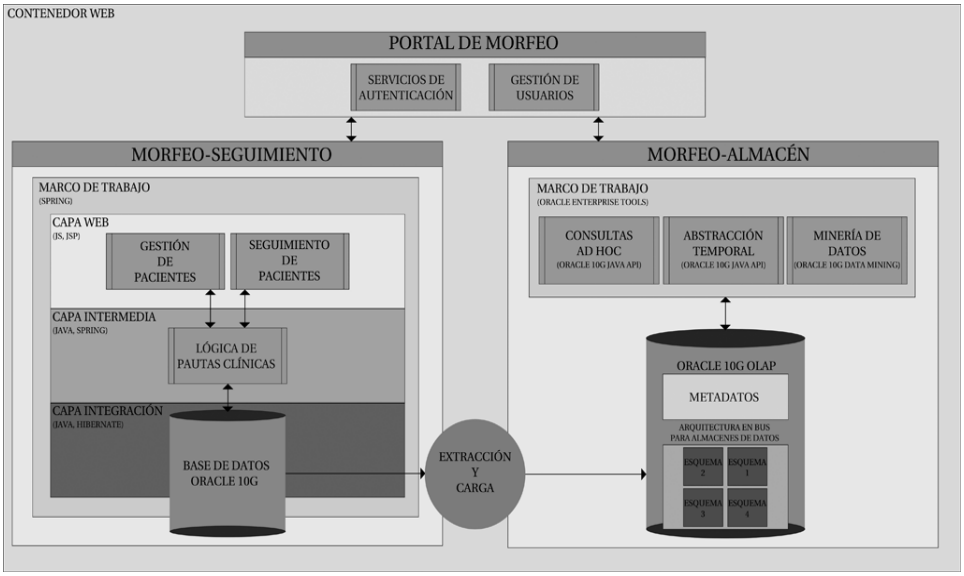

Figura 2: Arquitetura de MORFEO

El Cuaderno de Seguimiento de cada paciente organiza las actuaciones médicas susceptibles de realización en varias etapas, que comprenden una o más visitas del paciente al Servicio de Neumología. MORFEO-Seguimiento distingue, en concreto, cinco etapas:

- *Primera consulta*: durante la primera visita de un paciente a la unidad, se realiza su historia clínica, una exploración física y se solicitan una serie de exámenes complementarios para poder realizar un diagnóstico que se encuadre dentro de una de las 4 categorias que componen las alteraciones cardiopulmonares del sueño indicadas en el capítulo 1.1.
- *Evaluación específica*: esta etapa permite recopilar información más específica sobre el paciente, una vez clasificado dentro de uno de los grupos de dolencias.
- *Nivel de gravedad*: esta etapa trata de determinar el grado de afectación cardiopulmonar del paciente respecto a su dolencia
- *Tratamiento*: el objetivo de esta etapa es programar y proporcionar los distintos tratamientos administrar a un determinado paciente, deacuerdo con el tipo de alteración que padezca y el nivel de gravedad de la misma.
- *Revisión*: en esta etapa se revisarán los efectos de los tratamientos administrados al paciente y su evolución para comprobar la eficacia de la terapia administrada.

Así, *MORFEO-Seguimiento* se constituye como la herramienta encargada de recopilar y almacenar los datos correspondientes al seguimiento de los pacientes del Servicio de Neumología y tiene como principal objetivo el facilitar el seguimiento de los mismos, de acuerdo con la pauta clínica diseñada para cada uno de los protocolos instaurados en el mismo. Para ello organiza la obtención de los datos médicos relevantes para los objetivos planteados, tanto asistenciales como de investigación, poniendo a disposición del personal clínico un conjunto de herramientas web que permiten la gestión y consulta de la información recogida.

1.3.2. MORFEO-Almacén

El seguimiento de pacientes durante el estudio de las alteraciones cardiopulmonares del sueño da lugar a un conjunto heterogéneo de datos, procedente de la realización de cuestionarios, de la evaluación médica o de la realización de pruebas hospitalarias. El proceso de seguimiento obliga a repetir la realización de ciertos cuestionarios y pruebas (entre ellas la polisomnografía) para comprobar la respuesta del paciente a la administración de la terapia correspondiente. Por tanto, se dispone de un amplio conjunto de datos de 151 pacientes del servicio, llegando al orden de varias decenas de Gigabytes, en los que su disposición temporal es fundamental para la interpretación y estudio del estado y evolución de los pacientes.

MORFEO-Almacén permite estructurar los datos disponibles en el Servicio de Neumología con el fin de aplicar técnicas analíticas avanzadas. Los almacenes de datos no son imprescindibles para la aplicación de técnicas de minería, que pueden ser implementadas directamente sobre un simple fichero. Esta es la solución adoptada en proyectos como SIESTA [6], de vocación marcadamente normativa, donde se busca generar una base de datos de referencia en la comunidad científica. En cambio, MORFEO-Almacén parte de premisas distintas:

- Responde a una realidad clínica en la que los procesos de adquisición de datos son incompletos y defectuosos.
- Las técnicas de abstracción proporcionan nueva información derivada que luego es clínicamente validada e incorporada a la rutina clínica en *MORFEO-Seguimiento*, tal es el caso de la aplicación de algoritmos de reconocimiento de apneas, o de patrones en la caída de la saturación de oxígeno.

1.4. Objetivos

Los objetivos de *MORFEO-Almacén* básicamente radican en diseñar e implementar un sistema de almacenamiento con datos adquiridos a pacientes que:

- Permita seleccionar, transferir y reorganizar dichos datos en un nuevo repositorio especialmente configurado para tareas de análisis.
- Permita consultar, agregar y minar de manera eficiente y sofisticada la información reorganizada en el anterior repositorio.
- Permita la aplicación de técnicas de abstración para descubrir nuevo conocimiento, para que sea evaluado e incorporado a las pautas clínicas de *MORFEO-Seguimiento*.
- Garantice la seguridad, confidencialidad e integridad de la información clínica almacenada.

2. Arquitectura de almacenes de datos

2.1. Definición y justificación de un almacén de datos

La arquitectura básica de MORFEO se fundamenta en "software especializado" formalmente denominado *Sistema Gestor de Bases de Datos* que se encarga de gestionar los datos almacenados, de manera masiva y eficiente. La consulta y recuperación de información básica o derivada simple (agregada, interrelacionada) se realiza mediante lenguajes como SQL. Con esta capa de software el proyecto MORFEO es capaz de solucionar los problemas de almacenamiento y acceso a datos.

Cabe hacer una breve aclaración entre los términos *SGBD* y *base de datos*, ya que normalmente su uso entraña cierta confusión:

Una posible definición de base de datos [15] haría referencia a una colección de datos recopilados y estructurados que existe durante un periodo de tiempo. Generalmente, un sistema de información hace referencia a un conjunto de una o más bases de datos, junto con los medios para almacenarlas y gestionarlas (software SGBD), sus usuarios y sus administradores. La figura 3 detalla un ejemplo básico sobre la arquitectura de un sistema de información basado en software SGBD.

Las funciones básicas de un SGBD se alinean con la finalidad del proyecto MORFEO, en cuanto a almacenamiento y gestión de los datos de pacientes, que según la referencia básica de *Ullman* y *Widom* [15] son:

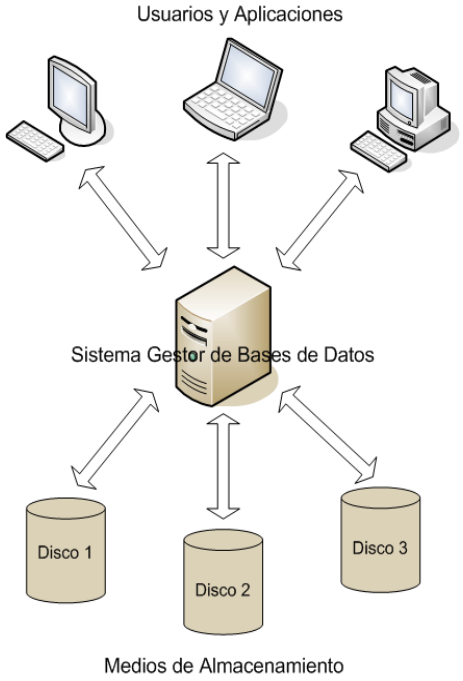

Figura 3: Arquitectura básica de un sistema de información

- Permitir a los usuarios crear nuevas bases de datos y especificar su estructura
- Permitir la posibilidad de consultar los datos, es decir, recuperarlos parcial o totalmente y modificarlos
- Permitir el almacenamiento de grandes cantidades de datos durante un largo periodo de tiempo, manteniéndolos seguros de accidentes o uso no autorizado y permitiendo un acceso eficiente a los datos para consultas y modificaciones.
- Controlar el acceso a los datos de muchos usuarios a la vez, impidiendo que las acciones de un usuario puedan afectar a las acciones de otro y que el acceso simultáneo no cree inconsistencias en la base de datos, es decir, tiene que haber un equilibrio entre el requisito de acceso a los datos y la necesidad de asegurar la integridad de los mismos.

Es importante recalcar la diferencias existentes entre los subsistemas que componen MORFEO, en cuanto a funcionalidades y propósitos. A pesar de que compartan *tecnología relacional* [3], ambos están pensados para desempeñar tareas distintas a la hora de gestionar los datos, y esto se refleja en su diseño.

La funcionalidad básica de *MORFEO-Seguimiento*, es la recopilación y almacenamiento de datos sobre pacientes, historias clínicas, tratamientos, radiografías, polisomnografías etc, que se realiza a través de un interfaz web conectado con el SGBD Oracle 10g. El personal del hospital es el encargado de introducir dichos datos, de manera que la base de datos está continuamente actualizada y preparada para visualizar las consultas sobre historias clínicas, pruebas realizadas a pacientes, tratamientos, etc. Esta funcionalidad se indentifica con el paradigma de **procesamiento transaccional** en bases de datos, que según *Gray* y *Reuter* [11] se podría definir como un conjunto de operaciones lógicas sobre los datos que deben completarse o no como un grupo, es decir, el SGBD debe comportarse de manera determinística, garantizando la integridad y disponibilidad de los datos almacenados.

Teniendo en cuenta esto último, *MORFEO-Seguimiento* se ha configurado en modo OLTP *(On-Line Transaction Processing)* o modo transaccional, cuya meta es maximizar el número de operaciones de consulta, inserción, actualización y borrado.

El modo transaccional satisface las necesidades de gestión de datos de *MORFEO-Seguimiento*. Sin embargo, la configuración anterior resulta insuficiente para otras funciones más complejas como el análisis, la planificación o la predicción, en general, para el desarrollo de sistemas de ayuda a la decisión.

Debido a lo anterior, surge un nuevo tipo de configuración utilizada para **procesar analíticamente** los datos almacenados en un sistema transaccional. Este nuevo tipo de procesamiento se denomina OLAP *(On-Line Analytical Processing)* y su tecnología aplicada, *almacén de datos*, que difiere de un sistema transaccional en su propósito, funcionamiento, estructura y rendimiento.

W.H.Inmon [17], define un almacén de datos como "un repositorio de información heterogénea coleccionada desde una o distintas fuentes, almacenada e integrada bajo un esquema unificado que normalmente reside en un único emplazamiento, variable a lo largo del tiempo y no volátil".

Generalmente, dos ideas básicas dirigen la creación de un almacén de los datos [17] según Inmon:

- Integración de los datos procedentes de bases de datos distribuidas y diferentemente estructuradas, que facilita una descripción global y un análisis comprensivo en el almacén de datos.
- Separación física de los datos usados en operaciones diarias y de los datos usados en el almacén para propósitos de análisis y toma de decisiones.

Por esta razón MORFEO consta de dos subsistemas:

- *MORFEO-Seguimiento*: Un subsistema OLTP para la adquisición de datos
- *MORFEO-Almacén*: Un subsistema OLAP para el posterior análisis de los mismos

Los almacenes de datos qproporcionan almacenamiento, funcionalidad y receptividad a consultas complejas, que no tienen cabida en sistemas transaccionales. Así, bajo un sistema OLTP como *MORFEO-Seguimiento* se podría consultar el número de pacientes a los cuales se les ha realizado un estudio polisomnográfico durante el año. Pero consultas como: "Se ha incrementado el número de pacientes con EPOC en los últimos 5 años" sólo son posibles bajo un sistema OLAP.

MORFEO-Almacén parte de la idea de integrar fuentes heterogéneas de datos (historias de pacientes, ficheros de señal, pruebas, cuestionarios etc), cuyos procesos de adquisicion son incompletos y/o defectuosos. Una vez seleccionados los datos a analizar, se transfieren y reorganizan en un repositorio común para poder agregar, cruzar y minar eficientemente la información de manera sofisticada. Además, dicho repositorio permite tener información altamente sumarizada convirtiéndose así en una herramienta esencial para poder, entre otras cosas, analizar, predecir y tomar decisiones acerca del contexto que esa información representa.

En la actualidad, las anteriores actividades de prediccion empiezan a semiautomatizarse gracias a la evolución de las herramientas de generación de modelos estadísticos hacia los sistemas de **minería de datos y simulación predictiva** [9], que facilitan y automatizan gran parte de los procesos de análisis, predicción y toma de decisiones. Este paradigma también se añade a la lista de funcionalidades soportadas por *MORFEO-Almacén* en las capas de Minería de Datos y Abstracción Temporal.

Las tareas de análisis y prediccion que se realizarán en *MORFEO-Almacén* se detallan gráficamente en la figura 4.

En definitiva, la tecnología OLAP se perfila como el complemento ideal a un sistema OLTP, para poder realizar actividades de análisis y minería de datos[8] en *MORFEO-Almacén*.

2.2. Visión general de un almacén de datos

Una vez aclarada la finalidad y uso de *MORFEO-Almacén*, se identificarán las partes diferenciadoras que intervienen en el diseño e implementación del mismo.

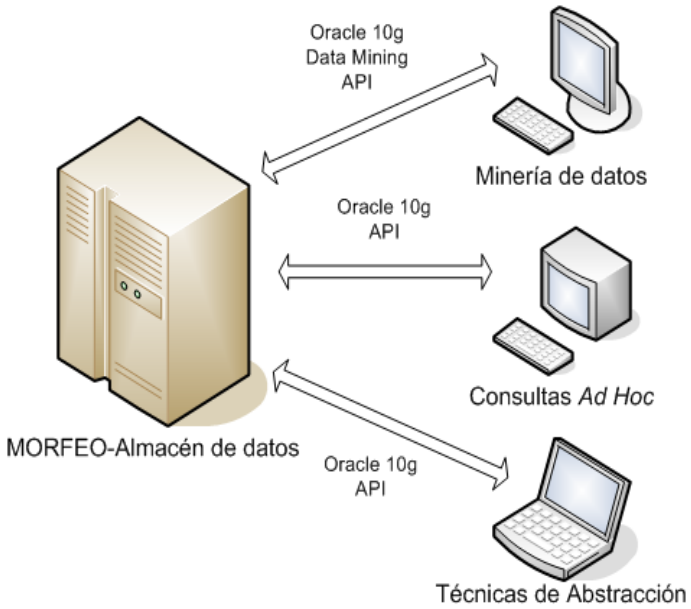

Figura 4: Acceso a *MORFEO-Almacén*

La estructura de *MORFEO-Almacén* comprende cinco etapas:

1) *Fuente de datos*, desde la cual se extraen los datos relacionados con los pacientes, tales como ficheros binarios con la señal obtenida de pruebas a pacientes (polisomnografías, electrocardiogramas ...), datos del historial clínico del paciente (fecha de nacimiento, enfermedades, tratamientos, etc), datos sobre las pautas clínicas seguidas (EPOC, SAHS, enfermedades restrictivas, etc) etc. Toda esta información está almacenada en el sistema transaccional *MORFEO-Seguimiento*, reduciéndose así la complejidad y el tiempo en la fase de extracción.

2) *Extracción, transformación y carga*, en donde se procede a extraer la información del sistema transaccional de seguimiento médico y se consolida en el almacén de datos de MORFEO. En esta fase se procede a convertir, normalizar, limpiar, etc o realizar cualquier tipo de operación con los datos para aumentar su calidad. Este proceso se realiza mediante herramientas software (ETL [14] del inglés *Extraction, Tranformation and Load*) que extraen los datos de las fuentes, los transforman y cargan en el almacén. Por ejemplo, los datos de tipo `DATE` o `TIMESTAMP` no pueden ser minados correctamente por la herramienta de minería de datos de Oracle. Así, la herramienta ETL realiza una conversión de dichos tipos de datos a tipos alfanuméricos como `NUMBER` o `VARCHAR` y poder disponer de esa componente temporal, tan

importante para el análisis de los datos del almacén. Un ejemplo de transformación y limpieza sería el que se realiza con los nombres de los tratamientos medicamentosos, ya que se convierten a minúsculas y se eliminan las posibles reseñas sobre envases (crema, comprimido, capsula etc) o dosis (mg).

3) *Diseño del almacén de datos*, en donde se definen las desnormalizaciones y agregaciones necesarias a realizar en cada caso y la implementación del modelo dimensional que describe las etapas de las pautas clínicas implementadas en *MORFEO-Seguimiento*. El diseño de *MORFEO-Almacén* está basado en el paradigma de *arquitectura en bus para almacenes de datos*, método de diseño incremental desarrollado por R. Kimball[12], que se explicará con mas detalle en la siguiente sección.

4) *Acceso a datos*. En esta fase se contemplan las tareas de análisis de los datos del almacén y como consultarlos. Las pricipales operaciones que se desarrollan en esta fase consisten en ejecutar consultas *ad hoc* y aplicar técnicas de abstracción temporal y de minería de datos para descubrir nuevo conocimiento.

5) *Metadatos*, hacen referencia a la información que se deberá guardar sobre el almacén es decir, toda la información derivada de la actividad del entorno del almacén de datos (proceso ETL, análisis e implementación). Esta información, que describe los datos almacenados y los procesos de gestión de los mismos, es muy importante y se tiene en cuenta a la hora de diseñar el almacén de datos. Algunos ejemplos:

- mapa de datos entre las fuentes y el almacén destino y planificación de las tareas de extracción y carga.
- procesos de transformación, filtrado y limpieza de datos.
- el diccionario de términos del sistema, es decir, todos y cada uno de los términos del dominio médico que entiende el usuario y con que datos y etapas del Cuaderno de Seguimiento se encuentran enlazados.
- plantillas de consultas predefinidas por el usuario, características de seguridad y privacidad, características de acceso a datos
- copias de seguridad (*backups*).

En la figura 5 se detallan las fases y componentes a desarrollar de *MORFEO-Almacén*.

2.3. Tecnologías y herramientas para almacenes de datos

Una vez justificada la necesidad de un almacen de datos como plataforma de análisis para MORFEO, es necesario explicar qué tecnología y soluciones

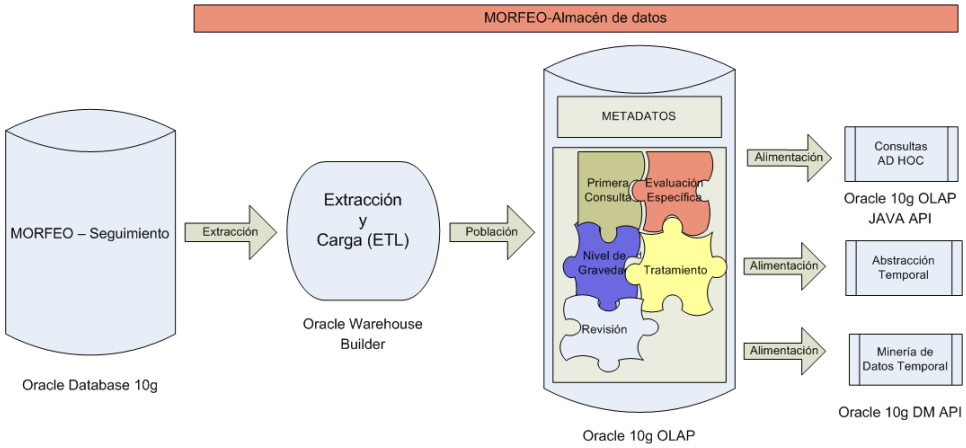

Figura 5: Elementos básicos de *MORFEO-Almacén*

se han utilizado para implementar *MORFEO-Almacén*.

Un SGBD en modo OLTP, está configurado para un rendimiento óptimo en inserciones, actualizaciones, o borrado de registros. *MORFEO-Seguimiento* es un sistema crítico, ya que su funcionamiento está vinculado al proceso asistencial a pacientes del Servicio de Neumología, en donde se actualizan a diario historias clínicas de pacientes, pruebas polisomnográficas, cuestionarios, etc. El hecho de sobrecargar este servicio lanzando consultas que en ocasiones pueden ser complejas y que consumen tiempo de procesamiento haría que el rendimiento del sistema no fuese aceptable. Esto hace necesaria la creación de otro repositorio cuyos datos se extraen y cargan desde *MORFEO-Seguimiento*.

Otra razón que justifica la creación de un almacén, es que *MORFEO-Seguimiento* mantiene modelos Entidad/Relación (E/R) en Tercera Forma Normal, lo que supone la dispersión de datos entre multitud de tablas, hecho que exige un esfuerzo a la hora de sumarizar los datos para un informe (operaciones *join* en álgebra relacional).

Técnicamente, la tipología de operaciones que se ejecutan en *MORFEO-Almacén* son consultas estándar (*queries*) sobre el modelo de datos OLAP [16] del almacén, el cual está optimizado para la ejecución de dichas consultas, haciendo que la dispersión de datos sea mínima (normalmente una consulta no reliza más de 4 o 5 operaciones *join*). Estas operaciones, también denominadas consultas *Ad Hoc*, permiten obtener un análisis de primer nivel, de una manera eficaz y simple, de los datos almacenados.

Algunas tecnologías OLAP tienen la restricción de que aquellas columnas de sumarización que no se hayan previsto, y programado previamente, no

podrán ser utilizadas. Así, para la implementación de *MORFEO-Almacén* se han barajado dos tipos de tecnologías:

- *MOLAP*: La tecnología MOLAP [1] (Multidimensional OLAP) se implementa sobre bases de datos multidimensionales también denominados motores MOLAP. Resuelven ese inconveniente calculando todas las posibles combinaciones de distintos niveles asociados a las métricas. Así, se vaya a utilizar o no, existe una celda donde se encuentra el valor del dato precalculado para cualquier elemento de dimensión. El conjunto de celdas constituye lo que se denomina un *cubo*[10]. Esta tecnología tiene evidentes limitaciones producidas por el desmesurado incremento de ocupación en disco, y el tiempo que se llega a consumir para recalcular el cubo al añadir nuevas columnas de sumarización; pero tiene otras ventajas, como el hecho de que los datos están almacenados en una estructura de índices y almacenamiento óptima, reduciéndose así el tiempo en cálculos complejos. En la figura 6 se describe una posible implementación de un cubo MOLAP para *MORFEO-Almacén*
- *ROLAP*: La tecnología ROLAP (Relational OLAP) implementa las métricas en tablas de bases de datos relacionales (llamadas tablas de hechos), que a su vez apuntan a otras tablas relacionales donde se encuentran los datos dimensionales. No es necesario implementar un cubo, sino que los distintos valores de las métricas, son calculados sobre la marcha mediante consultas SQL. Esta tecnología ha mejorado mucho con los años, habiendo productos en el mercado que éstan a la altura de un motor MOLAP en términos de rendimiento y superándolos en el manejo de grandes volúmenes de datos.

Ambas tecnologías utilizan herramientas para el análisis de los datos bajo modelos dimensionales, pero ROLAP difiere de MOLAP en que no es necesario precalcular y almacenar la información. Las herramientas ROLAP acceden a los datos a través de un motor relacional, y generan consultas que calculan la información bajo demanda. Esta ventaja hace que ROLAP proporcione mayor escalabilidad a la hora de almacenar grandes volúmenes de datos, especialmente en modelos donde las dimensiones tienen una cardinalidad muy alta (cientos de miles de registros).

Debido a lo anterior, se ha seleccionado *Oracle 10g* como plataforma ROLAP para la implementación de *MORFEO-Almacén*, ya que posee mejoras específicamente diseñadas para la gestión de los mismos, siendo su rendimiento similar al de un motor MOLAP.

El ahorro en costes a la hora de utilizar un motor relacional como *Oracle 10g* juega a favor de MORFEO, además de poder combinar técnicas de análisis, consultas y minería de datos sobre el mismo repositorio. Su reutilización

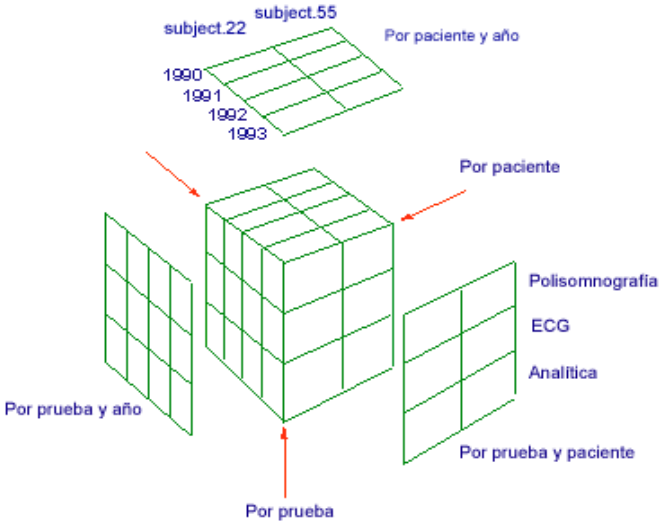

Figura 6: Cubo MOLAP

como motor OLTP para *MORFEO-Seguimiento* también ha inclinado la balanza a favor de esta solución comercial. La alternativa de utilizar software libre, también ha sido estudiada pero no ha cristalizado debido a que los dos SGBDs libres seleccionados (MySQL y PostgreSQL) no poseen las funcionalidades y características técnicas necesarias para construir un sistema de esta magnitud (almacenamiento de grandes cantidades de datos, disparadores, procedimientos almacenados, herramientas ETL, para consultas, minería de datos, etc).

2.4. Modelo dimensional y Arquitectura en Bus para Almacenes de Datos

El almacén de MORFEO se ha diseñado mediante técnicas dimensionales [13] orientadas al modelado de procesos, habiendo tenido en cuenta los flujos de procesos y actuaciones médicas que se realizan durante el siguimiento del paciente.

El modelado dimensional es una técnica de diseño que presenta los datos de una manera intuitiva y estándar, proporcionando un acceso a datos altamente eficiente. Está basada en las técnicas de modelado relacional clásico pero con una componente inherentemente dimensional. Cada esquema dimensional está formado por una tabla *central* con una clave foránea múltiple, llamada *tabla de hechos* y un conjunto de tablas que la rodean

llamadas *tablas de dimensiones*, que no son mas que tablas generalmente desnormalizadas.

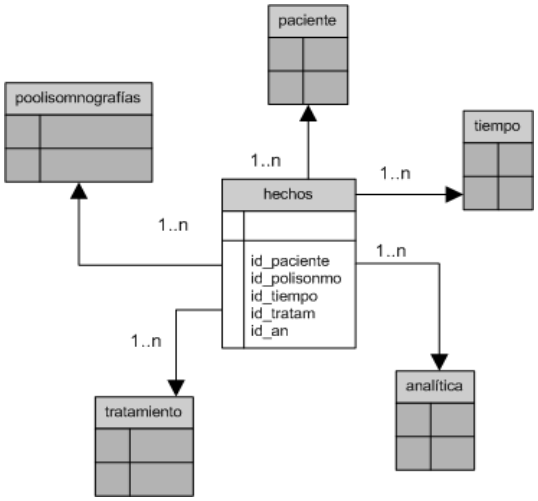

Figura 7: Modelo dimensional en estrella

Cada tabla de dimensión tiene una clave primaria que se corresponde exactamente con uno de los componentes de la clave foránea múltiple de la tabla de hechos, es decir una tabla de hechos está relacionada con `n` tablas de dimensiones. En la figura 7 se puede observar un modelo dimensional ejemplo que explica lo anterior de manera gráfica.

Una tabla de hechos, debido a que su clave primaria es de tipo múltiple y está compuesta por varias claves foráneas o externas, puede implementar relaciones de tipo múltiple (relaciones `m..n`) con otras dimensiones. Además, dichas tablas pueden contener atributos numéricos (`INTEGER, FLOAT`) o de texto simple (`VARCHAR`). Los atributos de las dimesiones pueden ser más complejos y de distintos tipos (`DATE, CLOB, TEXT,` etc).

Uno de los debates actuales más extendidos a la hora de construir un almacén de datos es cómo afrontar su diseño. Se podría plantear desde una perspectiva generalista (construir todo el almacén modelando todos los procesos que tienen lugar durante el seguimiento del paciente) o acogerse a una visión más simplista e inconexa (construir una parte del almacén, aislando y modelando cada etapa del seguimiento médico por separado). Mediante esta última solución se construye lo que común y erróneamente se denomina *data mart* y que normalmente se constituye como una de entre varias partes aisladas e inconexas de un almacén, entre las cuales la interoperabilidad es nula.

La verdadera naturaleza de un *data mart* radica en conseguir que sea un

subconjunto natural y completo del almacén de datos, siendo el mismo la unión lógica de todos y cada uno de los marts.

Es por ello que la metodología para obtener el modelo de *MORFEO-Almacén* se enfoca desde una perspectiva basada en *data marts* como subconjuntos naturales del almacén, dando lugar a un paradigma *bottom-up* de diseño incremental e interoperable denominado *arquitectura en bus para almacenes de datos* [12]. Este paradigma, basado en el modelo dimensional, está compuesto por "*un conjunto de dimensiones conformadas y definiciones de hechos estándar*" [1, P. 156].

Las dimensiones conformadas son dimensiones compartidas entre los distintos *data marts* del almacén, los cuales, describen las propiedades de los procesos y flujos de trabajo relizados en una organización determinada. Es por ello, que el término *arquitectura en bus* hace referencia a una analogía entre las dimensiones compartidas por los distintos *data marts* del almacén y los distintos periféricos o memorias conectadas a un bus de un computador, mediante el cual pueden comunicarse y compartir información.

La finalidad de esta técnica es la de proporcionar la flexibilidad necesaria al modelo dimensional para que el almacén evolucione y pueda ser extendido, integrando nuevo conocimiento procedente de la aplicación de técnicas de abstracción temporal y minería de datos. Este nuevo conocimiento (nuevas pruebas médicas, etapas, etc) se cristaliza en forma de *data marts* o dimensiones.

El ejemplo gráfico detallado en la figura 8 refleja, de manera visual, este concepto de extensibilidad. Si pensamos en un almacén de datos como un puzzle, el cual está formado por varias piezas que son los *data marts*, podremos obtener un ejemplo bastante visual de la *arquitectura en bus para almacenes de datos*. En dicha figura también se puede apreciar que cada uno de los *data marts* se corresponde con una de las etapas del mencionado cuaderno, facilitando la extensibilidad del almacén, con nuevos *data marts* o dimensiones (*data mart* n).

Un *data mart*, o abreviadamente *mart*, está formado por una tabla de hechos y varias dimensiones (ambos conformados). Dichas dimensiones son únicas y son compartidas entre *data marts*. Por ejemplo, la dimensión `PACIENTE` está formada por una tabla de pacientes con una clave primaria única de paciente y con un conjunto de atributos bien diferenciados y construidos que describen a cada paciente. Esto significa que una dimensión conformada es exactamente la misma para cada uno de los *data marts* que configuran el almacén. En la figura 9 se detalla un esquema ejemplo de dos *data marts* (Nivel de gravedad y Tratamiento) que comparten dimensiones conformadas, reflejándose de una manera visual el concepto de *arquitectura en bus*.

Las tablas de hechos se identifican y configuran una vez definidas todas

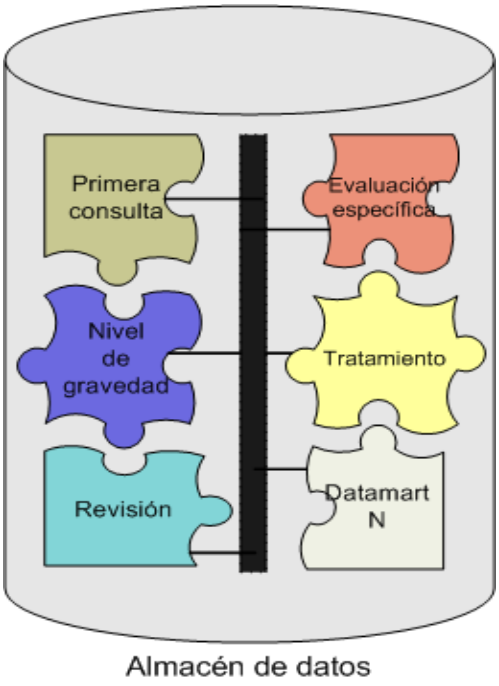

Figura 8: Arquitectura en bus para *MORFEO-Almacén*

las dimensiones. Normalmente, los hechos conformados constan de claves externas y medidas naturales como, por ejemplo, la edad de un paciente. Pero también se pueden utilizar hechos complejos, fruto de una serie de cálculos entre distintas medidas como, por ejemplo, el número de tomas al día de un determinado medicamento para un paciente, o el número de consultas realizadas a un paciente en un mes.

A la hora de escoger estas métricas simples o agregadas como un hecho, se debe tener en cuenta lo que se denomina *granularidad*, un factor diferencial a la hora de diseñar un almacén de datos. Las dimensiones conformadas en el almacén de MORFEO son atómicas, es decir, cada registro de la tabla se corresponde con un paciente, un electrocardiograma, una polisomnografía, un medicamento o un día. Un ejemplo de granularidad más gruesa podría ser utilizar métricas de hechos complejas como por ejemplo el número de pacientes que son consultados al mes/año por una unidad médica o dimensiones como tratamientos medicamentosos administrados a pacientes diagnosticados en una de las dolencias cardiopulmonares.

Esta granularidad tan fina facilita que una tabla de hechos pueda ser la intersección de varias dimensiones atómicas y además permite que los marts puedan ser fácilmente ampliados añadiendo nuevas métricas de hechos, nuevos atributos a una dimensión o incluso dimensiones completas. Esto hace

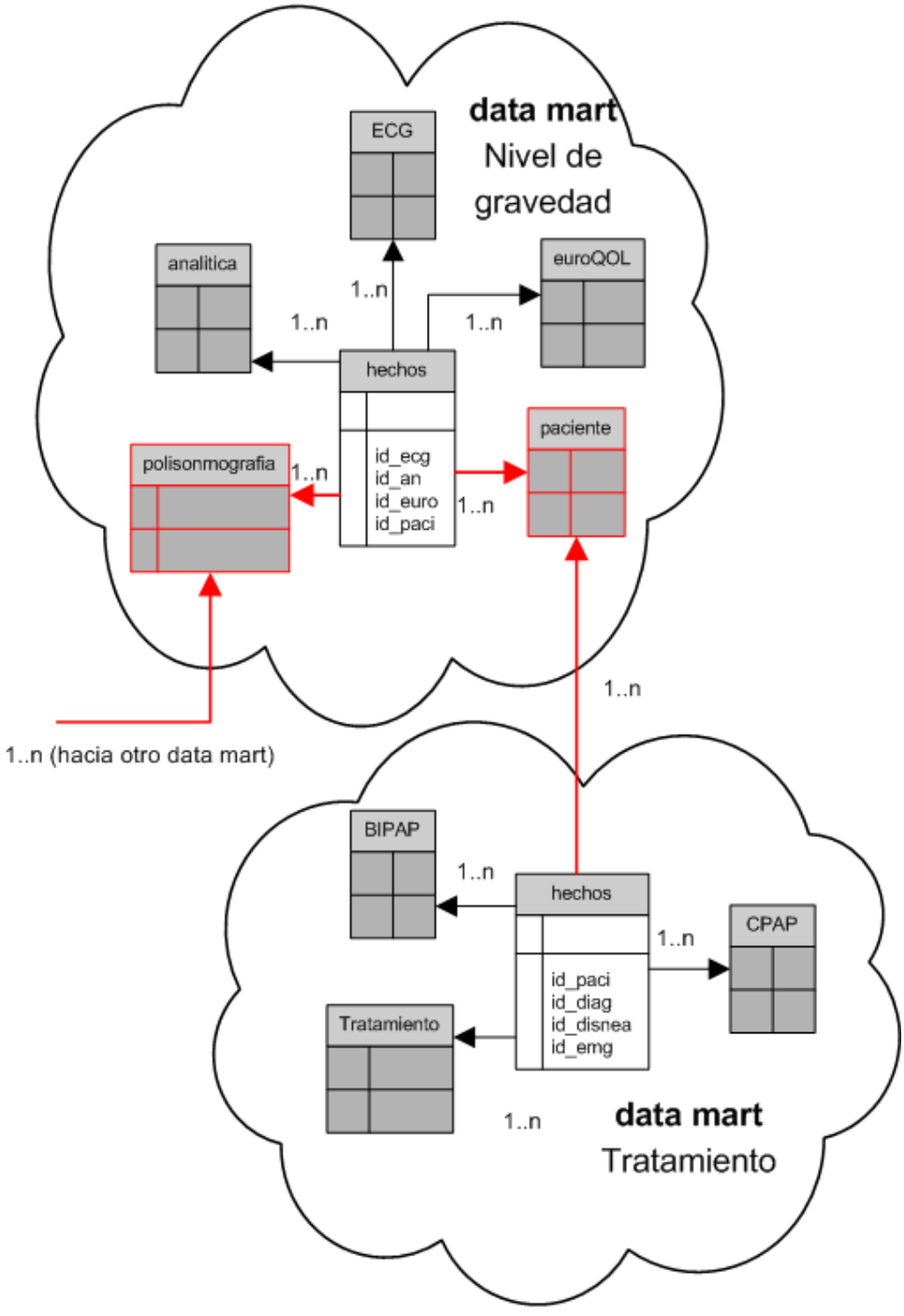

Figura 9: Dimensiones y hechos conformados

que las anteriores aplicaciones y consultas sigan funcionando a pesar de los cambios realizados y que se puedan implementar nuevas técnicas de minería de datos que exploten las nuevas extensiones del almacén.

La previsión de espacio para *MORFEO-Almacén* es que disponga de cientos gigabytes y que vaya creciendo exponencialmente con el tiempo. En caso de tener tener problemas de rendimiento, podría ser necesario utilizar un grano más grueso, aunque la mejor opción es utilizar técnicas de *agregados* para obtener mejoras en el rendimiento de las consultas. Los *agregados* son resúmenes de registros precalculados y almacenados en la base de datos que se utilizan para mejorar el tiempo de respuesta de determinadas consultas. Podría considerarse como una caché de consultas, pero con todos los datos precalculados de antemano. *MORFEO-Almacén* contempla la posibilidad de utilizar *agregados* para determinado perfil de consultas.

Como conclusión, el objetivo final del modelo que se utilizará en MORFEO es poder diseñar un sistema cuya lógica se parezca lo máximo posible al modelo de procesos del Cuaderno de Seguimiento. Este modelo deberá estar basado en una serie de técnicas de diseño avanzado que faciliten el acceso a datos para todo tipo de usuarios, la interoperabilidad entre sistemas y la extensibilidad y escalabilidad a corto y medio plazo.

3. Análisis y diseño del sistema

En este capítulo se obtendrá un conjunto de modelos lógicos que conforman el modelo dimensional del almacén, basándonos en algunos de los requisitos de *MORFEO-Seguimiento* y en los objetivos de *MORFEO-Almacén*.

Una vez identificados los procesos que tienen lugar en MORFEO-Seguimiento y su naturaleza, se procede al diseño del almacen, el cual a su vez se puede dividir en varias fases, encargándose cada una de ellas de la construcción de una parte del modelo final del almacén. Este proceso se puede considerar como una secuencia entre sus fases componentes, de manera que cada fase requiere de los resultados generados en la anterior para poder continuar con el proceso. Por ejemplo, la fase del modelo de datos genera una lista de todas las entidades fuente que servirán para crear las dimensiones conformadas en la siguiente fase. Una entidad, contextualizada en el modelo E/R, se define como un objeto concreto o abstracto, que existe en el problema a modelar, y sobre el cual se desea almacenar información. Para el caso de MORFEO, ejemplos de entidades son `PACIENTE`, `TRATAMIENTO`, `POLISOMNOGRAFIA` o `ECG`.

Una o varias de estas entidades (implementadas como tablas) se convertirán en una dimensión conformada, heredando los atributos de dicha entidad. La disposición de estas dimensiones conformadas, en torno a una

tabla de hechos, da como resultado un *data mart* y mediante la combinación de todos los marts generados, se obtiene el modelo final del almacén de datos. La figura 10 describe gráficamente el proceso de modelado para *MORFEO-Almacén*

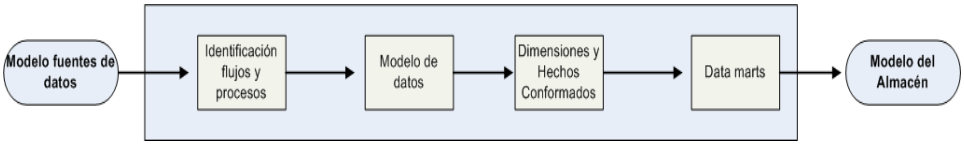

Figura 10: Proceso de modelado del almacén de MORFEO

3.1. Identificación de flujos y procesos

Como se explicó en el capítulo 1.3.1, el sistema *MORFEO-Seguimiento* resuelve la gestión clínica de pacientes mediante la aplicación de una serie de pautas clínicas, y la implementación de un *Cuaderno de Seguimiento*, que se puede definir como un conjunto de flujos de procesos que organizan las actuaciones médicas (pruebas, cuestionarios, diagnósticos y tratamientos) que se realizan a los pacientes. Así, a cada paciente ingresado en la unidad se le asigna un Cuaderno de Seguimiento que se divide en un conjunto de etapas secuenciales y a su vez, cada una de ellas, se corresponde con una serie de objetivos médicos a cumplir.

Cada de las anteriores etapas engloba un conjunto de pruebas y exámenes específicos que podrán repetirse a lo largo del proceso de seguimiento. Es por ello que cada uno de los *data marts* que componen el almacen siguen una correspondencia lógica secuencial con cada una de las etapas del Cuaderno de Seguimiento. Además cada una de las actuaciones médicas se corresponde con una dimensión conformada.

Esta correlación entre etapas, actuaciones, *data marts* y dimensiones permite generar un modelo dimensional extensible y adaptable a los cambios en el Cuaderno de Seguimiento (debido a la aplicación de Técnicas de Abstracción), de manera que si se crean nuevas etapas y actuaciones médicas, el almacén, como se ha mencionado anteriormente, podrá extendense creando nuevos *data marts*, dimensiones y hechos respectivamente, sin afectar a los objetos ya existentes en el almacén.

3.2. Modelo de datos

El modelo de datos es la segunda iteración en el proceso de consecución del modelo final de un almacén de datos. Esta fase trata de describir las

entidades fuente y su comportamiento.

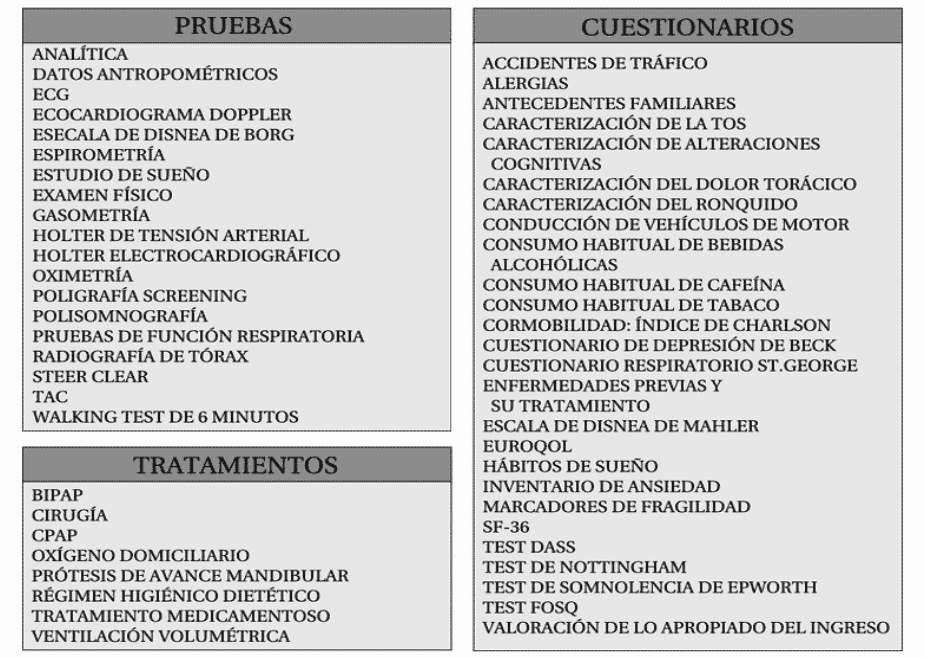

Figura 11: Actuaciones médicas de *MORFEO-Seguimiento*

Despues de analizar el modelo E/R de *MORFEO-Seguimiento*, se puede comprobar que existe un número elevado de entidades (más de 50) y que se podrían clasificar en dos grupos:

- 62 entidades que reflejan los resultados de las actuaciones médicas realizadas a pacientes.
- 7 entidades que implementan los flujos de las etapas del Cuaderno de Seguimiento.

La figura 11 recoge todas las actuaciones médicas (pruebas, cuestionarios y tratamientos) realizadas a pacientes bajo seguimiento. Cada una de estas actuaciones se corresponde con una o varias entidades del modelo de *MORFEO-Seguimiento*. Además, hay otras entidades como `PACIENTE`, `SEGUIMIENTO` o `DIAGNOSTICO`, que no son actuaciones médicas pero forman parte del proceso de seguimiento, y se combinan entre si para generar nuevas dimensiones.

El modelo E/R presentado en la figura 12 refleja las entidades y relaciones que componen el proceso de seguimiento clínico. Cada una de estas

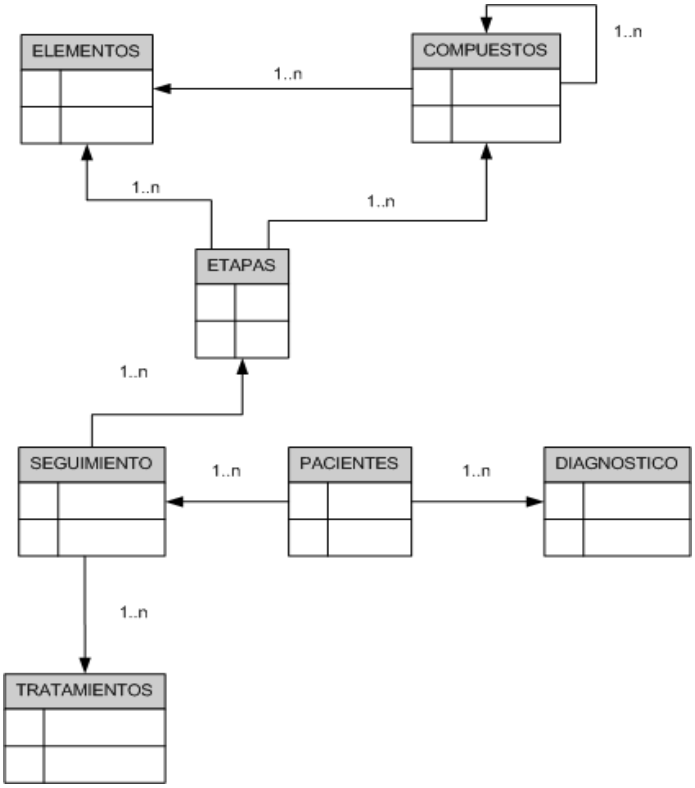

Figura 12: Modelo E/R de *MORFEO-Seguimiento*

entidades está formada por una serie de atributos que la definen. A continuación, y como ejemplo, se describe la entidad `PACIENTE`:

- Identificador de clave primaria: `ID NUMBER(19,0)`
- Iniciales: `IDENTIFICADOR VARCHAR2(255)`
- Sexo: `SEXO VARCHAR2(255)`
- Edad: `EDAD NUMBER(10,0)`
- Fecha de ingreso: `FECHA_INGRESO DATE`
- Procedencia del paciente (si se trata de su médico de cabecera o de urgencias): `PROCEDENCIA VARCHAR2(255)`
- Unidad del CHUS (de la que se ha desviado si procede): `UNIDAD VARCHAR2(255)`
- Identificador de diagnóstico: `DIAGNOSTICO_ID NUMBER(19,0)`

La tabla 1 presenta el modelo de datos completo en que se basa el modelo dimensional de *MORFEO-Almacén* y contiene una lista con todas las entidades que intervienen en el proceso de diseño del almacén:

Entidad	Tipo	Etapa
Analítica	Actuación médica	1 y 5
Datos antropométricos	Actuación médica	1
Electrocardiograma	Actuación médica	1 y 5
Ecocardiograma Doppler	Actuación médica	2
Escala de disnea de Borg	Actuación médica	2
Espirometría	Actuación médica	2
Estudio del sueño	Actuación médica	2
Examen físico	Actuación médica	1
Gasometría	Actuación médica	2
Holter de tensión arterial	Actuación médica	2
Holter electrocardiografico	Actuación médica	2
Oximetría	Actuación médica	1, 3 y 5
Poligrafía screening	Actuación médica	1, 3 y 5
Polisomnografía	Actuación médica	1, 3 y 5
Pruebas de función respiratoria	Actuación médica	2
Radiografía torácica	Actuación médica	1 y 5
Steer Clear	Actuación médica	2
Tomografía axial computerizada	Actuación médica	1
Walking test	Actuación médica	2
Accidentes de tráfico	Actuación médica	2
Alergias	Actuación médica	1 y 5
Antecedentes familiares	Actuación médica	1 y 5
Caract. de alteraciones cognitivas	Actuación médica	1 y 5
Caracterización de la tos	Actuación médica	1 y 5
Caracterización del dolor torácico	Actuación médica	1 y 5
Caracterización del ronquido	Actuación médica	1 y 5
Índice de comorbilidad de Charlson	Actuación médica	3
Conducción de vehículos	Actuación médica	2
Hábitos de consumo de alcohol	Actuación médica	1 y 5
Hábitos de consumo de tabaco	Actuación médica	1 y 5
Hábitos de consumo de cafeína	Actuación médica	1 y 5
Cuestionario de depresión	Actuación médica	2
Cuestionario respiratorio	Actuación médica	2
Enfermedades previas	Actuación médica	1 y 5
Disnea de Mahler	Actuación médica	2
EuroQoL	Actuación médica	2
Hábitos del sueño	Actuación médica	2
Inventario de ansiedad	Actuación médica	2
Marcadores de fragilidad	Actuación médica	3
SF-36	Actuación médica	1 y 5
Test DASS	Actuación médica	2
Test de Nottingham	Actuación médica	2
Test de Epworth	Actuación médica	2
Valoracion apropiada del ingreso	Actuación médica	2
BIPAP	Actuación médica	4
CPAP	Actuación médica	4
Cirugía	Actuación médica	4
Oxígeno domiciliario	Actuación médica	4
Prótesis de avance mandibular	Actuación médica	4
Régimen higiénico-dietético	Actuación médica	4
Tratamiento medicamentoso	Actuación médica	4
Ventilación volumétrica	Actuación médica	4
Pacientes	Proceso	Todas
Seguimiento	Proceso	Todas
Etapas	Proceso	Todas
Compuestos	Proceso	Todas
Elementos	Proceso	Todas
Diagnóstico	Proceso	Todas

Tabla 1: Modelo de datos

3.3. Dimensiones y hechos conformados

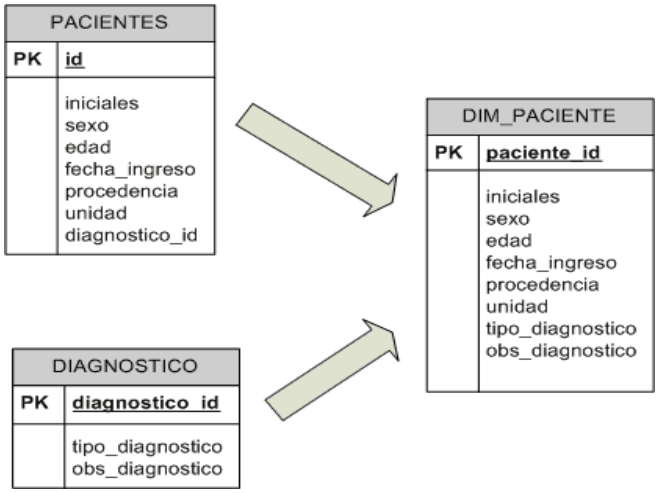

Figura 13: Construcción de la dimensión paciente

Las entidades del modelo de datos sirven como base para construir las dimensiones conformadas, según la arquitectura en bus para almacenes de datos. El proceso lógico en esta fase consiste en definir primero las dimensiones y a continuación los hechos surgirán de manera natural. A nivel conceptual y para el caso de MORFEO, cada prueba, cuestionario o tratamiento se corresponde con una dimensión conformada. Descendiendo a nivel de entidad se puede comprobar que esta correspondencia unitaria no se cumple para todos los casos, es decir varias entidades pueden conformar una dimensión. Generalmente, cuando dos entidades están vinculadas por una relación `1..n` suelen desnormalizarse y formar una dimensión.

Como ejemplo aclaratorio del anterior razonamiento: la dimensión `DIM_BIPAP` posee todos los atributos de la entidad `BIPAP`, completándose con un atributo de tipo `DATE`, pero también se da el caso de que una dimensión pueda aglutinar atributos de varias entidades, como por ejemplo el caso de `DIM_PACIENTES`, conformada por los atributos de las entidades `PACIENTE` y `DIAGNOSTICO`.

Debido a lo anterior, el proceso de creación de una dimensión conformada conlleva la selección de una serie de atributos bien definidos y la desnormalización de algunas entidades que componen el modelo relacional de *MORFEO-Seguimiento*. A nivel físico, los atributos de varias tablas se combinan en una sola tabla de dimensión, produciéndose el anterior fenómeno de desnormalización relacional aunque, como se ha comentado en secciones anteriores, el modelo dimensional no es más que un modelo relacional masivamente desnormalizado.

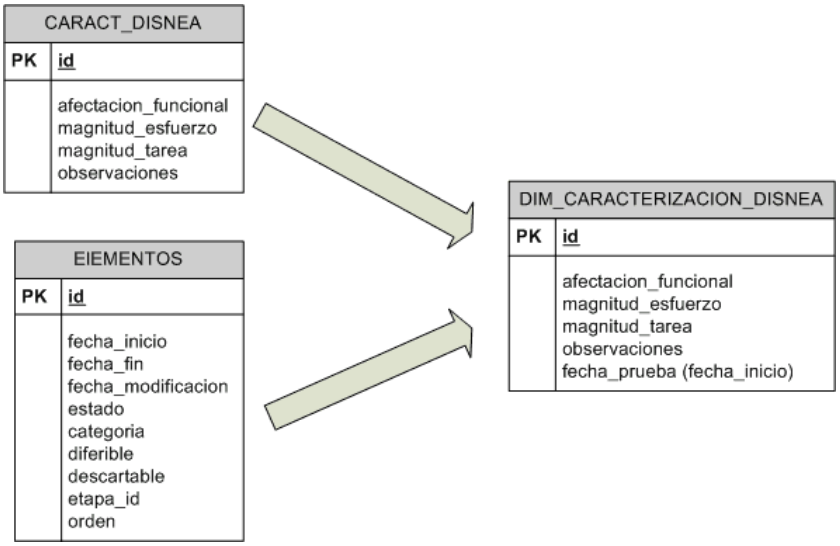

Figura 14: Construcción de la dimensión caracterización disnea

A continuación, y a modo de ejemplo, se describen dos dimensiones tipo de un total de 51 dimensiones conformadas que componen el modelo de *MORFEO-Almacén*. La primera es la dimensión `PACIENTE`, compuesta por dos entidades. En la figura 13 se pueden observar las entidades que conforman la dimensión y los atributos resultantes de la unión de ambas entidades.

La dimensión `CARACTERIZACION_DISNEA` se corresponde con un cuestionario específico. Esta dimensión estará formada por todos los atributos de su entidad correspondiente completada con un atributo de tipo `DATE` que indica la fecha de realización de dicho cuestionario. La figura 14 describe los atributos de dicha dimensión. La entidad `ELEMENTOS` aglutina todas las fechas de las actuaciones médicas realizadas, así que el atributo *fecha_inicio* se añade a la dimensión con el nombre de *fecha_prueba*. Esta tipología de dimensión es la más extendida en el almacén.

Una parte importante del tiempo empleado en esta fase se ha invertido en diseñar una dimensión tiempo o calendario que proporcione una perspectiva histórica del almacén además de servir como punto de referencia para integrar y analizar eventos médicos basados en fechas. Esta dimensión no está basada en ninguna entidad de *MORFEO-Seguimiento* y tiene que ser construida desde cero.

Es necesario analizar con detalle los requisitos temporales del almacén para poder diseñar una dimensión tiempo que se ajuste a los procesos de minería de datos temporal (MDT) y a la tipología de consultas *Ad Hoc* que se van a implementar.

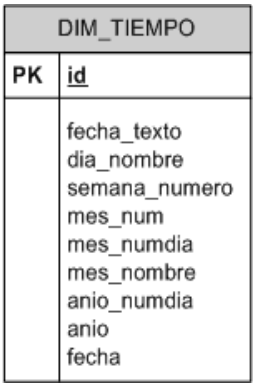

Figura 15: Construcción de la dimensión tiempo

Los atributos seleccionados para esta dimensión son:

- Identificador de clave primaria.
- Fecha completa en formato texto (dd/mm/aaaa), para poder ser utilizada en tareas de minería de datos.
- Nombre del día en formato texto (LUNES, MARTES, MIERCOLES etc).
- Número de semana con respecto al total de semanas del año.
- Número de mes con respecto al total de meses del año.
- Número de día respecto al mes correspondiente.
- Nombre del mes en formato texto (ENERO, FEBRERO etc).
- Número de día respecto al total de días de un año.
- Año actual.
- Fecha completa en formado `DATE` para poder ser referenciada desde funciones del SGBD que trabajan con fechas.

La figura 15 describe de manera gráfica la dimensión tiempo.

Mediante la combinación de los atributos de esta dimensión y una serie de cálculos sencillos, se pueden obtener consultas *Ad Hoc* con una componente temporal más compleja. Por ejemplo, se podría verificar rápidamente si el Servicio de Neumología ha tenido un número elevado de ingresos debido a determinados problemas respiratorios vinculados con la época de polinización de las gramíneas.

La tabla 2 contiene una lista con todas las dimensiones conformadas de *MORFEO-Almacén*:

Nombre	Descripción
DIM_ANALITICA	Analísis de sangre y orina
DIM_DATOS_ANTROPO	Talla, peso y medidas asociadas
DIM_ECG	Electrocardiograma
DIM_ECODOPPLER	Ecocardiograma Doppler
DIM_DISNEA	Escala de Disnea de Borg
DIM_ESPIROMETRIA	Prueba espirométrica
DIM_EST_SONO	Cuestionario de hábitos de sueño
DIM_EXAMEN_FISICO	Examen físico
DIM_GASOMETRIA	Gasometría
DIM_HOLTER	Holter de tensión arterial
DIM_HT_ELECTRO	Holter electrocardiografico
DIM_OXIMETRIA	Oximetría
DIM_SCREENING	Poligrafía screening
DIM_POLISOMNOGRAFIA	Polisomnografía
DIM_PRB_RESP	Pruebas de función respiratoria
DIM_RX_TORAX	Radiografía torácica
DIM_STEERCLEAR	Test de vigilancia de conducción de automóviles
DIM_TAC	Tomografía axial computerizada
DIM_WALKING_TEST	Prueba de resistencia
DIM_ACCIDENTES_TRAFICO	Cuestionario sobre accidentes de tráfico
DIM_ALERGIAS	Cuestionario de alergias
DIM_ANT_FAMIL	Cuestionario de antecedentes familiares
DIM_CARACT_TOS	Cuestionario para caracterizar la tos
DIM_CAR_ALT_COG	Caracterizar alteraciones cognitivas
DIM_CARACT_DOLOR_TORAC	Caracterizar el dolor torácico
DIM_CARACT_RONQ	Caracterizar el ronquido
DIM_VEHICULOS	Tipología de vehiculos conducidos
DIM_ALCOHOL	Hábitos de consumo de alcohol
DIM_TABACO	Hábitos de consumo de tabaco
DIM_CAFEINA	Hábitos de consumo de cafeina
DIM_COMORBILIDAD	Comorbilidad de Charlson
DIM_DEPRESION	Cuestionarrio de depresión
DIM_C_RESP	Cuestionario respiratorio
DIM_ENF_PREVIAS	Cuestionario sobre enfermedades previas
DIM_DISNEA	Disnea de Mahler
DIM_EUROQUOL	Cuestionario europeo sobre calidad de vida
DIM_SUENO	Hábitos del sueño
DIM_ANSIEDAD	Inventario de ansiedad
DIM_FRAGILIDAD	Marcadores de fragilidad
DIM_SF36	Cuestionario sociosanitario
DIM_DASS	Escala de ansiedad, depresion y stress
DIM_NOTTINGHAM	Auditoría respiratoria
DIM_FOSQ	Cuestionario de funcionalidad del sueño
DIM_EPWORTH	Escala de somnolencia
DIM_INGRESO	Valoracion apropiada del ingreso de un paciente
DIM_BIPAP	Tratamiento con dispositivo BIPAP
DIM_CPAP	Tratamiento con dispositivo CPAP
DIM_CIRUGIA	Tratamiento quirúrgico
DIM_OXIGENO	Tratamiento con oxígeno domiciliario
DIM_PROT_AVMAN	Trat. con prótesis de avance mandibular
DIM_HIXDIE	Trat. con régimen dietético
DIM_TRATAMIENTOS	Tratamiento medicamentoso
DIM_VENT_VOLUM	Tratamiento con ventilación volumétrica
DIM_PACIENTES	Registro de pacientes ingresados
DIM_TIEMPO	Registro de fechas

Tabla 2: Listado de Dimensiones Conformadas

3.4. Data marts

La arquitectura en bus para almacenes de datos permite obtener un modelo extensible y actualizable a lo largo del ciclo de vida del almacén. Esto es especialmente importante para MORFEO ya que actualmente se están definiendo nuevas pruebas y cuestionarios que formarán parte del Cuaderno de Seguimiento. Como consecuencia de esto, se crearán nuevas dimensiones conformadas, a partir de las anteriores pruebas, que se integrarán dentro del modelo dimensional. Si fuera necesario crear una nueva etapa, el mecanismo sería parecido al anterior, con la diferencia de que el modelo dimensional se vería extendido con un nuevo *data mart*.

Cada pauta clínica está constituida por un conjunto de etapas, que a su vez están formadas por varias actuaciones médicas. Para la implementación del Cuaderno de Seguimiento se han utilizado estructuras declarativas descritas en una gramática XML que determinan en consecuencia aquel conjunto de actuaciones médicas a realizar en cada etapa. Esto permite conocer a priori el número de dimensiones que intervienen en un *data mart* concreto.

Así pues, el Cuaderno de Seguimiento de MORFEO, engloba 5 etapas y se establecen correspondencias entre etapas, *data marts* y dimensiones. Esto quiere decir que cada etapa se modela mediante un *data mart* como mímino y que cada *data mart* consta de varias dimensiones.

Como se ha explicado anteriormente, un *data mart* está compuesto, a nivel físico, de una tabla de hechos y `n` tablas de dimensiones. La tabla de hechos está formada por un conjunto de claves externas, siendo cada una de éstas la clave primaria de una tabla de dimensión. Además, dicha tabla puede poseer varios *hechos*, que no son más que atributos de carácter numérico que reflejan una medida determinada. En el caso de *MORFEO-Almacén*, reflejan las fecha de inicio de una etapa determinada o el identificador de seguimiento asignado a un paciente en concreto.

En la figura 16 se describe el modelo dimensional del *data mart* `Primera Consulta` que cuenta con más de 26 dimensiones. El modelo del mart de `Evaluación Específica` se presenta en la figura 17. Las figuras 18, 19 y 20 respectivamente reflejan los modelos de las etapas restantes: `Nivel de Gravedad`, `Tratamiento` y `Revisión`.

Debido a la complejidad de ciertas etapas en términos de cardinalidad dimensional y granularidad, hay ciertas partes del Cuaderno que requieren la implementación de un mart. La regla más importante que tiene que cumplir un *data mart* es que debe ser un subconjunto natural del almacén, es decir, el conjunto de los marts modelados deben constituir el almacén al completo. Este es el caso del *data mart* `Tratamiento Medicamentoso`. Este último complementa al de `Tratamiento` con la parte de medicamentos

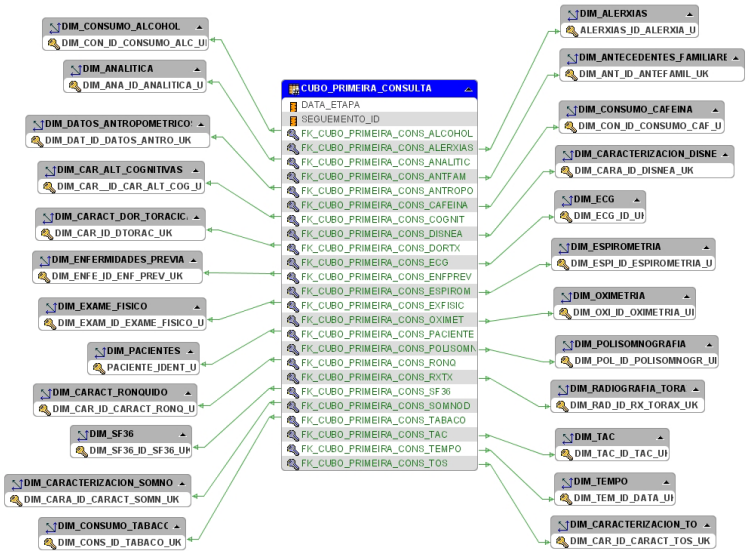

Figura 16: Data Mart de Primera Consulta

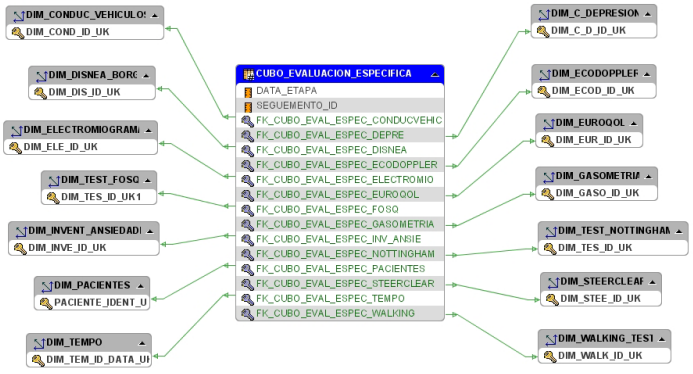

Figura 17: data mart de Evaluación Específica

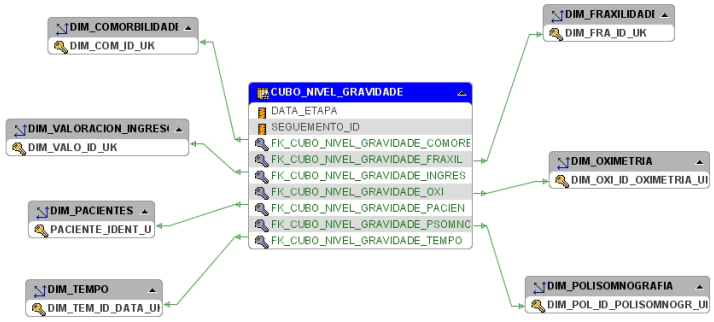

Figura 18: Data Mart de Nivel de Gravedad

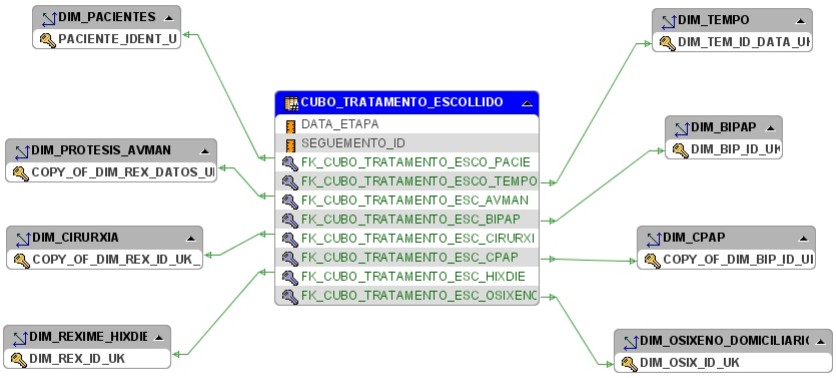

Figura 19: Data Mart de Tratamento

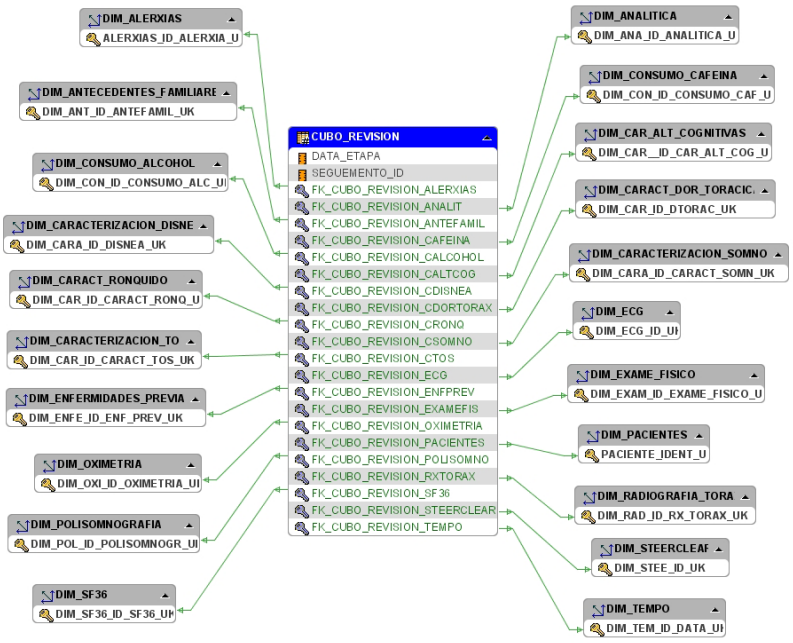

Figura 20: Data Mart de Revisión

administrados.

La granularidad del *data mart* `TRATAMIENTO_MED` es distinta de la del resto de *data marts*, donde un paciente, bajo un determinado seguimiento, es sometido a una serie de actuaciones médicas bien definidas. Esta relación es de tipo `1..1`, es decir, una fila en la tabla de dimensión `PACIENTES` se corresponde con una fila en la dimensión `CPAP`, para el *data mart* `TRATAMIENTOS`.

Para el caso de `TRATAMIENTO_MED`, la cardinalidad es del tipo `1..n`, es decir, una fila en la dimensión `PACIENTES` se corresponde con varias en la dimensión `TRATAMIENTO`. Debido a esta diferencia de granularidad, es necesario generar un *data mart* adicional que gestione los tratamientos medicamentosos administrados a pacientes bajo seguimiento. La figura 21 muestra el modelo del *data mart* `TRATAMIENTO_MED`.

3.5. Modelo dimensional del almacén

Las dimensiones conformadas son la base de la *arquitectura en bus para almacenes de datos* y es por ello que su análisis y definición forman una de las partes más importantes del diseño del almacén.

Una vez definidas las dimensiones y hechos conformados se puede com-

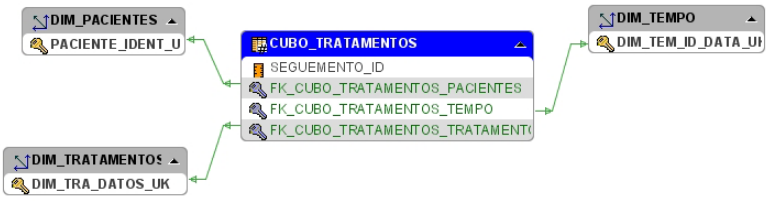

Figura 21: Data Mart Tratamiento Medicamentoso

probar que todos los *data marts* comparten dimensiones comunes. Éstas son las que dan coherencia y solidez al modelo, actuando como nexos de unión entre *data marts* y evitando así que éstos se conformen como entidades aisladas e inconexas. Ejemplos de dimensiones compartidas son `PACIENTES`, `TIEMPO` o `POLISOMNOGRAFIA` entre otras.

Además, las dimensiones conformadas permiten a las consultas *Ad Hoc* recorrer los distintos *data marts* que conforman el almacén en busca de datos. La información resultante de dichas consultas proporciona al usuario una visión general del almacén como un todo y no como piezas separadas.

La figura 22 representa un ejemplo del modelo dimensional o modelo *lógico* del almacén. Como se puede apreciar, las tres dimensiones ejemplo situadas en medio de la imagen, son compartidas entre los distintos marts. Aquí se puede distinguir claramente como éstas dimensiones articulan el concepto de *bus*, al que se refiere Kimball, como un sistema de comunicación entre *data marts*. La dimensión `DIM_PACIENTES`, al igual que la de `DIM_TIEMPO`, es compartida por los 6 *data marts* y la de `DIM_POLISOMNOGRAFIA` es compartida por 3. El resto de dimensiones coloreadas en gris son únicas para un *data mart* en concreto. Tal es el caso de `DIM_DASS` para el mart de `Evaluación específica` o de `DIM_ALCOHOL` para el mart de `Primera consulta`.

Estas dimensiones han sido seleccionadas como ejemplo didáctico sobre la totalidad que componen el almacén. Para tener una visión mas detallada de todas las dimensiones existentes y a que *data mart* pertenecen, se recomienda consultar:

- **Data marts**: Modelo dimensional detallado de los *data marts* en las figuras 16, 17, 18, 19, 20 y 21

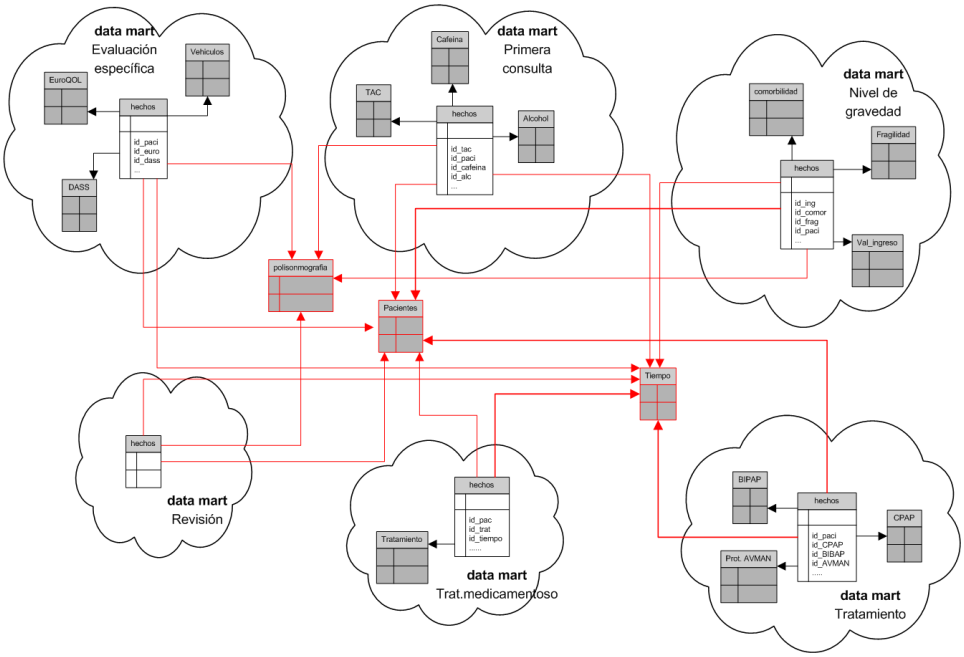

Figura 22: Modelo dimensional del almacén

- **Dimensiones**: Listado de entidades junto con sus correspondientes etapas y listado de dimensiones en las tablas 1 y 2

4. Implementación del sistema

En la sección anterior se han aplicado una serie de técnicas y patrones de diseño que han contribuido a la creación de varios modelos (de datos y dimensional) que conforman el modelo general del almacén. Dichos modelos sientan las bases para la implementación de los procesos ETL y la estructura física del almacén.

La implementación fisica del almacén mediante la tecnología ROLAP proporciona una serie de ventajas como el uso de estructuras y técnicas relacionales (tablas, índices, consultas, agregados) que facilitan la gestión y el mantenimiento del almacén, además de poder utilizar vistas minables en forma de tablas, lo cual facilita y agiliza el proceso de minado.

4.1. Granularidad y Hechos

Como se ha comentado en secciones anteriores, la granularidad es un parámetro muy importante a la hora de diseñar un almacén de datos. La granularidad afecta de distintas maneras al comportamiento de un almacén de datos:

- Desde una perspectiva de procesos, afecta a la flexibilidad y capacidad del almacén, es decir, si se escoge una granularidad demasiado alta, el almacén será incapaz de responder a las preguntas con un nivel de detalle inferior a la granularidad seleccionada.
- Desde una perspectiva técnica, el nivel de granularidad es uno de los mayores determinantes del tamaño de un almacén, el cual tiene un impacto significativo en cuanto a costes operativos y rendimiento.

Para el caso de *MORFEO-Seguimiento*, el nivel de granularidad más fino recae a nivel de actuación médica. Uno de los objetivos de MORFEO radica en estudiar determinados parámetros de las pruebas y cuestionarios realizados a pacientes e intentar descubrir conocimiento, mediante la aplicacion de técnicas de minería de datos y abstracción temporal. Este requerimiento condiciona el nivel de granularidad, siendo necesario el mayor nivel de detalle posible. Como consecuencia de esto, es necesario tener en cuenta el tamaño del almacén y sus repercusiones.

A pesar de esto y debido a que la cantidad de pacientes del almacén está creciendo lenta y sostenidamente, no hay problemas de rendimiento a corto plazo. Si en un futuro existiese algún problema de rendimiento, podrían aplicarse técnicas de *particionado*. Estas técnicas se aplican sobre el modelo físico del almacén, dividiéndolo en múltiples partes que lo hacen más manejable. El modelo físico (ROLAP) subyacente se ve alterado, no siendo lo mismo para el modelo lógico, que permanece intacto.

Las tablas de hechos de los *data marts* comparten la misma granularidad, pero puede ser necesario construir un *data mart* de distinta granularidad como el caso de `TRATAMENTO_MED`, explicado en la seccion anterior.

4.2. Dimensiones y Cubos

Hasta el momento, se ha hecho referencia a las dimensiones como estructuras lógicas que almacenan información categorizada para posteriormente ser consultada. En términos de implementación física en Oracle, una dimensión se corresponte con una tabla y un objeto dimensión.

En Oracle, el objeto dimensión permite organizar y agrupar la información dimensional en *jerarquías*. Esto representa relaciones naturales `1..n`,

entre columnas o grupos de columnas, denominadas *jerarquías de niveles*. Un nivel suele estar formado por una o varias columnas y un conjunto de niveles forman una jerarquía. Estas nuevas estructuras permiten que las consultas utilicen operadores OLAP como:

- **Detallar (*Drill*)**: se trata de disgregar los datos (mayor nivel de detalle o desglose, menos sumarización) siguiendo los caminos de una o varias dimensiones.
- **Agregar (*Roll*)**: se trata de agregar los datos (menor nivel de detalle o desglose, más sumarización) siguiendo los caminos de una o varias dimensiones.
- **Sumarizar (*Slice&Dice*)**: se seleccionan y proyectan datos de distintas dimensiones en una sola vista, formando un *collage* de información de las distintas dimensiones.
- **Pivotar (*Pivot*)**: se reorientan las dimensiones, cambiando filas por columnas. Imprescindible para tareas de minería de datos.

Así para una estructura dimensional en Oracle, subir de nivel en una jerarquía se denomina *roll up* y a bajar *drill down*. Por ejemplo:

- Para la dimensión tiempo, la jerarquía `TIEMPO_ROLLUP` define que los dias asciendan (roll up) a meses, los meses a años y los años a todos los años. En la figura 23 se representa la totalidad de la jerarquía.
- Para la dimension paciente, su identificador asciende al grupo de columnas con datos de su sexo y edad, éstos su diagnóstico y éste al hospital/unidad de procedencia.
- Para la dimensión tratamiento, un seguimiento de un paciente asciende a varios tratamientos medicamentosos.

Las dimensiones no tienen porqué ser definidas con la sentencia SQL `CREATE DIMENSION`. Pueden implementarse con sólo una tabla común. Pero teniendo en cuenta la plataforma relacional que utiliza MORFEO, parece razonable utilizar las estructuras dimensionales de Oracle, ya que es beneficioso en términos de rendimiento para determinados mecanismos de reescritura de consultas o para agregados (en Oracle se denominan vistas materializadas). El siguiente cuadro refleja un ejemplo de cómo se crea una estructura dimensional que alberga niveles y jerarquías para la dimensión tiempo.

```
CREATE  DIMENSION "DIM_TEMPO"
LEVEL  "MESES" IS  "DIM_TEMPO"."meses_mes_num"
```

Figura 23: Jerarquía ROLLUP de la dimensión Tiempo

```
LEVEL  "DATA" IS  "DIM_TEMPO"."data_fdata"
LEVEL  "SEMANAS" IS  "DIM_TEMPO"."semanas_semana_num"
LEVEL  "ANOS" IS  "DIM_TEMPO"."anos_ano"
LEVEL  "DIAS" IS  "DIM_TEMPO"."dias_mes_numdia"
LEVEL  "ID_DATA" IS  "DIM_TEMPO"."id_data_id"
HIERARCHY "TEMPO_ROLLUP"(
"DIAS" CHILD OF
"SEMANAS" CHILD OF
"MESES" CHILD OF
"ANOS" )
ATTRIBUTE "MESES" DETERMINES ("meses_mes_nome" )
ATTRIBUTE "DATA" DETERMINES ("data_datatexto" )
ATTRIBUTE "DIAS" DETERMINES ("dias_ano_numdia","dias_dia_nome" );
```

La tabla 2, presentada en la seccion anterior, dispone una lista completa de las dimensiones utilizadas en *MORFEO-Almacén*.

Sólo queda describir los *cubos* que acompañan a las dimensiones, que no son más que la denominación que utiliza Oracle para las tablas de hechos del almacén (previamente descritas en la sección de *data marts*). Dichos cubos o tablas de hechos, junto con las dimensiones, son construidos desde la herramienta Oracle Warehouse Builder que permite definir, crear y cargar las dimensiones y cubos que forman los *data marts*.

4.3. Proceso ETL

El siguiente paso en la implementación de un almacén es la carga de datos o proceso ETL. En realidad, la carga y mantenimiento del almacén es uno de los aspectos más delicados y que más esfuerzo requiere (casi la mitad del esfuerzo invertido en la implementación del almacén), y, de hecho, existe un sistema especializado para realizar estas tareas, denominado sistema ETL *(Extraction, Transformation and Load)*.

Este sistema es de tipo híbrido, ya que se han utilizado herramientas ETL como OWB (Oracle Warehouse Builder 10gR1) y además, se han codificado, en el lenguaje procedural de Oracle (PL/SQL), determinados procesos de generación y ordenamiento de datos que no han podido ser implementados mediente esta herramienta de Oracle.

El sistema ETL de MORFEO se encarga de realizar las siguientes tareas:

- Lectura de los datos transaccionales de *MORFEO-Seguimiento*.
- Creación de los objetos de base de datos de apoyo al proceso de carga (vistas, tablas temporales, funciones, procedimientos almacenados, triggers etc.)
- Creación de claves: generalmente es recomendable crear nuevas claves para el almacén (*surrogate keys*), ya que las utilizadas en el sistema transaccional suelen estar formadas por números y caracteres. Para el caso de la parte de seguimiento, sus claves son enteramente numéricas y están generadas por una secuencia Oracle (función que genera claves numéricas para objetos de base de datos), así que es interesante utilizar el mismo conjunto de claves para el almacén que para el sistema transaccional.
- Limpieza y transformación de datos: se realizan las tareas de limpieza y transformacion necesarias para organizar el almacén en base al modelo dimensional. Se trata de evitar datos redundantes, inconsistentes, estandarizar medidas, formatos, fechas, tratar valores nulos, etc. Por ejemplo, determinadas dimensiones como `DIM_CPAP` o `DIM_BIPAP` poseen campos de tipo `TIMESTAMP` que almacenan una fecha y una hora pero sólo hacen referencia lógica a una fecha. En el almacén sólo es relevante la granularidad fecha así que dicho campo se reformatea a un tipo `DATE`.
- Creación y mantenimiento de metadatos: para que el proceso ETL funcione es necesario crear y mantener los metadatos sobre el propio proceso ETL y los pasos realizados y por realizar. Esta tarea está automatizada por la herramienta Oracle Warehouse Builder, ya que al generar los procesos de carga se encarga de almacenar dichos metadatos.

Los procesos codificados manualmente también se pueden importar al repositorio de metadatos mediante la herramienta ETL.

- Planificación de la carga y mantenimiento: consiste en definir las fases, el orden y las ventanas de carga del almacén, con el objetivo de realizar la carga sin saturar el sistema transaccional y que no se violen las restricciones de integridad de los datos almacenados.

- Indexación: finalmente se crean los índices sobre las claves y atributos del almacén de datos que se consideran relevantes. Esta tarea también esta automatizada en parte por la herramienta ETL. Para las dimensiones, se crean índices en campos numéricos de los niveles que forman una jerarquía así como en la clave primaria de cada dimensión. Para las tablas de hechos se ha optado por un índice multicolumna que engloba a todas las columnas con claves foráneas y a su vez cada una de éstas columnas poseen un índice de tipo único (en Oracle `UNIQUE INDEX`).

Para poder satisfacer todas las funcionalidades anteriores es necesario utilizar la herramienta ETL de Oracle, con la cual se definen las tablas de hechos, dimensiones, procesos de limpieza, transformación y carga.

Las dimensiones ya han sido definidas en el modelo dimensional del almacén, con lo que sólo hay que crearlas desde la herramienta ETL e importarlas al repositorio del almacén.

El siguiente paso es la creación de los *cubos*[1], mediante OWB. La implementación física de un cubo se realiza mediante una tabla con tantos campos como dimensiones haya asociadas al cubo y `n` campos para las métricas de hechos. Actualmente `n=2` para todos los cubos siendo:

- Un campo de tipo `DATE` para la fecha de inicio de la etapa a la que haga referencia el cubo.

- Un identificador del proceso de seguimiento de pacientes de tipo `NUMBER`.

A medida que se vayan detectando nuevas métricas de hechos, mediante técnicas de abstracción o minería de datos, se irán añadiendo a los distintos cubos, propagándose los cambios en la siguiente carga del almacén.

En la figura 24 se pueden observar los cubos existentes en el almacén y en la 25, las propiedades (índices multicolumna y únicos) del cubo Evaluación Específica.

[1] Oracle denomina *cubos* a las tablas de hechos

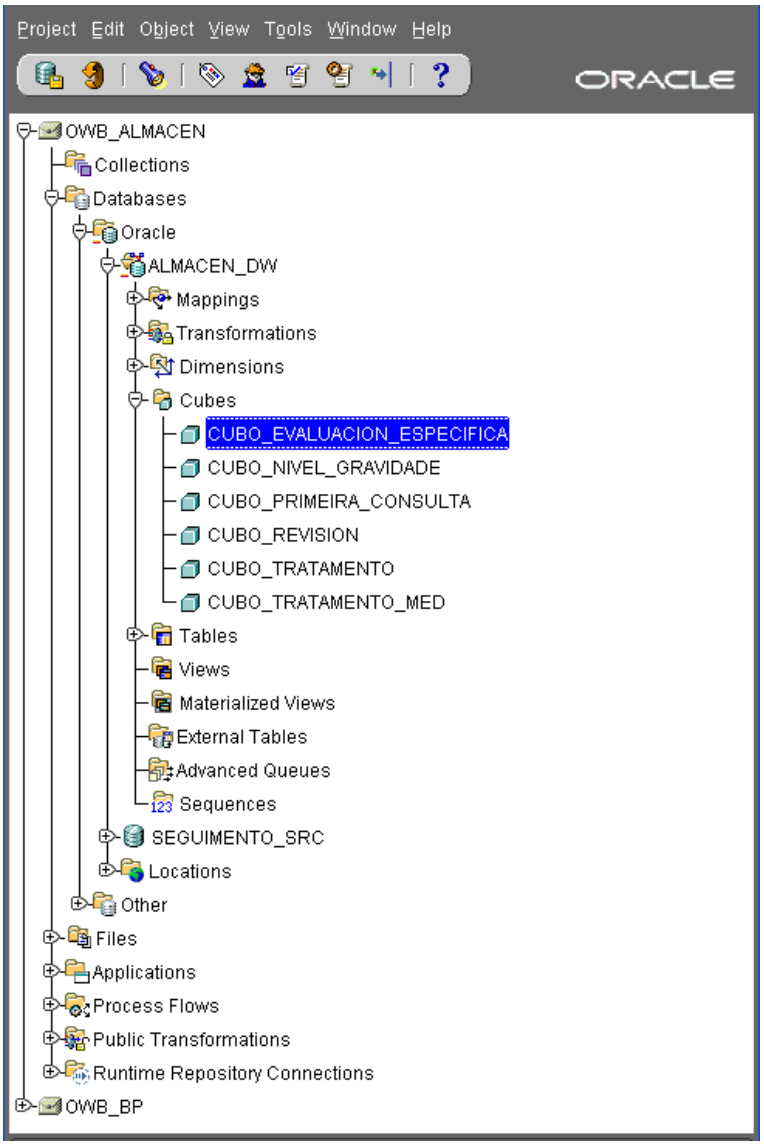

Figura 24: Navegación a través de los cubos que forman el almacén

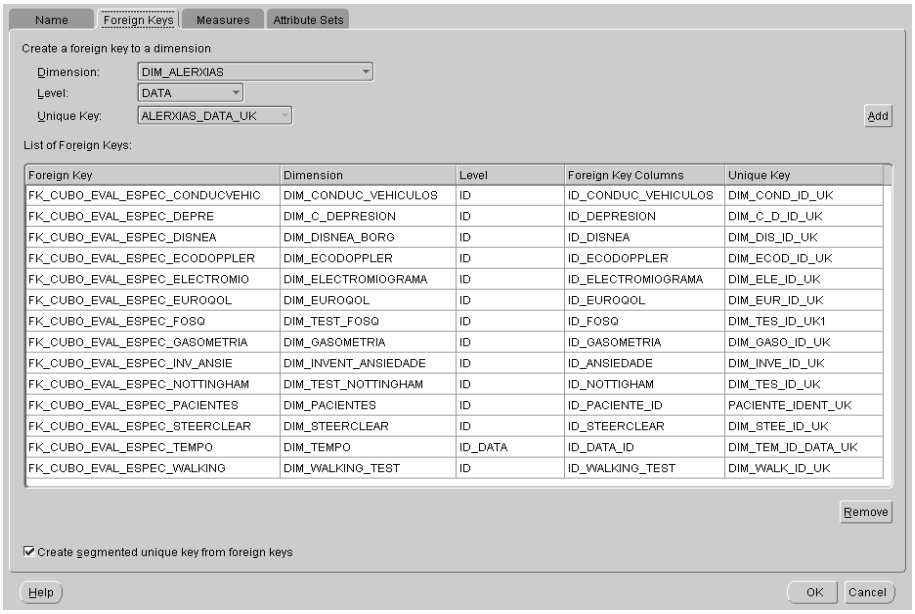

Figura 25: Propiedades del cubo Evaluación Específica

Una vez definidas las dimensiones y cubos (que forman los marts y que a su vez constituyen el almacén) se pasa a definir los *mapeos*, que serán los encargados de extraer, limpiar y cargar la información en el almacén.

La herramienta OWB permite crear dichos procesos ETL o mapeos, que describen una serie de operaciones a realizar sobre los datos para extraerlos de sus fuentes, transformarlos y cargarlos en el almacén. Los mappings proporcionan una representación visual del flujo de los datos y las operaciones a las que se someten. Una vez finalizada la creación de un mapeo, OWB lo convierte en una serie de objetos de base de datos (funciones y procedimientos almacenados) que luego ejecuta para realizar la carga.

Los *mapeos* del almacén están divididos en dos grupos, uno para la carga de las dimensiones y el otro para la carga de los cubos. La tarea de `CARGA_DIM` extrae los datos de la fuente trasaccional, los transforma y carga en las correspondientes dimensiones. La figura 27 muestra una parte del mapeo con las dimensiones `DIM_PACIENTES`, `DIM_ALERGIAS` y `DIM_ANTE_FAMIL` en donde se puede ver la correspondencia entre los campos de las tablas fuente y los de la dimensión.

La figura 26 muestra el interfaz principal de navegación de OWB y los cubos que han sido creados.

A continuación, en la figura 27, se detalla un ejemplo visual de la carga de la dimensión `DIM_ALERGIAS`,

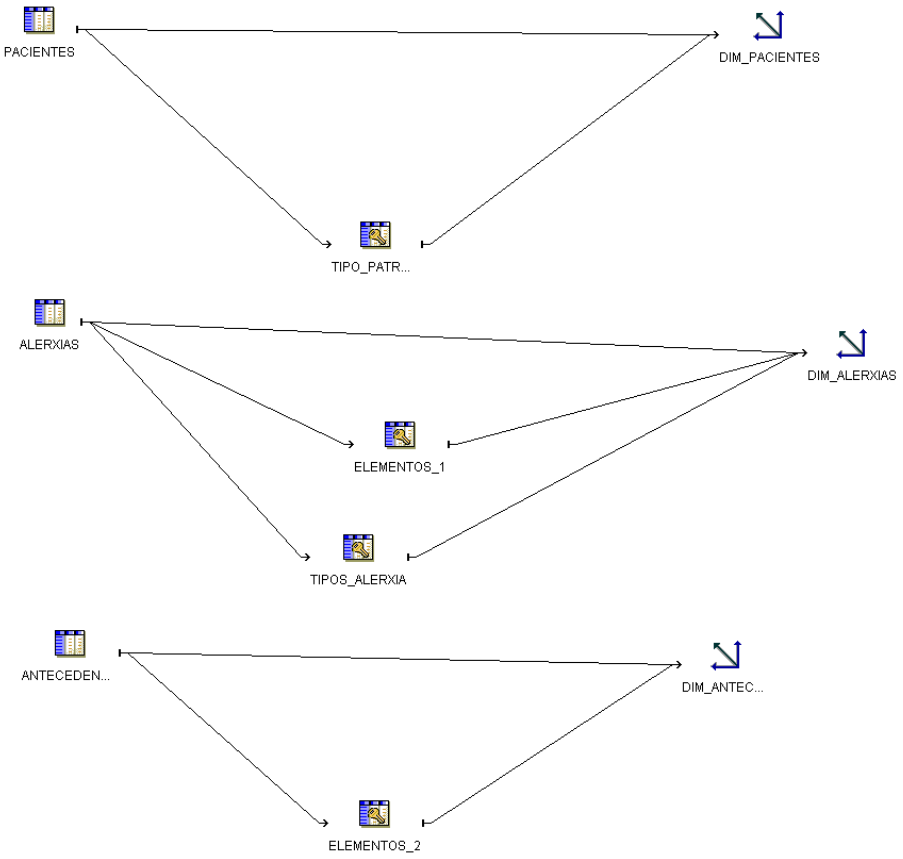

Figura 26: Mapeo de carga varias dimensiones: Alergias, Pacientes y Antecedentes Familiares

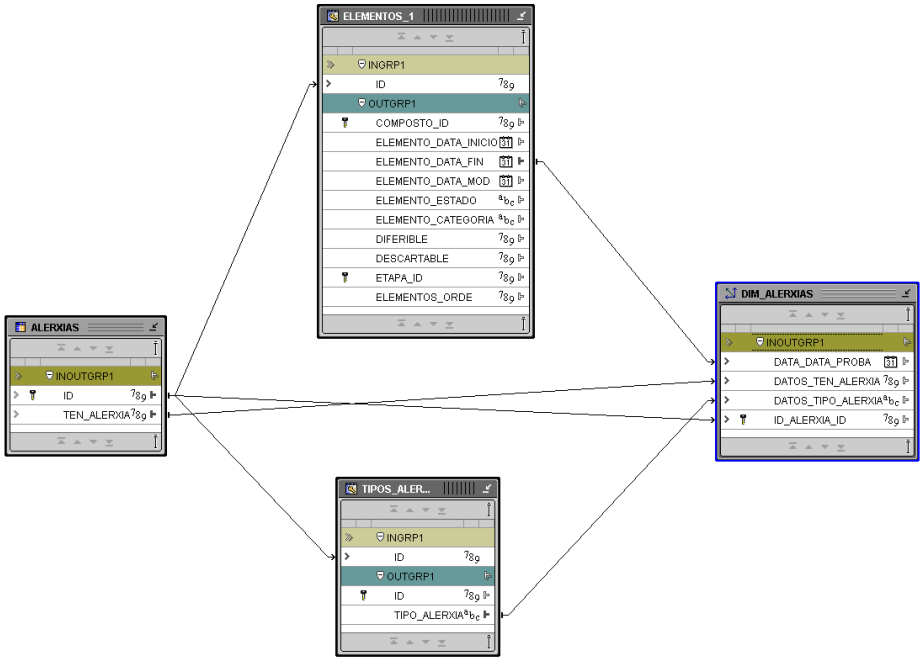

Figura 27: Carga de la dimensión Alergias

El siguiente grupo de mapeos engloba a los encargados de poblar los cubos del almacén. Hay tantos mappings como cubos o marts se definan, así que a medida que se extienda el almacén, deberán añadirse nuevos mapeos para implementar el proceso de carga de dichos cubos.

La carga de un cubo es un proceso un poco más complicado, ya que hay que mantener la integridad refencial de las dimensiones que lo rodean. Además hay que utilizar como fuente del proceso de carga varias tablas como `ELEMENTOS`, `COMPUESTOS` o `ETAPAS` y las dimensiones relacionadas. Se ha creado una vista por cubo que engloba los campos más relevantes de éstas tres tablas en base a unos criterios que son: la etapa a la que pertenecen determinadas pruebas y si han sido realizadas o descartadas.

En la figura 28 se muestra la estructura de la vista utilizada como fuente para iniciar el proceso de carga del cubo *Tratamiento*. Cada fila de dicha vista contendrá un identificador de paciente y un identificador de la actuación médica que se ha realizado para dicho paciente, además de datos muy útiles como la fecha de inicio de la etapa y la de la actuación entre otros.

El proceso de carga pasa por un estadio intermedio para el que se crea una tabla que contendrá todos los datos del cubo sin ordenar. Dicha tabla contiene una fila por actuación, lo cual genera una cardinalidad demasiado

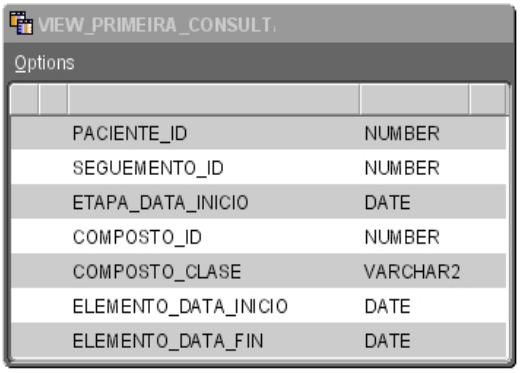

Figura 28: Vista de la etapa de Primera Consulta

alta para la estructura del cubo, cuya cardinalidad óptima debería ser una fila por paciente con todas las pruebas que se le hayan realizado en dicha etapa.

Es por ello que se ha creado un procedimiento almacenado denominado `SortMerge`, para optimizar dicha tabla intermedia ya que, se encarga de ordenar y fusionar las filas con el mismo identificador de paciente. De esta manera se reduce considerablemente el número de filas de la tabla y se procede a cargar el cubo de manera óptima y sin desperdiciar espacio en disco.

En la figura 29 se describe visualmente el proceso de carga del cubo mediante el diseño del mapeo correspondiente.

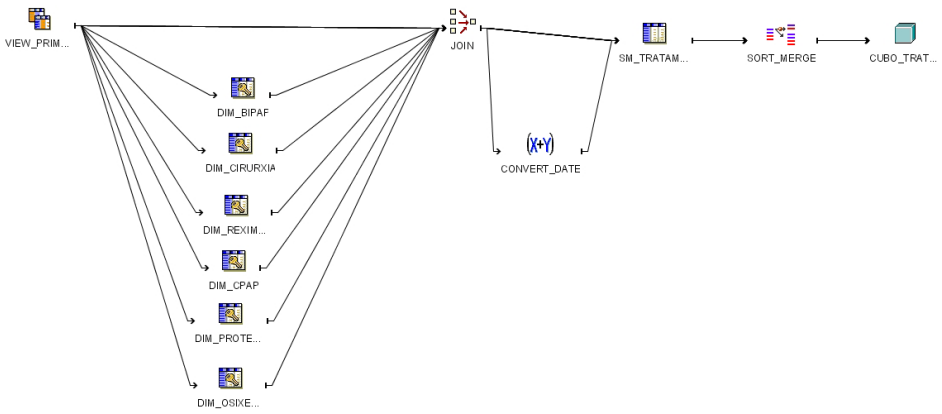

Figura 29: Carga del cubo Tratamiento

El procedimiento `SortMerge` ha sido codificado manualmente, utilizan-

do PL/SQL, e importado a la definición de procedimientos y funciones del OWB. Forma parte del proceso ETL junto con otros procedimientos y funciones. Consta de un esqueleto básico que se utiliza como base para codificar procedimientos de carga para nuevos cubos. Sólo cambia el número de variables locales que se utilizan (tantas como campos de claves foráneas tenga el cubo) en el cuerpo del procedimiento. Este procedimiento se modifica para adaptarlo a las necesidades de cada cubo y se importa al repositorio de metadatos de OWB para que pueda ser utilizado en el mapeo de carga del cubo.

Lo mismo ocurre con la función `convert_date_to_number`, la cual es la encargada de convertir una variable de entrada de tipo `DATE` en un entero que se utiliza como clave primaria en la dimensión tiempo. La definición de dicha función se importa en el repositorio de metadatos de OWB para poder ser utilizada en el proceso de carga de los distintos cubos.

```
CREATE OR REPLACE FUNCTION convert_date_to_number
(start_date IN DATE) RETURN NUMBER
IS
  datenum NUMBER(19,0);
BEGIN
  datenum:=to_number(to_char(start_date,'YYYYMMDD'),'99999999');
  RETURN datenum;
  EXCEPTION WHEN OTHERS THEN
  raise_application_error(-20001,'An error was encountered');
END;
```

Y por último, cabe destacar el procedimiento `Timedimension_Load`, el cual se encarga de cargar la dimensión tiempo, generando el rango de fechas con sus campos asociados como, por ejemplo, el número de semana y día del año. También es importado de la misma manera que los anteriores procedimientos.

4.4. Implementación física

En secciones anteriores se ha presentado la arquitectura del sistema y los modelos dimensionales de los marts. Debido al gran volumen de datos que maneja *MORFEO-Almacén*, es importante obtener un diseño físico que sea eficiente a la hora de acceder y almacenar los datos. Las técnicas de indexado, particionado y agregados son de gran utilidad para conseguir este objetivo.

4.4.1. Indexado

El *indexado* es una técnica muy importante para obtener un procesamiento eficiente de las consultas. A diferencia de los sistemas transaccionales como *MORFEO-Seguimiento*, que tienen un alto porcentaje de operaciones de actualización, los almacenes tienden a leer grandes cantidades de datos para poder responder a las consultas. Es por ello que es importante comprender las necesidades de indexado [7] de un almacén como el de MORFEO y así poder decidir el tipo de índices y en qué tablas de dimensión o hechos deberán ser creados.

Las desventajas de los índices son:

- Consumen bastante espacio en disco, que suele ser igual o superior al espacio que ocupan los datos almacenados.
- Sobrecargan los procesos de mantenimiento del almacén, ya que estas estructuras consumen tiempo de procesamiento a la hora de generarlas y gestionarlas.

Lo anterior es especialmente cierto para las tablas de hechos con índices de tipo *B*tree*. Este tipo de índice se implementa mediante estructuras en árbol que permiten buscar rápidamente una fila, la cual tiene asignado un valor de la clave del índice. Este tipo es muy utilizado en *MORFEO-Seguimiento*, en donde cada clave primaria o foránea implementa un índice de este tipo.

La tipología de índices *Bitmap* [2] está diseñada para consultas que seleccionan columnas con un rango reducido de valores (por ejemplo sexo del paciente) pero con un número de filas muy elevado. Mientras que los *B*tree* almacenan punteros a las filas de las tablas, los *Bitmap* guardan un mapa de bits por cada valor distinto de una columna, ocupando así menos espacio en disco. En entornos OLAP es muy recomendable utilizar siempre índices *Bitmap*, por su ahorro en espacio y rendimiento aceptable.

Las columnas de una dimensión, que formen parte del criterio de selección, o que sean niveles, suelen ser buenas candidatas para la creación de índices. También lo son las que frecuentemente son referenciadas en la cláusula `WHERE` del lenguaje SQL.

Es por ello que la estrategia de indexado para *MORFEO-Almacén* comprende la creación de índices en:

- columnas seleccionadas y referenciadas por consultas.
- todos los niveles de las dimensiones de *MORFEO-Almacén*.

- atributos de tipo `BOOLEAN` o con pocos valores, como el sexo de un paciente.
- tablas de hechos con columnas con claves foráneas. Además cada una de éstas tablas tendrá un índice multicolumna formado por todas las claves foráneas.

Para complementar la anterior estrategia de indexado, se han utilizado varias técnicas de optimización de consultas para mejorar el rendimiento, entre las que destaca la *transformación en estrella* (*star transformation*). Esta técnica, esta basada en la premisa de combinar columnas de la tabla de hechos con índices *Bitmap* y columnas de las tablas de dimensión que intervienen en la consulta. Primero se utilizan los índices *Bitmap* para recuperar las filas necesarias de la tabla de hechos. Al anterior conjunto de filas se le aplica una operación *join* con las tablas de dimension para obtener los resultados de la consulta.

Una consulta que ejemplifica la técnica anterior: listado de los pacientes ingresados en el año 2007 y que les ha sido realizado un examen físico en la fase de primera consulta. La tabla de hechos de Primera Consulta implementa un índice *Bitmap* en cada una de sus columnas con clave foránea (PACIENTE_ID, EXAMEN_ID e ID_DATA). Éstos índices son utilizados por el optimizador para obtener un conjunto de filas a las que luego aplicar una operación *join* con las dimensiones de Paciente, Examen Físico y Tiempo. A continuación se detalla el código de la consulta ejemplo:

```
SELECT p.sexo_id, p.ident_id, p.diagnostico_obs
FROM dim_paciente p, dim_examen_fisico e,
   dim_tiempo t, cubo_primera_consulta cp
WHERE p.paciente_id = cp.paciente_id AND
     cp.id_examen NOT NULL AND
     cp.id_data = t.id_data AND
     t.anio = '2007'
GROUP BY sexo_id;
```

El optimizador de Oracle, mencionado en el párrafo anterior, es el encargado de utilizar esta técnica para las consultas que se ejecutan en el almacén, proporcionando transparencia a usuarios y aplicaciones.

4.4.2. Agregados

Anteriormente había hablado de los *agregados* como una técnica de optimización de consultas utilizada para obtener unos tiempos de respuesta aceptables. Oracle 10g denomina a este tipo de tecnología *vistas materializadas*, que define como un objeto de base de datos que precalcula y almacena el resultado de una consulta, como una tabla resumen. Una tabla resumen

suele ser generada por una consulta agregada, aunque se pueden crear vistas materializadas para cualquier consulta. A estos efectos, una vista materializada es muy similar a una vista convencional. La diferencia es que la vista materializada almacena los resultados en la base de datos mientras que la vista los calcula en tiempo de consulta. Otra ventaja de las vistas materializadas es que pueden utilizar un mecanismo de reescritura de consultas propio de Oracle llamado *query rewrite*. Este mecanismo permite al SGBD reescribir las consultas para que transparentemente utilizen las vistas materializadas que se hayan creado, con el consiguiente

A continuación se detalla un trozo de código de una vista materializada creada para la consulta: "Listado de los pacientes ingresados en los últimos 4 años agrupados por sexo y que les ha sido realizado un examen físico en la fase de primera consulta". En este ejemplo también se comentan los parámetros básicos que definen el comportamiento de dicha vista materializada, como el tiempo de refresco de los datos o si la vista utiliza el mecanismo de reescritura de consultas.

```
CREATE MATERIALIZED VIEW consulta
PCTFREE 0 TABLESPACE summary
STORAGE (initial 64k next
64k pctincrease 0)   <- Parámetros de almacenamiento
BUILD IMMEDIATE      <- Cuando se construye
REFRESH FORCE        <- Como se refresca
ON DEMAND            <- Cuando se refresca
ENABLE QUERY REWRITE <- Uso de query rewrite
AS
SELECT p.sexo_id, p.ident_id, p.diagnostico_obs
FROM dim_paciente p, dim_examen_fisico e,
   dim_tiempo t, cubo_primera_consulta cp
WHERE p.paciente_id = cp.paciente_id AND
     cp.id_examen NOT NULL AND
     cp.id_data = t.id_data AND
     (t.anio = '2007' OR
     t.anio = '2006' OR
     t.anio = '2005' OR
     t.anio = '2004')
GROUP BY sexo_id;
```

No todas las consultas que se realizan en *MORFEO-Almacén* implementan una vista materializada. Las consultas, que se repiten con frecuencia entre las distintas tareas de análisis del almacén y que suelen consumir muchos recursos, se perfilan como claras candidatas para implementar una vista materializada. Además los datos de dicha vista materializada pueden ser utilizados por otras consultas gracias al mecanismo de reescritura anteriormente mencionado, con el consiguiente ahorro de tiempo y recursos del sistema.

4.4.3. Particionado

El *particionado*, es otro factor que juega un papel muy importante en la obtención de una implementación física eficiente. Ésta técnica se basa en el paradigma de "divide y vencerás", en donde los objetos de base de datos como, tablas, índices, dimensiones etc, se dividen en partes más pequeñas y manejables por el SGBD. La ventaja más importante del particionado es que muchas operaciones de mantenimiento como carga de datos, creación de índices, limpieza de datos, recolección de estadísticas para el optimizador de consultas, *backups*, etc pueden ser realizadas a nivel de partición, no siendo necesario hacerlo a un nivel superior.

La eficiencia a la hora de acceder a datos se ve incrementada ya que por ejemplo, no es lo mismo en términos de rendimiento que una consulta recorra una tabla con 20 millones de registros buscando una fila, que hacerlo sobre una partición de dicha tabla con sólo unos cuantos miles de registros. Es por ello que el particionado está directamente relacionado con la cantidad de datos almacenados y su distribución entre las dimensiones.

El particionado se puede aplicar en base a distintos criterios, como por ejemplo el temporal (particionando los pacientes por años, meses etc) o por género (particionando los pacientes por sexo) etc, dependiendo en gran medida de las tareas de análisis y tipología de consultas que se estén aplicando. En la figura 30 se puede observar un posible particionado temporal en el que cada partición aglutina los datos de un año.

Actualmente y debido a que la cantidad de datos de *MORFEO-Almacén* está creciendo lenta pero sostenidamente, no es necesario aplicar ninguna estrategia de particionado, ya que el número máximo de registros en las tablas de hechos es del orden de decenas de miles y todavía no se ha superado el umbral del millón.

El optimizador de consultas de Oracle es el encargado de sugerir si es necesario particionar determinadas tablas, haciendo que el sistema esté preconfigurado para responder en un futuro a esta demanda.

5. Acceso a datos

Esta sección proporciona una visión detallada de los tipos de usuarios y herramientas que explotan los datos de *MORFEO-Almacén* en busca de información que mas tarde se puede transformar en conocimiento. Dichas herramientas ejecutan transparentemente las consultas Ad Hoc que hayan sido definidas y presentan sus resultados de manera visual y/o con múltiples elementos gráficos.

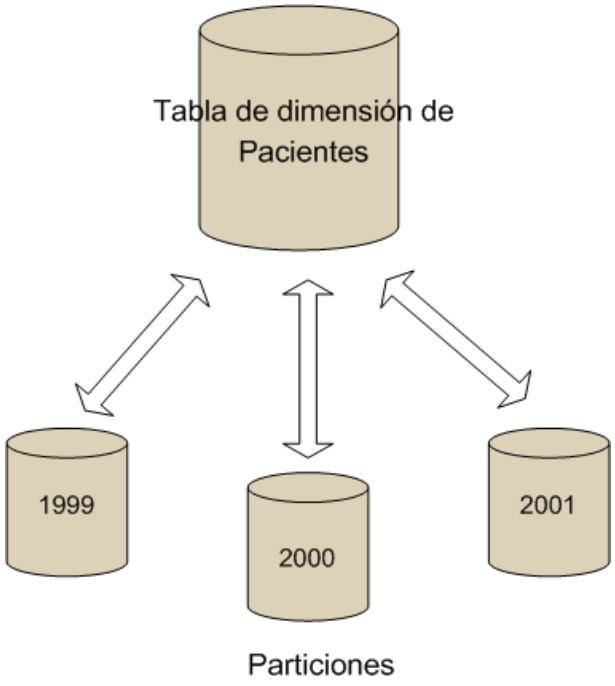

Figura 30: Particionado temporal de la dimensión paciente

Como se ha comentado en el capítulo anterior, el tiempo de respuesta las consultas está intimamente ligado a las optimizaciones y a la implementación física del almacén. Se utilizan técnicas como la transformación en estrella o técnicas de agregados para obtener tiempos de respuesta bajos en las consultas más utilizadas en *MORFEO-Almacén*. A medida que se vayan definiendo nuevas consultas se irán creando nuevos agregados, en función del tiempo de respuesta de las mismas y de su complejidad.

5.1. Tipología de usuarios

En *MORFEO-Almacén*, nos encontramos con que hay varios perfiles [17] de usuarios que realizan tareas distintas. Esta tipología de usuarios permite catalogar las consultas realizadas en el almacén en base a tres perfiles:

- **Exploradores**
- **Granjeros**
- **Mineros**

El comportamiento de los exploradores es totalmente impredecible, ya que se dedican a buscar patrones que pueden existir o no en base a su propio criterio y experiencia. Los exploradores suelen generar información de calidad y en pocas cantidades, es decir, normalmente no obtienen resultados, pero cuando los obtienen, suelen descubrir patrones significativos que retroalimentan el almacén y sirven de plataforma para reenfocar las tareas de análisis. Un explorador se formula preguntas del tipo "¿Como son los pacientes ingresados en el servicio? (tipología de pacientes). ¿Existe estacionalidad en el ingreso de pacientes?."

Las anteriores razones conducen a que este tipo de usuario ejecute consultas sobre grandes volúmenes de datos que consumen recursos y tiempo del sistema. Además, no es posible construir agregados debido al comportamiento *heurístico* del explorador (el investigador no sabe cúal será el siguiente paso en el análisis hasta que estén listos los resultados del paso actual). Es por ello que la herramienta más útil para esta tipología de usuario es *SQL*Plus* u *Oracle Discoverer*, ya que permiten al médico explorar y consultar los datos de una manera parcialmente interactiva (roll y drill).

Los granjeros, se dedican fundamentalmente a realizar informes periódicos, ver la evolución de indicadores, controlar valores anómalos, etc. Se corresponden con un perfil más *previsible* ya que suelen ejecutar la misma tipología de consultas de manera rutinaria. Dichas consultas suelen ser localizadas (acceden directamente a los datos que necesitan), consumen pocos recursos y están optimizadas con agregados. Ejemplos de preguntas que se formula un granjero: "¿Cuales son los medicamentos más frecuentemente recetados para un diagnóstico de Síndrome de Apnea/Hipopnea del Sueño?. ¿Cuantas polisomnografías se realizan semanalmente? (para evaluar el rendimiento de la máquina polisomnográfica)".

La herramienta utilizada por este perfil es el Oracle Discoverer, que proporciona una lista de consultas predefinidas. Dichas consultas son definidas previamente por el administrador de *MORFEO-Almacén* en conjunción con el granjero y posteriormente son ejecutadas de una manera rutinaria por este último.

El minero trata de buscar patrones y secuencias en los datos para identificar nuevo conocimiento. Suele operar de manera coordinada con el explorador, es decir, el explorador define las hipótesis y aserciones y el minero caracteriza la probabilidad de validez las mismas. Las preguntas que el minero se suele formular son: "En base a la tipología de pacientes ingresados, ¿Empeoran a lo largo del tiempo?, ¿Mejoran tras la aplicación de determinado tratamiento?". Actualmente se está trabajando con técnicas de clasificación (árboles de decisión y técnicas de *clustering*) de pacientes en uno de los cuatro grupos de dolencias cardiopulmonares del sueño. La herramienta que se está utilizando es Oracle Data Miner.

Para finalizar, a pesar de que ambos perfiles (explorador y minero) comparten determinadas tareas de análisis, cabe destacar las diferencias entre ambos perfiles para *MORFEO-Almacén*. El explorador realiza un *análisis clásico*, basado en la agregación, la visualización y técnicas de estadística descriptiva . Las tareas del minero entran dentro del perfil genuino de la *minería de datos*, que no transforma los datos en otros datos sino que los transforma en conocimiento (reglas o modelos). Además, el nivel de agregación para los requerimientos de un análisis OLAP puede ser mucho más grueso que el necesario para la minería de datos. Por ejemplo para un análisis OLAP puede ser suficiente utilizar el día (realización de una prueba polisomnográfica) como unidad mímina de tiempo y para tareas de minería de datos suele ser más interesante tener un nivel más fino como el minuto o segundo (duración en segundos de una apnea registrada durante una polisomnografía).

5.2. Ejemplos de consultas

Actualmente, tanto el pefil de explorador como el de granjero utilizan *Oracle Discoverer* y *SQL*Plus* para poder ejecutar consultas Ad Hoc sobre el almacén. Antes de detallar algunas de las consultas que ambos perfiles están realizando al almacén, es necesario tener en cuenta que *MORFEO-Almacén* tiene cargados a día de hoy 151 pacientes, junto con sus actuaciones médicas y tratamientos asociados. La capacidad del sistema ronda los 12 Gb y crece sostenidamente. Con este nivel de carga de datos, se pueden abordar tareas de análisis preliminar; a medida que vaya creciendo el almacén se podrán obtener resultados de mayor relevancia estadística.

Una consulta típica del perfil de un granjero es: "*¿Cuales son los medicamentos más frecuentemente recetados para pacientes en la etapa de Primera Consulta con un diagnóstico de Síndrome de Apnea/Hipopnea del Sueño?*". A continuación se detalla el código SQL de la misma:

```
SELECT tratamento_nome_comercial, count(*)
FROM dim_tratamentos t, cubo_primeira_consulta cp, dim_pacientes p
WHERE
t.seguemento_id = cp.seguemento_id AND
cp.id_paciente_id = p.id_paciente_id AND
p.diagnostico_patron_diag = 'apneaSueno'
GROUP BY t.tratamento_nome_comercial
```

La ejecución de dicha consulta produce los siguientes resultados (Para limitar el tamaño de la tabla se ha utilizado una muestra pequeña, pero significativa, del conjunto resultado):

Tratamiento	Frecuencia
adiro	2
cardyl	1
zarator	1
aprovent	1
plusvent	2
rivotril	1
sirdalud	1
tenormyn	1
liparison	1
pulmicort	1
alprazolam	1
carduran neo	1
tromalyt	1
lexatin	3
antihipertensivos	3
norvas	2
renitec	4
lantanon	2
trankimazin	2

Tabla 3: Listado de medicamentos

Esta consulta abre una línea de investigación interesante ya que con estos resultados se pueden deducir varias premisas:

- Que muchos pacientes afectados por esta dolencia toman medicamentos como Renitec o Lexatin que, son medicamentos generalmente caros para la sanidad pública. Es por ello que dichos medicamentos podrían ser sustituidos por medicamentos genéricos con el mismo principio activo pero con un coste mucho menor.
- También se puede observar que los enfermos toman bastantes ansiolíticos y tranquilizantes (trankimazin, alprazolan, lantanon, etc) que podría ser debido a la falta de sueño nocturno asociada a las apneas.

Otra consulta interesante lanzada por exploradores: "*¿Cuál es el porcetaje de pacientes que pertenece a cada uno de los 4 tipos de enfermedades cardiopulmonares del sueño?*". Los resultados de la consulta se presentan en la figura 31. Estos resúmenes o informes con elementos gráficos (*OracleBI SpreadSheet* y *Oracle discoverer*) proporcionan una visión global sobre la tarea de análisis y una interpretación de los resultados más precisa.

Como se puede observar, hay un alto porcentaje de pacientes con el Síndrome de Apnea Hipopnea del Sueño. Este resultado puede estar asociado a factores como hábitos alimenticios, predisposición genética, etc. Esta consulta sirve como plataforma de salto hacia otras líneas de análisis, que

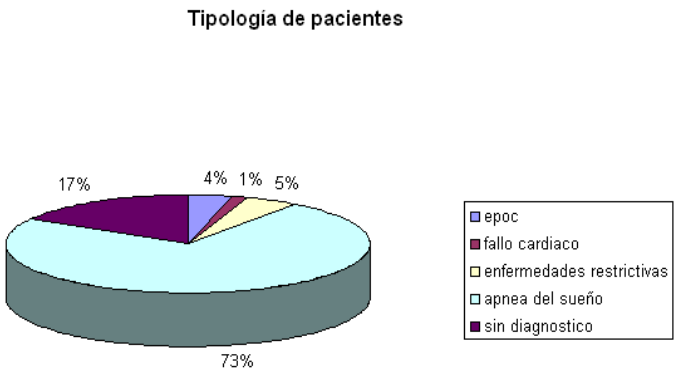

Figura 31: Tipología de pacientes

traten de descubrir más información sobre la problemática de tan alto porcentaje.

Mediante estos ejemplo de consultas se puede obtener una visión global de las posibilidades que *MORFEO-Almacén* aporta como plataforma el análisis de los datos adquiridos a pacientes ingresados en un servicio de Neumología.

6. Conclusiones y trabajo futuro

MORFEO-Almacén es un sistema orientado al estudio de pacientes con alteraciones cardiopulmonares el sueño, ha sido implantado, junto con *MORFEO-Seguimiento*, en el Servicio de Neumología del Hospital Clínico Universitario de Santiago de Compostela y está siendo utilizado por el personal clínico de este servicio.

La implementación de *MORFEO-Almacén* ha sido una tarea complicada ya que está sujeto a un gran número de requisitos y precondiciones a evaluar, así como satisfacer una serie de objetivos relacionados con el análisis de los datos almacenados.

Como se ha comentado en la sección 1.3.2, el proyecto SIESTA aporta un marco de trabajo insuficiente, en términos de almacenamiento de datos. Las fuentes de datos son heterogéneas, y es necesario almacenar todos los resultados de cuestionarios y pruebas médicas realizadas a pacientes, así como los tratamientos administrados.

La tipología de dichas pruebas es muy amplia, desde cuestionarios donde la información es recogida en formato texto, hasta pruebas médicas en las que se registran parámetros fisiológicos de pacientes como polisomnografías o ECGs. Además, dichas pruebas suelen repetirse en distintos períodos de tiempo, para poder evaluar el estado fisiopatológico del paciente y comprobar su mejoría o empeoramiento, despues de aplicar un tratamiento. Debido a lo anterior es necesario un sistema de almacenamiento masivo, estructurado, que soporte datos de distinta naturaleza (numéricos, texto, binarios, etc) y con una componente temporal de gran importancia. Es por ello que un sistema de ficheros no es la solución más adecuada para este problema. Un SGBD satisface la anterior problemática de almacenamiento además de proporcionar escalabilidad y rendimiento.

MORFEO-Almacén es un sistema complejo, pensado para aplicar técnicas computacionales y de abstracción temporal sobre los datos adquiridos a pacientes. *MORFEO-Seguimiento* ha sido diseñado para que aglutine los procesos de adquisición de datos bajo un mismo entorno OLTP. Este sistema no está pensado para tareas de caracter analítico, las cuales necesitan un modelo estructural distinto al utilizado en la parte de seguimiento, y está optimizado para inserciones, borrados y actualizaciones. Por el contrario, los sistemas OLAP están optimizados para operaciones de consulta además de proporcionar un modelo estructural y un conjunto de operadores específicamente diseñados para análisis de datos en sistemas de almacenamiento masivo.

Las pautas clínicas asociadas al Cuaderno de Seguimiento pueden cambiar con el tiempo, modificándose o creándose nuevas pautas, etapas o actuaciones médicas. A su vez, *MORFEO-Almacén* está pensado para aplicar técnicas computacionales y de abstracción temporal sobre los datos almacenados. Dichas técnicas proporcionan información que posteriormente es validada e incorporada a la rutina clínica en *MORFEO-Seguimiento*, retroalimentando ambos sistemas. Es por ello que *MORFEO-Almacén* ha sido diseñado con técnicas orientadas al modelado de procesos (Arquitectura en Bus para Almacenes de Datos). Esto hace que el almacén se articule como un conjunto de marts que modelan cada una de las etapas y procesos que implementa el Cuaderno de Seguimiento, dando como resultado un modelo totalmente extensible.

MORFEO-almacén proporciona un acceso a datos universal y transparente mediante distintas interfaces (JAVA, PL/SQL, interfaces gráficas como Oracle Data Miner o Oracle Discoverer, etc). Dichas interfaces facilitan la interoperabilidad entre sistemas ya que están basadas en distintos estándares comunmente adoptados por los SGBDs. Además éste proporciona escalabilidad en el acceso a datos, ya que multiplexa y cachea las conexiones al sistema haciendo que el rendimiento del mismo no se degrade por el aumento del

número de clientes conectados. Los distintos usuarios del sistema (médicos, investigadores, etc) pueden optar por cualquiera de estas interfaces, en función de las tareas que vayan a realizar (Oracle Discoverer o SQL*Plus para consultas Ad Hoc, JAVA para crear nuevos modelos de minería de datos, etc), lo cual proporciona flexibilidad a la hora de escoger el tipo de herramienta que mejor se adapte a la tarea a realizar. El requisito de acceso a datos también ha sido decisivo a la hora de seleccionar un SGBD.

La suite Oracle 10g integra distintas herramientas gráficas (ETL, consultas Ad Hoc, minería, etc), APIs (JAVA, PL/SQL, etc), opciones y configuraciones (OLAP y minería de datos) y demás software adicional que la convierten en un marco de trabajo idóneo para desarrollar las distintas tareas que componen el ciclo de vida del almacén. Las soluciones Open Source evaluadas no disponen de herramientas y software adicional, siendo necesario recurrir a herramientas de terceros cuyos costes, en algunos casos, eran muy superiores al de la suite adquirida.

Y por último, *MORFEO-almacén* ha sido configurado y preparado para realizar tareas de minería de datos. Actualmente se están realizando pruebas con distintos modelos predictivos. Además se está explorando la posibilidad de implementar nuevos modelos de minería mediante JAVA, que complementen a los integrados en la opción Oracle Data Mining (ODM).

Referencias

[1] R. Agrawal, A. Gupta, and S. Sarawagi. Modeling multidimensional databases. pages 232–243, 7–11 1997.

[2] Chee-Yong Chan and Yannis E. Ioannidis. Bitmap index design and evaluation. pages 355–366, 1998.

[3] E. F. Codd. A relational model of data for large shared data banks. *Communications of the ACM*, 13(6):377–387, 1970.

[4] P.A. de Clercq, J.A. Blom, H.H.M. Korsten, and A. Hasman. Approaches for creating computer-interpretable guidelines that facilitate desision support. *Artificial Intelligence in medicine*, 5(31):1–27, 2004.

[5] U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy (eds.). *Advances in knowledge discovery and data mining*. MIT Press, 1996.

[6] G.Klösch and col. The siesta project polygraphic and clinical database. *IEEE Engineering in medicine and biology*, 20(3):51–57, 2001.

[7] H. Gupta, V. Harinarayan, A. Rajaraman, and J.D. Ullman. Index selection for OLAP. pages 208–219, 1997.

[8] J. Han. OLAP mining: Integration of OLAP with data mining. pages 1–11, 1997.

[9] D. Hand, H. Mannila, and P. Smyth. *Principles of Data Mining*. MIT, 2001.

[10] V. Harinarayan, A. Rajaraman, and J.D. Ullman. Implementing data cubes efficiently. pages 205–216, 1996.

[11] J.Gray and A.Reuter. *Transaction Processing: Concepts and Techniques*. Morgan Kaufmann, 1992.

[12] R. Kimball, L. Reeves, M. Ross, and W. Thornthwaite. *The data warehouse lifecycle toolkit: expert methods for designing, developing and deploying data warehouses*. Wiley publishing, 2003.

[13] R. Kimball and M. Ross. *The data warehouse toolkit: the complete guide to dimensional modelling*. Wiley publishing, 2002.

[14] R.Kimball and J.Caserta. *The data warehouse ETL toolkit: practical technines for extracting, cleaning, conforming, and delivering data*. Wiley publishing, 2004.

[15] J.D. Ullman and J. Widom. *A First Course in Database Systems.* Prentice Hall, 1997.

[16] Panos Vassiliadis and Timos K. Sellis. A survey of logical models for OLAP databases. *SIGMOD Record*, 28(4):64–69, 1999.

[17] W.H.Inmon. *Building the Data Warehouse.* Wiley publishing, 2002.

[18] W.Korn. Ai in medicine on its way from knowledge-intensive to data-intensive systems. *Artificial Intelligence in medicine*, 2(23):5–12, 2001.