



Magíster en Ciencias de Computación

Facultad de Ciencias Empresariales

Universidad del Bio Bio

Definición de un Marco de Trabajo para la obtención de un modelo de base de  
datos relacional desde sistemas heredados

Tesis para optar a grado de Magister en Ciencias de Computación

Alumno: Lic. Fabián Villagrán Gutiérrez

Directora: Dra. Angélica Caro

Codirector: Dr. Gilberto Gutiérrez

Mayo 2012

## Resumen

Uno de los activos más importantes de las organizaciones es la información, para algunas incluso puede ser crítico y esencial para su funcionamiento, en consecuencia es necesario que los datos sean mantenidos a lo largo del tiempo, inclusive si los sistemas de información evolucionan. Este problema lo viven muchas organizaciones de distintas áreas, que poseen sistemas de larga data denominados Sistemas de Información Heredados (SIH). La evolución de estos sistemas se puede realizar mediante estrategias como reingeniería, reemplazo del sistema o alguna solución híbrida de las anteriores. Sin embargo, independiente de la solución adoptada, los datos del sistema heredado deben ser conservados, aunque no necesariamente con la misma estructura de datos, que puede resultar obsoleta para las nuevas aplicaciones. Puesto que en los sistemas modernos el uso de bases de datos es esencial como una forma de optimizar su procesamiento, es deseable que en un proceso de modernización de un sistema de información heredado se puedan migrar los datos heredados a una base de datos relacional. Consecuentemente con esto, esta tesis define un marco de trabajo que facilita la generación de modelos de datos relacionales a partir de distintas fuentes de datos de un SIH. El marco de trabajo está compuesto por tres etapas que en su conjunto permiten la extracción de la estructura y datos desde las fuentes de datos originales del SIH hacia un Almacén de Datos Intermedio (ADI) desde donde se obtienen las Dependencias Funcionales (DFs) mediante un proceso semiautomático a través de expertos y herramientas ad-hoc, y a partir de estas DFs generar un modelo de datos relacional que represente la realidad actual del SIH. Esta investigación cobra especial relevancia debido a que se propone un marco de trabajo que establece un mecanismo ordenado que, utilizando herramientas, algoritmos y expertos, obtiene un modelo de datos a partir de un SIH relacional y no relacional. Este último punto es un factor diferenciador con las investigaciones encontradas en la literatura. El marco de trabajo fue aplicado en un caso de estudio en el Hospital Clínico Herminda Martín de Chillán de una manera exitosa, puesto que contribuyó con la concepción del modelo de datos para la generación del nuevo sistema de información de atención de emergencias.

## **Abstract**

One of the most important assets of organizations is the information, and for some of them it may even be critical and essential for their operation, therefore it is necessary that their data is kept over time, even if their Information Systems evolve.

This problem is experienced by many organization in different areas, wich have long lasting systems, called Legacy Information System (LIS).

The evolution of these systems may be conducted according to strategies such as reengineering, system replacement or a hybrid solution of both.

However, regardless of the implemented solution, the data of the Legacy System must be kept, although not necessarily with the same data structure, wich can be obsolete for the newer applications.

Since in modern systems, the use of databases is essential, as a way to optimize the processing, it is desirable that in a modernization process of a Legacy System. The data can be migrated to a relational database.

Consequently with this, this thesis provides a framework that facilitates the generation of relational data models from different source of data in a LIS.

The framework consists of three stages, wich all together allow the extraction of the structure and data from the original sources of a LIS to a Intermediate Data Storage (IDS). From this IDS, the functional dependencies are obtained with a semiautomatic process that includes experts and ad-hoc tools. Then, from this FD, is possible to generate a data model that represents the actual state of the LIS.

This research is particularly relevant because it proposes a framework that provides an ordered mechanism, which by means of tools, algorithms and experts, gets a data model from a relational and nonrelational LIS.

This last point is a differentiating factor of our work with other research found in literature. The framework was applied in a successful case study in the Clinical Hospital Herminda Martin of Chillán, where it contributed to the conception of the data model for the generation of a new information system for the emergency unit.

*Gracias a Dios, por darme la salud que me ha permitido cumplir mis sueños, a mis hijos por la gran tajada de tiempo que les he robado, a mi esposa, compañera y amiga, por levantarme y ponerme de pie mil veces, a mis padres por las sabias lecciones y apoyo incondicional, a mis amigos que con su ayuda desinteresada y consejos me han acompañado en cada uno de mis proyectos y especialmente a mis directores de tesis por su paciencia y comprensión.*

## Índice de Contenidos

Capítulo 1 Introducción.....	1
1.2    Hipótesis, .....	4
1.3    Objetivos, Alcance y Limitaciones.....	4
1.3.1    Objetivos .....	4
1.3.2    Alcances y Limitaciones .....	5
Capítulo 2 Metodología de Investigación .....	4
2.1    Investigación Acción .....	4
2.2    Revisión Sistemática de la Literatura .....	5
2.3    Caso de Estudios .....	6
2.3.1    Identificación del problema .....	7
2.3.2    Establecimiento de Hipótesis.....	7
2.3.3    Recolección de datos.....	7
2.3.4    Análisis de datos e interpretación de resultados .....	7
2.4    Aplicación de métodos de trabajo .....	8
Capítulo 3 Estado del Arte .....	9
Capítulo 4 Marco de Trabajo EOG.....	21
4.1    Marco de Trabajo EOG.....	21
4.2    Roles del Equipo de Trabajo .....	22
4.3    Etapas de EOG .....	24
4.3.1    Etapa de Extracción .....	25
4.3.2    Etapa de Obtención .....	39
4.3.3    Etapa de Generación .....	51
4.4    Conclusiones del Capítulo .....	54
Capítulo 5 Herramientas para EOG .....	55
5.1    Etapa de Extracción .....	55
5.1.1    Sistemas de Gestión de Base de Datos .....	56
5.1.2    Herramientas ETL.....	57
5.2    Etapa de Obtención.....	57
5.2.1    Herramientas para obtención de DFs .....	58
5.3    Etapa de Generación.....	59

5.3.1	Herramientas Generadoras de Modelo .....	60
5.3.2	Herramientas para Modelado de Datos .....	61
5.4	Conclusiones del Capítulo .....	62
Capítulo 6 Caso de Estudio .....		63
6.1	Descripción de SIH.....	63
6.2	Utilizando EOG.....	64
6.2.1	Conformación de Equipo de Trabajo .....	64
6.2.2	Etapa de Extracción .....	65
6.3.3	Etapa de Obtención .....	76
6.3.4	Etapa de Generación .....	89
6.4	Conclusiones del Capítulo .....	96
Capítulo 7 Conclusiones .....		97
Capítulo 8 Bibliografía.....		100
Anexos .....		104
Anexo A Revisión Sistemática de la Literatura .....		104
Anexo B Fuentes de Información de Sistema SAM .....		111
Anexo C Dependencias Funcionales Automáticas .....		135
Glosario de Términos.....		146

## Índice de Figuras

Figura 3.1.:	Diagrama del Modelo de Datos.....	4
Figura 3.2:	Esquema de funcionamiento de la metodología. ....	14
Figura 3.3:	Repositorio de meta-atributos. ....	15
Figura 3.4:	Esquema funcional del Método M-FF2RDB.....	16
Figura 4.1:	Esquema de funcionamiento del Marco de Trabajo. ....	22
Figura 4.2:	Esquema general de EOG con flujos de información.....	24
Figura 4.3:	Esquema de funcionamiento de la etapa de Extracción.....	25
Figura 4.4:	Estructura de tabla de origen y ADI. ....	38
Figura 4.5:	Esquema de funcionamiento Etapa de Obtención de DFs.....	40
Figura 4.6:	Ejemplo de estructura de archivos XML con DFs.....	50

Figura 4.7: Esquema de funcionamiento Etapa Generación de Modelo.....	51
Figura 4.8: Modelo relacional propuesto.....	53
Figura 5.1: Posibles combinaciones de los atributos A, B, C, D y E.....	58
Figura 6.1: Estructura de Formularios.....	69
Figura 6.2: Esquema resumen del modelo de datos SAM.....	70
Figura 6.3: DFs que generan nuevas entidades. . . . .	90
Figura 6.4: Diagrama de datos relacional propuesto.....	95

## Índice de Tablas

Tabla 4.1: Clasificación de Expertos.....	27
Tabla 4.2: Clasificación de Componentes del SIH.....	28
Tabla 4.3: Documentación que contribuyen al Marco de Trabajo.....	29
Tabla 4.4: Ejemplo de Fuentes de Información a utilizar en EOG.....	30
Tabla 4.5: Ejemplo de registro de entidades a extraer.....	31
Tabla 4.6: Fuentes de información y Actividades.....	41
Tabla 4.7: Tabla de Registro de DFs.. . . . .	42
Tabla 4.8: Tabla Ejemplo de Registro de DFs establecidas en los Atributos.....	44
Tabla 4.9: Tabla de Registro de inconsistencias.....	46
Tabla 4.10: Atributos eliminados.....	47
Tabla 4.11: Ejemplo de registro numerado de DF. . . . .	48
Tabla 4.12: Registro de DFs eliminadas. . . . .	50
Tabla 6.1: Descripción de entidades. . . . .	68
Tabla 6.2: Fuentes de Información de SAM. . . . .	71
Tabla 6.3: Entidades a extraer desde SAM. . . . .	72
Tabla 6.4: Estructura tabla SAPAT001. . . . .	74
Tabla 6.5: Estructura tabla SABST030. . . . .	75
Tabla 6.6: Estructura tabla SAURT009. . . . .	75
Tabla 6.7: DFs establecidas en el Dominio. . . . .	77
Tabla 6.8: Tabla de Registro de DFs establecidas en los Atributos. . . . .	78
Tabla 6.9: Resultados inconsistentes . . . . .	79
Tabla 6.10: Limpieza de datos. . . . .	80

Tabla 6.11: Atributos a eliminar. ....	81
Tabla 6.12: Ejemplo de DFs obtenidas Automáticamente desde tabla SAPAT001. ....	83
Tabla 6.13: Ejemplo de DFs obtenidas Automáticamente desde tabla SABST030. ....	83
Tabla 6.14: Ejemplo de DFs obtenidas Automáticamente desde tabla SAURT009. ....	83
Tabla 6.15: Resumen de DFs obtenidas por entidad.....	84
Tabla 6.16: DFs Eliminadas en tabla SAPAT001. ....	84
Tabla 6.17: DFs Eliminadas en tabla SABST030. ....	85
Tabla 6.18: DFs Eliminadas en tabla SAURT009. ....	86
Tabla 6.19: DFs depuradas para entidad SAPAT001. ....	87
Tabla 6.20: DFs depuradas para entidad SABST030. ....	88
Tabla 6.21: DFs depuradas para entidad SAURT009.....	89
Tabla 6.22: Modelo de datos en DDL entidad SAPAT001.....	91
Tabla 6.23: Modelo de datos en DDL entidad SABST030.....	92
Tabla 6.24: Modelo de datos en DDL entidad SAURT009.....	93

# **Capítulo 1**

## **Introducción**

---



Un SIH se puede definir como "cualquier sistema de información que se resiste significativamente a la modificación y evolución" (BRODIE and STONEBRAKER 1998). "Los SIH son sistemas de generaciones anteriores que son incompatibles con las normas y sistemas de la generación actual" (SILBERSCHATZ, KORTH et al. 1991). Los SIH suelen ser sistemas críticos que son mantenidos debido al alto riesgo de reemplazarlos (SOMMERVILLE 2005). Generalmente, los SIH son críticos para la organización y operan las 24 horas del día, en efecto, suelen ser la columna vertebral y el principal vehículo para la consolidación de información, en consecuencia su mal funcionamiento puede tener un grave impacto (BISBAL, LAWLESS et al. 1999). Por ejemplo, para una institución financiera como un banco es impensable realizar operaciones sin que estas estén apoyadas y registradas en un sistema de información (SI). Un SIH es una aplicación de software que posee un importante valor de negocio, aunque se ha debilitado por el constante cambio tecnológico, un pobre soporte arquitectónico, alto costo de mantenimiento y documentación a veces inexistente, y a pesar de esto se mantiene en producción (O'CALLAGHAN 2009), asimismo la escasa documentación hace necesaria la comprobación del código fuente para entender su funcionalidad (BORONAT, PÉREZ et al. 2004), y por otro lado tiene limitadas o nulas posibilidades de interactuar con plataformas modernas orientadas a interfaces de usuario amigables.

Los SIH, en general, utilizan estructuras de datos no relacionales para almacenar su información generando como consecuencia problemas frecuentes como: la redundancia e inconsistencia de los datos, ya que éstos se encuentran en diferentes formatos y duplicados, dificultad de acceso a los datos debido a que existen aplicaciones que controlan de manera independiente el acceso a estos (MALDONADO, GUTIÉRREZ et al. 2009), complicación para utilizar los datos desde nuevas aplicaciones o nuevos programas debido a la incompatibilidad, ó la actualización simultánea que no se controla adecuadamente y suele producir información inconsistente.

En conclusión, debido a la no existencia de procedimientos claros que controlen el ingreso de datos incorrectos o incompletos se producen problemas de inconsistencia. Asimismo, los datos heredados podrían estar contenidos en bases de datos que hayan quedado obsoletas o no representen el modelo de negocios de toda la organización.

A raíz de lo anterior, el mayor problema surge con la independencia entre los datos y la lógica de negocio. Dada la estrecha relación que existe, resulta difícil modificar los programas sin que los datos se vean afectados y viceversa. Por otro lado, en la actualidad la migración de datos de los SIH se suele hacer de forma manual, sin apoyo de procedimientos establecidos en la literatura,

transformándolo en un proceso largo y tedioso, que tiene directa relación con el número de fuentes de datos y tamaño de éstas. Las empresas, tarde o temprano tienen que volver a inventar sus sistemas heredados, añadir mejoras a los mismos o modificar significativamente sus capacidades (PEDERSEN and GOLDBERG 2009).

El estudio de investigaciones por medio de una Revisión Sistemática de la Literatura (RSL) nos ha permitido detectar la falta de guías o marcos de trabajo que indiquen claramente por medio de etapas, procedimientos y herramientas el cómo obtener un modelo de datos relacional a partir de un SIH. Del mismo modo se han conocido herramientas, técnicas y algoritmos que nos han facilitado la definición de un marco de trabajo destinado a generar un esquema de base de datos relacional normalizado en tercera forma normal (3FN) a partir de las estructuras de datos de sistemas heredados, el que está dirigido esencialmente a administradores del sistema, sin embargo su utilidad se extiende a toda la organización.

Esta tesis se organiza de la siguiente forma. En este capítulo se abordan los objetivos y alcance de la investigación, posteriormente en el capítulo 2, se presenta la metodología de trabajo y las actividades asociadas. El capítulo 3, da cuenta el estado del arte obtenido a través de una revisión sistemática de la literatura. El capítulo 4, plantea una propuesta que explica y define el funcionamiento del marco de trabajo. Posteriormente, el capítulo 5, contiene un conjunto de herramientas, aplicaciones y algoritmos que pueden ser utilizados en el marco de trabajo, mientras en el capítulo 6 se muestran los resultados obtenidos en un caso de estudio y, finalmente, el capítulo 7, abarca las conclusiones y trabajos futuros.

## **1.1 Fundamentos**

El objetivo final es la generación de un modelo de datos relacional en 3FN que facilita la labor de los ingenieros mediante una disminución del tiempo utilizado para ello. De este modo, un modelo es una representación de la realidad que contiene las características generales de algo que se desea desarrollar. Regularmente dicha representación en base de datos se desarrolla de manera gráfica. Más específicamente un modelo de datos según (DE MIGUEL, PIATTINI et al. 2000) es “un conjunto de conceptos, reglas convenciones bien definidos que nos permiten aplicar una serie de abstracciones a fin de describir y manipular los datos de un cierto mundo real que deseamos almacenar en la base de datos”. Es decir una colección de herramientas conceptuales que permiten describir los datos, las relaciones existentes entre ellos, la semántica asociada a los datos y restricciones de consistencia.

Los modelos de datos se dividen básicamente en tres grupos: modelos lógicos basados en objetos que permiten describir datos a nivel conceptual y visual, modelos lógicos basados en registros que facilitan la descripción a nivel conceptual y físico, y en último lugar modelos físicos de datos utilizados para describir datos a nivel más bajo.

En la categoría de modelos lógicos basados en registros se encuentra el modelo de datos relacional que proporciona una base de alto nivel para datos lingüísticos que entrega la máxima independencia entre los programas, por una parte representa la máquina y por otra la organización de los datos sobre el otro. Otra ventaja de la perspectiva relacional es que constituye una base sólida para el tratamiento de derivabilidad, la redundancia, y la coherencia de las relaciones (CODD 1970). Este modelo ganó mucha popularidad a principios de los noventa y un factor clave de ese éxito fue la simplicidad de su estructura (CASTELLANOS and SALTOR 1993).

La estructura básica y única del modelo relacional es la relación (también llamada tabla) que sirve para representar tanto los objetos como las asociaciones entre ellos (DE MIGUEL, PIATTINI et al. 2000). Las filas o tuplas equivalen a cada uno de los registros que contendrá la base de datos y las columnas corresponden a los atributos de cada registro localizado en la tupla.

El modelo de datos relacional se caracteriza principalmente por poseer:

- Independencia física: el modo en el que se almacenan los datos no influye en su manipulación lógica y, por tanto, los usuarios que acceden a esos datos no tienen que modificar sus programas por cambios en el almacenamiento físico.
- Independencia lógica: el añadir, eliminar o modificar objetos de la base de datos no repercute en los programas y/o usuarios que están accediendo a subconjuntos parciales de los mismos (vistas).
- Flexibilidad: en el sentido de poder presentar a cada usuario los datos de la forma en que éste prefiera.
- Uniformidad: las estructuras lógicas de los datos presentan un aspecto uniforme, lo que facilita la concepción y manipulación de la base de datos por parte de los usuarios.
- Sencillez: las características anteriores, sumado al uso de lenguajes de usuario sencillos, generan como resultado que el modelo de datos relacional sea fácil de comprender y de utilizar por parte del usuario final.

La Figura 3.1 muestra el modelo relacional expresado en un diagrama en el que se aprecian las entidades con sus atributos destacando las claves primarias, foráneas y las relaciones entre ellas.

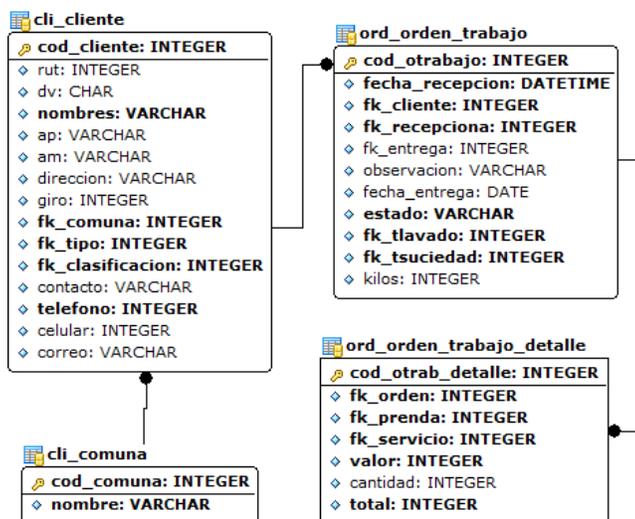


Figura 3.1: Diagrama de Modelo de Datos.

## 1.2 Hipótesis,

Es posible determinar un método que permita obtener un modelo de datos relacional en 3FN a partir de un SIH.

## 1.3 Objetivos, Alcance y Limitaciones

En este punto se identifica la hipótesis y los objetivos que persigue esta investigación, así como también el establecimiento del alcance y limitaciones que permitirán delimitar de una manera clara la circunscripción de esta tesis de grado.

### 1.3.1 Objetivos

Los objetivos de esta tesis reflejan los resultados que se esperan de la investigación, de este modo se han organizado entre objetivos generales y específicos, y se describen a continuación.

- **Objetivo General**

Definir un marco de trabajo que permita obtener un esquema de base de dato relacional normalizado en tercera forma normal a partir de las estructuras de datos y de los datos de un SIH.

- **Objetivos Específicos**

Para alcanzar el objetivo general se requiere cumplir con los objetivos específicos planteados a continuación.

- Definición de métodos y etapas para la extracción y transformación de estructuras de datos de sistemas de información heredados.
- Búsqueda y/o desarrollo, selección y prueba de herramientas y algoritmos que serán propuestos en el Marco de Trabajo.
- Integración de métodos, etapas y herramientas necesarias para la generación de esquemas relacionales normalizados.
- Comprobar y validar marco de trabajo mediante un caso de estudio.

### **1.3.2 Alcances y Limitaciones**

La generación de este marco de trabajo implica el establecimiento de varias etapas que en su conjunto generan un esquema de base de datos relacional. El marco de trabajo no pretende imponer herramientas específicas ni artefactos. En consecuencia su definición pretende plantear una estructura conceptual y tecnológica de soporte que propone un conjunto estandarizado de conceptos, prácticas y criterios.

Asimismo, como punto de partida para la definición y aplicación del marco de trabajo se considera que las estructuras de datos originales de un SIH pueden ser heterogéneas, tales como archivos, bases de datos relacionales y no relacionales.

## **Capítulo 2**

# **Metodología de Investigación**

---

En este capítulo se presenta la metodología de trabajo utilizada en el desarrollo de esta tesis. Inicialmente se realizó la búsqueda de investigaciones científicas por medio de una RSL, que permitió establecer un completo estado del arte y conjuntamente encontrar métodos y herramientas que facilitan la obtención de DFs y generación de modelo de datos relacional. Posteriormente, las herramientas y métodos fueron evaluados con el objeto de medir los resultados obtenidos, y de este modo se elaboró el marco de trabajo utilizando el método de investigación acción, en último lugar, se aplicó el marco de trabajo en una institución por medio de un caso de estudio.

## **2.1 Investigación Acción**

La investigación acción, es un método de investigación cualitativo, que plantea la unión del enfoque práctico y teórico, permitiendo englobar etapas de manera sistemática e iterativa. Investigación Acción es un término acuñado por (LEWIN 1946), quien en ese entonces era profesor del MIT (Massachusetts Institute of Technology) la definió como un proceso de investigación orientado al cambio social, caracterizado por una participación democrática en la toma de decisiones. Es una forma de investigar que permite enlazar el enfoque experimental de las ciencias sociales con programas de acción social que respondan a los problemas sociales esenciales. El concepto tradicional está basado en tres etapas del cambio social: descongelación, movimiento y recongelación.

En este tipo de exploración el investigador como sujeto de la investigación aborda un aspecto de la realidad (objeto de la investigación) con el propósito de describir, interpretar, establecer relación o explicar el fenómeno estudiado.

En rigor, investigación acción no es un método de investigación sino más bien una clase de métodos que poseen las siguientes características en común:

- Orientación a la acción y el cambio.
- Focalización en un problema particular.
- Un modelo de proceso “orgánico” que abarca etapas sistemáticas e iterativas.

La investigación acción también puede ser realizada por las grandes organizaciones o instituciones, con asistencia o guiados por investigadores, con el objetivo de mejorar sus estrategias, prácticas y conocimientos de los entornos en los que ejercen su profesión. De acuerdo a lo anterior, este tipo de investigación puede ser adoptada en una variedad de contextos para una amplia gama de propósitos, por ello algunas definiciones encontradas en la literatura la definen como:

- Un proceso por el cual los profesionales intentan estudiar sus problemas científicamente con el fin de guiar, corregir y evaluar sus decisiones y acciones (COREY 1988)
- La participación de todas la partes involucradas en la investigación examinado la situación existente, con el objetivo de cambiarla y mejorarla (WADSWORTH 1998).
- La forma que tienen los grupos de personas que preparan las condiciones necesarias para aprender de sus propias experiencias, y hacer estas accesibles a otros (MCTAGGART 1991)

Por lo tanto, investigación acción es una forma de investigar de carácter colaborativo que busca unir teoría y práctica entre investigadores y profesionales mediante un proceso de naturaleza cíclica.

## **2.2 Revisión Sistemática de la Literatura**

La investigación se inició estableciendo un estado del arte sobre estudios relacionados con métodos o marcos de trabajo que faciliten la extracción de DFs y la generación de modelos de datos relacionales, para lo cual se utilizó la RSL como una metodología que facilita la búsqueda de material bibliográfico. En (KITCHENHAM and CHARTERS 2007) se define una revisión sistemática de la literatura como “una manera de evaluar e interpretar toda la investigación disponible, que sea relevante respecto de una interrogante de investigación particular, en un área temática o fenómeno de interés”.

En palabras sencillas se puede definir la RSL como un diseño investigativo que permite observar el resultado de muchas investigaciones primarias y secundarias, entendiendo que estas son necesarias para comprender el tema a investigar. En particular esta revisión permitió identificar, evaluar e interpretar los estudios considerados relevantes para satisfacer las preguntas de investigación.

Una RSL sintetiza el trabajo existente de una manera que parece ser justa y completa, es decir, si esta carece de ambas características posee poco valor científico. La estrategia de búsqueda debe permitir que la totalidad de la exploración sea evaluada. Quizás el problema central al que se enfrentan todas las ciencias de la computación, es cómo se debe hacer para construir sobre la base de trabajos anteriores. La RSL presenta las siguientes ventajas:

- La metodología bien definida hace menos posible omitir resultados de la literatura.
- Puede dar información acerca de los efectos de algunos fenómenos.
- En caso de estudios cuantitativos, es posible combinar datos, usando técnicas meta-analíticas. Esto aumenta la probabilidad de detectar efectos reales.

Sin embargo, requiere de un esfuerzo mayor al de los métodos tradicionales y un mayor poder de meta-análisis, lo que puede ser una desventaja, ya que es posible detectar pequeñas fluctuaciones, así como también efectos reales.

Esta RSL comienza con la definición de un protocolo de revisión que especifica las preguntas de investigación que se tienen, y los métodos que se utilizarán para realizar la revisión. Además, se elaboró una estrategia de búsqueda definida cuyo objetivo fue detectar la mayor cantidad posible de literatura relevante. Y por otro lado, se consideró la inclusión y exclusión explícita de los criterios para evaluar cada uno de los potenciales estudios primarios.

### **2.3 Caso de Estudios**

En definitiva, esta tesis contempló una investigación que fue llevada a cabo en la acción, específicamente, con la ejecución de la teoría y por consecuencia su comprobación por medio de un caso de estudio.

El caso de estudio es un método de investigación cuyo origen proviene de la investigación médica y psicológica. Es una herramienta valiosa de investigación en el área de las ciencias sociales, educación, dirección de empresas y desarrollo tecnológico, entre otras. Además, los datos pueden ser obtenidos desde diversas fuentes, tanto cuantitativas como cualitativas, esto es, documentos, registros de archivos, entrevistas y observación directa. El caso de estudio no tiene especificidad, por lo que puede ser usado en cualquier disciplina para dar respuesta a preguntas de la investigación (YIN 2009).

El método de caso de estudio es una metodología rigurosa que es adecuada para investigar fenómenos en los que se busca dar respuesta a cómo y por qué ocurren. Permite estudiar los fenómenos desde múltiples perspectivas y no desde la influencia de una sola variable. Además, permite explorar en forma más profunda y obtener un conocimiento más amplio sobre cada fenómeno, lo cual permite la aparición de nuevas señales sobre los temas que emergen.

El caso de estudio requiere protocolizar las tareas, instrumentos y procedimientos que se van a ejecutar, y el protocolo del caso de estudio se convierte en un documento que materializa el diseño de la investigación y las reglas generales y específicas que se deben seguir, lo cual redundará en el aumento de la calidad de la investigación. Sin embargo, no existen guías precisas acerca del número de casos que deben ser incluidos, por lo que “esta decisión queda en manos del investigador”. Sin embargo, para la aplicación del caso de estudio en ésta investigación se han considerado las siguientes etapas.

### **2.3.1 Identificación del problema**

Las preguntas de investigación son el primer elemento del diseño de cualquier indagación según (YIN 2009), específicamente las interrogantes "cómo" y "por qué" permiten concretar el problema inicial de una investigación de caso de estudio donde será necesario identificar un sistema integrado que constituirá el fenómeno objeto de estudio.

Además (YIN 2009) propone sofisticar el diseño de investigación aplicando diferentes unidades de análisis sobre el mismo caso. Según este autor, las unidades de análisis permiten definir qué es el caso. Cuando el caso de estudio se realiza sobre fenómenos o acontecimientos más complejos de definir, es necesario considerar una o varias unidades de análisis que permitan dar un paso más en la concreción de la investigación. Las unidades de análisis permiten definir los límites del caso para diferenciarlos de su contexto y orienten la elaboración de los resultados, estableciendo los límites de la argumentación.

### **2.3.2 Establecimiento de Hipótesis**

Es la etapa donde se enuncian proposiciones o hipótesis de investigación. Se trata de afirmaciones sobre el problema identificado a partir de las bases teóricas de la investigación. Su función es encaminar la investigación en la dirección correcta, mostrando aquello que es necesario observar para obtener evidencias y evitar "recogerlo todo". El análisis de los datos no se produce nunca a partir del vacío. Es necesario un marco teórico donde situar las hipótesis provisionales de solución del problema detectado o, como mínimo, algunas presuposiciones sobre cómo tratarlo.

### **2.3.3 Recolección de datos**

En (YIN 2009) se recomienda el uso de múltiples fuentes de datos, las que permitirán verificar si los datos obtenidos a través de las diferentes fuentes de información guardan relación entre sí (cumplimiento del principio de triangulación), para garantizar la validez interna de la investigación. De manera similar, se requiere la aplicación de distintos instrumentos de recolección de información, tales como: entrevista personal no estructurada y revisión de documentos, entre otros.

### **2.3.4 Análisis de datos e interpretación de resultados**

El objetivo del análisis es generar una comprensión del problema de investigación. Es importante que los datos sean analizados en forma inductiva, guiado por la literatura inscrita en el marco teórico de la investigación.

## 2.4 Aplicación de métodos de trabajo

Para el desarrollo del marco de trabajo se utilizaron diversos métodos y técnicas que forman parte de la metodología de investigación, para lo cual se han considerado los siguientes participantes:

- **Investigador:** el grupo de investigación Pehuen, formado por profesores del Departamento de Ciencias de la Computación y Tecnologías de Información de la Universidad del Bio Bio, en la ciudad de Chillán y el candidato a magister.
- **Objeto Investigado:** Obtención de modelo de datos relacional a partir de SIH.
- **Beneficiarios:** Todas aquellas instituciones e investigadores que requieran de un marco de trabajo que contribuya a obtener un modelo de datos normalizado a partir de un SIH.

Finalmente, La aplicación de los métodos conlleva la ejecución de las siguientes actividades.

- Realizar un estudio de la literatura existente mediante una RSL, de manera rigurosa e imparcial que permita identificar, evaluar, interpretar y sintetizar los trabajos relacionados existentes con el objeto de conocer el estado del arte y, a partir de este, proponer mejoras en métodos encontrados.
- Búsqueda de herramientas, algoritmos y artefactos propuestos en la literatura que permitan realizar:
  - Extracción de estructuras y datos desde fuentes de datos obsoletas.
  - Generación de Dependencias Funcionales (DFs) a partir de estructuras de datos y datos del SIH.
  - Obtención de Modelo de datos relacional a partir de DFs.
  - Selección de herramientas, algoritmos y artefactos mediante pruebas de campo.
  - Integración coherente y ordenada de métodos, etapas, herramientas y artefactos que generen como resultado el modelo de datos relacional.
- Validación de marco de trabajo mediante caso de estudio.

## **Capítulo 3**

### **Estado del Arte**

---

En el presente capítulo se describe el estado del arte obtenido a través de la aplicación de una RSL que ha permitido conocer la literatura descrita en libros e investigaciones científicas publicadas en revistas y conferencias de ingeniería de software relacionadas con SIHs, en particular sobre la migración de éstos hacia nuevas plataformas, herramientas o algoritmos que faciliten la obtención de DFs y la generación de modelo de datos a partir de DFs. Para mayor referencia, en el Anexo A se describen detalladamente las etapas que conformaron la RSL. A continuación, se muestran los resultados obtenidos en la RSL y un análisis de las investigaciones.

### **3.1 Revisión de la Literatura**

Los sistemas heredados no relacionales son problemáticos por varias razones. En primer lugar, son difíciles de mantener y expandir porque hay una falta general de conocimiento sobre cómo los flujos de trabajo y reglas de negocio se reflejan en estos sistemas no relacionales. En segundo lugar, son difíciles de integrar con los nuevos sistemas en una plataforma moderna debido a la no extensibilidad, incompatibilidad, y menos la apertura del hardware y software subyacentes de estos sistemas heredados (BISBAL, LAWLESS et al. 1999) y finalmente, la aplicación de un desarrollo más rápido en un entorno no relacional es difícil de obtener porque los sistemas no relacionales funcionan a un menor nivel de abstracción y requieren una programación extensa.

Adicionalmente, el costo de mantener operativo un SIH y usuarios, en condiciones aceptables, consume una parte importante, a menudo entre el 60% y 90%, de los costos totales del ciclo de vida (ALKHATIB 1992). Existen diversos estudios que apuntan a identificar los costos de mantenimiento en el ciclo de vida del software (FOSTER 1993; SNEED 2002). Sumado a esto, las empresas destinan gran parte de sus recursos al mantenimiento de sus sistemas, actividad contemplada dentro de los modelos de desarrollo en la ingeniería de software.

Uno de los problemas que poseen los SIH es que a menudo no existe una comprensión conceptual de los datos (JOHANNESON 1994), y esto generalmente es debido al paso de los años, generando una obstaculización en la utilización efectiva de los datos y, por otro lado, la dificultad para abordar la modernización de un SIH.

Para muchas organizaciones, comprender completamente la lógica de desarrollo de los SIH, en particular de la base de datos (SULTAN and PETROUNIAS 2009) no es una opción, debido a su alto costo, en consecuencia se ven obligadas a buscar formas alternativas de hacer frente a la problemática.

En efecto, para las organizaciones es necesario abordar los SIH con el objeto de evolucionar. La tecnología y el mundo de los negocios avanza a pasos agigantados, tomando estos antecedentes en la actual era de Internet, los sistemas relacionales han demostrado ser más eficaces que los no relacionales en el apoyo a procesos de comercio electrónico debido a su accesibilidad superior, escalabilidad y apertura (LIN 2008).

El proceso de cambio en los sistemas debe promover la evolución de estos. Esta debe consistir en dar respuesta rápida, preparada y eficiente a los cambios que se produzcan en el entorno, ya sean estos de índole tecnológico o de gestión del propio negocio (RODRÍGUEZ, MÁRQUEZ et al. 2001). La evolución es un concepto amplio y abarca desde una modificación para la corrección de errores hasta la reimplantación completa del sistema. El reto en la evolución de los sistemas es lograr comprender la funcionalidad, diseño, operación y funcionamiento del sistema heredado y prever los tipos de cambios que serán necesarios durante la vida útil del sistema (BERGEY, NORTHROP et al. 1997).

En la literatura se proponen distintos métodos para hacer frente a los sistemas de información heredados, estos van desde soluciones típicas como el abandono, reingeniería a través de wrapping, migración del SIH, y hasta soluciones híbridas (CIMITILE, FASOLINO et al. 1997). Definitivamente, las organizaciones se ven obligadas a migrar sus sistemas a arquitecturas modernas para aumentar su competitividad en el mercado (COLOSIMO, DE LUCIA et al. 2007), (BERGEY, NORTHROP et al. 1997). A continuación se describen algunas de estas técnicas o métodos.

La reingeniería se presenta como una opción y se define como la mejora de sistemas existentes mediante la inversión rentable de los recursos; el costo, tiempo y riesgo de reingeniería deben ser bajos. La necesidad de llevar a cabo un proceso de reingeniería depende de si es posible el mantenimiento del propio sistema. En general, los problemas más comunes son: documentación escasa u obsoleta, abandono del proyecto por parte de los desarrolladores, comprensión limitada de todo el sistema, duplicidad de código, tiempos excesivos de integración o la necesidad de corrección constante de errores (CERVIÑO and MORO 2009).

Muchas de esas aplicaciones que llevan varios años funcionando, están desactualizadas tecnológicamente, son monolíticas, y, aunque se erigen como islas muy eficientes en la estructura vertical de las organizaciones, su evolución e integración horizontal con otros sistemas de software se han vuelto un verdadero reto (GARCÍA 2008).

El wrapping es una técnica de la reingeniería, que en el mejor de los escenarios analiza y modifica las interfaces de usuario, ignorando detalles internos, en la que se entrega una "mejor y nueva" mirada que permita mejorar las operaciones (BISBAL, LAWLESS et al. 1999). El wrapping permite la reutilización de componentes de probada eficacia, y aprovecha la fuerte inversión inicial y la confianza de la organización.

El cambio de la interfaz gráfica es un método económico y eficaz, sin embargo, el wrapping de interfaz de usuario no siempre es suficiente, por lo tanto, es necesario acudir al wrapping de datos en el que se acceden a los datos del SIH a través de interfaces propietarias estándares como ODBC, JDBC u otros mecanismos de integración como XML o replicación de bases de datos que posibilitan el acceso a los datos desde los nuevos lenguajes de desarrollo.

La migración es una alternativa que también permite abordar los SIH. En esencia la migración de un SIH significa mover un sistema existente y operativo a una nueva plataforma conservando la funcionalidad de este, causando el menor trastorno a la operaciones existentes (BISBAL, LAWLESS et al. 1999).

Un sistema se abandona cuando no es capaz de seguir el ritmo de las necesidades del negocio y su reingeniería es imposible (BISBAL, LAWLESS et al. 1997).

Un estudio realizado en (LIN 2008) comprobó que la migración a sistemas relacionales se ha convertido en una opción de transición para más de 10.000 organizaciones en Estados Unidos, que siguen empleando los datos no relacionales base como archivos planos, bases de datos jerárquica o de red para sus sistemas heredados.

No obstante, la migración de un SIH es un procedimiento muy costoso que conlleva un riesgo de fracaso (THIRAN and VAN DEN HEUVEL 2004). Justificar una migración es más difícil que el desarrollo de aplicaciones comunes pero, en muchos casos, la migración puede ser vista como una última opción. Por lo tanto, antes de tomar la decisión de migrar es necesario, realizar un estudio intensivo, que debe llevarse a cabo para cuantificar los riesgos, beneficios, y justificar plenamente el desarrollo de un nuevo (SNEED 1995).

Consistente con lo anterior, se ha encontrado en la literatura diversos trabajos que consideran la migración de datos desde un SIH hacia una base de datos relacional (PRATHER, LOBACH et al. 1995; ALHAJJ 2003; HENRARD, CLEVE et al. 2004; LIN 2008) adoptándolo como un estándar de facto para el desarrollo de sistemas de información. En general, la mayoría de las estrategias

consideran la proposición de un nuevo modelo de datos que mantengan la semántica contenida en los datos heredados sin considerar las características propias del SIH (HENRARD, CLEVE et al. 2004).

La proposición de nuevos modelos implica la extracción del modelo de datos de un SIH, y para realizar esta tarea la técnica más utilizada es la ingeniería inversa. La ingeniería inversa permite la construcción de especificaciones de un mayor nivel de abstracción a partir de los datos de un SIH, códigos fuentes u otros. La ingeniería inversa de datos se ocupa principalmente de comprender la base de datos heredada y la extracción de su diseño (semántica del dominio) (CHIANG, BARRON et al. 1996).

En (ANDERSSON 1994) se propone un método para la extracción de un modelo entidad relación a partir de una base de datos relacional, en la que la información del modelo se obtiene por medio de consultas SQL establecidas en la propia aplicación y en las instancias de las relaciones. La ingeniería inversa y, en particular, la recuperación de la arquitectura, ayuda a la extracción de representaciones de más alto nivel de sistemas heredados, y ofrece soporte a los ingenieros para la evaluación, el mantenimiento y el desarrollo de sistemas software a gran escala (GARCÍA-RODRIGUEZ, POLO et al. 2004). Otras investigaciones como (HENRARD, ROLAND et al. 2007) proponen la obtención del esquema de base de datos heredada CODASYL (modelo de red) a partir de un proceso semiautomático mediante instrucciones DDL (Data Description Lenguaje) que no considera la normalización adicional del esquema de base de datos relacional.

Por otro lado en (CHIANG, BARRON et al. 1993; CHIANG, BARRON et al. 1994) también proponen una metodología para la extracción de un modelo de entidad relación extendido (MEE) a partir de una base de datos relacional. Este modelo ofrece una mayor comprensión para el mantenimiento, debido a que es semánticamente más rico. El principal aporte de esta investigación es la generación de dependencias de inclusión y justificar las transformaciones aplicadas a la base de datos para generar el MEE. Más tarde los autores proponen un marco de trabajo en (CHIANG, BARRON et al. 1996) para el diseño y evaluación de los métodos de ingeniería inversa para bases de datos relacionales. Este marco está compuesto por ocho criterios relevantes para la generación del modelo a partir de la base de datos. Igualmente en (JOHANNESSON 1994) se propone un método que convierte el modelo relacional en un esquema de modelo conceptual que permite mejorar la comprensión conceptual de los datos mediante un modelo de datos semántico. Esta investigación evidencia el modo en que las estructuras de objeto son identificadas en un esquema relacional, basado en las interacciones entre las claves y dependencias de inclusión, además del rol fundamental que cumplen los usuarios en la toma de decisiones. Del mismo modo en (PETIT, KOULOUMDJIAN et al. 1994) hacen lo propio con un método que incluye la normalización en el esquema relacional.

En otra dirección en (ABDELSALAM, AKHTAR et al. 2008) proponen un marco de trabajo denominado MIGROX, que facilita la migración de una base de datos relacional hacia bases de datos orientadas a objeto o XML. Resulta interesante que el marco de trabajo entrega como resultado dos esquemas de bases de datos (ORDB y XML Schema) que ayudan en la evaluación y elección de la base de datos de destino más apropiada. Igualmente, en (Wang, Lo et al. 2004) proponen una metodología tendiente a facilitar la obtención de un esquema XML, en lugar de un esquema DTD, a partir de una base de datos relacional. Esta investigación es una continuidad en la investigación iniciada en (ALHAJJ 2003), por lo que se podrían utilizar ambos estudios para migrar desde una base de datos heredada hacia XML.

En otra dirección, los autores (COLOSIMO, LUCIA et al. 2009) y (DE LUCIA, FRANCESE et al. 2008) presentan experimentos a profesionales que utilizan MELIS, una herramienta desarrollada como plugin de eclipse que permite la migración de sistemas desarrollados en plataformas COBOL hacia la Web. En dichos trabajos se realizaron pruebas con el objetivo de medir el grado de efectividad de la herramienta, logrando comprobar la disminución de tiempo que lleva la migración, tomando un cuarto del tiempo al haber utilizado otras herramientas tradicionales. En conclusión, MELIS reduce en promedio a menos de un quinto del tiempo normalmente utilizado para esta labor y, además, logró disminuir la brecha entre ingenieros expertos y novatos.

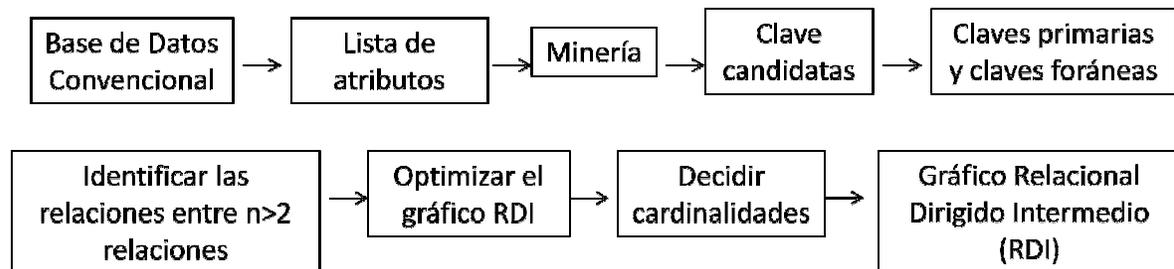
Del mismo modo, en (AVERSANO, CANFORA et al. 2001) se presenta una herramienta para migrar aplicaciones desarrolladas en COBOL hacia la web utilizando para ello la metodología MORPH descrita en (MOORE and MOSHKINA 2002). La metodología consiste en la descomposición inicial del SIH entre la lógica de la interfaz de usuario y la base de datos, posteriormente se realiza la migración de la interfaz de usuario a la web utilizando lenguaje ASP .NET y Visual Basic Script.

Mientras en (BODHUIN, GUARDABASCIO et al. 2003) proponen una estrategia incremental, descomponiendo la aplicación en una nueva arquitectura basada en el patrón de diseño Modelo Vista Controlador (MVC), dicha estrategia contempla un análisis y posterior reestructuración del SIH.

Por otro lado, en (ZOU and KONTOGIANNIS 2001) proponen la generación de un marco de trabajo para la migración de sistemas de información heredados hacia la Web mediante el uso de servicios Web y para lograrlo proponen tres pasos: primero, se identifican las partes de un SIH, luego se utiliza CORBA para generar el contenido a mostrar en la Web y, finalmente, un SOAP/CORBA que sirve como un puente para reducir las brechas tecnológicas y resuelve problemas de integración universal entre SOAP y HTTP.

En (ALHAJJ 2003) se presenta un enfoque para la obtención de un modelo de datos relacional que también utiliza ingeniería inversa. El modelo de datos relacional es resultado de la aplicación de varias etapas, tal como se muestra en la Figura 3.2, comenzando con la identificación de los atributos, a continuación se aplica técnica de minería de datos en búsqueda de las claves candidatas para luego, indicar las claves primarias y foráneas, luego se deciden las cardinalidades y en último lugar, se identifican las relaciones entre las entidades.

Esta investigación hace hincapié en que ante la imposibilidad de obtener información de los desarrolladores y la falta de documentación o inexactitud de ésta, se hace necesaria la implementación de una serie de algoritmos que permitan obtener conocimiento suficiente para lograr una comprensión de la estructura y sus características.



**Figura 3.2.:** Esquema de funcionamiento de la metodología.

El primer algoritmo se encarga de identificar las posibles claves candidatas de cada una de las relaciones, dichas claves son esenciales para el funcionamiento del segundo algoritmo que identifica las claves foráneas determinando la cardinalidad de estas, además de sugerir la optimización de claves foráneas, basada en el mejoramiento de referencias simétricas y transitivas, en resumen, se plantean algoritmos que eliminan la simetría y la transitividad.

El autor (ALHAJJ 2003) propone reducir al mínimo la dependencia de usuarios mediante el desarrollo, proporcionando agentes autónomos que puedan ayudar en este proceso de decisión, toda vez que sea posible. Además, supone que todas las relaciones de la base de datos se encuentran en tercera forma normal, es decir, el procedimiento descrito no está diseñado para solucionar problemas de diseño en las tablas y sus relaciones. Ésta es probablemente la diferencia mayor entre esta investigación y la propuesta, puesto que en este marco de trabajo se obtienen las DFs inicialmente, y a partir de ellas, se establece un modelo de datos relacional que hasta antes del proceso no existía.

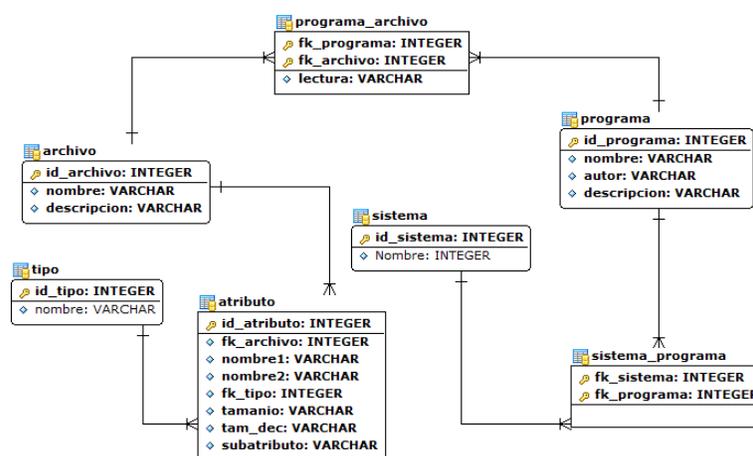
Por otro lado, se proponen específicamente algoritmos destinados a la obtención de claves foráneas y DFs, por lo que la contribución de esta investigación es el método de extracción, prueba de

ello es que otros autores como (Wang, Lo et al. 2004; LUBYTE and TESSARIS 2007; YEH, LI et al. 2008) han basado su trabajo en dichos algoritmos.

En otros trabajos como (MALDONADO, GUTIÉRREZ et al. 2009) se propone la obtención de un modelo de datos relacional desde sistemas heredados basados en archivos planos, considerando para ello las diversas fuentes de información (descripción de archivos, código e interfaces). En dicho trabajo se propone el método M-FF2RDB y se describe un caso práctico aplicado en una empresa del rubro eléctrico en el que se generó un nuevo esquema relacional para los datos del sistema heredado de la organización construido en lenguaje de segunda generación COBOL. Dicho trabajo tiene una relación estrecha con esta tesis debido a que la metodología para la obtención de modelos relacionales descrita representa una base conceptual para la definición del marco de trabajo.

Dicho método se compone de las etapas que se detallan a continuación.

*Etapas 1. Obtención de Meta-Atributos:* Etapa inicial cuyo objetivo es la extracción de información descriptiva contenida en los esquemas de la fuente de datos. La obtención de estos metadatos debe ser registrada en un repositorio que permita establecer una base descriptiva sobre la estructura del SIH, datos que son esenciales para las siguientes etapas. Los esquemas deben ser rescatados desde el propio SIH con ayuda de usuarios expertos que conozcan la estructura de la aplicación, y luego éstos serán procesados por una aplicación ad-hoc que permita extraer meta-atributos hacia el repositorio, donde se consignarán los datos que permitan identificar la estructura de datos y que contienen la estructura de los atributos. La Figura 3.3 muestra un esquema del repositorio de meta atributos.



**Figura 3.3:** Repositorio de meta-atributos.

*Etapas 2. Obtención de Sinónimos:* Habiendo terminado exitosamente la etapa anterior se procede a la depuración de la información almacenada en el repositorio, identificando posibles atributos duplicados.

La aplicación ad-hoc utilizada en la etapa anterior presentará las coincidencias de nombres, tipos y tamaños encontradas en los atributos de las estructuras de datos, de esta manera, los usuarios expertos podrán identificar y clasificar dichos atributos con el objeto de evitar duplicidad. Los atributos serán clasificados en la propia aplicación ad-hoc según la coincidencia de éstos otorgando, un porcentaje establecido y parametrizado en la aplicación. Finalmente, la aplicación generará un reporte con meta-atributos a partir de los sinónimos propuestos y posterior refinamiento de los usuarios expertos.

*Etapa 3. Obtención de Dependencias Funcionales:* Etapa en que se obtiene un repositorio de datos en forma automática mediante programas ad-hoc con las DFs a partir del análisis de los esquemas de la estructuras de datos mediante la combinación de las claves primarias e índices definidos en los esquemas de las estructuras de datos con el resto de los atributos. Posteriormente, se realiza un refinamiento manual por parte de los usuarios expertos.

*Etapa 4. Obtención de Modelo de Dato:* Etapa consistente en la obtención de un esquema de base de datos relacional normalizado, hasta tercera forma normal. Para la normalización, no se requiere intervención inicial de usuarios, puesto que con programas ad-hoc como RENO (DE MIGUEL and PIATTINI 1999) permite la obtención, a partir de las DFs, de un esquemas de base de datos relacional en tercera forma normal. Posteriormente, y con el fin de refinar el modelo obtenido en forma automática, los expertos podrán realizar los cambios necesarios que permitan mejorar el modelo resultante. En la Figura 3.4 se muestra el proceso completo, desde que se poseen los archivos hasta la obtención del modelo de datos relacional.



**Figura 3.4:** Esquema funcional del Método M-FF2RDB.

La etapa de obtención de DFs de (MALDONADO, GUTIÉRREZ et al. 2009) utiliza programas ad-hoc para obtener la estructura de datos en un repositorio, no obstante, esto no es suficiente para pensar en la extracción de DFs si se piensa en una base de datos obsoleta en la que gran parte de la lógica de negocio se encuentra en procedimientos almacenados, triggers, funciones u otros. Por otro lado, no se especifica claramente el rol que cumplen las personas ni los productos, sino más bien se especifica como una instancia específica y no genérica.

En efecto, la obtención de DFs es una técnica de minería de datos utilizada en la ingeniería inversa, por lo que se convierte en una práctica esencial en el ámbito de los SIHs. Durante esta investigación se han encontrado algunos algoritmos que facilitan la obtención de DFs (SAVNIK and FLACH 1993; CHIANG, BARRON et al. 1996; HUHTALA, KÄRKKÄINEN et al. 1999; LOPES, PETIT et al. 2000; NOVELLI, CICCHETTI et al. 2001; WYSS, GIANNELLA et al. 2001; LOPES, PETIT et al. 2002; BAIXERIES 2004). Sin embargo, en general, la problemática más común en estos algoritmos es el tiempo de respuesta cuando las entidades poseen una gran cantidad de atributos. Por lo tanto, la propuesta enfrenta dicha problemática implementando técnicas que faciliten la obtención de DFs en tiempos razonables, mediante la acotación del problema a resolver por estos algoritmos (por medio de la poda de atributos), y no mediante la creación o modificación de algoritmos.

Finalmente, para la generación del modelo de datos relacional a partir de DFs se requiere del uso de herramientas altamente especializadas. De este modo, se han encontrado dos aplicaciones que facilitan la generación del modelo; RENO (DE MIGUEL and PIATTINI 1999) y DBNormalizer (FESSENMEYER 2008). Ambas herramientas son resultado de experimentaciones y tesis de maestría tendientes a proponer un modelo de datos relacional a partir de DFs. El primero de ellos es una versión menos elaborada en relación a la Interfaz Gráfica de Usuario (IGU), sin embargo, es totalmente funcional, mientras el segundo posee una IGU altamente amigable y ofrece una amplia compatibilidad con herramientas que facilitan la obtención de DFs, como (RAFREIDER 2008) .

La obtención de un modelo de datos a partir de un SIH, por medio de un marco de trabajo, cobra especial relevancia debido a la gran cantidad de sistemas heredados que se mantienen en funcionamiento, para los que existe la necesidad de enfrentar mediante técnicas o métodos comprobados. De este modo, el aporte de esta investigación hacia la comunidad científica es el establecimiento de un marco de trabajo, compuesto por etapas que contribuyen a la generación de un modelo de datos relacional en 3FN, adoptando una estrategia basada en la semántica, y que actúa como una estructura tecnológica y conceptual en la que se plantean conceptos, criterios y prácticas estandarizadas.

La mayoría de las investigaciones estudiadas proponen algoritmos o métodos de extracción y generación de modelos de datos a partir de bases de datos relacionales (CHIANG, BARRON et al. 1993; ANDERSSON 1994; CHIANG, BARRON et al. 1994; HENRARD, ROLAND et al. 2007), mientras otros como (AVERSANO, CANFORA et al. 2001; BODHUIN, GUARDABASCIO et al. 2003; DE LUCIA, FRANCESE et al. 2008; COLOSIMO, LUCIA et al. 2009; MALDONADO, GUTIÉRREZ et al. 2009) lo hacen desde estructuras de datos basadas en archivos, principalmente desde COBOL.

Lo anterior se convierte en una motivación para emprender esta investigación, debido a que no se ha hallado investigaciones que faciliten un marco de trabajo referencial que permita la obtención de un modelo de datos a partir de un sistema heredado independiente de la estructura de datos utilizada.

Por otro lado, se han encontrado diversas investigaciones en que se proponen herramientas y algoritmos (FLACH and SAVNIK 1999; HUHTALA, KÄRKKÄINEN et al. 1999; LOPES, PETIT et al. 2000; NOVELLI, CICCHETTI et al. 2001; WYSS, GIANNELLA et al. 2001; LOPES, PETIT et al. 2002; BAIXERIES 2004) especializados que se adaptan a las distintas etapas del marco de trabajo propuesto aquí.

En definitiva, el estado del arte ha permitido descubrir estudios relacionados con esta investigación, y de este modo han impulsado al desarrollo de esta tesis y a la obtención de los resultados esperados.

## **Capítulo 4**

### **Marco de Trabajo EOG**

---

En este capítulo se describe detalladamente la propuesta de un Marco de Trabajo destinado a obtener un modelo de datos relacional a partir de un sistema heredado. El marco de trabajo está compuesto de etapas y actividades, que ejecutadas secuencialmente contribuyen a conseguir un modelo de datos relacional, que, en efecto, es una representación actualizada de los datos de un SIH. La ejecución de este marco típicamente deberá estar a cargo de un equipo de trabajo con apoyo de expertos y fuentes de información, de hecho, en su aplicación intervienen, expertos, componentes del SIH y documentación.

En nuestra tesis utilizamos como base algunas etapas del método M-FF2RDB (MALDONADO, GUTIÉRREZ et al. 2009), tales como la obtención de meta-atributos a partir esquemas de la fuente de datos y la obtención de sinónimos. Sin embargo, en ésta propuesta se han unido ambas etapas de manera que se logre extraer información sobre la estructura de datos, almacenarla en un repositorio y proveer de una interfaz de usuario que permita a los expertos interactuar con los meta-atributos y realizar un refinamiento inmediato de estos, descartando sinónimos en los nombres, tipos y tamaños de los atributos de las entidades involucradas.

A diferencia del método planteado por (MALDONADO, GUTIÉRREZ et al. 2009) éste marco de trabajo no se limita a fuentes de datos específicas sino más bien a fuentes heterogéneas, es decir, bastaría con tener una base de datos obsoleta, sea esta relacional o no, y como resultado de la aplicación del marco de trabajo se entregaría un modelo de datos normalizado, lo que serviría para encontrar y mejorar errores de diseño o en otro escenario confirmar que el modelo actual es consistente y no posee errores.

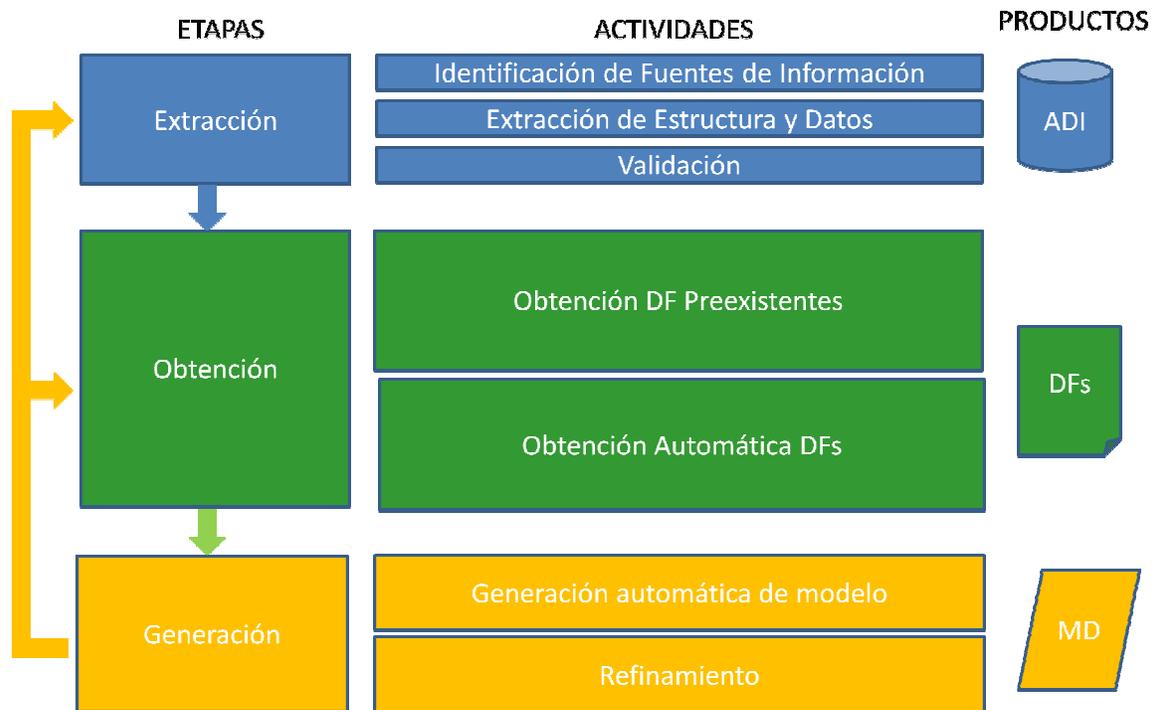
A continuación, en el puntos 4.1, 4.2 y 4.3 se describen en extenso las etapas y sub etapas que lo componen.

#### **4.1 Marco de Trabajo EOG**

La obtención de un modelo de datos relacional a partir de sistemas heredados, es una tarea que ha sido estructurada en un marco de trabajo denominado EOG (Extracción, Obtención y Generación) compuesto de tres etapas que, a su vez, se componen de actividades. El funcionamiento de EOG requiere del conocimiento de varios ámbitos, entre los cuales destacan el conocimiento sobre el dominio del negocio asociado a la fuente de datos del SIH, las fuentes de información, y la extracción de la estructura y migración de datos. Tanto la extracción de la estructura como la migración de los datos conllevan a la creación de tablas o eventuales modificaciones en las existentes, mediante cambios

en algunos tipos de datos, entre otros, a través de herramientas ad-hoc. La aplicación del marco de trabajo requiere de la participación de un equipo de trabajo, los sistemas heredados, herramientas y fuentes de información.

La Figura 4.1 muestra las etapas y actividades de EOG, así como los productos que se obtienen luego de aplicar cada etapa. Además, las etapas y las actividades se pueden ejecutar de manera iterativa.



**Figura 4.1:** Esquema de funcionamiento del Marco de Trabajo.

*ADI:* Almacén de Datos Intermedio que contiene la estructura original del SIH

*DFs:* Dependencia funcionales

*MD:* Modelo de datos relacional en 3FN

## 4.2 Roles del Equipo de Trabajo

La ejecución del Marco de Trabajo exige contar con un equipo de trabajo encargado de aplicarlo, el que debe poseer conocimientos sólidos sobre el ámbito del negocio, herramientas ETL

(Extract, Transform and Load) y bases de datos. Éste debe estar conformado por un jefe de grupo, quien estará encargado de dirigir la aplicación del marco.

La elección del jefe de equipo de trabajo debe tener directa relación con el nivel de conocimiento sobre el SIH y ante la ausencia de un liderazgo claro en ese ámbito se debería aplicar un criterio jerárquico basado en el liderazgo del grupo de trabajo. Los demás integrantes deben asumir funciones relacionadas con las etapas propias del marco de trabajo, dependiendo del tamaño del SIH. Los roles de los integrantes del equipo de trabajo son los siguientes:

- **Rol de Extracción:**

Función asociada a las actividades de extracción de la(s) fuente(s) de datos del SIH, con un fuerte apoyo de expertos. Para lograr este objetivo, se requiere el uso de herramientas que facilitan la extracción de la estructura y datos desde el SIH, en este sentido los integrantes del equipo deben dominar las herramientas ETL, así mismo, comprender conceptos propios de la creación y mantención de base de datos, que les permitan utilizar las fuentes de información, tales como documentación y componentes del SIH que contribuyan en este proceso. Además, deben contribuir con fuentes de información y reunir datos sobre las mismas. Esta labor debe ser liderada por el administrador de base de datos o la persona con mayor conocimiento sobre la fuente de datos del SIH, con apoyo de los demás integrantes del equipo y expertos.

- **Rol de Obtención:**

Labor ligada a las actividades propias de la etapa de obtención de DFs. Por ello debe existir un conocimiento acabado sobre el dominio del SIH y colaborar en el establecimiento de DFs preexistentes. Además, se debe comprender la documentación y componentes del SIH clasificadas como fuente de información. En este sentido la contribución de los expertos mediante el conocimiento específico sobre el dominio del problema es fundamental.

- **Rol de Generación:**

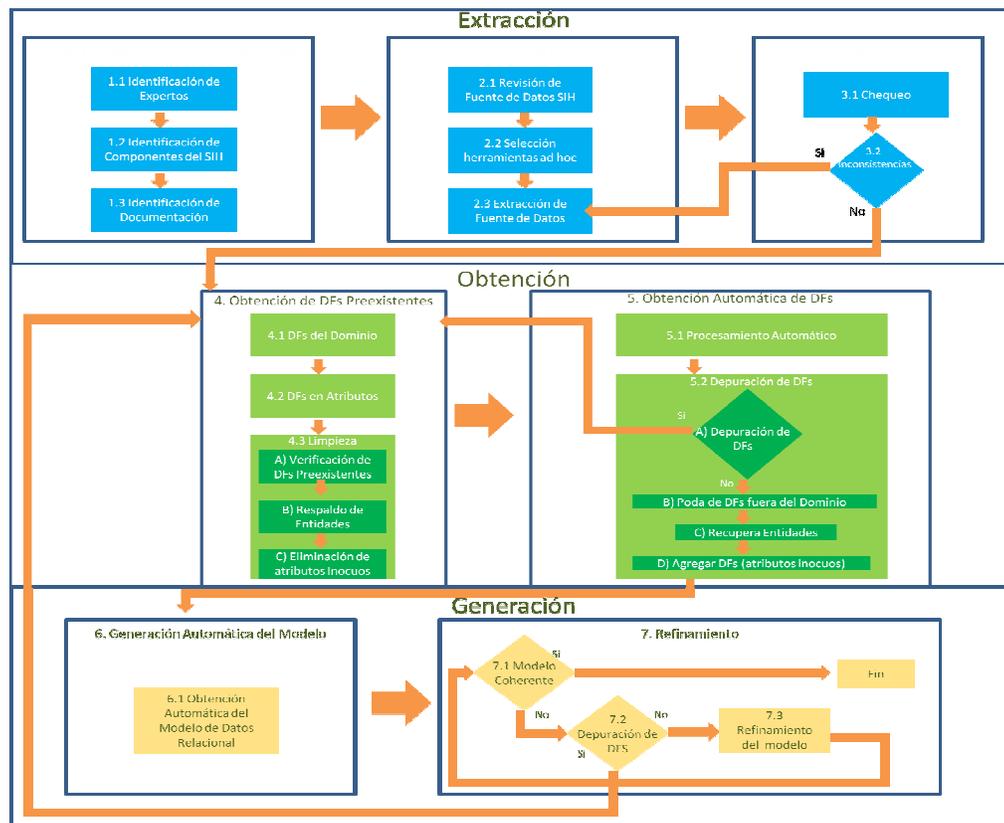
Facilita la generación del modelo de datos relacional a partir de las DFs obtenidas. Además, se chequean y proponen mejoras al modelo de datos preliminar. Consecuentemente y conforme a los resultados de la revisión del modelo propuesto, es posible volver a ejecutar la etapa de obtención para mejorar el modelo, de esta forma es necesario ejecutar ambas etapas de manera iterativa la cantidad de veces que sea necesario para satisfacer el criterio técnico de los expertos. Por esta razón, debe existir una comunicación fluida y coordinada entre todos los integrantes del equipo y expertos.

Finalmente, luego de haber conformado el equipo de trabajo se aplica EOG, comenzando por la etapa de extracción. A continuación, en los próximos puntos se detallan cada una de la etapas que contiene el marco de trabajo.

### 4.3 Etapas de EOG

Las etapas de EOG han sido diseñadas con el objeto de que sean ejecutadas por el equipo de trabajo en orden secuencial. Esto debido a que el resultado de una etapa es el insumo necesario para la siguiente.

La Figura 4.2 resume las principales etapas y actividades de EOG. Se observa que EOG está conformado por tres grandes etapas, las que se ejecutan de manera secuencial. La primera etapa (Extracción) tiene por objetivo generar un repositorio, lo más depurado posible, con los datos del SIH. La segunda etapa (Obtención), a partir de los datos del repositorio, más la información sobre DFs propias del dominio del problema, se ocupa de obtener un conjunto de DFs. En último lugar, la tercera etapa (Generación) crea un modelo de datos relacional en 3FN utilizando las DFs obtenidas en la etapa anterior.



**Figura 4.2:** Esquema general de EOG con flujos de información.

Se observa que el flujo en la ejecución del marco de trabajo contiene decisiones que pueden volver a ejecutar una actividad o una etapa. En lo que sigue de este capítulo, se describe en detalle cada una de las etapas y actividades de EOG.

### 4.3.1 Etapa de Extracción

Es la etapa inicial del marco y consiste en la extracción de la estructura y datos contenidos en las fuentes de datos de origen del SIH. Es un proceso semiautomático que debe ser llevado a cabo por el equipo de trabajo, apoyados por expertos que posean conocimientos técnicos específicos sobre el SIH.

En esta etapa se han definido tres actividades que deben ser ejecutadas secuencialmente. Sin embargo, pueden iterar, de esta manera el resultado de una actividad permite el inicio de la otra, tal como se exhibe en la Figura 4.3.



**Figura 4.3:** Esquema de funcionamiento de la etapa de Extracción.

La primera actividad permite identificar las fuentes de información del SIH y verificar la disponibilidad de éstas. En un SIH, luego de años de mantenimiento, actualizaciones y mejoras, los manuales de usuario y la documentación del diseño de sistemas, la mayoría de las veces, se encuentran desactualizados, inexactos y no reflejan la capacidad del sistema actual y su funcionamiento (BERGEY, NORTHROP et al. 1997).

La segunda actividad consiste en la extracción de la estructura y los datos contenidos en las fuentes de datos de origen del SIH, hacia un almacén de datos sobre el cual se realizarán posteriormente, operaciones para la obtención de DFs.

En último lugar, se debe validar el resultado de la extracción, consistente en realizar una verificación de la estructura y datos extraídos.

Como resultado final de esta etapa, se obtiene una réplica de la estructura de datos original, o parte de esta, en una nueva estructura de datos, y la identificación de las fuentes de información que contribuirán a realizar operaciones tendientes a generar el modelo de datos relacional. A continuación, se describen en detalle cada una de las actividades que la componen la etapa.

#### **4.3.1.1 Identificación de Fuentes de Información**

Esta actividad tiene por objeto identificar las fuentes que contienen información sobre el SIH. Para comenzar, se debe realizar un catastro de los elementos disponibles que resulten relevantes en términos de información descriptiva sobre el SIH. Las fuentes de información son diversas, y la cantidad, calidad y disponibilidad de éstas dependerá de cada SIH. Por ejemplo, la mayoría de las veces la documentación no existe o está desactualizada, por otro lado los equipos de trabajo que dieron origen al SIH ya se encuentran disueltos o inaccesibles (ALHAJJ 2003). La cantidad de fuentes de información disponibles tendrá una directa relación con el grado de desarrollo y mantenimiento que el SIH haya tenido durante el tiempo de funcionamiento.

Para el óptimo funcionamiento del marco de trabajo, es deseable contar con la mayor cantidad de fuentes de información, concretamente, códigos fuentes documentados que faciliten la comprensión, de los integrantes del equipo de trabajo y expertos. Evidentemente, este escenario no siempre es factible en todas las organizaciones que poseen SIH. En (ALHAJJ 2003), se indica que las fuentes naturales de información son los desarrolladores y/o la documentación. En efecto, contar con ellos es un factor diferenciador favorable. En orden a identificar el rol que cumplen en el SIH, se deben describir sus funciones básicas y la información que proveen. En consecuencia, se dispone de una clasificación de fuentes de información compuesta por expertos, documentación y componentes del SIH, las que se describen detalladamente a continuación.

##### **4.3.1.1.1 Identificación de Expertos**

Actividad que facilita la identificación y clasificación de personas reconocidas como una fuente confiable, es decir, un experto o perito que posea la capacidad técnica, habilidad o amplio dominio de conocimientos sobre el SIH o parte de éste.

Los expertos cumplen un rol importante, debido a que poseen información de primera fuente sobre el funcionamiento y flujos de trabajo del SIH, que no siempre se encuentran debidamente

documentados en códigos fuentes, diccionarios de datos, diagramas de flujo u otros. Por lo tanto, contar con personas especialistas con amplio conocimiento y dominio del SIH, es un gran aporte, puesto que han participado en el desarrollo del SIH o se han familiarizado con éste, ya sea como usuario, tal como lo haría el cajero de un banco que debe interactuar con el sistema de depósitos y cobros de documentos, ó como administrador de sistemas, quien tiene una relación más directa con la gestión y mantención de los datos. En la Tabla 4.1 se muestra la identificación de roles, descripción de los expertos, e información que éstos pueden proveer al equipo de trabajo.

<b>Rol</b>	<b>Descripción</b>	<b>Información que Provee</b>
Administrador del SIH	Profesional con fuerte formación y conocimientos sobre ingeniería de software y administración de recursos, que le permite evaluar y analizar el comportamiento del SIH.	- Flujos de Trabajo. - Roles de Usuario.
Analista del SIH	Profesional que juega un rol fundamental en el proceso de desarrollo del SIH y sus conocimientos están vinculadas principalmente en la toma de requerimientos.	- Flujos de Información. - Roles de Usuario.
Programador del SIH	Profesional que posee altos conocimientos sobre el desarrollo del SIH, o mantenimiento de este, dominando la sintaxis y semántica de los códigos fuente.	- Interpretación de códigos fuentes.
Administrador de Base de Datos del SIH	Persona responsable de manejar aspectos relacionados con el desempeño, integridad, seguridad, disponibilidad, desarrollo y soporte a pruebas, por lo tanto posee conocimientos sobre la estructura de datos del SIH.	- Interpretación de estructura y datos.
Jefe de Proyecto del SIH	Profesional con amplios conocimientos sobre el diseño de alto nivel y arquitectura del sistema, amplio dominio sobre patrones y abstracciones coherentes.	- Flujos de Trabajo. - Flujos de Información. - Roles de Usuario. - Visión general.
Usuarios del SIH	Están directamente relacionados con el sistema, lo que les permite participar y aportar activamente en el proceso de comprensión de flujos de trabajo y de información entre los módulos que comprenden el sistema heredado.	- Uso de programas (entrada y salida). - Dominio del SIH.

**Tabla 4.1:** Clasificación de Expertos.

#### **4.3.1.1.2 Identificación de Componentes del SIH**

Los componentes del SIH son diversos y estos contribuyen con un conjunto de información que facilita la comprensión de las fuentes de datos del SIH. Por ejemplo, las aplicaciones comúnmente pueden ser consultadas con el objeto de obtener información sobre las entradas y salidas en los formularios; obtener alguna descripción de las estructuras de datos por medio de documentación incluida dentro de los códigos fuentes. Los componentes del SIH agrupan todas aquellas fuentes de datos electrónicas que permiten interactuar al equipo de trabajo y expertos con el SIH mediante interfaces gráficas de usuario, códigos fuentes, la propia estructura y los datos del SIH. Dichos

componentes se deben estudiar con el objeto de deducir información que permita explicar el comportamiento y flujo de los datos en el sistema. La Tabla 4.2 establece una clasificación de los componentes.

<b>Fuente</b>	<b>Descripción</b>	<b>Información que Provee</b>
Códigos Fuentes	El código fuente de un programa es un componente del SIH y en él está descrito por completo su funcionamiento. Puede ser estudiado o analizado por los miembros del equipo de trabajo y expertos. Por esta razón, es relevante averiguar el lenguaje en que fue desarrollado para realizar un análisis semántico.	Semántica de los programas del SIH.
Programas Ejecutables	Los programas o aplicaciones ejecutables son un componente esencial de los SIH; entregan acceso a las interfaces gráficas de usuario, y de este modo es posible analizar la funcionalidad y flujo de información mediante la contrastación de los datos en las propias fuentes de datos del SIH.	Información requerida y entregada por interfaces de usuario del SIH.
Estructura de datos	Fuente principal de información que permite la extracción de los datos y metadatos de la estructura de la fuente de datos del SIH.	Esquemas y datos del SIH.

**Tabla 4.2:** Clasificación de Componentes del SIH.

Comúnmente, los componentes son conocidos por los expertos, ya sea porque han participado en el desarrollo de las aplicaciones, las utilizan como usuarios o bien realizan mantenimiento en sus fuentes de datos.

#### **4.3.1.1.3 Identificación de Documentación**

La documentación es un conjunto de elementos registrados sobre algún soporte que facilita el conocimiento, interpretación, comprensión y divulgación del sistema. Ésta provee los antecedentes esenciales, concretos y permanentes y se constituye como un medio idóneo de consulta general para tomar conocimiento actual del SIH.

La documentación que describe el diseño o funcionamiento de un SIH es relevante para la comprensión y sociabilización con conceptos que se encuentran intrínsecos en el dominio del problema. A continuación, en la Tabla 4.3, se muestra una clasificación de la documentación posible de hallar en un SIH.

<u>Documentación</u>	<u>Descripción</u>	<u>Información que Provee</u>
Diccionarios de Datos	Este conjunto de metadatos permite descubrir de manera directa las características lógicas de los atributos de las tablas, tales como claves primarias, índices, tamaño, tipo de datos. Los elementos más importantes son flujos de datos, almacenes de datos y procesos.	Descripción del modelo de datos (Entidades y Atributos) del SIH.
Modelo de Datos	Permite describir detalladamente las estructuras de datos, sus tipos de datos y restricciones de integridad. Estas estructuras se encuentran definidas en dos lenguajes: DDL (Data Definition Language) el que está orientado a describir en forma abstracta y el DML (Data Manipulation Language) que permite describir las operaciones de manipulación de datos normalmente con lenguaje SQL por medio de un SGBD. En el caso de archivos planos se utilizan ficheros File Description.	Descripción general de la estructura de datos del SIH.
Especificaciones de Programa	Facilita la comprensión de los programas debido a la representación gráfica de los flujos de información. Particularmente, un DFD facilita la visualización de los flujos en distintas dimensiones, obteniendo mayor granularidad en el diagrama de expansión.	Flujo de información en SIH.

**Tabla 4.3:** Documentación que contribuyen al Marco de Trabajo.

De acuerdo a la clasificación de las fuentes de información, es posible inferir que uno de los factores relevantes para una aplicación exitosa del marco de trabajo es la cantidad y disposición de las fuentes de información. Bajo esta premisa, la aplicación de EOG se podrá ver enfrentada a distintos escenarios, con mayor o menor cantidad de fuentes de información que podrían alterar los resultados.

El escenario ideal es que todas o al menos la mayoría de las fuentes de información mencionadas se encuentren disponibles, además de otras que no hayan sido clasificadas en EOG. Sin embargo, los años de operación de algunos SIH hace poco probable que se cuente con todos ellos, en especial la documentación. De esta forma, contar con expertos cobra mayor relevancia debido a que estos poseen información que puede suplir en parte la documentación inexistente.

El peor de los escenarios se daría ante la ausencia de expertos al interior de la organización. En tal caso, la comprensión de la estructura de datos del SIH se hace más compleja. Sin embargo, se podría recurrir a expertos que se encuentren fuera de la organización, que posean un grado de experiencia en el negocio o en el propio SIH.

Ante la ausencia definitiva de expertos desarrolladores, se deberá recurrir a los propios usuarios del SIH que posean conocimientos avanzados sobre su funcionamiento a nivel operativo. De esta manera, el conocimiento operacional aportará aspectos propios del funcionamiento de las aplicaciones.

Los resultados de la aplicación de esta actividad se deben registrar en una tabla que contenga datos sobre las fuentes de información encontradas tal como se muestra en el ejemplo de la Tabla 4.4, el registro facilitará la identificación de las fuentes de información durante la aplicación del marco de trabajo. Es relevante tener como referencia la calidad de las fuentes de información, para lo cual se debe clasificar del siguiente modo:

*Buena:* Fuentes de información que se encuentren legibles y accesibles de manera rápida y sencilla.

*Regular:* Fuentes de información que se encuentren legibles pero con un acceso más restringido, por lo que contar con ella demandaría más tiempo.

*Mala:* Fuentes de información que se encuentren ilegibles y/o inaccesibles.

<u>Clasificación</u>	<u>Sub Clasificación</u>	<u>Descripción</u>	<u>Ubicación</u>	<u>Calidad</u>
Aplicación	Base de datos	Dbasic	servidor Fenix II	Buena
Documentación	Modelo de Datos	M.D. Sistema RRHH	PC gerente	Regular
Experto	Desarrollador	Administrador De BD	Unidad de Informática	Buena

**Tabla 4.4:** Ejemplo de Fuentes de Información a utilizar en EOG.

#### **4.3.1.2 Extracción de Estructura y Datos**

Actividad semiautomática que tiene por objeto la extracción de los datos y estructura original de la fuente de datos heredada y la información registrada en ésta. Este proceso se realiza utilizando herramientas ad-hoc, y en general se podrá efectuar mediante algún ETL (VASSILIADIS, SIMITSIS et al. 2002; HENRARD, CLEVE et al. 2004) que debe ser manejado por el equipo de trabajo y apoyado por los expertos.

La primera actividad tiene relación con la revisión de las fuentes de datos originales del SIH. Esta información es relevante para la búsqueda de interfaces y protocolos de conexión que faciliten la comunicación entre las fuentes de datos originales del SIH y el ADI. La segunda actividad es complementaria a la anterior, y permite seleccionar las herramientas que serán utilizadas para realizar la extracción. En las próximas secciones se describen en detalle cada una de las actividades.

##### **4.3.1.2.1 Revisión de la(s) Fuente(s) de Datos Original(es) del SIH**

Actividad en la cual se exploran las características técnicas de las fuentes de datos originales del SIH y se seleccionan las entidades que serán extraídas hacia el ADI.

Inicialmente se debe localizar el origen y proveedor de los datos, luego obtener toda la información posible acerca de la naturaleza de los datos. Se deben comprender las características de los datos y averiguar si existe algún tipo de restricción de acceso a los datos.

Esta revisión permite conocer información relacionada con la descripción técnica de la fuente de datos original, y la arquitectura en que se encuentra funcionando. Es decir, aspectos relevantes como tipo de estructura de datos, nombre del software de aplicación, tipo de arquitectura bajo la que se encuentra funcionando y métodos de acceso a ésta.

Esta información es relevante, debido a que facilita la búsqueda de mecanismos que permiten el acceso a las fuentes de datos originales del SIH, comúnmente son herramientas y drivers propietarios los que facilitan la conexión por medio de algún protocolo utilizado como estándar de acceso a los datos.

La elección de las entidades que serán extraídas es un proceso completamente manual, y el criterio de inclusión y exclusión es particularmente técnico, es decir, de un universo de entidades en la(s) fuente(s) de datos original(es) del SIH se deben seleccionar las que tienen relación con el sistema de información para el cual se requiere el modelo de datos relacional. De este modo, se requiere de conocimientos avanzados sobre la fuente de datos, y contar con fuentes de información suficientes.

El equipo de trabajo y expertos deben identificar en las propias fuentes de datos originales las entidades que serán extraídas, apoyándose fundamentalmente en la información que se pueda inferir desde los códigos fuentes de las aplicaciones, diccionarios de datos y especificaciones de programas. Esta revisión es meramente técnica y tiene su fundamento en la optimización de los resultados, puesto que las entidades seleccionadas serán procesadas y, posteriormente, facilitarán la obtención de DFs. En orden a mantener un adecuado registro de las entidades que se extraerán, éstas se deben registrar en una tabla, tal como se exhibe un ejemplo en la Tabla 4.5.

<u>Servidor</u>	<u>Entidad</u>	<u>Descripción</u>
Abraxas	Paciente	Tabla que registra datos personales de pacientes
Abraxas	Cuentas	Tabla que registra la generación de cuentas corrientes

**Tabla 4.5:** Ejemplo de registro de entidades a extraer.

En general, los datos pueden provenir de ubicaciones externas en distintos formatos, es decir, archivos planos, bases de datos relacionales y no relacionales que operan sobre plataformas que van desde computadores de escritorio hasta mainframes. Por otro lado, los medios de almacenamiento podrían incluir discos duros, cintas o discos ópticos. Los medios de transmisión pueden incluir

protocolos de transferencia de archivos, o mapeo de unidad. En ciertas ocasiones los archivos de datos contienen información de diseño de registro.

Frecuentemente, se deben obtener datos desde agentes externos o de otros departamentos de la organización. En consecuencia, se deben considerar las restricciones organizativas respecto de la seguridad de los datos.

Comúnmente, el control, procesamiento y distribución de los datos está a cargo del equipo de sistemas de información de la organización. Por lo tanto, se deben solicitar acceso a las fuentes de datos del SIH de manera absoluta, identificando el método de acceso, procurando utilizar un acceso de forma nativa, con el objeto de mejorar la velocidad y calidad de la conexión.

Las fuentes de datos basadas en archivos planos almacenan información en registros y campos simples y no en estructuras de base de datos complejas. Además, los archivos contienen solo datos separados por algún carácter o espacio predeterminado y un carácter CR/LF<sup>1</sup> al final de cada registro. Por lo que es primordial contar con los diseños de registro, puesto que proporcionan características tales como las longitudes de registro, posiciones de los campos, tipos de datos y descripciones de los campos. Por ejemplo, en las tablas de DB2, todos los ficheros de mainframe contienen un archivo de registro asociado, mientras que otros poseen archivos con definiciones externas que contienen el diseño de registro para archivos de datos específicos. Por ejemplo, en OS/400 el diseño se denomina Formato de Definición de Campo (FDF), mientras que para COBOL se conoce como Descripción de Archivos (FD). Por otro lado, los estándares de acceso a datos ofrecen una comunicación entre fuentes de datos actuando como un mediador entre éstos.

En particular, ODBC es uno de los estándares de acceso a datos más comunes y su objetivo es facilitar el acceso a cualquier dato desde cualquier aplicación, sin importar la marca del producto utilizado como fuentes de datos. ODBC es utilizado como una capa intermedia, denominada nivel de interfaz de cliente entre los sistemas y la fuente de datos.

JDBC posee la misma filosofía que ODBC, y actúa como una interfaz de comunicación, es decir, como Interfaz de Programación de Aplicaciones (IPA), permitiendo la ejecución de operaciones sobre fuentes de datos específicamente desde aplicaciones diseñadas en lenguaje JAVA, independientemente de la plataforma donde se ejecute o resida la aplicación.

---

<sup>1</sup> Combinación de dos códigos de control: CR (retorno de carro) y LF (salto de línea), permiten crear una nueva línea.

#### 4.3.1.2.2 Selección de ADI y Herramientas

Esta actividad permite seleccionar el ADI y la herramienta ETL que facilitará la extracción de la estructura y datos original desde el SIH hacia el ADI.

Luego de identificar los datos de origen del SIH se debe decidir cómo trasladarlos hacia una unidad accesible de una manera sencilla. Si los datos de origen provienen de un microcomputador o un mainframe éstos se deben transferir a un computador personal que posea un sistema de gestión de base de datos (SGBD) representados como un esquema de base de datos.

La elección del ADI es meramente técnica y práctica, debido a que debe existir un adecuado dominio del software de aplicación donde residen los datos y se realizarán las operaciones que darán paso a la obtención del modelo de datos relacional. Por otro lado, el ADI debe ofrecer compatibilidad con la herramienta ETL que facilitará la extracción. Muchas herramientas ETL son propietarias de la firma desarrolladora y poseen compatibilidad únicamente con sus productos, por lo tanto, es un aspecto que se debe considerar al momento de realizar la elección.

Las herramientas ETL son el complemento ideal para realizar la extracción, puesto que típicamente permiten extraer (Extract) datos desde fuentes de datos diversas y heterogéneas, transformar (Transform) para ajustar los datos a la fuente de destino y cargar (Loading) que es el proceso de poblar datos en el ADI. Las herramientas ETL son necesarias, puesto que integran datos desde distintas fuentes de datos, las que permiten limpiarlos, es decir, remover datos transaccionales e introducir consistencia, además de reestructurar los datos para un proceso analítico. A continuación, se detallan las características de cada uno de los procesos involucrados.

- *Extracción:* Permite acceder a las fuentes de datos. De este modo el equipo de trabajo selecciona los elementos que serán extraídos. Esta actividad se puede ejecutar la cantidad de veces que sea necesaria, con el objeto de satisfacer las necesidades requeridas. Trabaja con formatos de datos compatibles con los procesos de transformación y carga.
- *Transformación:* Transforma los datos extraídos y les da un formato para la nueva plataforma de destino. Esto involucra el resumen de valores de múltiples registros, asegura la calidad de los datos y excepciones de procesamiento, separa columnas específicas para la carga, decodifica valores, codifica según la fuente de destino y realiza cálculo de datos.
- *Carga:* Proceso que traslada los datos a la plataforma de datos de destino (ADI) y permite establecer una secuencia en la carga. Además, existe un proceso de “bulk loading”,<sup>2</sup> que

---

<sup>2</sup> Término inglés que significa carga masiva de datos.

facilita el poblado de datos para entidades con grandes volúmenes de datos y facilita la reindexación de datos antes y/o después de la carga.

#### **4.3.1.2.3 Extracción de Fuente de datos**

La extracción de estructura y datos se ha caracterizado como proceso semiautomático, ya que requiere de la intervención directa y manipulación guiada de las herramientas por el equipo de trabajo. Es necesario planificar dicho proceso manteniendo una claridad sobre la ubicación del ADI, de esta manera se indica correctamente dicha información a las herramientas ad-hoc.

La extracción se llevará a cabo mediante el uso y configuración de la herramienta ad-hoc, tales como ETLs, comandos y herramientas propias de los SGBDs, para lo cual es necesario establecer una conexión mediante algún protocolo de acceso a datos conocido. Posteriormente, es necesario seleccionar los elementos de la fuente de datos que serán extraídos, es decir, aquellos que fueron identificados en la Tabla 4.5.

En general, la extracción se realiza desde fuentes de datos de origen del SIH heterogéneas, es decir, archivos de texto plano, bases de datos relacionales y no relacionales de distintas marcas comerciales y versiones.

Frecuentemente, algunas entidades en la(s) fuente(s) de datos del SIH poseen una gran cantidad de atributos, y un gran volumen de tuplas, lo que provoca dificultades en la extracción debida, principalmente, a que en ocasiones existen inconsistencias sintácticas en los datos de origen del SIH, producto del bajo mantenimiento o limpieza de los mismos. Las herramientas ETLs facilitan la extracción de los datos y la propia estructura de cada entidad, realizando un análisis de los datos de origen, y, de este modo, proponen un tipo, longitud y rango de datos. En efecto, es común que esta automatización genere incompatibilidad entre los datos de origen y la estructura del ADI, debido a que los datos se encuentran fuera de rangos aceptados, lo que genera inconsistencias indetectables e incontrolables por las herramientas ETL a causa del alto volumen de datos.

En tal caso, será necesario recurrir a otros métodos de extracción propios a la fuente de datos. Por ejemplo, si la fuente de datos original del SIH es una base de datos, se pueden utilizar herramientas propias del SGBD, o comandos que faciliten el volcamiento (vaciamiento) de los datos mediante consultas SQL en un archivo plano o XML. Informix 5 puede utilizar la sentencia “unload to” a continuación de cualquier consulta SQL y facilita la exportación de datos en un formato estándar. Igualmente MySQL permite extraer datos mediante la sentencia mysqldump. En ambos ejemplos, se

genera un archivo electrónico en un formato conocido que puede ser utilizado posteriormente para agregar los datos al ADI.

Tal como se ha indicado, este es un proceso semiautomático y puede generar errores, por lo que debe haber un especial cuidado en la codificación de las fuente de datos de destino, es decir, se debe verificar que los datos extraídos coincidan con los contenidos en las fuentes de destino. Las dificultades más comunes en la extracción de datos están relacionadas con los tipos, tamaños y formatos de los atributos. Por ejemplo, los atributos de tipo fechas, numéricos, cadenas de carácter y multimedia, generalmente provocan dificultades en el proceso de extracción, debido a que los tipos de datos no son compatibles entre una fuente de datos y otra.

En la medida que la fuente de datos de origen contenga atributos que almacenen objetos más complejos que posean una gran magnitud, las probabilidades de incurrir en inconsistencias son mayores. Las herramientas ETL no poseen utilidades destinadas a comprobar los resultados de la extracción. Por esta razón la comprobación se debe realizar manualmente y por medio de herramientas ad-hoc teniendo especial cuidado en los atributos ya mencionados. Las herramientas ETL poseen asistentes que facilitan su uso y permiten generar acciones siguiendo sólo algunos pasos de manera guiada.

El proceso de extracción comienza con la identificación de la fuente de origen, es decir, desde donde se desean copiar los datos. En general, las herramientas proveen interfaces de usuario amigables que facilitan el acceso a fuentes de datos heterogéneas. Dependiendo del tipo de fuente de datos, se deben especificar más datos. Por ejemplo, en el caso de archivos planos se debe identificar adicionalmente los delimitadores de columnas y el fin de línea. Luego, se selecciona la estructura de datos de destino (ADI), es decir, donde se extraerán los datos. En este proceso se pueden realizar transformaciones, comúnmente llamadas “mapping” donde los destinos pueden ser tablas, archivos o servicios web.

Seguidamente, se deben seleccionar los elementos a extraer desde la fuente de datos de origen del SIH, conforme a lo establecido previamente en la Tabla 4.5. Frecuentemente, las herramientas ETL proveen de interfaces gráficas de usuario que permiten explorar el contenido de la fuente de datos original del SIH lo que mejora la selección. Además, permiten la inclusión de consultas SQL que facilitan el filtrado de datos según el criterio técnico del equipo de trabajo y expertos.

Seguidamente, se debe configurar la asignación de columnas o transformaciones, indicando si desea que el propio ETL cree la(s) tabla(s) en el ADI de manera automática, de acuerdo a un análisis

que realiza la herramienta sobre los datos del origen, mediante la inferencia en los tipos de datos originales. Esta opción permite disminuir tiempo en la creación de grandes estructuras que posean muchos atributos, sin embargo, se asume un riesgo de incompatibilidad en los atributos. No obstante se puede realizar manualmente por medio del mapeo de los atributos de origen con los del ADI, el que puede materializar con herramientas ad-hoc indicadas en el capítulo 5.

En último lugar, luego de realizar los pasos descritos previamente, están dadas las condiciones para realizar la extracción. Como resultado de esta actividad se obtiene una réplica del modelo de datos de origen del SIH, o parte de éste, en una nueva estructura definida como ADI.

#### **4.3.1.3 Validación**

Actividad manual en que interviene el equipo de trabajo apoyado por expertos, con el objetivo de validar y contribuir a la estructura de datos contenida en el ADI. Es necesario realizar una revisión de las entidades y sus atributos, descartando que se hayan producido inconsistencias.

Los tipos de datos almacenados en los atributos son diversos, y cobra especial relevancia corroborar que éstos han sido correctamente extraídos. Por ejemplo, los tipos de datos numéricos se diferencian esencialmente en el nivel de exactitud que poseen, por lo tanto, es posible encontrar tipos numéricos enteros, decimales y otros. Por otro lado, los tipos de atributo fecha y hora representan valores temporales que pueden ser almacenados en distintos formatos, según la ubicación geográfica. Consecuentemente, si se ha encargado a la propia herramienta la creación de entidades en el ADI, podrían existir inconsistencias, producto de errores en la inferencia que la propia herramienta realiza sobre el tamaño y tipo de los datos. Por lo tanto, podrían haberse extraído datos truncados que no representen la realidad de la fuente de datos del SIH.

En caso de encontrar inconsistencias tanto en la estructura como en los datos, es necesario tomar una decisión que permita solucionar la problemática, dicha solución corresponde a la realización de actualizaciones sobre los datos registrados, o realizar nuevamente el proceso de extracción de la estructura y/o datos. La validación está compuesta sólo por una actividad, la que tiene relación con el chequeo de inconsistencias producidas al momento de realizar la extracción. A continuación, se describe en detalle dicha actividad.

##### **4.3.1.3.1 Verificación de Inconsistencias**

El principio de la verificación tiene por objeto validar que la etapa de extracción sea una contribución para la etapa siguiente, donde se obtendrán las DFs. Comúnmente, la falta de consistencia

en los datos contenidos en el ADI es provocada por errores en los datos originales del SIH. Estos errores por lo general, son producto de problemas en la validación de la entrada de datos del SIH, y un bajo o nulo mantenimiento de los datos. Consecuentemente, como resultado del proceso de extracción, se pueden generar cambios en la nueva estructura, esto es, diferencias entre la fuente de datos de origen del SIH y el ADI.

La verificación consiste en realizar inspecciones visuales y/o automáticas por medio de una aplicación ad-hoc habilitada para este fin. Por ejemplo, un atributo de tipo decimal puede aparecer truncado en el ADI con 2 decimales, mientras que la fuente de datos de origen indica 4 decimales. Esto provocaría una incongruencia de la realidad de los datos, generando una alteración en los resultados del marco de trabajo.

La Figura 4.4 muestra la estructura de una entidad de un SIH contrastada con la nueva tabla establecida en el ADI. Tal como se muestra, existen algunas inconsistencias respecto de los tipos de datos, debido a que la estructura de la entidad del ADI no posee todos los tipos de datos establecidos en el origen, así por ejemplo, el atributo “codpacie” se encuentra establecido en la fuente de datos del SIH como serial, es decir un atributo entero auto numérico, sin embargo, en la estructura de destino se observa como Varchar. Para este ejemplo, es probable que no existan dificultades en la obtención de DFs, pero en otros más complejos como fechas mal formadas o números truncados, se podría generar una distorsión de los datos obteniendo como resultado DFs que no responden a la realidad del problema.

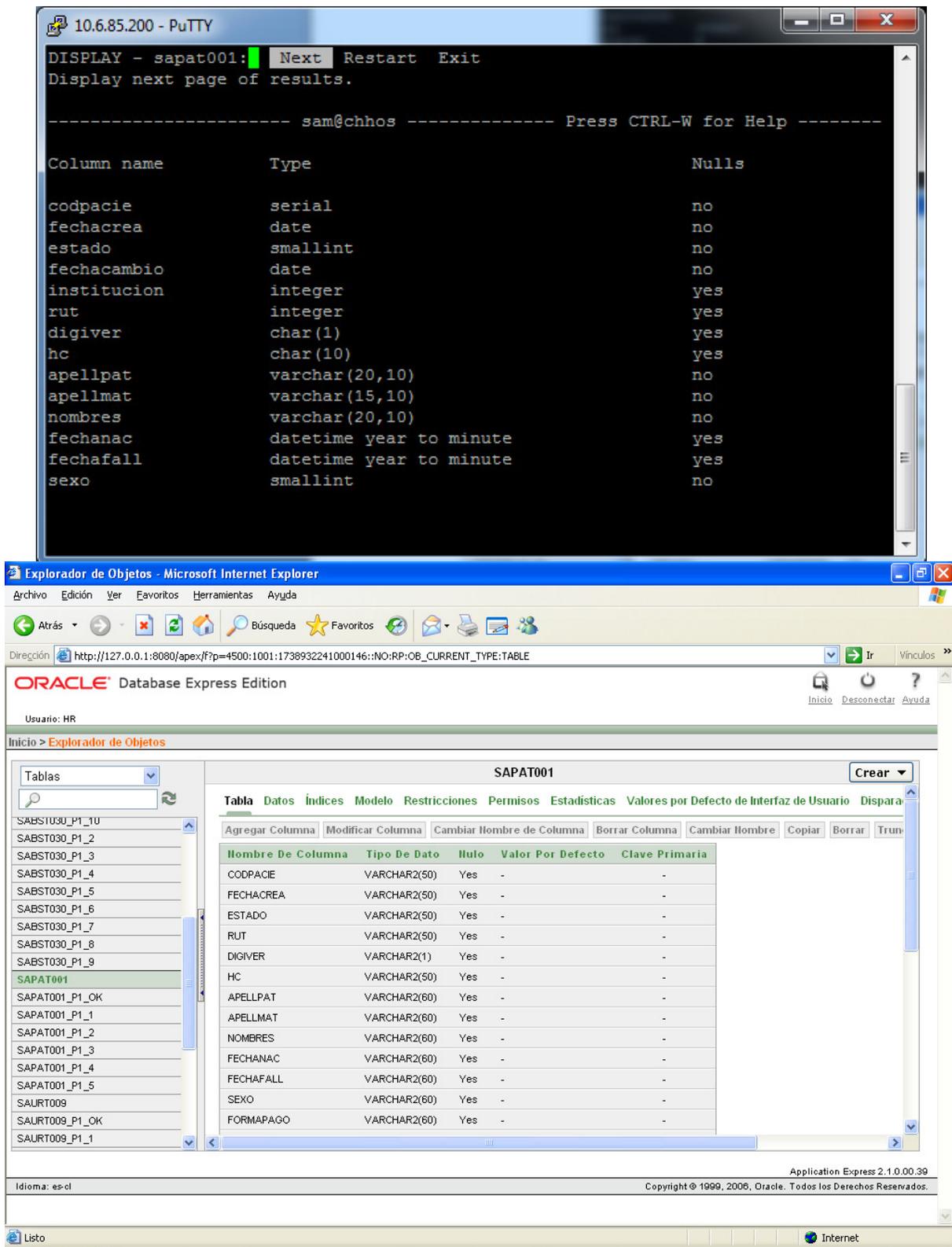


Figura 4.4: Estructura de tabla de origen y ADI.

Es posible que la estructura de una entidad extraída no coincida en su forma debido a la omisión o cambio en uno o varios de sus atributos, por ello es necesaria la intervención para corregir dicha inconsistencia. Cada vez que es detectada una inconsistencia se debe corregir mediante soluciones manuales, esto es, cambiar el tipo o tamaño de los atributos o realizar nuevamente la extracción de la entidad inconsistente.

Finalmente, y luego de obtener una réplica o parte de la fuente de datos original del SIH en el ADI, están dadas las condiciones para ejecutar la etapa de Obtención. Los resultados de la etapa de Extracción son suficientes para continuar con la aplicación del marco de trabajo. En concreto, dichos resultados reflejan:

- La identificación de fuentes de información.
- Un ADI que contiene las entidades que han sido extraídas desde la fuente de datos original del SIH. Dichas entidades se deben identificar indicando los atributos y datos básicos propios de ésta, tales como cantidad de atributos y tuplas.

#### **4.3.2 Etapa de Obtención**

Esta etapa facilita la obtención de DFs a partir del ADI, obtenido como resultado de la etapa anterior. Estas DFs son obtenidas mediante un proceso híbrido y semiautomático en que intervienen el equipo de trabajo, expertos, fuentes de información y herramientas ad-hoc. Esta etapa concentra su esfuerzo en la obtención de DFs, aplicando algunas acciones sobre la nueva fuente, y para cumplir dicho objetivo se consideran actividades y subactividades que deben ser aplicadas en orden secuencial tal como se muestra en la Figura 4.5. No obstante, algunas pueden iterar.



**Figura 4.5:** Esquema de funcionamiento Etapa de Obtención de DFs.

La primera actividad es un proceso manual que facilita la identificación de DFs mediante el conocimiento específico de los expertos que aportan información relacionada con la estructura de las entidades, dicha información es conocida o establecida en el dominio del problema. La siguiente actividad tiene por objeto la obtención automática de DFs no triviales a partir de un esquema de base de datos, en lo posible, podada, es decir, con menos atributos que los originales.

La fuerte participación de expertos apoyados con fuentes de información, facilitan la obtención de DFs, debido al amplio conocimiento sobre el negocio. En los siguientes puntos se describe detalladamente cada una de las actividades estableciendo de una manera clara el insumo que requieren y el resultado que éstas proveen.

#### 4.3.2.1 Obtención de DFs Preexistentes

La identificación de DFs preexistentes es una actividad manual, que debe ser implementada por los integrantes del equipo de trabajo en la que predomina el conocimiento específico de los expertos y la información que pueda emanar desde las fuentes de información, con el fin de facilitar la tarea a las herramientas encargadas de obtener de manera automática las DFs. No obstante, el sólo conocimiento específico de los expertos no es suficiente para establecer una DF preexistente, de esta forma, es necesario realizar comprobaciones mediante revisiones a la documentación y componentes

del SIH. Las fuentes de información identificadas en la etapa anterior son diversas, y aportan distintos niveles de información, las que serán útiles para confirmar o contradecir una DF establecida por un experto. En la Tabla 4.6 se muestran las fuentes de información junto a las actividades y los resultados esperados.

<u>Fuente</u>	<u>Actividades</u>	<u>Resultados</u>
Códigos Fuentes	-Revisar los archivos electrónicos que facilitarán la comprensión de procesos de validación de información en las aplicaciones y comentarios de los desarrolladores que lograrían aclarar el flujo de los datos. -Realizar el seguimiento en este tipo de fuente puede ser muy útil, pero requiere de un conocimiento muy específico para su análisis, además se podría llegar a necesitar un tiempo excesivo para encontrar los resultados buscados.	Comprensión sobre validación de información por medio interfaces de usuario y estructura de datos.
Programas Ejecutables	-Realizar pruebas de software, particularmente en la interfaz gráfica proveyendo datos, con el objeto de corroborar o refutar que los programas poseen las validaciones suficientes para asegurar que se cumplan las DFs encontradas. Esta acción puede ser complementada con un análisis a las tuplas de relación ubicada en la fuente de datos.	Comprensión sobre validación de información por medio de interfaces de usuario.
Diccionarios de Datos	-Analizar los metadatos que facilitarían una comprensión de la semántica de las entidades.	Análisis de atributos en búsqueda de DFs.
Modelo de Datos	-Interpretar la descripción de las estructuras de datos encontradas en script SQL o FD (File Description) en caso de archivos, colocando especial énfasis en las observaciones y comentarios de los desarrolladores que faciliten la comprensión del dominio de la relación.	Análisis de atributos en búsqueda de DFs.
Especificaciones de Programa	-Observar las especificaciones de programas haciendo hincapié en el análisis gráfico de los flujos de datos (DFD) en distintas dimensiones, obteniendo una mayor granularidad en el diagrama de expansión.	Comprensión sobre el flujo de datos en el SIH.

**Tabla 4.6:** Fuentes de información y Actividades.

La obtención de DFs preexistentes está compuesta por dos sub actividades tendientes a facilitar la identificación de DFs propias del dominio del problema y de los mismos atributos, de este modo se obtiene un visión general para luego llegar a una específica.

A continuación, se describe en detalle cada una de las actividades mencionadas.

#### **4.3.2.1.1 DFs Establecidas en Dominio del Problema**

Actividad que debe ser ejecutada por los integrantes del equipo de trabajo con un fuerte apoyo de expertos y fuentes de información. El ADI debe ser analizado con el objeto de obtener manualmente DFs propias del dominio general del problema, es decir, dependencias del ámbito del problema

asociadas a la entidad, que son conocidas por los integrantes del equipo de trabajo y los expertos, es decir, son preexistentes. Concretamente, son DFs que se pueden encontrar en otros sistemas de información relacionados con el dominio del problema, en consecuencia, son genéricas y no necesariamente se encuentren en la realidad estudiada. Por ejemplo, una DF propia del dominio en el sector salud indica que el padre de un paciente determina al mismo (PADRE->RUT), es decir, a partir de los datos propios del padre, es posible determinar la identidad del paciente, sin embargo, esta situación podría no satisfacerse en los datos, producto de que en una entidad sólo se almacene como referencia el nombre el padre, de este modo no es único, además que el mismo (padre) podría tener más de un hijo.

La identificación de DFs preexistentes sustenta su base en el conocimiento específico de los expertos sobre la estructura de las relaciones y el dominio de sus datos, sumado a las fuentes de información obtenidas previamente. Todo lo anterior permitirá identificar claramente el dominio de la entidad, es decir, qué datos debe albergar en la entidad. Este punto cobra especial importancia debido a que el diseño inicial y el objetivo con que fueron creadas las entidades puede distar mucho de lo que se pueda encontrar al analizarlas luego de muchos años de funcionamiento, particularmente se presenta este problema cuando no están bien controladas las restricciones del modelo en la estructura de datos ni en las interfaces de usuario que permiten el registro de información.

La identificación de DFs se debe llevar a cabo estableciendo una exploración exhaustiva de la estructura de la entidad, esto es, revisar el nombre de ésta y los datos que contiene, procurando identificarlas de acuerdo al conocimiento específico de los expertos quienes deberán apoyarse en las fuentes de información propias del SIH idealmente, con la documentación asociada a la fuente de datos como diccionario de datos, descripción de estructura y los datos de la fuente de datos original del SIH.

La exploración consiste esencialmente en la búsqueda del dominio de la entidad, es decir, qué datos son los que debieran estar registrados en ella. Particularmente, se debe indicar una breve descripción de los atributos tendiente a establecer cuál es el conjunto de posibles valores en los datos.

Las DFs del dominio identificadas en esta actividad deben ser registradas tal como se muestra en la Tabla 4.7, en la que se debe consignar el nombre de la entidad, el o los atributo(s) determinante(s) y el atributo determinado. De este modo, se facilitará la comprobación posterior del nivel de satisfacción de éstas en el ADI.

<u>Entidad</u>	<u>Determinante</u>	->	<u>Determinado</u>
Paciente	RUT	->	Todos los atributos con datos personales del paciente

**Tabla 4.7:** Tabla de Registro de DFs.

Los datos registrados en esta tabla serán utilizados posteriormente con el fin de comprobar que éstas se satisfagan en la obtención automática de DFs y en la etapa de obtención del modelo de datos. Estos datos son de suma relevancia puesto que contienen intrínseco el criterio técnico de los expertos, el que deberá preponderar en las siguientes etapas.

#### **4.3.2.1.2 DFs Establecidas en los Atributos**

Esta actividad complementa los resultados obtenidos anteriormente, agregando datos más específicos sobre las entidades, a nivel de atributos. Con dicha información existe una mayor familiaridad con cada una de las entidades, puesto que ya existen lineamientos generales sobre el dominio de la entidad, esto facilitará la obtención de DFs de los atributos, sin embargo, es necesario hacer un estudio más profundo sobre cada uno de los atributos de las entidades albergadas en el ADI.

La identificación manual de DFs propias de los atributos se debe realizar sobre aquellos que posean un cierto grado de obviedad semántica, es decir, atributos sobre los cuales se puede establecer fehacientemente que forman parte de una DF. Para esto, se debe recurrir comúnmente a la ayuda de expertos con el afán de utilizar su conocimiento sobre el negocio o dominio del problema. Adicionalmente, se podrán utilizar las fuentes de información del SIH que aporten datos relevantes sobre la identificación.

En ocasiones, los atributos de las entidades se encuentran relacionados entre sí de manera más específica que la de pertenecer a una misma relación. A veces es posible determinar que un atributo depende de otro gracias al conocimiento específico que poseen los propios expertos. Por ejemplo, en el contexto de la salud, en una relación en que se registren los datos personales de pacientes es evidente que el atributo de grupo sanguíneo no determinará el nombre ni la dirección del paciente, no obstante el RUT, sí determina el grupo sanguíneo, nombre, apellidos y dirección.

Las fuentes de información juegan un rol importante, debido a que, entre otros, aportan la descripción de los atributos de la fuente de datos original del SIH, es decir, metadatos, por ejemplo los archivos DDL en general poseen comentarios en su interior que evidencian la naturaleza de los datos, o en el caso de archivos, los ficheros FD describen la estructura de los mismos. Estos documentos descriptivos deben ser analizados detalladamente, según se indica en la Tabla 4.7, por el equipo de trabajo en conjunto con expertos. El análisis de las fuentes de datos descriptivas de entidades se debe realizar poniendo especial cuidado en la información contenida en los documentos, puesto que puede estar desactualizada, de esta manera, la comparación con la realidad encontrada en el ADI es fundamental. Sin embargo, y tal como ya se ha mencionado el dominio original de los atributos, puede

distar de la realidad encontrada en los datos, de este modo el criterio técnico de los expertos debe primar por sobre la información emanada de los datos actuales.

Por ejemplo, si tenemos un documento descriptivo de una entidad que posea los datos personales de pacientes, en éste se podría indicar si el paciente es institucional, es decir, funcionario de la institución, automáticamente tiene asociada una forma de pago predeterminada. Esto se debe analizar realizando una consulta SQL sobre la entidad que alberga los datos de pacientes para realizar el análisis, y asegurarnos de que el atributo “forma de pago” dependa funcionalmente del atributo “tipo de paciente”.

La comprobación de una DFs preexistente se debe llevar a cabo mediante el establecimiento de consultas SQL en el ADI. Dichas consultas deben ser elaboradas con el objeto de comprobar que una determinada DF obtenida desde los expertos se satisfaga en los datos. Concretamente, la consulta debe revisar que el atributo del lado derecho dependa siempre funcionalmente del atributo establecido en el lado izquierdo de la DF. Por ejemplo, si en una entidad que alberga datos inherentes a pacientes se estableció por los expertos que la Historia Clínica (HC) depende funcionalmente del RUT del paciente, esto se debe comprobar mediante las siguientes consultas: (1) “select count(distinct RUT,HC) from paciente” y (2) “select count (distinct HC) from paciente”. Si ambos resultados son iguales entonces la DF RUT-> HC se verifica.

Luego, tanto las DFs comprobadas y las que no, se registran en una tabla con el objeto de utilizar dicha información posteriormente en las etapas de Obtención y Generación. La Tabla 4.8 muestra los datos de ejemplo que deben ser consignados; el código de identificación de la DF, que en general será un código auto numérico antecedido de la sigla “DFP” (Dependencia Funcional Preexistente), el nombre de la entidad, atributo determinante, atributo determinado y el estado, que indica si se encuentra satisfecha en los datos.

<b>DF</b>	<b>Entidad</b>	<b>Determinante</b>	<b>-&gt;</b>	<b>Determinado</b>	<b>Estado</b>
DFP1	Paciente	RUT	->	Nombre	No Satisfecha
DFP2	Paciente	RUT	->	DV	Satisfecha

**Tabla 4.8:** Tabla Ejemplo de Registro de DFs establecidas en los Atributos.

En conclusión, la obtención de DFs preexistentes contribuirá información que facilitará a la generación de un modelo de datos más adecuado y conforme a la realidad del dominio del problema.

#### 4.3.2.1.3 Limpieza

Las DFs preexistentes obtenidas en la actividad anterior deben ser satisfechas completamente en los datos contenidos en el ADI, por lo tanto, se deben realizar acciones sobre los datos en aquellas DFs que posean el estado de no satisfechas. Dichas acciones tienen relación con la ejecución de instrucciones SQL en el propio ADI, con el objeto de modificar o agregar datos en las tuplas. Esto es, forzar que las DFs se cumplan en los datos mediante modificaciones o inserciones. Estas acciones conllevan al ingreso de datos al ADI de manera conveniente para dar cumplimiento a las DFs.

Es un hecho común que producto de los años de funcionamiento en un SIH se encuentren fallas en la validación de datos en interfaces de usuario, o por falta de mantención en los datos originales, en consecuencia, es muy usual hallar inconsistencias entre las DFs preexistentes y los datos contenidos en el ADI. Como consecuencia, la estructura de las entidades y los datos contenidos en éstas se podrían ver afectados producto de la poda de atributos y/o eliminación o modificación en los valores de las tuplas. Por esta razón, la aplicación del proceso de limpieza desencadena principalmente tres acciones que deben ser ejecutadas en estricto orden secuencial: satisfacción de DFs preexistentes, respaldo de entidades, y, finalmente, eliminación de atributos inocuos. A continuación, se describen cada una de estas acciones.

- **Satisfacción de DFs Preexistentes:** Dicha comprobación se debe llevar a cabo mediante el establecimiento de consultas SQL en el propio ADI. Debido a errores en los datos, existe la posibilidad de que las DFs preexistentes no se satisfagan. A partir de esta realidad, debe primar el criterio técnico de los expertos utilizado para obtener las DFs preexistentes y, en consecuencia, se deben realizar las operaciones necesarias para hacer que éstas se satisfagan, esto podría requerir que, eventualmente, se deban eliminar tuplas, o realizar modificaciones en los datos. Tanto la eliminación como modificaciones se deben llevar a cabo mediante el uso de comandos propios de la herramienta utilizada como ADI, tales como update y delete, acompañado de estructuras de control tales como cursores, que facilitan el recorrido y procesamiento de registros del resultado de una consulta.

La Tabla 4.9 muestra un ejemplo del uso de un cursor que actualiza en forma automática los atributos que se encuentran nulos. Este procedimiento se debe ejecutar con el objeto de subsanar las inconsistencias encontradas. Las instrucciones que realizan limpieza en los datos podrían satisfacer a más de una comprobación. Tal como se muestra en Tabla 4.10, en que se identifica la DFP no satisfecha y la acción a realizar para hacer que esta se cumpla.

<u>DFP</u>	<u>Limpieza</u>
DFP1	<pre> DECLARE p_cp SAPAT001.codpacie%TYPE; p_rut SAPAT001.rut%TYPE; CURSOR c_sapat001 IS SELECT codpacie FROM SAPAT001 WHERE rut is null; BEGIN select max(rut) into p_rut from SAPAT001; OPEN c_sapat001; LOOP   FETCH c_sapat001 INTO p_cp;   p_rut:= p_rut+1;   UPDATE SAPAT001 SET RUT= p_rut WHERE codpacie=p_cp;   EXIT WHEN c_sapat001%NOTFOUND; END LOOP; CLOSE c_sapat001;END; </pre>

**Tabla 4.9:** Tabla de Registro de inconsistencias.

- **Respaldo de Entidades:** Luego de haber realizado las acciones sobre el ADI tendientes a satisfacer las DFs en los datos, se debe realizar un respaldo de las entidades involucradas, debido a que luego de esta actividad algunos atributos serán podados con el objeto de mejorar el rendimiento posterior de la herramienta que obtendrá las DFs.
- **Eliminación de atributos Inocuos:** Posteriormente, se ejecuta la eliminación de todos los atributos determinados que pertenecen a una DFP, pero además, existe la certeza que de acuerdo a las DFs del dominio no son capaces de determinar a ningún otro atributo, en concreto, la eliminación de la relación no causaría ningún daño, por ejemplo el dígito verificador es claramente una DF del RUT, y además, no determina a ningún atributo, por lo que puede ser eliminado. Los tipos de atributos que podrían ser eliminados debido a su inocuidad son los que se describen a continuación:
  - *Atributos Propios del Registro (Fecha):* Los atributos que generalmente registran la fecha y hora en que los datos de una tupla, fueron agregados. No tienen mayor relevancia (dependiendo del dominio de la relación), puesto que no determinan a ningún otro atributo.
  - *Atributos Observación:* Son atributos que generalmente albergan una cantidad heterogénea de caracteres y éstos no responden a un patrón común, es decir, son descriptivos de la propia tupla, pero ambiguos, puesto que los usuarios pueden escribir libremente de acuerdo a su forma de redacción.
  - *Atributos Derivados:* Son aquellos atributos que existen como producto o consecuencia de alguna fórmula aplicada a otros atributos. Inclusive podrían ser catalogados como redundantes, por ejemplo, la edad actual es un atributo derivado de la diferencia entre la fecha actual y la fecha de nacimiento.

La poda de un atributo es una acción que afecta directamente a las entidades y se fundamenta en la generación de mejores resultados en la obtención de DFs en un tiempo razonable. Sin embargo, es absolutamente necesario identificar claramente qué atributos fueron eliminados para que, previo a la generación del modelo de datos, sean nuevamente reincorporados en la entidad.

Previo a ejecutar la eliminación de los atributos, se deben registrar el nombre, entidad, clasificación y justificación de la eliminación conforme a los criterios planteados previamente, tal como muestran los ejemplos en la Tabla 4.10.

<b>Id</b>	<b>Atributo</b>	<b>Entidad</b>	<b>Clasificación</b>	<b>Justificación</b>
1	FECHACREA	SAPAT001	Atributo propio del Registro	Atributo asociado a la fecha y hora en que se realizó el registro de la tupla.
2	ESTADO	SAPAT001	Atributo propio del Registro	Atributo que indica el estado de la tupla
3	DOMICILIO	SAPAT001	Atributo Observación	Atributo en que se describe el lugar donde habita el paciente, es un campo abierto.

**Tabla 4.10:** Ejemplo de atributos eliminados.

Luego de la identificación de los atributos que deben ser podados en las entidades indicadas en la tabla anterior, se ejecuta la eliminación de tales atributos por medio de instrucciones SQL.

Estas acciones responden a la experiencia adquirida durante ésta investigación, puesto que se detectó que al obtener DFs automáticamente desde las relaciones es muy recurrente encontrar que los datos estén incorrectos y esto genera una cantidad excesiva de DFs que sólo son válidas a nivel de datos, pero no representan la realidad del dominio.

Como resultado de la aplicación de esta actividad, se obtienen entidades con menos atributos o tuplas, pero que responde al dominio establecido por los expertos. De esta manera se podrán obtener resultados de mejor calidad al momento de identificar DFs por medio de algoritmos ad-hoc.

#### **4.3.2.2 Obtención automática de DFs**

La aplicación de las actividades predecesoras permiten conseguir entidades acotadas y consensadas conforme al criterio técnico de los expertos, lo que contribuye a disminuir el tiempo de procesamiento y mejorar el rendimiento de los algoritmos ad-hoc específicos que permiten la obtención automática de DFs.

La obtención automática de DFs técnicamente realiza un procesamiento de los datos contenidos en el ADI mediante algoritmos especializados que infieren DFs a partir de un análisis de las tuplas, averiguando la dependencia entre los atributos que componen la relación.

En general, al conocer las DFs mediante herramientas ad-hoc se evidencian problemas de redundancia e integridad existente en los datos, tal como se pudo establecer en la actividad de limpieza para las DFs preexistentes. Posteriormente, las DFs obtenidas deben ser analizadas detalladamente con el objetivo de descartar, confirmar o añadir otras. Por otro lado, las herramientas proveen un ambiente sencillo y de fácil manejo y, a pesar de ser un proceso automático, requiere de la intervención de los integrantes del equipo de trabajo puesto que es necesario configurar ciertos parámetros que permitan establecer la comunicación con el ADI.

#### 4.3.2.3.1 Procesamiento Automático

Inicialmente, se debe establecer la comunicación entre la herramienta ad-hoc y el ADI, para lo cual la propia herramienta debe proveer los métodos de acceso a datos mediante algún protocolo conocido. De lo contrario, se deberá recurrir a otros protocolos de acceso a datos compatibles con la herramienta de obtención de DFs.

Posteriormente se deben seleccionar las entidades que serán evaluadas individualmente en la búsqueda de DFs. Este punto es importante, puesto que la evaluación se realizará en forma individual, debido principalmente a la alta carga de trabajo que esta acción demanda.

Para concluir, se ejecuta el proceso de obtención automático por medio de la herramienta que explora las entidades del ADI en busca de DFs. Luego, sólo resta exportar los resultados encontrados hacia un archivo en algún formato conocido, comúnmente archivo plano o XML. El documento electrónico con los resultados obtenidos es el insumo necesario para continuar con la etapa en que se generará el modelo de datos. Dichos datos se deben registrar en una tabla indicando un número secuencial que identifica la DF, atributo determinante y atributo determinado, tal como se muestra en la Tabla 4.11.

N°	Determinante	->	Determinado	N°	Determinante	->	Determinado
1	APELLMAT	->	FONO_INSTITUCION	3	DCOMUNA, DFORMAPAGO, HC, SEXO	->	FECHAFALL
2	APELLMAT	->	INSTITUCION	4	DCOMUNA, DOCUPACION, FACTORRH	->	GRUPSANG

**Tabla 4.11:** Ejemplo de registro numerado de DF.

La identificación de las DFs permite individualizar las mismas de un modo sencillo en las actividades siguientes.

#### **4.3.2.3.2 Depuración de DFs**

Es la última actividad de la etapa de obtención de DFs, y su finalidad es realizar un refinamiento de los resultados de la actividad anterior, esto es, explorar los documentos electrónicos que contienen las DFs obtenidas automáticamente, con el objeto de corroborar que éstas sean satisfechas en el ámbito del problema y no solamente en los datos. Adicionalmente, se deben eliminar DFs según el criterio de los expertos y agregar las DFs de los atributos inocuos que fueron eliminados anteriormente, esto significa que el archivo electrónico debe ser editado, manualmente o por medio de una herramienta ad-hoc, con el fin de agregar dicha información.

La actividad comienza con la evaluación de las DFs obtenidas automáticamente, esto es, se evalúan las DFs en función de la cantidad obtenida para cada entidad. Es común que se obtenga una gran cantidad de DFs en la medida que no se haya realizado una prolija eliminación de atributos inocuos. De este modo, se encontrará un volumen excesivo de DFs, lo que puede provocar una inviabilidad en la depuración producto de la gran cantidad. La evaluación por cantidad de DFs es exclusivamente técnica y de conocimiento experto, de modo que no existe una regla general para determinar si la cantidad obtenida es excesiva o no, sin embargo, cuando estas son varios cientos o miles, es necesario realizar una revisión rápida de las DFs poniendo especial cuidado en aquellas que existe certeza de su existencia y en otras que, con seguridad, no se deben cumplir en el ámbito del problema.

Luego de la evaluación visual de las DFs, se debe decidir si es necesario volver a la actividad de obtención de DFs preexistentes o sólo realizar la poda de las DFs manualmente de conformidad al criterio de los expertos. La decisión es puramente técnica y tiene relación con el tiempo que demanda la depuración de las DFs. Si este tiempo es excesivo es conveniente volver a la obtención de DFs preexistentes y eliminar algunos atributos inocuos.

La poda de DFs consta del registro y eliminación de éstas, que según el criterio de los expertos y equipo de trabajo no corresponden al dominio de la entidad. Regularmente, estas DFs son evaluadas aplicando meramente conocimiento técnico apoyado en las fuentes de información.

La exclusión de una DF es una actividad que debe ser registrada en una tabla indicando de manera clara el número de la DF y el argumento por el cual ha sido eliminada. Típicamente los argumentos de los expertos responden a casos comunes como:

- *Atributo Indeterminante*: Son DFs en que el atributo determinante no posee la capacidad de determinar a otro atributo, por ejemplo: el nombre de un paciente no es suficiente para determinar su apellido ni ningún otro atributo.
- *Determinante erróneo*: Son DFs que no se cumplen, debido a que el atributo determinante no determina al atributo determinado, por ejemplo: el código de la comuna no determina al nombre de un paciente.
- *Conjunto Indeterminante*: Son DFs que no se cumplen, debido a que el conjunto de atributos determinantes no son capaces de determinar al atributo determinado, es decir, la combinación de atributos del lado izquierdo no determina al atributo del lado derecho.

La Tabla 4.12 muestra un ejemplo del formato en que se registran las dependencias funcionales y el argumento de eliminación.

<u>Argumento</u>	<u>Dependencias Funcionales</u>
Atributo no determina a ningún otro atributo	1,2,3,4,44,45,46,47,48,155,156,157,158,164,165
Atributo no determina al determinado	101,102,103,104,116,117,118,119,130,132,133.

**Tabla 4.12:** Ejemplo del registro de DFs eliminadas.

A continuación, es necesario recuperar las entidades originales, es decir, se deben renombrar las entidades que previamente fueron respaldadas en la actividad de limpieza del punto 4.3.2.1.3. Esta acción permite recuperar los atributos podados previamente. Conjuntamente, se deben agregar las DFs que incluyen a los atributos que fueron podados previamente. Por ejemplo, si el atributo TELEFONO fue considerado inocuo y, por consecuencia eliminado, se deben agregar las DFs que lo incluyan (RUT->TELEFONO), de este modo figurará en el nuevo modelo de datos, de lo contrario quedará excluído.

Para agregar datos al archivo que contiene las DFs depuradas, se debe colocar especial cuidado en mantener la estructura jerárquica original del documento. En caso de documentos XML, se deben conservar las etiquetas que componen el documento manteniendo especial cuidado en la sintaxis del mismo, esto es, asegurarse que una etiqueta debe estar correctamente incluida en otra, es decir, deben estar correctamente anidadas, tal como se muestra en el ejemplo de la Figura 4.6.

```

<fd>
  <lhs> <attribute>RUT</attribute> </lhs>
  <rhs> <attribute>CODCOMUNA</attribute> </rhs>
</fd>

```

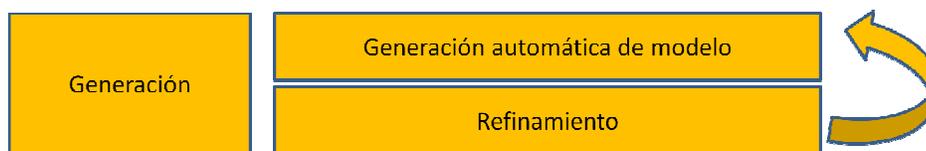
**Figura 4.6:** Ejemplo de estructura de archivos XML con DFs.

En último lugar, la etapa de extracción concluye, aportando principalmente dos elementos:

- DFs Preexistentes y DFs obtenidas automáticamente en un documento en formato electrónico que será el insumo necesario para continuar con la etapa de obtención del modelo de datos.
- DFs propias del dominio y atributos de las entidades en una tabla que facilitan su verificación en la etapa de generación, debido a que poseen el criterio técnico de los expertos.

### 4.3.3 Etapa de Generación

Corresponde a la última etapa del Marco de Trabajo, y tiene por finalidad la generación semiautomática del modelo de datos a partir de las DFs obtenidas en la etapa anterior. Su ejecución conlleva el cumplimiento de dos actividades, la primera de ellas es la generación automática del modelo de datos preliminar a partir de DFs por medio de herramientas ad-hoc. En la segunda, se realiza un refinamiento del modelo de datos inicial basado en los conocimientos específicos de expertos y fuentes de información. En orden a clarificar y tal como se muestra en la Figura 4.7, esta etapa puede funcionar iterativamente, es decir, la obtención del modelo de datos no necesariamente podrá satisfacer el criterio de los expertos y en tal circunstancia se deberán realizar los ajustes necesarios en esta etapa, o, eventualmente, en las etapas anteriores. El análisis de los expertos es relevante, debido a que se revisan los resultados de la aplicación del marco de trabajo. Si bien EOG posee etapas bien definidas que deben ser ejecutadas en estricto orden secuencial, esto no asegura que los resultados obtenidos satisfagan el criterio experto, debido a esto no es posible evaluar el modelo de datos resultante sino hasta la aplicación de la generación automática del modelo.



**Figura 4.7:** Esquema de funcionamiento Etapa Generación Modelo.

A pesar de ser un proceso altamente automatizado, requiere de la constante intervención de expertos que manejen las herramientas. A continuación, se describen detalladamente cada una de las actividades que componen esta etapa.

#### **4.3.3.3 Generación automática del modelo:**

Actividad que facilita la generación automática de un modelo de datos relacional en 3FN, mediante el uso de herramientas ad-hoc a partir de DFs contenidas en documentos electrónicos en formatos conocidos, como XML o archivos planos, en consecuencia requiere de la intervención de usuarios que guíen el proceso, manejando la aplicación, indicando inicialmente la ubicación de los documentos electrónicos y, posteriormente, verifiquen que todas las DFs se satisfagan de acuerdo al criterio técnico establecido por los expertos en la etapa anterior.

En general, las herramientas ad-hoc facilitan la edición de las DFs y, de este modo, se chequean previamente que éstas se satisfagan en la fuente de datos de manera automática.

El proceso comienza con la identificación de la fuente de información que proveerá las DFs, por lo que se debe proporcionar a la herramienta ad-hoc la ubicación del documento electrónico y, posteriormente, indicar los datos de configuración básica del ADI. Comúnmente, las propias herramientas proveen de protocolos de acceso a datos.

A continuación, se deben verificar las DFs, es decir, chequear que estas concuerden con los criterios técnicos establecidos en la obtención de DFs preexistentes. Consecuentemente, se podrán eliminar aquellas que no se satisfacen o agregar otras que pudieron haber sido olvidadas. Generalmente, estas acciones pueden estar debidamente corroboradas por algunas herramientas ad-hoc (que serán detalladas en el capítulo 5), debido a que tienen acceso a los datos del ADI. Por lo tanto, comprueba que cada DF indicada se satisfaga conforme a los datos contenidos en la fuente de datos.

Otro aspecto relevante es la generación de claves candidatas y clave principal, en este sentido es la propia herramienta la que provee de una serie de claves candidatas, de las cuales se debe seleccionar sólo una para convertirse en clave primaria. Tanto las claves candidatas como la clave principal son inferidas de acuerdo las DFs y la fuente de datos. Al proponer más de una clave candidata debe primar el criterio de los expertos para su elección.

Los resultados obtenidos en esta actividad tiene el carácter de preliminar, puesto que comúnmente se requiere de la intervención de usuarios para agregar DFs o eliminar aquellas que no contribuyan a la generación de un modelo que se condiga con las DFs propias del dominio.

#### **4.3.3.4 Refinamiento**

Actividad final que tiene por objeto la revisión del modelo de datos preliminar obtenido en la actividad anterior. El proceso de obtención de DFs es complejo desde el punto de vista de la

computación de datos, en consecuencia, requiere de una revisión y refinamiento de los resultados obtenidos. La revisión del modelo se debe llevar a cabo minuciosamente procurando corroborar la coherencia entre las DFs propias del dominio, atributos y las nuevas entidades propuestas.

El refinamiento comienza con la revisión de las entidades y atributos propuestos por la herramienta. La revisión consiste principalmente en evaluar si las DFs del dominio y atributos son consecuentes con el modelo propuesto, dicha revisión se debe llevar a cabo realizando una comparación entre las DFs preexistentes registradas en la Tabla 4.10 y la proposición de la aplicación, tal como se muestra en la misma figura.

En orden a explicar el procedimiento, por ejemplo en caso de que se haya definido una DF del dominio relacionada con la ciudad donde reside el paciente, esto es RUT->CIUDAD, dicha DF debe mantenerse en el modelo propuesto, si esto no sucede se puede deber a que no fue definida o bien otra DF ha provocado que no figure en la misma relación que el RUT del paciente.

A continuación, se ejecuta nuevamente el procesamiento de las DFs en búsqueda de un nuevo modelo, el que es evaluado, y dependiendo del grado de satisfacción que éste posea con las DFs preexistentes, se podría efectuar de modo iterativo la etapa de obtención de DFs. La decisión de volver a ejecutar la etapa recae en los expertos, y tiene relación con las inconsistencias del modelo, el que evidencia las fallas en la etapa de obtención de DFs.

Luego que el modelo propuesto satisfaga el criterio técnico, es factible exportar dicho modelo mediante un script DDL con la estructura y datos, tal como se muestra en la Figura 4.8. Este script podría ser exportado directamente a algún SGBD.

```
CREATE TABLE HR.EMPLEA2_1 (
    CODCOMUNA NUMBER(10) PRIMARY KEY, COMUNA NVARCHAR2(510));

CREATE TABLE HR.EMPLEA2_2 (
    CODDPTO NUMBER(10) PRIMARY KEY, DPTO NVARCHAR2(510), UNIQUE (DPTO));

-- data migration (copy table contents)

INSERT INTO HR.EMPLEA2_1 (CODCOMUNA, COMUNA ) SELECT DISTINCT CODCOMUNA, COMUNA FROM HR.EMPLEA2;

INSERT INTO HR.EMPLEA2_2 (CODDPTO, DPTO ) SELECT DISTINCT CODDPTO, DPTO FROM HR.EMPLEA2;

-- drop old Table (with constraints)

DROP TABLE HR.EMPLEA2 CASCADE CONSTRAINTS;

ALTER TABLE HR.EMPLEA2_3 ADD FOREIGN KEY (CODCOMUNA) REFERENCES HR.EMPLEA2_1 (CODCOMUNA);
```

**Figura 4.8:** Modelo relacional propuesto.

#### **4.4 Conclusiones del Capítulo**

Durante este capítulo se ha revisado detalladamente las etapas y actividades de EOG, las que han sido fruto de las necesidades que fueron detectadas durante el desarrollo de esta tesis. Por otro lado, es relevante destacar que la participación de expertos y equipo de trabajo facilitan en gran medida los resultados obtenidos en las etapas de Obtención y Generación, sin embargo, es necesario considerar que ante la ausencia de expertos al interior de la organización, es necesario recurrir a usuarios con experiencia o expertos externos que apoyados en las fuentes de información aportarán conocimiento al proceso de obtención de DFs y la depuración de las mismas.

Del mismo modo, las herramientas ad-hoc que intervienen en cada etapa son un apoyo esencial para el proceso de generación del modelo de datos relacional, y no contar con ellas haría imposible la aplicación de EOG. Por este motivo, el Capítulo 5 aborda con detención las herramientas disponibles para ser utilizadas en las etapas de Extracción, Obtención y Generación del marco de trabajo.

## **Capítulo 5**

### **Herramientas para EOG**

---

La aplicación del marco de trabajo EOG conlleva el uso de algoritmos y herramientas pertinentes, que sean capaces de dar solución a problemáticas específicas, por lo tanto, su uso no es generalizable para otros propósitos. La finalidad de este capítulo es proveer al lector de información técnica suficiente que facilite la elección de herramientas que permitirán la ejecución del marco de trabajo.

EOG propone un conjunto estandarizado de conceptos, prácticas y criterios que sirven como referencia para obtener un modelo de datos relacional en 3FN a partir de un SIH. De este modo, esta estructura conceptual y tecnológica plantea específicamente el uso de herramientas ad-hoc para cada una de las etapas del marco de trabajo.

Específicamente, en la actividad de Extracción se requiere el uso de un ADI que albergará los datos extraídos desde la fuente de datos original del SIH mediante el uso de herramientas ETL. Además, en la etapa de Obtención es necesario el uso de herramientas que faciliten la extracción de DFs a partir de datos contenidos en la ADI. Finalmente, en la etapa de Generación se requiere el uso de herramientas que permitan la obtención del modelo de datos relacional a partir de DFs. Las herramientas propuestas para aplicar en el marco de trabajo son detalladas a partir del punto 5.1 y se han organizado por etapas.

## **5.1 Etapa de Extracción**

Etapa en que cobra especial relevancia el ADI, puesto que actúa como un repositorio de datos en el que se podrán realizar transformaciones. En consecuencia, se debe seleccionar una plataforma de datos que sea capaz de soportar los datos y las transacciones que permitirán realizar modificaciones sobre éste.

En general, el almacén de datos podrá ser cualquier producto de base de datos propietaria de alguna marca en particular, típicamente un SGBD, no obstante, se debe corroborar que la herramienta ETL posea los protocolos de acceso a datos compatibles con el ADI.

Concretamente, se debe seleccionar la aplicación que funcionará como ADI y la herramienta ETL, que en su conjunto ayudarán a extraer la estructura de datos desde la fuente de datos original de un SIH. A continuación, se especifican las aplicaciones propuestas en esta investigación.

### **5.1.1 Sistemas de Gestión de Base de Datos**

Los SGBD son aplicaciones que facilitan la definición, creación y mantención de bases de datos, además de proporcionar acceso controlado a las mismas. En efecto, es una herramienta ideal para ser usada como ADI, puesto que el manejo de datos es natural y sencillo. Algunos de los SGBD más comunes que podrían ser utilizados en esta etapa son los que se describen a continuación.

#### **5.1.1.1 MySQL**

Es uno de los SGBD relacional más comunes, sobre todo en aplicaciones web, es compatible con plataformas Windows y Unix. Es un producto de fácil dominio puesto es muy utilizado en el ambiente académico y estudiantil. Su principal característica es la velocidad de respuesta al realizar consultas, sin embargo, en sistemas de alta concurrencia su velocidad se ve disminuida al realizar modificaciones en los registros. Además, posee un amplio subconjunto de ANSI SQL 99, permite el uso de procedimientos almacenados, triggers, vistas actualizables, sub consultas, entre otros.

Habitualmente, se puede utilizar MySQL en conjunto con otras herramientas que facilitan su administración. Tal es el caso de phpMyAdmin, que es una herramienta de uso libre que permite manejar la administración de la base de datos a través de un ambiente web.

#### **5.1.1.2 Oracle 10g Express Edition (EE)**

Es una edición limitada del ORDBMS (Sistema de Gestión de Base de Datos Objeto Relacional) Oracle. Permite desarrollar e implementar aplicaciones de modo inicial, está orientada principalmente, a desarrolladores de código abierto, administradores de base de datos que requieran entrenamiento inicial, instituciones educativas, entre otros. Oracle EE se puede instalar en cualquier equipo, independiente de la cantidad de CPUs, pero se utilizará sólo una, además almacenará sólo hasta 4GB de datos, usando un máximo de 1GB de memoria. Sin embargo, y a pesar de las limitaciones, se transforma en un excelente aliado, puesto que presenta una amplia compatibilidad con Tane-Java (RAFREIDER 2008), una de las herramientas que facilita la obtención de DFs.

Oracle EE provee de una interfaz de usuario navegable intuitiva que permite administrar la base de datos, creación de tablas, vistas y otros objetos de la base de datos, importación, exportación y visualización de datos de las tablas, ejecución de consultas y script SQL y generación de informes.

## **5.1.2 Herramientas ETL**

Las herramientas ETL típicamente permiten la extracción desde una fuente de datos hacia una fuente de destino, proporcionando de manera general control de la extracción de los datos y la automatización del proceso, además del acceso a fuentes de datos heterogéneas y manejo de excepciones. Comúnmente, se utilizan para integrar SIH hacia nuevas plataformas. En el mercado existe una gran variedad de estas herramientas, en distintas modalidades, por esta razón, se presentan a continuación algunas herramientas que podrían ser utilizadas durante esta etapa.

### **5.1.2.1 Microsoft DTS**

Facilita la extracción, transformación y carga de datos heterogéneos con OLE DB (Object Linking and Embedding for Databases.), Open Database Connectivity (ODBC), o archivos de texto a cualquier base de datos OLE DB soportada o almacenamiento multi-dimensional. También, automatiza la transformación de datos permitiendo a los usuarios importar o transformar datos automáticamente sobre una base regular programada, además, posee control de flujo y errores de importación y exportación.

### **5.1.2.2 MySQL Migration Toolkit**

Es un conjunto de herramientas gráficas que permiten migrar esquemas y datos desde bases de datos relacionales heterogéneas (Oracle, Microsoft SQL Server, Informix, etc.) hacia MySQL 5.0 (o posterior) por medio de asistentes que proveen un ambiente sencillo y rápido para operar.

Las herramientas descritas permiten realizar, sin dificultades, la extracción, sin embargo MySQL Migration Toolkit sólo permite extraer hacia un tipo de base de datos. No obstante, existen otras alternativas que permiten extraer hacia cualquier destino, tal es el caso de DTS de Microsoft.

## **5.2 Etapa de Obtención**

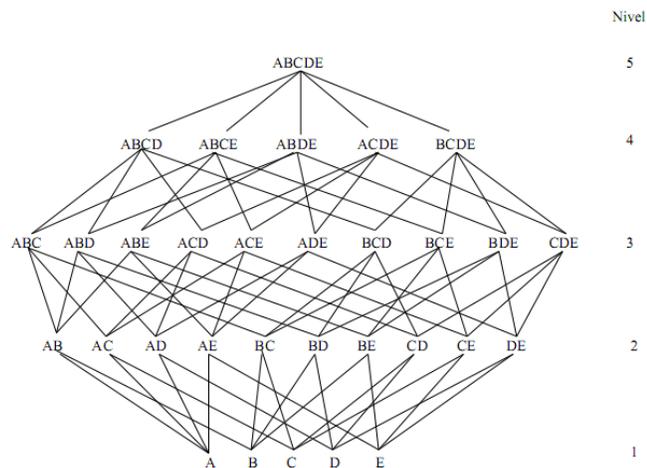
La etapa de obtención tiene por propósito la extracción de DFs a partir de un esquema de base de datos albergado en un ADI. En consecuencia, la herramienta que se utilice para este fin debe poseer compatibilidad con el ADI, esto es, que desde la propia herramienta se pueda acceder de una manera sencilla a los datos. Esto no es trivial si consideramos que algunas herramientas poseen compatibilidad predeterminada con algunas fuentes de datos. A continuación, se describen las herramientas que favorecen la obtención de DFs.

## 5.2.1 Herramientas para obtención de DFs

En la literatura es posible encontrar algunas investigaciones que proponen herramientas y algoritmos que facilitan la obtención de DFs infiriendo desde los datos de la fuente de origen. Sin embargo, no todas poseen interfaces de usuario amigables y sencillas de utilizar, de manera que el grado de dificultad en esta etapa se encuentra supeditado a las propias herramientas o algoritmos. Por un lado, existe un algoritmo que no posee mayor desarrollo de interfaces gráficas de usuario, pero que ha sentado las bases suficientes para la creación de otras herramientas. A continuación, se especifican dos herramientas que ayudan a conseguir DFs a partir de los datos.

### 5.2.1.1 TANE

El algoritmo TANE (HUHTALA, KÄRKKÄINEN et al. 1999) permite encontrar las DFs a partir de grandes bases de datos, y se basa en la búsqueda de DFs por niveles dentro del retículo booleano, obtención de DFs aproximadas de acuerdo a un factor de error y la utilización de técnicas de pruning para optimizar la búsqueda, lo que hace que las pruebas de la validez de las dependencias funcionales sean rápidas, incluso para un gran número de tuplas. El uso de particiones también hace que el descubrimiento de DFs aproximadas sea rápido, y las filas erróneas o excepcionales se pueden identificar fácilmente. Sin embargo, su rendimiento es bajo para un gran número de atributos en una relación, esto debido a que el retículo booleano crece exponencialmente con la cantidad de atributos de la relación, tal como se muestra en el ejemplo de la Figura 5.1 (FUENTES, SÁEZ et al. 2011).



**Figura 5.1:** Posibles combinaciones de los atributos A, B, C, D y E

Si tenemos un conjunto de atributos  $R=\{A,B,C,D,E\}$ , el espacio de búsqueda de determinantes es exponencial al tamaño de  $R$ , ya que se deben formar  $2^n-1$  combinaciones, descartando el nivel 0 (en el ejemplo  $2^5-1 = 31$  posibles subconjuntos de atributos candidatos). En la Figura 5.1 se muestran todas las posibles combinaciones. Estas combinaciones forman un retículo booleano.

En definitiva, el algoritmo se aplica también a conjuntos de datos mucho mayores que métodos anteriores (MANNILA and RÄIHÄ 1986; BITTON, MILLMAN et al. 1989; SAVNIK and FLACH 1993; SCHLIMMER 1993; MANNILA and RÄIHÄ 1994; BELL and BROCKHAUSEN 1995).

### 5.2.1.2 TANE-Java

Tane-Java (RAFREIDER 2008), se basa en el algoritmo descrito en (HUHTALA, KÄRKKÄINEN et al. 1999), es una herramienta desarrollada bajo licencia GPLv2<sup>3</sup> con fines académicos posee una interfaz gráfica amigable e intuitiva, desarrollada con lenguaje Java, provee una conexión predeterminada con la base de datos Oracle 10g y no mediante archivos CSV como se indica en (HUHTALA, KÄRKKÄINEN et al. 1999). No obstante, la herramienta podría conectarse a otras base de datos realizando algunas modificaciones en los códigos fuentes.

Tane-Java facilita el descubrimiento de todas las DFs mínimas no triviales de las relaciones, ofreciendo algunas alternativas relacionadas con la optimización en el procesamiento, entre otras, permite la disminución del número de atributos a analizar, uso de variantes de los algoritmos, tales como uso de memoria o espacio en disco.

El algoritmo TANE basa su funcionamiento en la división del conjunto de tuplas con respecto a sus valores de atributos, lo que genera que las pruebas de validez de DFs se ejecuten más rápido, facilitando, de esta manera, la obtención en relaciones con una gran cantidad de filas. El uso de particiones permite también una mejor aproximación de DFs y una fácil identificación de filas erróneas. Los resultados se obtienen en archivos XML con una estructura que es comprendida de manera natural por la aplicación (FESENMEYER 2008) que permite la generación del modelo de datos.

## 5.3 Etapa de Generación

La última etapa del marco de trabajo requiere de una herramienta o algoritmo que tenga la capacidad de generar un modelo de datos a partir de DFs. Si bien esta no es una tarea sencilla de

---

<sup>3</sup> Licencia Pública General creada por Free Software Foundation orientada a proteger la libre modificación y uso de software libre.

realizar, existen al menos dos herramientas que facilitan la generación de un modelo de datos normalizado, las que se describen a continuación.

### **5.3.1 Herramientas Generadoras de Modelo**

Están disponibles dos herramientas que facilitan la generación de un modelo de datos requieren como insumo las DFs en formato compatible con la aplicación, alguna de las diferencias entre ambas herramientas es que la primera utiliza formato electrónico XML, mientras la otra archivos planos, además de una evidente diferencia en el modo de uso e interfaces de usuario. A continuación, se describen las dos herramientas que permiten realizar esta tarea.

#### **5.3.1.1 DBNormalizer**

Esta herramienta (FESENMEYER 2008) es resultado de una tesis de Maestría desarrollada con el objetivo de apoyar a los ingenieros en la mejora de los sistemas mediante el perfeccionamiento de las base de datos. La herramienta posee una IGU muy intuitiva y amigable que facilita la importación de esquemas de base de datos por medio de metadatos obtenidos como una instancia de la base de datos.

DBNormalizer (FESENMEYER 2008) permite la obtención de un modelo de datos a partir de DFs contenidas en documentos electrónicos XML que sólo hayan sido obtenidos mediante la herramienta Tane-Java (RAFREIDER 2008). Adicionalmente, admite la edición de las DF contenidas en el documento XML chequeando previamente que éstas se satisfagan en la fuente de datos. La herramienta, entre otros, calcula la cobertura mínima, determina la clausura de atributos para un conjunto determinado de atributos, calcula todas las claves candidatas, verifica las formas normales desde la 1FN hasta FNBC, genera una propuesta de normalización, garantizando al menos que las relaciones se encuentren en 3FN y en lo posible en forma normal Boyce Codd (FNBC). Concretamente, la normalización podría generar la creación de nuevas relaciones entregando como resultado un script con instrucciones DDL que permitirían la creación de tablas y su eventual intervención por parte de los expertos.

#### **5.3.1.2 Reno 3.0**

Reno (Relaciones Normalizadas) de (DE MIGUEL, PIATTINI et al. 2000) es una aplicación que permite utilizar algoritmos de manejo de DFs y de normalización (hasta FNBC) a esquemas relacionales. El objetivo primordial de Reno es la contribución al diseño de Bases de Datos relacionales. Su uso principal es realizar operaciones de normalización sobre un esquema de relación

del modelo relacional de manera automática. Una vez introducidos los datos del esquema se puede operar con los mismos. Las operaciones permitidas son:

- *Operaciones Básicas:* Cálculo de Claves, cierre de un descriptor y cálculo de recubrimiento minimal.
- *Test:* Forma normal actual (2FN, 3FN, FNBC)
- *Normalización:* Síntesis, análisis (hasta FNBC, test: Análisis sin pérdida de Dependencias, test: análisis sin pérdida de Información).

Estas operaciones pueden realizarse de secuencialmente o simultáneamente, según la preferencia del usuario. El resultado de estas operaciones puede guardarse en un archivo con extensión 'DOC', el cual puede editarse desde cualquier editor de texto. El procedimiento para ejecutar estas operaciones está basado en una serie de algoritmos que han sido clasificados en dos grandes grupos: Algoritmos tradicionales y los nuevos algoritmos mejorados.

A pesar de ser una aplicación que intenta proveer un entorno amigable, no es muy sencillo ni cómodo de operar, de modo que, la interfaz grafica de usuario es muy limitada. Además, no posee métodos de conexión a datos por medio de ODBC.

### **5.3.2 Herramientas para Modelado de Datos**

Son aplicaciones altamente especializadas y muy útiles para realizar un diseño de datos nuevo o comprender y estudiar un modelo existente (poseen características de ingeniería inversa). Su principal ventaja es la capacidad de realizar un diseño de modelo visualizándolo de forma gráfica, a través de diagramas entidad relación, o por medio de la obtención de un diagrama y documentación de una base de datos existente, mejorando la eficiencia de éste trabajo, al reducir el tiempo en las tareas de diseño, análisis y estudio.

#### **5.3.2.1 SQL Power Architect 1.0.6**

Es una herramienta de modelado de datos que posee muchas características dirigidas específicamente a arquitectos de almacenamiento de datos. Permite realizar ingeniería inversa en bases de datos existentes, realizar perfiles de datos en bases de datos de origen y generar automáticamente los metadatos de ETL. Se encuentra disponible comercialmente y en una versión comunitaria, es decir, de código abierto, esta última es muy cómoda en su uso. Además, ofrece bastantes opciones gráficas para la presentación, es una herramienta rápida y amigable. Algunas de las características más relevantes son:

- Permite acceder a las bases de datos a través de JDBC.
- Admite conexión a múltiples bases de datos al mismo tiempo.
- Compara modelos de datos y estructuras de bases de datos e identifica las discrepancias.
- Drag-and-drop de las tablas origen y las columnas en el área de trabajo.
- Ingeniería directa/inversa para PostgreSQL, Oracle, MS SQL Server, entre otras.

#### **5.3.2.2 PowerDesigner**

Es otra herramienta de diseño que ofrece una interfaz amigable que contempla todo el ciclo de modelado (lógico y físico) para una gran gama de bases de datos, además de otras tareas de modelado para objetos y esquemas XML, que permiten generar código. Resuelve las tareas de modelado e ingeniería directa e inversa. Su uso es muy sencillo y está disponible sólo para plataformas Windows.

#### **5.4 Conclusiones del Capítulo**

Finalmente, se ha revisado en este capítulo las herramientas propuestas por EOG encontradas en la literatura para ser utilizadas en el marco de trabajo. Por otro lado, es evidente que las herramientas son fundamentales para la aplicación la obtención del modelo de datos relacional. Del mismo modo, las herramientas planteadas podrían ser reemplazadas por otras que cumplan con la misma función.

## **Capítulo 6**

### **Caso de Estudio**

---

De conformidad a los objetivos planteados en esta investigación, en este capítulo se describe detalladamente la validación del marco de trabajo por medio de un caso de estudio aplicado a un SIH de una institución de salud pública que posee una antigüedad de 15 años. En particular, la validación se realiza en el módulo de Urgencias del Hospital Clínico Herminda Martin de Chillán (HCHM). Este sistema es considerado estratégico para la institución, puesto que se utiliza las 24 horas del día y los 365 días del año.

## **6.1 Descripción de SIH**

Los sistemas de información en salud evolucionaron, migrando su foco de atención desde el acto administrativo hacia el acto médico (MAIRA, NAVARRO et al. 2008). Las primeras experiencias de informatización se enfocaron específicamente en el “Componente administrativo”. Como ejemplos se hallan el sistema de turnos o agendamiento, el sistema de admisión o censo de pacientes, también llamado ADT (Admission, Discharge, Transfer), sistemas de farmacia y todos los sistemas relacionados a la gestión contable financiera (REICHERTZ 2006). Bajo este escenario, muchos procesos de negocio clínicos se realizan de forma manual, provocando ineficiencia, ya que los datos son digitados desde formularios escritos manualmente por funcionarios clínicos hacia los sistemas de información, induciendo de este modo a errores, lentitud e inexactitud de los datos. Por estas razones se ha hecho imperiosa la necesidad de migrar el sistema a plataformas que cumplan con los requerimientos actuales. Por lo tanto, se usará EOG para analizar y traspasar las estructuras de datos actuales a bases de datos relacionales normalizadas.

El sistema de la unidad de emergencia forma parte del HIS (Hospital Informatic System) de la unidad hospitalaria denominado SAM (Sistema de Administración Médica) el que fue desarrollado en los años 90 por IBM en Chile, este proyecto tuvo una duración de 4 años de construcción y puesta en marcha. En la actualidad, sigue siendo la piedra angular de los sistemas de la organización y existe una alta dependencia de éste. Con el transcurrir de los años, se le han realizado mantenciones preventivas tanto a nivel de software como también de hardware. Sin embargo, la tecnología de base de datos e interfaz gráfica sigue siendo la misma.

El volumen de información generada a diario en un recinto hospitalario que funciona las 24 horas del día, los 365 días de año, es del orden de los 20.000 registros diarios, en consecuencia posee una carga de trabajo intensa, sin embargo, el índice de operatividad es bastante aceptable gracias a que

es una aplicación que se ha robustecido durante el tiempo. Por otro lado, una de las dificultades que posee es su interfaz gráfica basada principalmente en textos y con una visión de registro posterior a los eventos clínicos por medio de una gran carga de trabajo mediante personal que digita en dicha aplicación.

Por otro lado, existe la constante necesidad de mostrar información hacia la web e interoperar con otros sistemas de información mediante protocolos establecidos con posterioridad al desarrollo del sistema. En consecuencia, se ha hecho necesaria la mejora no sólo de las interfaces de usuario mediante la técnica de wrapper, sino también cambios más profundos a nivel de los datos.

Estos cambios a nivel de estructura de datos están fundamentados, debido a problemas de inconsistencia en los datos, producto de un diseño redundante que no considera relaciones entre sus entidades. Es así como gran parte de la lógica de negocio se encuentra ubicada en la propia base de datos, es decir, muchas de las restricciones se encuentran validadas mediante el uso de triggers y procedimientos almacenados, programados por los desarrolladores de la época.

## **6.2 Utilizando EOG**

La aplicación del Marco de trabajo al caso de estudio indicado se realizó siguiendo rigurosamente cada una de las etapas y actividades señaladas en éste.

La cantidad y calidad de las fuentes de información encontradas son suficientes para aplicar el marco de trabajo con el apoyo de expertos. De acuerdo a la metodología, inicialmente se conforma el equipo de trabajo que llevará a cabo su ejecución y, posteriormente la aplicación de cada una de las etapas y actividades, tal como se describen detalladamente a continuación.

### **6.2.1 Conformación de Equipo de Trabajo**

Ajustado a los requisitos preliminares del marco de trabajo, y previa a la ejecución del mismo, se conformó el equipo de trabajo compuesto por el propio investigador de esta tesis y 2 colaboradores que poseen formación profesional del área informática con amplio dominio de bases de datos relacionales, desarrollo de sistemas de información y herramientas ETL.

Con el propósito de mantener un mayor control sobre los resultados del caso de estudio, la conducción del equipo de trabajo recayó en el propio investigador, en efecto, los demás integrantes desarrollaron tareas propias de cada etapa del marco de trabajo. Sin perjuicio de lo anterior, La

participación del investigador no se concentró meramente en supervisión de las actividades, sino también en la ejecución de cada una, es decir participó activamente en las etapas de Obtención y Generación. A continuación, se detallan las actividades desarrolladas por el equipo de trabajo.

## **6.2.2 Etapa de Extracción**

Etapa que esencialmente permitió extraer los datos y estructura desde la base de datos del módulo de Emergencia del HCHM. Las actividades se desarrollaron sin dificultades, debido a que, tanto los expertos como el equipo de trabajo, poseen un amplio conocimiento sobre el dominio del problema, y en particular un entendimiento sobre SIH. Adicionalmente, la cantidad y calidad de las fuentes de información favorecieron fundamentalmente a la validación de los datos extraídos desde la fuente de datos original. Seguidamente, se describen los pormenores de cada una de las actividades que componen la presente etapa.

### **6.2.2.1 Identificación de Fuentes de Información**

La presente actividad se realizó inicialmente por los integrantes del equipo de trabajo y posteriormente en conjunto con los expertos. Esto debido a que preliminarmente se realizó la búsqueda de expertos y luego en conjunto la identificación de otras fuentes de información que son de conocimiento de los expertos. De acuerdo a la clasificación propuesta por EOG, no se logró identificar la totalidad de la documentación, sin embargo, fue suficiente para llevar a cabo esta actividad. A continuación, se describen las fuentes de información halladas en éste caso de estudio.

#### **6.2.2.1.1 Identificación de Expertos**

Tal como se define en este marco de trabajo, los expertos cuentan con información relevante sobre el funcionamiento del sistema, no obstante y según la propia clasificación de éstos, no siempre se encuentran disponibles, debido a los años de funcionamiento y el estado de madurez del SIH. Particularmente, fue imposible mantener contacto con miembros del equipo de desarrollo de la aplicación, debido a que éstos no pertenecían a la organización, en consecuencia, los expertos identificados son el administrador de sistemas, administrador de base de datos y usuarios avanzados del SIH. A continuación, se identifican en detalle cada uno de estos.

- *Administrador de Sistemas:* La dependencia de los sistemas de información en la institución ha permitido la continuidad de este cargo. Actualmente esta responsabilidad recae en el jefe de la unidad de Ingeniería de Sistemas del HCHM, sus labores son las de mantener y asegurar el correcto funcionamiento de las aplicaciones, adaptándose a los cambios y evolución de la

Institución y la tecnología que da soporte a los sistemas, además de las labores propias de la jefatura de una unidad que está compuesta por siete ingenieros dedicados a la mantención del SIH y al desarrollo de nuevas aplicaciones.

El experto posee formación profesional en el área de la informática y una vasta experiencia en la mantención del sistema, además de un amplio dominio y conocimiento técnico sobre la estructura de las aplicaciones, documentación y códigos fuentes. En efecto, su participación en la búsqueda de fuentes de información fue trascendental. Por otro lado, la experiencia acumulada durante los años y el conocimiento sobre la evolución del sistema lo convierten en un componente clave como apoyo en la obtención de DFs y la generación del modelo de datos relacional.

*Administrador de Base de datos:* Al igual que el caso anterior, la estructura orgánica de la institución no considera el cargo de Administrador de Base de Datos, no obstante las labores propias de un administrador son asumidas por el jefe de la unidad de Soporte y Comunicaciones del Servicio de Salud Ñuble. Su gestión como administrador está asociado principalmente a la mantención de la base de datos, vale decir, verificar la integridad de los datos, definir e implementar controles de acceso a los mismos, así como asegurar un buen desempeño, mayor disponibilidad y recuperabilidad del SIH. Además, de las ocupaciones inherentes a la jefatura de una unidad compuesta por tres técnicos ocupada esencialmente de brindar soporte técnico a equipos y redes.

El experto posee formación profesional en el área de computación e informática y una dilatada experiencia en la puesta en marcha, mantención y soporte tecnológico del SIH. Además, posee las competencias suficientes para la interpretación de códigos fuentes, modelos de datos, estructuración de aplicaciones, debido a que este funcionario participó en la puesta en marcha del sistema de información hospitalario en los años 90. En consecuencia, contar con su experiencia y conocimiento es fundamental sobre todo en la obtención de DF preexistentes.

*Usuarios del sistema:* Las entidades públicas chilenas presentan una estructura que ha propiciado la mantención de funcionarios a lo largo del tiempo, por esta razón no fue difícil encontrar empleados con suficientes años de servicio y conocimiento que les permitiría aportar como expertos en la aplicación del marco de trabajo. Los usuarios identificados poseen competencias suficientes para realizar configuraciones básicas de las aplicaciones, así como la obtención de informes de gestión, entre otros. Más aún, de los funcionarios identificados sólo

se ha seleccionado el que posee mayor conocimiento acumulado y disponibilidad para aportar al grupo de expertos.

El experto posee una formación técnica del área administrativa y participó como uno de los usuarios líderes en la puesta en marcha del SIH. Fruto de aquello, está en condiciones de contribuir con conocimiento específico sobre aspectos relacionados con la mantención de parámetros base, además de la información que emanan de los reportes y datos que deben ser registrados en los formularios. En concreto su participación es vital debido al amplio grado de conocimiento del SIH.

#### **6.2.2.1.2 Identificación de Componentes del SIH**

La estructurada implementación y puesta en marcha del SIH hace más de una década en el HCHM, sumada a la conservación parcial de las fuentes de información y la adecuada cooperación de los expertos, permitieron que esta actividad se haya desarrollado sin sobresaltos.

De acuerdo a la categorización propuesta en el marco de trabajo fue posible encontrar una gran cantidad de información que contribuye de manera esencial a las etapas de obtención de DF y generación del modelo de datos. Específicamente, los códigos fuentes y programas ejecutables se encontraban en un repositorio de una forma bastante ordenada y fácil de manipular. A continuación, se describen las fuentes encontradas bajo esta clasificación.

- **Códigos fuentes**

Los documentos electrónicos se encuentran disponibles y debidamente organizados bajo una estructura jerárquica en medios magnéticos. El Sistema SAM alberga todas las aplicaciones, de esta forma la propia disposición de los códigos fuentes sigue esta misma organización, tal como se muestra en el Anexo B.

La estructura de directorios de aplicaciones de SAM permite encontrar códigos fuentes de los programas y base de datos. De este modo, al interior de cualquier subdirectorio de sam (sam\ur) se encuentran los directorios “4gl” y “sql”. Por consiguiente, se encuentra la totalidad de los códigos fuentes del SIH. Para mayor detalle se especifica en el Anexo B un ejemplo de los script SQL.

Particularmente, el sistema de emergencia del HCHM utiliza módulos de otros sistemas de SAM, que se interrelacionan entre sí. Por ejemplo, el módulo de pacientes es transversal a la mayoría de los sistemas del HIS. Luego de explorar las entidades que componen el SIH, los expertos deciden extraer sólo las tres entidades más representativas del sistema de emergencias, debido a que estas

contienen gran parte de la lógica del negocio. A continuación, en la Tabla 6.1 se describen las funcionalidades de las tablas SABST030, SAURT009 y SAPAT001 redactada en su momento por los propios desarrolladores.

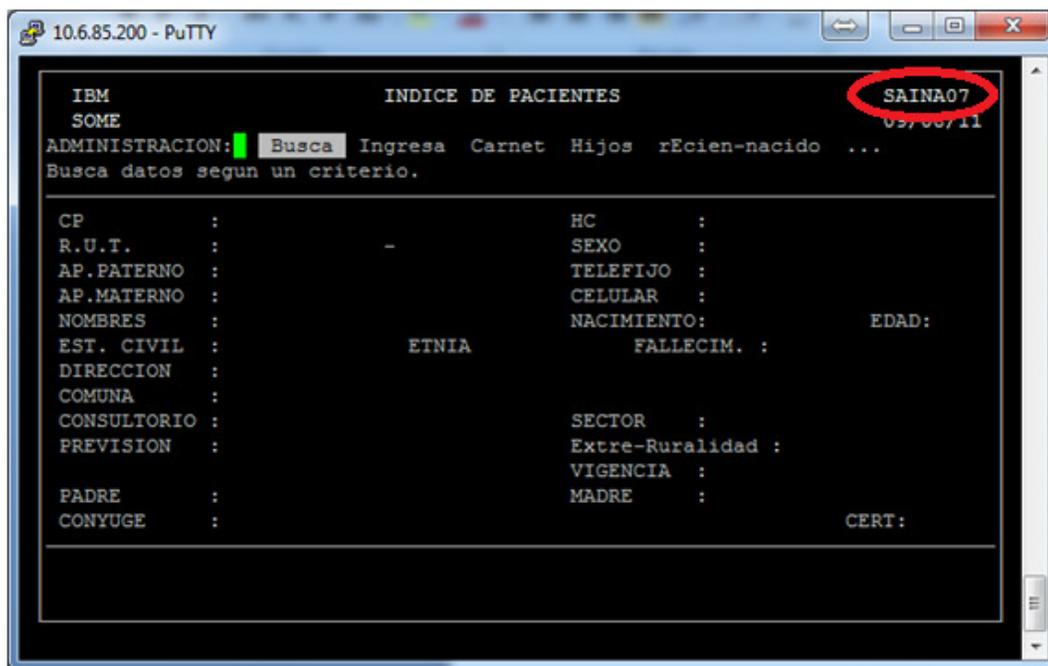
<u>Entidad</u>	<u>Descripción</u>	<u>Observaciones</u>
SABST030	Funcionarios	Tabla que contiene a los Profesionales del Establecimiento, Profesionales Externos al Establecimiento e Instituciones Externas (Empresas, Establecimientos hospitalarios u otros).
SAURT009	Documentos de Admisión de Urgencia (DAU)	Tabla que contiene los antecedentes propios (básicos relevantes) de la atención de Urgencia, representados en el DAU.
SAPAT001	Pacientes	Tabla que contiene los antecedentes básicos de un Paciente. Siempre que un paciente exista, debe haber una fila en esta tabla. En el Índice de pacientes pueden coexistir pacientes reales (personas) o pacientes ficticios (Instituciones).

**Tabla 6.1:** Descripción de entidades.

- **Programas Ejecutables**

Estas fuentes de información se encuentran disponibles y facilitan el análisis de la entrada de datos desde las interfaces gráficas de usuario de las aplicaciones. El análisis consiste en realizar comparaciones entre los datos ingresados desde la aplicación y los datos almacenados en la propia base de datos, esto con el objeto de comprobar el estado final de los datos luego del ingreso.

Particularmente, todas las aplicaciones de SAM exhiben en el extremo superior derecho el nombre del formulario, tal como se muestra en el Anexo B. De este modo, se facilita la ubicación de las fuentes en la estructura de directorios descrita en la Figura 6.1. El acceso a las aplicaciones se realiza mediante un emulador de terminal que actúa como cliente telnet y facilita la conexión con el host. Mayor información respecto de las interfaces de usuario se pueden encontrar en el AnexoB.



**Figura 6.1:** Estructura formularios.

### 6.2.2.1.3 Identificación de Documentación

En relación a la documentación disponible del SIH, es preciso indicar que no existen diagramas que sugieran modelos de datos expresados en documentos DDL que evidencien la estructuración de las entidades. Sin embargo, existe documentación asociada a una descripción de los datos, más específicamente diccionario de datos, información que es trascendental para la obtención de DFs preexistentes. Dicha información se encuentra en documentos impresos que a continuación se describen detalladamente.

- **Diccionario de datos**

Documentación que se encontró parcialmente disponible, debido a que los manuales de desarrollo y usuarios no han sido conservados adecuadamente, producto de lo anterior, sólo se cuenta con una parte de la información, particularmente sobre los datos asociados a pacientes. Mientras que para las entidades del sistema de emergencia, se encuentran extraviadas.

De este modo, la documentación asociada al modelo de datos se encuentra en archivos electrónicos en formato SQL y proveen información relevante sobre la descripción de los atributos.

- **Modelo de datos**

Documentación destinada a describir el modelo de datos del sistema SAM se obtuvo a partir de la definición de la estructura de datos por medio del lenguaje de definición de datos SQL. En particular, se utilizaron documentos electrónicos con instrucciones que permitieron crear las tablas y sus atributos. La Figura 6.2 muestra una parte del diagrama del modelo de datos de SAM obtenido a través de la herramienta SQL Power Architect 1.0.6.



**Figura 6.2:** Esquema resumen del modelo de datos SAM.

- **Especificaciones de programa**

Las especificaciones de programas se encuentran expresadas fundamentalmente en una descripción acuciosa de los procedimientos realizadas por cada módulo.

La existencia de un manual de uso de la aplicación ha permitido una familiarización de conceptos y el descubrimiento de la naturaleza y el objetivo con el que fueron creados los programas.

En el Capítulo 3 de dicho manual se describen, detalladamente, procedimientos tales como, oficina/ventanilla de índice de pacientes, generación de nóminas de pacientes, búsqueda de pacientes, instituciones con características de pacientes, definición de Instituciones a facturar. Del mismo modo, en el Capítulo 4 se describen los casos especiales para pacientes NN<sup>4</sup> en urgencia, creación de índice en admisión de urgencia y admisión hospitalaria, nóminas de pacientes. Mientras que en el Capítulo 5 se describe el flujo de pantallas. El manual de usuario de SAM se detalla en la documentación que se encuentra en el Anexo B.

#### 6.2.2.1.4 Registro de Fuentes de Información

Las fuentes de información se registraron electrónicamente, tal como se muestra en la Tabla 6.2, y permitieron obtener un resumen en que se especifican datos simplificados como clasificación, descripción y ubicación de las mismas.

<u>Clasificación</u>	<u>Sub clasificación</u>	<u>Descripción</u>	<u>Ubicación</u>	<u>Calidad</u>
Experto	Administrador de Sistemas	Jefe de Ingeniería de Sistemas	Oficina Ingeniería de sistemas	Buena
Experto	Administrador de Base de Datos	Jefe de Soporte y comunicaciones	Soporte Técnico Servicio Salud Nube	Buena
Experto	Usuario del SIH	Administrativo de SOME	Oficina de Admisión	Buena
Componentes SIH	Códigos Fuentes	Fuentes de aplicaciones y script de tablas	Servidor Producción CHHOS	Buena
Componentes SIH	Programas Ejecutables	Aplicaciones en producción	Servidor Producción CHHOS	Buena
Documentación	Diccionario de Datos	Manuales impresos	Soporte Técnico HCHM	Regular
Documentación	Modelo de Datos	Script SQL de creación	Servidor Producción CHHOS	Buena
Documentación	Especificaciones de Programas	Manuales impresos	Soporte Técnico HCHM	Regular

**Tabla 6.2:** Fuentes de Información de SAM.

<sup>4</sup> Del latín *Nomen nescio* (nombre desconocido) se utiliza para referirse a pacientes sin identidad, es decir se desconoce su nombre real

### 6.2.2.2 Extracción de Estructura y Datos

Actividad que permitió la revisión de aspectos generales de la base de datos, tales como la elección de herramientas que facilitan la extracción desde la fuente de datos de origen y la ejecución automática de la extracción. A continuación, se detallan cada una de las actividades de esta etapa.

#### 6.2.2.2.1 Revisión de la fuente de datos original del SIH

Siguiendo los procedimientos señalados por el marco de trabajo, se realiza inicialmente un catastro de las características de la fuente de datos del SIH y, posteriormente, el registro de entidades que serán extraídas en la etapa siguiente.

La fuente de datos es única, es decir, los datos residen en un sólo lugar, particularmente en una base de datos Informix versión 11.0. Esta base de datos opera sobre una arquitectura UNIX, específicamente sobre la distribución Red Hat Enterprise 5.0 de Linux. La base de datos junto a las aplicaciones se encuentran funcionando continuamente las 24 horas del día.

El modo inicial de acceder a los datos se realiza de manera nativa, utilizando una consola de emulación Unix que permite ingresar al host y posteriormente ejecutar la herramienta de administración de base de datos DBACCES de Informix 11.

La extracción desde SAM se realiza sobre una muestra representativa del sistema de Unidad de Emergencia del HCHM, que corresponde a tres grandes entidades que albergan datos relacionados con la admisión y atención de pacientes en dicha unidad. La elección de estas entidades se realizó considerando la relevancia de las mismas en el SIH, es decir, son representativas debido a la gran cantidad de atributos que poseen.

De este modo, y siguiendo las actividades propuestas en EOG, se realiza la revisión y registro de entidades que serán extraídas en la etapa siguiente. La Tabla 6.3 identifica la ubicación y el nombre de la entidad.

<u>Servidor</u>	<u>Entidad</u>	<u>Descripción</u>
CHHOS	SAPAT001	Tabla que registra datos personales de pacientes
CHHOS	SABST030	Tabla que mantiene datos personales de funcionarios
CHHOS	SAURT009	Tabla que registra las atenciones de urgencia

**Tabla 6.3:** Entidades a extraer desde SAM.

#### **6.2.2.2 Selección de ADI y Herramientas**

La elección del ADI, herramienta ETL y aplicación para modelamiento se realizó priorizando su alto nivel de compatibilidad técnica. Entre estos, el uso intuitivo, favorable control de errores y experiencia en el uso por parte del equipo de trabajo y expertos.

Oracle 10g EE se presenta como la base de datos más adecuada para su uso como ADI, debido a su versatilidad en el uso, fácil instalación y gran compatibilidad con la herramienta (RAFREIDER 2008) que facilita la obtención de DFs. Por otro lado, Microsoft DTS es la herramienta ideal para extraer datos, debido a su disponibilidad en la institución, control de eventos y experiencia del investigador. Finalmente, Power Architect 1.0.6 se convierte en la herramienta apropiada para generar el modelo de datos por medio de instrucciones SQL, además de ayudar a comparar las entidades obtenidas desde la(s) fuente(s) de datos original(es) y las nuevas entidades en el ADI.

Adicionalmente, el acceso a la fuente de datos de origen se realizó mediante el uso de un SDK (Software Development Kit) de Informix, que actúa como una herramienta que permite el acceso a esta fuente mediante el uso de drivers y librerías que facilitan el acceso a los datos por medio de ODBC desde cualquier aplicación. La configuración del SDK requiere de datos que fueron proporcionados por el administrador de base de datos.

#### **6.2.2.3 Extracción de Fuente de Datos**

Luego de establecer y comprobar la conexión entre la herramienta ETL y la fuente de datos se realiza la extracción, y para completar esta actividad, se utilizó la herramienta de transformación de datos Microsoft DTS en la que se identifica el origen de los datos (fuente de datos del SIH) y, posteriormente, el destino (ADI).

Concretamente, esta labor no representó complejidad puesto que sólo se indicaron el origen y destino de los datos, los elementos extraídos y, en último lugar, se indicó la extracción automática de la estructura y los datos contenidos en ésta.

#### **6.2.2.3 Validación**

Actividad desarrollada conjuntamente entre el equipo de trabajo, administrador de sistemas y administrador de base de datos, con el objeto de realizar la revisión de la estructura y los datos extraídos en Oracle. Particularmente, esta actividad no reportó mayores dificultades, salvo la tabla SAPAT001 que contiene los datos demográficos de pacientes, debido a que contiene una gran cantidad de datos, aproximadamente 500.000 registros.

### 6.2.2.3.1 Verificación de Inconsistencias

La verificación de inconsistencias permitió detectar ciertos datos incoherentes en el ADI que tienen su génesis en las fuentes de datos originales del SIH, debido a problemas de digitación. Dichos datos son inconsistentes para el formato de los atributos del ADI, particularmente, en aquellos de tipo fecha, debido a que se encuentran fuera del rango aceptado.

El proceso de validación fue realizado de modo semiautomático por el equipo de trabajo, y consistió en un análisis de la estructura y los datos de cada entidad extraída. El análisis de la estructura se realizó por medio de la herramienta Power Architect 1.0.6 e inspecciones visuales que permitieron verificar que los nombres, tipos de datos y tamaños de los atributos en el ADI coincidan con la fuente de datos de origen. Posteriormente, se observaron los datos en el ADI, con el objeto de corroborar que estos se hayan extraído correctamente, es decir, verificar que no se encuentren trancos, nulos o inconsistentes.

La verificación comienza con una comparación visual entre las entidades originales ubicadas en Informix 11.0 y las nuevas entidades albergadas en Oracle 10g. Esta labor no representó mayor complejidad, sin embargo, resulta rutinaria y agotadora debido a que se deben revisar cada uno de los atributos con sus respectivos tipos de datos y tamaños. Por esta razón, se apoyó dicha labor con la verificación automática de la herramienta. Las inconsistencias detectadas en la extracción de la fuente de datos original se concentraron principalmente en atributos de tipo fecha y en la precisión de datos numéricos. Lo anterior tiene relación con errores de digitación provocados generalmente por falta de rigurosidad de las validaciones en las interfaces de usuario del SIH. En conclusión, las tablas extraídas están individualizadas en las Tablas 6.4, 6.5 y 6.6.

Atributos		
CODPACIE	FECHANAC	FACTORRH
FECHACREA	FECHAFALL	OCUPACION
ESTADO	SEXO	DOCUPACION
RUT	FORMAPAGO	NOMBREPAD
DIGIVER	DOMICILIO	NOMBREMAD
HC	COMUNA	OBSERVACION
APELLPAT	DCOMUNA	INSTITUCION
APELLMAT	TELEFONO	RUT_INSTITUCION
NOMBRES	GRUPSANG	
DFORMAPAGO	RAZSOCIAL_INSTITUCION	
	FONO_INSTITUCION	

**Tabla 6.4:** Estructura tabla SAPAT001.

La tabla SAPAT001 está compuesta por 29 atributos y posee 570.000 tuplas, su objetivo es mantener el registro de los datos demográficos de todos los pacientes del HCHM.

Atributos	
RUT	D_AFP
DV	CARGAS_FAM
PLANTA	LEY
D_PLANTA	NRO_HORAS
JURIDICA	F_NACIMIENTO
D_JURIDICA	UNIDAD
GRADO	D_UNIDAD
GENERO	CARGO
ESTADOCIVIL	D_CARGO
DIRECCION	TITULO
COMUNA	F_INICIO_CONTRATO
F_ING_GRADO	F_TERMINO_CONTRATO
F_ING_SERV	F_ALEJAMIENTO
F_ING_ADM_PUB	TOTAL_HABERES
ISAPRE	REMUNERACION
D_ISAPRE	USUARIOINGRESO
AFP	FECHAINGRESO

**Tabla 6.5:** Estructura tabla SABST030.

La tabla SABAST030 contiene los datos personales y contractuales de los funcionarios del HCHM, está compuesta por 34 atributos y 1500 registros.

Atributos		
CODIGO	DLUGAR	CARABINERO
CP	SERVEJE	DIAGPOS1
FP	DSERVEJE	DDIAGPOS1
DFP	COMENTARIO	DIAGPOS2
SERVSOL	ANULADO	DDIAGPOS2
DSERVSOL	CIERRE	PERSTRATANTE
PROFSOL	FCREA	PROFEJE
FECHALLEG	PRESION	SERVDEST
MOTIVO	PULSO	DSERVDEST
DMOTIVO	TEMPAXILAR	DPROFSOL
MEDIO	TEMPRECTAL	DPERSTRATANTE
DMEDIO	ALCOHOLEMIA	DPROFEJE
LUGAR	AMBULANCIA	

**Tabla 6.6:** Estructura tabla SAURT009.

La tabla SAURT009 contiene las admisiones de urgencia del HCHM, en ella se registran datos administrativos y clínicos de pacientes atendidos en dicha unidad.

Finalmente, con las tablas y datos en el ADI, sumado a las fuentes de información identificadas concluye la primera etapa de extracción. Con esto, están dadas las condiciones para dar paso a la etapa de obtención.

### **6.3.3 Etapa de Obtención**

El proceso de obtención de DFs se lleva a cabo mediante el análisis de las tres entidades hospedadas en el ADI, que se han obtenido desde la fuente de datos de origen en la etapa anterior. El proceso se inicia con la obtención de DFs manuales (preexistentes) y posteriormente de manera automática.

A continuación, se describen las actividades realizadas en esta etapa.

#### **6.3.3.1 Obtención de DFs Preexistentes**

Apegado al objetivo principal de esta actividad, se realizó la identificación de las DFs preexistentes de manera manual, por medio del esfuerzo conjunto entre el equipo de trabajo y expertos. El desarrollo de esta actividad implicó la ejecución de otras actividades tendientes a comprobar la satisfacción las DFs en los propios datos del ADI.

La obtención de DFs preexistentes se realizó tomando en consideración las fuentes de información y las actividades relacionadas con cada una de estas, indicadas en la Tabla 4.6 de la etapa de Obtención en el capítulo uno.

Los códigos fuentes encontrados fueron utilizados con el objeto de validar y mejorar la comprensión del dominio de las entidades. En general, algunas dudas fueron resueltas consultando las propias fuentes de la aplicación. Por otro lado, el modelo de datos expresado en formato DDL, facilitó la obtención de información respecto de la naturaleza de la entidad, es decir, determinar cuál es el dominio de los datos que se deben encontrar en la entidad. En definitiva, la documentación fue de vital importancia para comprender el rango de datos aceptados y los procedimientos para realizar acciones en el sistema.

#### **DFs establecidas en el dominio del problema**

Las DFs propias del dominio del problema fueron identificadas conjuntamente entre el equipo de trabajo y expertos. El conocimiento avanzado de los expertos sobre los datos del SIH, sumado a la

experiencia del equipo de trabajo en sistemas de información clínicos, facilitaron la obtención de dichas DFs. Se realizó un análisis detallado de cada una de las entidades contenidas en el ADI, más específicamente, una revisión de los atributos que permitiera identificar las DFs de acuerdo al criterio técnico asociado al dominio del problema y no al caso particular de estudio.

A continuación, en la Tabla 6.7, se identifican las DFs propias del dominio de las entidades SAPAT001, SAURT009 y SABST030.

<u>Entidad</u>	<u>Determinante</u>	<u>-&gt;</u>	<u>Determinado</u>
SAPAT001	RUT	->	Todos los atributos de la entidad, puesto que es un identificador único nacional de personas.
SAPAT001	HC	->	Todos los atributos de la entidad, puesto que es un identificador único de pacientes al interior de un hospital.
SAURT009	CODIGO	->	Todos los atributos, puesto que es un identificador único de la entidad.
SABST030	RUT	->	Todos los atributos de la entidad, puesto que es un identificador único nacional de personas.

**Tabla 6.7:** DFs establecidas en el Dominio.

En la Tabla 6.7 queda demostrado que la mayoría de las DFs encontradas son resultado de unos pocos atributos, de los cuales existe plena certeza que son determinantes. Tal es el caso de la entidad SAPAT001, en que tanto el RUT como la Historia Clínica (HC) determinan al resto de los atributos. Esto debido a que son únicos y propios de cada paciente. Del mismo modo el CODIGO es un atributo autonumérico y único que determina el resto de los atributos en la tabla SAURT009. En último lugar, e igualmente en la entidad SABST030, el RUT determina a todos los demás atributos.

#### **6.3.3.1.1 DFs establecidas en los Atributos**

Actividad que facilitó la obtención de DFs con un mayor grado de especificación que en la anterior, debido a que se logran identificar aquellas dependencias que son propias del dominio particular de SAM. La obtención de DFs se genera a partir de conocimientos puramente técnicos de los expertos sobre los datos almacenados en la base de datos de SAM. A continuación, se muestran en la Tabla 6.8 las DFs obtenidas manualmente conforme al criterio de los expertos y del equipo de trabajo.

DF	Entidad	Determinante	->	Determinado	DF	Entidad	Determinante	->	Determinado
DFP1	SAPAT001	CODPACIE	->	RUT	DFP56	SAPAT001	COMUNA	->	DCOMUNA
DFP2	SAPAT001	CODPACIE	->	HC	DFP57	SAPAT001	OCUPACION	->	DOCUPACION
DFP3	SAPAT001	CODPACIE	->	APELLPAT	DFP58	SAPAT001	INSTITUCION	->	RAZOCIAL_INSTITUCION
DFP4	SAPAT001	CODPACIE	->	APELLMAT	DFP59	SABST030	RUT	->	JURIDICA
DFP5	SAPAT001	CODPACIE	->	NOMBRES	DFP60	SABST030	RUT	->	GRADO
DFP6	SAPAT001	CODPACIE	->	FECHANAC	DFP61	SABST030	RUT	->	GENERO
DFP7	SAPAT001	CODPACIE	->	FECHAFALL	DFP62	SABST030	RUT	->	ESTADOCIVIL
DFP8	SAPAT001	CODPACIE	->	SEXO	DFP63	SABST030	RUT	->	DIRECCION
DFP9	SAPAT001	CODPACIE	->	FORMAPAGO	DFP64	SABST030	RUT	->	COMUNA
DFP10	SAPAT001	CODPACIE	->	DOMICILIO	DFP65	SABST030	RUT	->	ISAPRE
DFP11	SAPAT001	CODPACIE	->	COMUNA	DFP66	SABST030	RUT	->	AFP
DFP12	SAPAT001	CODPACIE	->	TELEFONO	DFP67	SABST030	RUT	->	CARGAS_FAM
DFP13	SAPAT001	CODPACIE	->	GRUPOSANG	DFP68	SABST030	RUT	->	UNIDAD
DFP14	SAPAT001	CODPACIE	->	FACTORRH	DFP69	SABST030	RUT	->	CARGO
DFP15	SAPAT001	CODPACIE	->	OCUPACION	DFP70	SABST030	RUT	->	TITULO
DFP16	SAPAT001	CODPACIE	->	NOMBREPAD	DFP71	SABST030	RUT	->	F_INICIO_CONTRATO
DFP17	SAPAT001	CODPACIE	->	NOMBREMAD	DFP72	SABST030	RUT	->	TOTAL_HABERES
DFP18	SAPAT001	CODPACIE	->	INSTITUCION	DFP73	SABST030	RUT	->	REMUNERACION
DFP19	SAPAT001	RUT	->	DV	DFP74	SABST030	RUT	->	FECHA_INGRESO
DFP20	SAPAT001	RUT	->	HC	DFP75	SABST030	RUT	->	DV
DFP21	SAPAT001	RUT	->	APELLPAT	DFP76	SABST030	RUT	->	PLANTA
DFP22	SAPAT001	RUT	->	APELLMAT	DFP77	SABST030	PLANTA	->	D_PLANTA
DFP23	SAPAT001	RUT	->	NOMBRES	DFP78	SABST030	JURIDICA	->	D_JURIDICA
DFP24	SAPAT001	RUT	->	FECHANAC	DFP79	SABST030	COMUNA	->	D_COMUNA
DFP25	SAPAT001	RUT	->	FECHAFALL	DFP80	SABST030	ISAPRE	->	D_ISAPRE
DFP26	SAPAT001	RUT	->	SEXO	DFP81	SABST030	AFP	->	D_AFP
DFP27	SAPAT001	RUT	->	FORMAPAGO	DFP82	SABST030	UNIDAD	->	D_UNIDAD
DFP28	SAPAT001	RUT	->	DOMICILIO	DFP83	SABST030	CARGO	->	D_CARGO
DFP29	SAPAT001	RUT	->	COMUNA	DFP84	SABST030	PROVINCIA	->	D_PROVINCIA
DFP30	SAPAT001	RUT	->	TELEFONO	DFP85	SABST030	REGION	->	D_REGION
DFP31	SAPAT001	RUT	->	GRUPOSANG	DFP86	SAURT009	CODIGO	->	CP
DFP32	SAPAT001	RUT	->	FACTORRH	DFP87	SAURT009	CODIGO	->	FP
DFP33	SAPAT001	RUT	->	OCUPACION	DFP88	SAURT009	CODIGO	->	SERVSOL
DFP34	SAPAT001	RUT	->	NOMBREPAD	DFP89	SAURT009	CODIGO	->	PROFSOL
DFP35	SAPAT001	RUT	->	NOMBREMAD	DFP90	SAURT009	CODIGO	->	FECHALLEG
DFP36	SAPAT001	RUT	->	INSTITUCION	DFP91	SAURT009	CODIGO	->	MOTIVO
DFP37	SAPAT001	HC	->	RUT	DFP92	SAURT009	CODIGO	->	DMOTIVO
DFP38	SAPAT001	HC	->	HC	DFP93	SAURT009	CODIGO	->	MEDIO
DFP39	SAPAT001	HC	->	APELLPAT	DFP94	SAURT009	CODIGO	->	LUGAR
DFP40	SAPAT001	HC	->	APELLMAT	DFP95	SAURT009	CODIGO	->	SERVEJE
DFP41	SAPAT001	HC	->	NOMBRES	DFP96	SAURT009	CODIGO	->	FCREA
DFP42	SAPAT001	HC	->	FECHANAC	DFP97	SAURT009	CODIGO	->	PRESION
DFP43	SAPAT001	HC	->	FECHAFALL	DFP98	SAURT009	CODIGO	->	PULSO
DFP44	SAPAT001	HC	->	SEXO	DFP99	SAURT009	FP	->	D_FP
DFP45	SAPAT001	HC	->	FORMAPAGO	DFP100	SAURT009	SERVSOL	->	D_SERVSOL
DFP46	SAPAT001	HC	->	DOMICILIO	DFP101	SAURT009	PROFSOL	->	D_PROFSOL
DFP47	SAPAT001	HC	->	COMUNA	DFP102	SAURT009	MOTIVO	->	D_MOTIVO
DFP48	SAPAT001	HC	->	TELEFONO	DFP103	SAURT009	MEDIO	->	D_MEDIO
DFP49	SAPAT001	HC	->	GRUPOSANG	DFP104	SAURT009	LUGAR	->	D_LUGAR
DFP50	SAPAT001	HC	->	FACTORRH	DFP105	SAURT009	SERVEJE	->	D_SERVEJE
DFP51	SAPAT001	HC	->	OCUPACION	DFP106	SAURT009	DIAGPOS1	->	D_DIAGPOS1
DFP52	SAPAT001	HC	->	NOMBREPAD	DFP107	SAURT009	DIAGPOS2	->	D_DIAGPOS2
DFP53	SAPAT001	HC	->	NOMBREMAD	DFP108	SAURT009	PROFEJE	->	D_PROFEJE
DFP54	SAPAT001	HC	->	INSTITUCION	DFP109	SAURT009	SERVDEST	->	D_SERVDEST
DFP55	SAPAT001	FORMAPAGO	->	DFORMAPAGO					

**Tabla 6.8:** Tabla de Registro de DFs establecidas en los Atributos.

La obtención manual resultó interesante, debido a que las DFs establecidas en los atributos no necesariamente se satisfacían en los datos contenidos en el ADI, por ejemplo, la DF (RUT->NOMBRE) resulta obvia para cualquier entidad que almacene datos personales en un Sistema de Información, sin embargo, al comprobar su existencia en los datos, nos encontramos con una incongruencia, debido a que existe un RUT duplicado, es decir, un paciente ingresado más de una vez. En general, esta situación tiene origen en la falta de rigurosidad en las validaciones de las interfaces de usuario, sumado a la baja mantención de los datos.

Consecuente con lo anterior, cada una de las DFs fueron comprobadas manualmente en el ADI en la actividad siguiente.

### 6.3.3.1.2 Limpieza

Actividad en la que se ejecutaron esencialmente dos acciones. La primera, orientada a la comprobación y satisfacción de las DFs preexistentes en los datos mediante la limpieza de datos. La segunda, tiene relación con la poda de atributos, con el objeto de disminuir la carga de trabajo de la herramienta que facilita la obtención de DFs. La comprobación se ejecutó por medio del uso de instrucciones SQL. Por ejemplo, para verificar la DF  $A \rightarrow B$ , con  $A \subset R$ ,  $B \in R$  y  $R$  un conjunto de atributos de la relación, entonces se ejecutaron las siguientes instrucciones SQL:

- (1) `select count (distinct A,B) from TABLA`
- (2) `select count (distinct A) from TABLA`

Si ambos resultados son iguales, entonces la DF  $A \rightarrow B$  se verifica. Es decir, si el número de tuplas  $\langle A;B \rangle$  distintas es igual al número de tuplas  $\langle B \rangle$  distintas, entonces la DF  $A \rightarrow B$  se cumple en la relación.

Posteriormente, los resultados inconsistentes de la comprobación de DFs preexistentes se registraron, tal como se muestra en la Tabla 6.9, con el objetivo de realizar la limpieza posteriormente.

<u>Comprobación</u>	<u>Resultado</u>
3 y 4	9996 RUT aparecen nulos, en consecuencia duplicados.
5 y 6	1048 HC aparecen nulos, en consecuencia duplicados.
12	201 RUT aparecen duplicados.
13	La planta "Directivos" aparece 2 veces con códigos distintos, códigos repetidos
14	Descripción de Calidad jurídica duplicada.
15	Una misma comuna con distinto código
19	El cargo "Directivo" aparece 2 veces con códigos distintos.

**Tabla 6.9:** Resultados inconsistentes.

La limpieza de los datos se lleva a cabo conforme a la comprobación de las DFs preexistentes indicadas en la tabla 6.9, por lo que cada comprobación se detalla en la Tabla 6.10.

Comprobación	Limpieza
3,4	<pre> DECLARE p_cp SAPAT001.codpacie%TYPE; p_rut SAPAT001.rut%TYPE; CURSOR c_sapat001 IS SELECT codpacie FROM SAPAT001 WHERE rut is null; BEGIN select max(rut) into p_rut from SAPAT001; OPEN c_sapat001; LOOP   FETCH c_sapat001 INTO p_cp;   p_rut:= p_rut+1;   UPDATE SAPAT001 SET RUT= p_rut WHERE codpacie=p_cp;   EXIT WHEN c_sapat001%NOTFOUND; END LOOP;CLOSE c_sapat001;END; </pre>
5,6	<pre> DECLARE p_cp SAPAT001.codpacie%TYPE; p_hc SAPAT001.hc%TYPE; CURSOR c_sapat001 IS SELECT codpacie FROM SAPAT001 WHERE hc is null; BEGIN select max(hc) into p_hc from SAPAT001; OPEN c_sapat001; LOOP   FETCH c_sapat001 INTO p_cp;   p_hc:= p_hc+1;   UPDATE SAPAT001 SET HC= p_hc WHERE codpacie=p_cp;   EXIT WHEN c_sapat001%NOTFOUND; END LOOP;CLOSE c_sapat001; END; </pre>
12	<pre> UPDATE SABST030_ SET RUT=' ' WHERE RUT IN(SELECT RUT FROM SABST030_ GROUP BY RUT HAVING (COUNT(1)&gt;1)); DECLARE p_codigo SABST030_.codigo%TYPE; p_rut SABST030_.rut%TYPE; CURSOR c_sabst030 IS SELECT codigo FROM SABST030_ WHERE rut is null; BEGIN select max(rut) into p_rut from SABST030_; OPEN c_sabst030; LOOP   FETCH c_sabst030 INTO p_codigo;   p_rut:= p_rut+1;   UPDATE SABST030_ SET RUT= p_rut WHERE codigo=p_codigo;   EXIT WHEN c_sabst030%NOTFOUND; END LOOP;CLOSE c_sabst030;END; </pre>
13	<pre> UPDATE SABST030_ SET PLANTA='99', DPLANTA='NUEVA PLANTA PARA DF' WHERE PLANTA=1 AND DPLANTA='DIRECTIVOS'; UPDATE SABST030 SET PLANTA='999' WHERE DPLANTA='ODONTOLOGOS'; UPDATE SABST030 SET PLANTA='9999' WHERE DPLANTA='QUIMICOS'; </pre>
14	<pre> UPDATE SABST030_ SET JURIDICA='99', DJURIDICA='NUEVA CALIDAD PARA DF1' WHERE JURIDICA=11 AND DJURIDICA='TITULARES'; UPDATE SABST030_ SET JURIDICA='999', DJURIDICA='NUEVA CALIDAD PARA DF2' WHERE JURIDICA=12 AND DJURIDICA='CONTRATADOS'; </pre>
15	<pre> UPDATE SABST030 SET COMUNA='99' WHERE DCOMUNA='PARRAL'; </pre>
19	<pre> UPDATE SABST030_ SET CARGO='9999', DCARGO='NUEVO CARGO PARA DF' WHERE CARGO=927 AND DCARGO='DIRECTIVO'; </pre>

**Tabla 6.10:** Limpieza de datos.

La limpieza se lleva a cabo a través de la ejecución de las instrucciones exhibidas en la tabla anterior. Las comprobaciones 3,4,5,6 y 12 reemplazan los valores faltantes con el máximo + 1. Esto,

debido a que no interesa el valor exacto del atributo, sino que es importante asegurar que todos los valores son distintos.

<u>Id</u>	<u>Atributo</u>	<u>Entidad</u>	<u>Clasificación</u>	<u>Justificación</u>
1	FECHACREA	SAPAT001	Atributo propio del Registro	Atributo asociado a la fecha y hora en que se realizó el registro de la tupla.
2	ESTADO	SAPAT001	Atributo propio del Registro	Atributo que indica el estado de la tupla.
3	DOMICILIO	SAPAT001	Atributo Observación	Atributo en que se describe el lugar donde habita el paciente, es un campo abierto.
4	TELEFONO	SAPAT001	Atributo Observación	Atributo que sólo indica el teléfono de contacto del paciente, es un campo abierto.
5	OBSERVACION	SAPAT001	Atributo Observación	Atributo que ofrece un campo abierto para realizar observaciones y comentarios sobre un paciente.
6	NOMBREPAD	SAPAT001	Atributo Observación	Atributo en que se indica el nombre del padre, dato no parametrizado.
7	NOMBREMAD	SAPAT001	Atributo Observación	Atributo en que se indica el nombre de la madre, dato no parametrizado.
8	DV	SAPAT001	Atributo Derivado	Atributo que solo depende funcionalmente del RUT y no determina a ningún otro atributo.
9	DV	SABST030	Atributo Derivado	Atributo que solo depende funcionalmente del RUT y no determina a ningún otro atributo.
10	FUNCIONARIO	SABST030	Atributo Observación	Atributo que describe los nombres y apellidos de los funcionarios, es un campo abierto.
11	DIRECCION	SABST030	Atributo Observación	Atributo en que se describe el domicilio del funcionario, es un campo abierto , este no es capaz de determinar a ningún otro atributo.
12	GENERO	SABST030	Atributo Observación	Atributo que indica el género del funcionario, este no es capaz de determinar a ningún otro atributo.
13	FNACIMIENTO	SABST030	Atributo Observación	Atributo que indica la fecha de nacimiento del funcionario, este no es capaz de determinar a ningún otro atributo.
14	TITULO	SABST030	Atributo Observación	Atributo que describe el título profesional del funcionario, este no es capaz de determinar a ningún otro atributo.
15	COMENTARIO	SAURT009	Atributo Observación	Atributo que registra los comentarios del funcionario, este no es capaz de determinar a ningún otro atributo.
16	ANULADO	SAURT009	Atributo Observación	Atributo que describe la fecha de anulación del registro, este no es capaz de determinar a ningún otro atributo.
17	CIERRE	SAURT009	Atributo Observación	Atributo que describe la fecha de cierre del registro, este no es capaz de determinar a ningún otro atributo.
18	FCREA	SAURT009	Atributo propio del Registro	Atributos que describen la fecha de creación del registro y llegada, por lo que no son capaces de determinar a ningún otro atributo.
19	FECHALLEG	SAURT009	Atributo Observación	
20	PRESION	SAURT009	Atributo Observación	Atributo que registra la presión registrada en la atención del paciente, este no es capaz de determinar a ningún otro atributo.
21	PULSO	SAURT009	Atributo Observación	Atributo que registra la presión registrada en la atención del paciente, este no es capaz de determinar a ningún otro atributo.
22	TEMPAXILAR	SAURT009	Atributo Observación	Atributo que registra la temperatura axilar registrada en la atención del paciente, este no es capaz de determinar a ningún otro atributo.
23	TEMPRECTAL	SAURT009	Atributo Observación	Atributo que registra la temperatura rectal registrada en la atención del paciente, este no es capaz de determinar a ningún otro atributo.
24	AMBULANCIA	SAURT009	Atributo Observación	Atributo que registra la patente del vehículo de urgencias que realizó el procedimiento previo a la atención del paciente, este no es capaz de determinar a ningún otro atributo.
25	ALCOHOLEMIA	SAURT009	Atributo Observación	Atributo que registra la fecha y hora de alcoholemia registrada en la atención del paciente, este no es capaz de determinar a ningún otro atributo.
26	CARABINERO	SAURT009	Atributo Observación	Atributo que registra el nombre del carabiniere que realizó el procedimiento previo a la atención del paciente, este no es capaz de determinar a ningún otro atributo.

**Tabla 6.11:** Registro de atributos a eliminar.

Luego de la limpieza de datos, las entidades son respaldadas en la propia base de datos. Esta acción tiene sustento en que éstas serán utilizadas posteriormente en la etapa de Generación. A continuación, y tal como lo indica EOG, se identifican los atributos de acuerdo a los criterios establecidos para la poda de atributos. En consecuencia, la Tabla 6.11, muestra los atributos que serán podados, así como la entidad a la que pertenece, la clasificación de la poda y la justificación de la eliminación.

Luego de la identificación de atributos, se concreta la eliminación de estos atributos por medio de instrucciones SQL.

En conclusión, las entidades quedan conformadas con una menor cantidad de atributos, lo que facilita la obtención de DFs de manera automática.

### **6.3.3.2 Obtención Automática DFs**

En general, la obtención automática de DFs es un proceso complejo, debido, en gran medida, a la cantidad excesiva de dependencias encontradas, que regularmente son propias de los datos y no del dominio de las entidades. Es decir, dichas DFs no siempre satisfacen el dominio de las entidades. En consecuencia, la labor de los expertos resulta relevante para refutarlas. La poda de atributos mediante la identificación de DF preexistentes resulta esencial debido a que se constató que al procesar una entidad que posee 38 atributos y 700 tuplas de manera automática mediante la herramienta TANE-Java (RAFREIDER 2008), no se lograron obtener DFs, producto que consume la totalidad de la memoria de un equipo que cuenta con 4GB de RAM en un tiempo inferior a dos horas, y un procesador Intel Dual Core.

En consecuencia, todas las acciones previas propuestas por EOG para la obtención automática de DFs facilitan la ejecución del proceso mediante la disminución de atributos y limpieza de datos, y las acciones posteriores permiten la depuración de las DFs encontradas. A continuación, se describen en detalle las actividades propias de la obtención automática de DFs.

#### **6.3.3.2.1 Procesamiento Automático**

El análisis de las entidades se llevó a cabo por medio de la herramienta TANE-Java (RAFREIDER 2008), de manera individual para cada una de ellas. Un resumen de los resultados de dicho procesamiento se muestran en la Tablas 6.12, 6.13 y 6.14, los detalles se encuentran en Anexo C.

El gran volumen de DFs encontradas, indica que muchas de estas probablemente no correspondan a la realidad puesto que sólo existen en los datos, pero no necesariamente en el dominio

de la entidad. Por ejemplo, como resultado, se aprecia en la Tabla 6.12. APELLMAT-> FONO\_INSTITUCION, lo que evidentemente no es real, debido a que el teléfono de una institución no es determinado por el apellido de un paciente. Consecuentemente, estas DFs son revisadas en la siguiente actividad.

N°	Determinante	->	Determinado
1	APELLMAT	->	FONO_INSTITUCION
2	APELLMAT	->	INSTITUCION
4	APELLMAT	->	RUT_INSTITUCION
5	APELLMAT, APELLPAT, COMUNA, RUT, SEXO	->	FECHAFALL

**Tabla 6.12:** Ejemplo de DFs obtenidas automáticamente desde tabla SAPAT001.

Las 266 DFs de la tabla SAPAT001 fueron obtenidas en tan solo 19,2 segundos. La Tabla 6.15 muestra sólo un ejemplo. Para mayor detalle consultar el Anexo C. Dicha tabla evidencia que muchas de ellas no corresponden al dominio de la entidad.

La Tabla 6.13 muestra un ejemplo de las DFs obtenidas para la tabla SABST030.

N°	Determinante	->	Determinado
1	AFP	->	DAFP
2	AFP, CARGO, DISAPRE, TOTAL_HABERES	->	DREGION
3	AFP, CARGO, DISAPRE, TOTAL_HABERES	->	REGION
4	AFP, CARGO, DJURIDICA, TOTAL_HABERES	->	DREGION

**Tabla 6.13:** Ejemplo de DFs obtenidas Automáticamente desde tabla SABST030.

Los resultados obtenidos en la tabla anterior dan cuenta de la gran cantidad de DFs obtenidas automáticamente, debido a que muchas de ellas sólo se satisfacen en los datos.

La tabla 6.14 exhibe una muestra las DFs obtenidas para la tabla SAURT009. Para mayor detalle consultar el Anexo C.

N°	Determinante	->	Determinado
1	CODIGO	->	CP
2	CODIGO	->	DDIAGPOS1
3	CODIGO	->	DDIAGPOS2
4	CODIGO	->	DFP
5	CODIGO	->	DIAGPOS1

**Tabla 6.14:** Ejemplo de DFs obtenidas Automáticamente desde tabla SAURT009.

En último lugar, las DFs obtenidas de manera automática mediante la herramienta son el insumo suficiente para continuar con la actividad de depuración. La Tabla 6.15 muestra un resumen que indica la cantidad de DFs encontradas para cada entidad.

Entidad	Cantidad
SAPAT001	266
SABST030	294
SAURT009	345

**Tabla 6.15:** Resumen de DFs obtenidas por entidad.

### 6.3.3.2.2 Depuración de DFs

Actividad que permitió refutar muchas DFs que aparecieron producto de los datos, que no pertenecen al dominio de los atributos ni entidades. En consecuencia, el establecimiento de DFs preexistentes cobra una relevancia adicional, debido a que queda expresado el conocimiento específico sobre las entidades, dicha información sumada a las fuentes de información, son suficientes para refutar las DFs.

El análisis se llevó a cabo manualmente por los expertos utilizando la herramienta DBNormalizer desarrollada por (FESENMEYER 2008), es decir, se examinaron las DFs obtenidas previamente con el objeto de verificar su satisfacción en el dominio. Esto es factible debido a que la herramienta propicia la edición de DFs por medio de una interfaz de usuario amigable y adecuada que facilita la incorporación y exclusión de DFs. Los resultados de este proceso se muestran en la Tabla 6.16, 6.17 y 6.18.

Argumento	Dependencias Funcionales
Atributo determinante no determina al atributo determinado	101,102,103,104,116,117,118,119,130,132,133,134,147,148,150,151,188,189,190,191,192,197,198,199,200,201,233,234,235,236,237,239,260,262
El atributo determinante no puede determinar a otro atributo	1,2,3,4,45,46,47,48,155,156,157,158,228,229,230,231,263,264,265,266
El conjunto de atributos determinantes no determinan otro atributo	5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,105,106,107,108,109,110,111,112,113,114,120,121,122,123,124,125,126,127,128,129,135,136,137,138,139,140,141,142,143,144,145,146,152,153,154,159,160,161,162,163,164,165,166,167,168,169,170,171,172,173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,193,194,195,196,202,203,224

**Tabla 6.16:** DFs Eliminadas en tabla SAPAT001.

El criterio de eliminación de DFs se ajustó a un análisis semántico de los expertos, quienes determinaron de acuerdo a su conocimiento específico tomando en consideración los argumentos propuestos en el marco de trabajo y la revisión de las fuentes de información y DFs preexistentes.

En particular, la eliminación de DFs para la tabla SAPAT001 se concentró esencialmente en aquellas DFs que en su determinante tenían más de un atributo. Esto, debido a que la combinación de estos no satisfacía el criterio técnico ni el dominio de las entidades.

A continuación en la Tabla 6.17 se muestran las DFs eliminadas para la tabla SABST030.

<b>Argumento</b>	<b>Dependencias Funcionales</b>
El atributo determinante no puede determinar a otro atributo	265,266
El conjunto de atributos determinantes no determinan otro atributo	2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,108,109,110,111,113,114,115,116,117,118,119,120,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,144,145,146,147,149,150,151,152,153,154,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,172,173,174,175,176,177,178,179,180,181,188,189,190,191,192,193,194,195,197,198,199,200,201,202,203,204,205,206,207,209,210,211,212,213,214,215,216,217,222,223,224,225,226,227,228,229,231,232,233,234,235,236,237,238,239,240,241,242,243,245,246,248,249,250,251,252,254,255,256,257,258,259,260,267,268,269,270,271,272,273

**Tabla 6.17:** DFs Eliminadas en tabla SABST030.

Al igual que la entidad SAPAT001, las DFs eliminadas para la tabla SABST030 son en su mayoría aquellas que el conjunto de atributos determinantes no son capaces de determinar otro atributo. Por ejemplo, DAFFP, DPROVINCIA, PLANTA, REMUNERACION -> CARGO indica que el nombre de la AFP, provincia, planta y remuneración de un funcionario, determina el cargo, lo que en base a la realidad de la institución no corresponde. A continuación, la Tabla 6.18 indica las DFs eliminadas para la tabla SAURT009.

<u>Argumento</u>	<u>Dependencias funcionales</u>
Atributo determinante no determina al atributo determinado	27,28,166,167,276,277,286,287,292,294,295,296,298,299,301,302,304,305,307,309,311,314,316,317,319,321,324,325,331,332,334,337,339,341,343
El conjunto de atributos determinantes no determinan otro atributo	29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,120,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,154,155,156,157,158,159,160,161,162,163,164,168,169,170,171,172,173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,208,209,210,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,226,227,228,229,230,231,232,233,234,235,236,237,238,239,240,241,242,243,244,245,246,247,248,249,250,251,252,253,254,255,256,257,258,259,260,261,262,263,264,265,266,267,268,269,270,271,272,273,278,279,280,281,282,283,284,288,289,290,291,308,312,313,326,327,328,329

**Tabla 6.18:** DFs Eliminadas en tabla SAURT009.

En definitiva la Tabla 6.18 deja en evidencia que la tendencia de eliminación de DFs tiene relación con aquellas en que los atributos determinantes no son capaces de determinar algún atributo. Tal es el caso de DDIAGPOS1, FP, MEDIO, PROFEJE, SERVSOL->DMOTIVO, en que se indica que el conjunto de atributos; diagnóstico, forma de pago, medio de llegada, profesional ejecutante y servicio solicitado determinan el motivo de consulta. Claramente esta DFs satisfacen el dominio de los datos pero no de la entidad. Sin embargo, podría darse este caso para otro modelo de negocio.

Luego de aplicar el criterio experto, basado en el conocimiento previo sobre las DFs preexistentes y la refutación de las DFs generadas automáticamente, se reincorporaron los atributos que previamente fueron podados. Dicha agregación se efectuó mediante la ejecución de dos acciones. La primera tiene relación con modificación de los documentos XML obtenidos automáticamente desde Tane-Java (RAFREIDER 2008) que contienen las DFs que luego son analizadas por DBNormalizer (FESSENMEYER 2008), teniendo especial cuidado en la sintaxis de estos documentos. Y la segunda, mediante la sustitución de las entidades actuales por los respaldos anteriores.

Posteriormente, las DFs que serán analizadas en la etapa de Generación están expresadas en las Tablas 6.19, 6.20 y 6.21, las que contienen, agregadas al final de éstas, las DFs correspondientes a los atributos podados previamente.

N°	Determinante	->	Determinado	N°	Determinante	->	Determinado
80	CODPACIE	->	APELLMAT	227	INSTITUCION	->	RUT_INSTITUCION
81	CODPACIE	->	APELLPAT	232	OCUPACION	->	DOCUPACION
82	CODPACIE	->	COMUNA	238	RAZSOCIAL_INSTITUCION	->	INSTITUCION
83	CODPACIE	->	DCOMUNA	240	RUT	->	APELLMAT
84	CODPACIE	->	DFORMAPAGO	241	RUT	->	APELLPAT
85	CODPACIE	->	DOCUPACION	242	RUT	->	CODPACIE
86	CODPACIE	->	FACTORRH	243	RUT	->	COMUNA
87	CODPACIE	->	FECHAFALL	244	RUT	->	DCOMUNA
88	CODPACIE	->	FECHANAC	245	RUT	->	DFORMAPAGO
89	CODPACIE	->	FONO_INSTITUCION	246	RUT	->	DOCUPACION
90	CODPACIE	->	FORMAPAGO	247	RUT	->	FACTORRH
91	CODPACIE	->	GRUPSANG	248	RUT	->	FECHAFALL
92	CODPACIE	->	HC	249	RUT	->	FECHANAC
93	CODPACIE	->	INSTITUCION	250	RUT	->	FONO_INSTITUCION
94	CODPACIE	->	NOMBRES	251	RUT	->	FORMAPAGO
95	CODPACIE	->	OCUPACION	252	RUT	->	GRUPSANG
96	CODPACIE	->	RAZSOCIAL_INSTITUCION	253	RUT	->	HC
97	CODPACIE	->	RUT	254	RUT	->	INSTITUCION
98	CODPACIE	->	RUT_INSTITUCION	255	RUT	->	NOMBRES
99	CODPACIE	->	SEXO	256	RUT	->	OCUPACION
100	COMUNA	->	DCOMUNA	257	RUT	->	RAZSOCIAL_INSTITUCION
115	DCOMUNA	->	COMUNA	258	RUT	->	RUT_INSTITUCION
131	DFORMAPAGO	->	FORMAPAGO	259	RUT	->	SEXO
149	DOCUPACION	->	OCUPACION	261	RUT_INSTITUCION	->	INSTITUCION
188	FORMAPAGO	->	DFORMAPAGO	267	CODPACIE	->	FECHACREA
204	HC	->	APELLMAT	268	CODPACIE	->	ESTADO
205	HC	->	APELLPAT	269	CODPACIE	->	DOMICILIO
206	HC	->	CODPACIE	270	CODPACIE	->	TELEFONO
207	HC	->	COMUNA	271	CODPACIE	->	OBSERVACION
208	HC	->	DCOMUNA	272	CODPACIE	->	NOMBREPAD
209	HC	->	DFORMAPAGO	273	CODPACIE	->	NOMBREMAD
210	HC	->	DOCUPACION	274	CODPACIE	->	DV
211	HC	->	FACTORRH	275	HC	->	FECHACREA
212	HC	->	FECHAFALL	276	HC	->	ESTADO
213	HC	->	FECHANAC	277	HC	->	DOMICILIO
214	HC	->	FONO_INSTITUCION	278	HC	->	TELEFONO
215	HC	->	FORMAPAGO	279	HC	->	OBSERVACION
216	HC	->	GRUPSANG	280	HC	->	NOMBREPAD
217	HC	->	INSTITUCION	281	HC	->	NOMBREMAD
218	HC	->	NOMBRES	282	HC	->	DV
219	HC	->	OCUPACION	283	RUT	->	FECHACREA
220	HC	->	RAZSOCIAL_INSTITUCION	284	RUT	->	ESTADO
221	HC	->	RUT	285	RUT	->	DOMICILIO
222	HC	->	RUT_INSTITUCION	286	RUT	->	TELEFONO
223	HC	->	SEXO	287	RUT	->	OBSERVACION
225	INSTITUCION	->	FONO_INSTITUCION	288	RUT	->	NOMBREPAD
226	INSTITUCION	->	RAZSOCIAL_INSTITUCION	289	RUT	->	NOMBREMAD
				290	RUT	->	DV

**Tabla 6.19:** DFs depuradas para entidad SAPAT001.

La tabla 6.19 es resultado de la depuración de DFs y la reincorporación de dependencias que corresponden a atributos eliminados por ser considerados inocuos. Los expertos deciden de acuerdo a

los antecedentes encontrados en las fuentes de información, que cada uno de ellos (atributos) es determinado por el RUT. A continuación, la Tabla 6.20, muestra las DFs que serán evaluadas posteriormente.

<u>Id</u>	<u>Determinante</u>	<u>-&gt;</u>	<u>Determinado</u>	<u>Id</u>	<u>Determinante</u>	<u>-&gt;</u>	<u>Determinado</u>
1	AFP	->	DAFP	264	REGION	->	DREGION
55	CARGO	->	DCARGO	274	RUT	->	AFP
103	COMUNA	->	DCOMUNA	275	RUT	->	CARGO
104	COMUNA	->	DPROVINCIA	276	RUT	->	COMUNA
105	COMUNA	->	DREGION	277	RUT	->	DAFP
106	COMUNA	->	PROVINCIA	278	RUT	->	DCARGO
107	COMUNA	->	REGION	279	RUT	->	DCOMUNA
112	DAFP	->	AFP	280	RUT	->	DISAPRE
148	DCARGO	->	CARGO	281	RUT	->	DJURIDICA
182	DCOMUNA	->	COMUNA	282	RUT	->	DPLANTA
183	DCOMUNA	->	DPROVINCIA	283	RUT	->	DPROVINCIA
184	DCOMUNA	->	DREGION	284	RUT	->	DREGION
185	DCOMUNA	->	PROVINCIA	285	RUT	->	DUNIDAD
186	DCOMUNA	->	REGION	286	RUT	->	ISAPRE
187	DISAPRE	->	ISAPRE	287	RUT	->	JURIDICA
196	DJURIDICA	->	JURIDICA	288	RUT	->	PLANTA
208	DPLANTA	->	PLANTA	289	RUT	->	PROVINCIA
218	DPROVINCIA	->	DREGION	290	RUT	->	REGION
219	DPROVINCIA	->	PROVINCIA	291	RUT	->	REMUNERACION
220	DPROVINCIA	->	REGION	292	RUT	->	TOTAL_HABERES
221	DREGION	->	REGION	293	RUT	->	UNIDAD
230	DUNIDAD	->	UNIDAD	294	UNIDAD	->	DUNIDAD
244	ISAPRE	->	DISAPRE	295	RUT	->	DV
247	JURIDICA	->	DJURIDICA	296	RUT	->	FUNCIONARIO
253	PLANTA	->	DPLANTA	297	RUT	->	DIRECCION
261	PROVINCIA	->	DPROVINCIA	298	RUT	->	GENERO
262	PROVINCIA	->	DREGION	299	RUT	->	FNACIMIENTO
263	PROVINCIA	->	REGION	300	RUT	->	TITULO

**Tabla 6.20:** DFs depuradas para entidad SABST030.

Las DFs agregadas en esta etapa corresponden a atributos inocuos de los cuales existe la certeza que son determinados por el atributo RUT. La Tabla 6.21 muestra las DFs que serán evaluadas posterior para la tabla SAURT009.

N°	Determinante	->	Determinado	N°	Determinante	->	Determinado
1	CODIGO	->	CP	297	DLUGAR	->	LUGAR
2	CODIGO	->	DDIAGPOS1	300	DMEDIO	->	MEDIO
3	CODIGO	->	DDIAGPOS2	303	DMOTIVO	->	MOTIVO
4	CODIGO	->	DFP	306	DPERSTRATANTE	->	PERSTRATANTE
5	CODIGO	->	DIAGPOS1	310	DPROFEJE	->	PROFEJE
6	CODIGO	->	DIAGPOS2	315	DPROFSOL	->	PROFSOL
7	CODIGO	->	DLUGAR	318	DSERVDEST	->	SERVDEST
8	CODIGO	->	DMEDIO	320	DSERVEJE	->	SERVEJE
9	CODIGO	->	DMOTIVO	322	DSERVSOL	->	SERVSOL
10	CODIGO	->	DPERSTRATANTE	323	FP	->	DFP
11	CODIGO	->	DPROFEJE	330	LUGAR	->	DLUGAR
12	CODIGO	->	DPROFSOL	333	MEDIO	->	DMEDIO
13	CODIGO	->	DSERVDEST	335	MOTIVO	->	DMOTIVO
14	CODIGO	->	DSERVEJE	336	PERSTRATANTE	->	DPERSTRATANTE
15	CODIGO	->	DSERVSOL	338	PROFEJE	->	DPROFEJE
16	CODIGO	->	FP	340	PROFSOL	->	DPROFSOL
17	CODIGO	->	GRUPO	342	SERVDEST	->	DSERVDEST
18	CODIGO	->	LUGAR	344	SERVEJE	->	DSERVEJE
19	CODIGO	->	MEDIO	345	SERVSOL	->	DSERVSOL
20	CODIGO	->	MOTIVO	346	CODIGO	->	COMENTARIO
21	CODIGO	->	PERSTRATANTE	347	CODIGO	->	ANULADO
22	CODIGO	->	PROFEJE	348	CODIGO	->	CIERRE
23	CODIGO	->	PROFSOL	349	CODIGO	->	FCREA
24	CODIGO	->	SERVDEST	350	CODIGO	->	FECHALLEG
25	CODIGO	->	SERVEJE	351	CODIGO	->	PRESION
26	CODIGO	->	SERVSOL	352	CODIGO	->	PULSO
165	DDIAGPOS1	->	DIAGPOS1	353	CODIGO	->	TEMPAXILAR
274	DDIAGPOS2	->	DIAGPOS2	354	CODIGO	->	TEMPRECTAL
275	DFP	->	FP	355	CODIGO	->	AMBULANCIA
285	DIAGPOS1	->	DDIAGPOS1	356	CODIGO	->	ALCOHOLEMIA
293	DIAGPOS2	->	DDIAGPOS2	357	CODIGO	->	CARABINERO

**Tabla 6.21:** DFs depuradas para entidad SAURT009

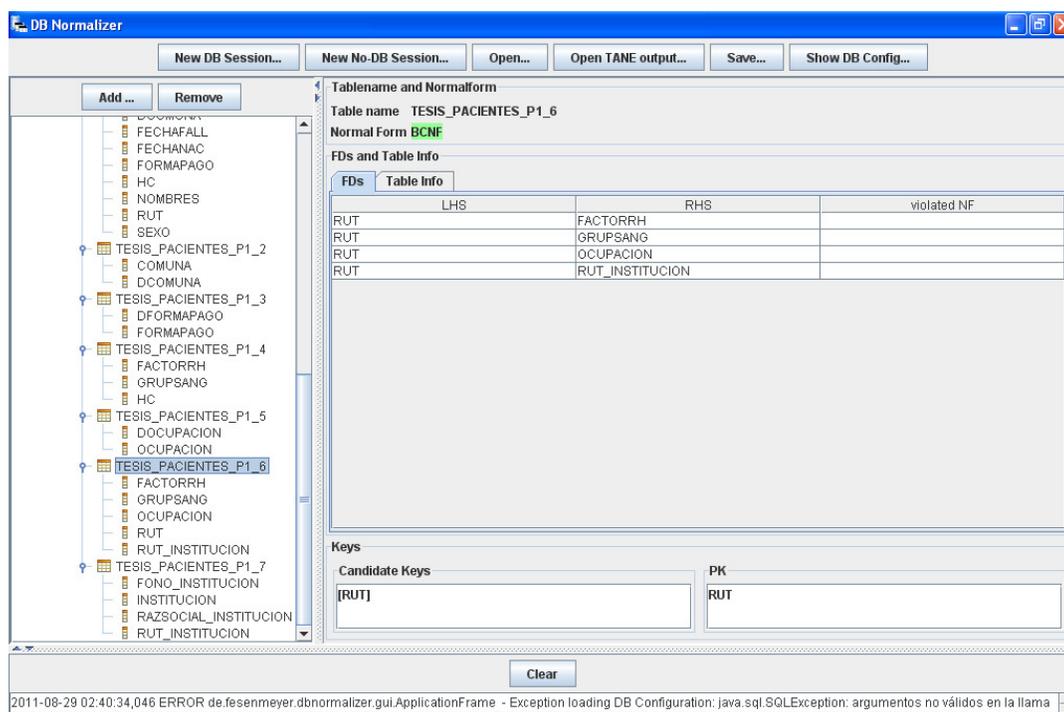
En conclusión, la aplicación de esta etapa permitió obtener las DFs que serán analizadas en la etapa siguiente, incluyendo que los atributos podados previamente por ser considerados inocuos fueron reincorporados en forma de DFs determinados por los atributos indicados por los expertos.

Teniendo las DFs depuradas, están las condiciones suficientes para que sean procesadas con el objeto de obtener un modelo de datos relacional.

#### 6.3.4 Etapa de Generación

La generación del modelo se realizó mediante un proceso automático utilizando la misma herramienta en la que se visualizaron y refinaron las DFs. Este es un proceso mecánico que genera un documento SQL como resultado. Posteriormente, se debe evaluar. Lo interesante de esta etapa es que los cambios realizados en las DFs o en el ADI generan un efecto inmediato en el modelo de datos

propuesto. La Figura 6.3 muestra las entidades planteadas para el modelo de datos relacional. En cada una se muestra un detalle de las DFs.



**Figura 6.3:** DFs que generan nuevas entidades.

### 6.3.4.1 Generación automática del modelo

La generación del modelo de datos se realiza por medio del procesamiento de las DFs depuradas previamente. En consecuencia, la labor de proponer un modelo de datos, es mecanizada y realizada por la herramienta, sin embargo requiere del control y supervisión del equipo de trabajo y expertos, una vez realizado un análisis de los resultados preliminares.

La dinámica de trabajo en la herramienta permite ir iterando el proceso de depuración de las DFs y posterior generación del modelo. De este modo, la actividad de refinamiento de esta etapa se realiza implícitamente en conjunto con la generación automática del modelo. En definitiva, sólo resta exportar las sentencias SQL hacia un SGBD. En las Tablas 6.22, 6.23 y 6.24 se muestran las instrucciones SQL que componen el modelo preliminar propuesto por DbNormalizer (FESENMEYER 2008) en base a las modificaciones iterativas que se realizan sobre las DFs.

SAPAT001	
<pre>-- create new tables (without FK-Constraints)  CREATE TABLE HR.SAPAT001_1 ( COMUNA VARCHAR2(60) PRIMARY KEY, DCOMUNA VARCHAR2(30) NOT NULL, UNIQUE (DCOMUNA));  CREATE TABLE HR.SAPAT001_2 ( FORMAPAGO VARCHAR2(60) PRIMARY KEY, DFORMAPAGO VARCHAR2(60) NOT NULL, UNIQUE (DFORMAPAGO));  CREATE TABLE HR.SAPAT001_3 ( FONO_INSTITUCION VARCHAR2(20), INSTITUCION VARCHAR2(60) PRIMARY KEY, RAZSOCIAL_INSTITUCION VARCHAR2(60), RUT_INSTITUCION VARCHAR2(20) NOT NULL, UNIQUE (RAZSOCIAL_INSTITUCION), UNIQUE (RUT_INSTITUCION));  CREATE TABLE HR.SAPAT001_4 ( DOCUPACION VARCHAR2(30) NOT NULL, OCUPACION VARCHAR2(60) PRIMARY KEY, UNIQUE (DOCUPACION));  CREATE TABLE HR.SAPAT001_5 ( APELLMAT VARCHAR2(60) NOT NULL, APELLPAT VARCHAR2(60) NOT NULL, CODPACIE VARCHAR2(50) PRIMARY KEY,</pre>	<pre>COMUNA VARCHAR2(30), DOMICILIO VARCHAR2(60), DIGIVER VARCHAR2(1), ESTADO VARCHAR2(1), FACTORRH VARCHAR2(60), FECHACREA VARCHAR2(60), FECHAFALL VARCHAR2(60), FECHANAC VARCHAR2(60), FORMAPAGO VARCHAR2(60), GRUPSANG VARCHAR2(60), HC VARCHAR2(50), NOMBRES VARCHAR2(60), NOMBREPAD VARCHAR2(60), NOMBREMAD VARCHAR2(60), OBSERVACION VARCHAR2(200), OCUPACION VARCHAR2(60), RUT VARCHAR2(50), INSTITUCION VARCHAR2(60), SEXO VARCHAR2(60), TELEFONO VARCHAR2 (20), UNIQUE (HC), UNIQUE (RUT));  ALTER TABLE HR.SAPAT001_5 ADD FOREIGN KEY (COMUNA) REFERENCES HR.SAPAT001_1 (COMUNA);  ALTER TABLE HR.SAPAT001_5 ADD FOREIGN KEY (FORMAPAGO) REFERENCES HR.SAPAT001_2 (FORMAPAGO);  ALTER TABLE HR.SAPAT001_5 ADD FOREIGN KEY (INSTITUCION) REFERENCES HR.SAPAT001_3 (INSTITUCION);  ALTER TABLE HR.SAPAT001_5 ADD FOREIGN KEY (OCUPACION) REFERENCES HR.SAPAT001_4 (OCUPACION);</pre>

**Tabla 6.22:** Modelo de datos DDL entidad SAPAT001

SABST030	
<pre>CREATE TABLE HR.SABST030_1 ( AFP VARCHAR2(255) PRIMARY KEY, DAFP VARCHAR2(255) NOT NULL,UNIQUE (DAFP));  CREATE TABLE HR.SABST030_10 (DUNIDAD VARCHAR2(255) NOT NULL, UNIDAD VARCHAR2(255) PRIMARY KEY,UNIQUE (DUNIDAD));  CREATE TABLE HR.SABST030_2 (CARGO VARCHAR2(255) PRIMARY KEY, DCARGO VARCHAR2(255) NOT NULL,UNIQUE (DCARGO));  CREATE TABLE HR.SABST030_3 (COMUNA VARCHAR2(255) PRIMARY KEY, DCOMUNA VARCHAR2(255) NOT NULL,PROVINCIA VARCHAR2(255), UNIQUE (DCOMUNA));  CREATE TABLE HR.SABST030_4 (DISAPRE VARCHAR2(255) NOT NULL, ISAPRE VARCHAR2(255) PRIMARY KEY,UNIQUE (DISAPRE));  CREATE TABLE HR.SABST030_5 (DJURIDICA VARCHAR2(255) NOT NULL, JURIDICA VARCHAR2(255) PRIMARY KEY,UNIQUE (DJURIDICA));  CREATE TABLE HR.SABST030_6 (DPLANTA VARCHAR2(255) NOT NULL, PLANTA VARCHAR2(255) PRIMARY KEY,UNIQUE (DPLANTA));</pre>	<pre>CREATE TABLE HR.SABST030_7 (DPROVINCIA VARCHAR2(255) NOT NULL,PROVINCIA VARCHAR2(255) PRIMARY KEY, REGION VARCHAR2(255) NOT NULL,UNIQUE (DPROVINCIA));  CREATE TABLE HR.SABST030_9 (DV VARCHAR2(1), DAFP VARCHAR2(255),DIRECCION VARCHAR2(60), DCARGO VARCHAR2(255),DCOMUNA VARCHAR2(255), FNACIMIENTO VARCHAR2(15),FUNCIONARIO VARCHAR2(100), GENERO VARCHAR2(1),ISAPRE VARCHAR2(255), JURIDICA VARCHAR2(255),PLANTA VARCHAR2(255), REMUNERACION VARCHAR2(255),RUT VARCHAR2(255) PRIMARY KEY, TITULO VARCHAR2(100), TOTAL_HABERES VARCHAR2(255), UNIDAD VARCHAR2(255));  ALTER TABLE HR.SABST030_1 ADD FOREIGN KEY (AFP) REFERENCES HR.SABST030_1 (AFP);  ALTER TABLE HR.SABST030_1 ADD FOREIGN KEY (UNIDAD) REFERENCES HR.SABST030_10 (UNIDAD);  ALTER TABLE HR.SABST030_1 ADD FOREIGN KEY (CARGO) REFERENCES HR.SABST030_2 (CARGO);  ALTER TABLE HR.SABST030_1 ADD FOREIGN KEY (COMUNA) REFERENCES HR.SABST030_3 (COMUNA);  ALTER TABLE HR.SABST030_1 ADD FOREIGN KEY (ISAPRE) REFERENCES HR.SABST030_4 (ISAPRE);  ALTER TABLE HR.SABST030_1 ADD FOREIGN KEY (JURIDICA) REFERENCES HR.SABST030_5 (JURIDICA);  ALTER TABLE HR.SABST030_1 ADD FOREIGN KEY (PLANTA) REFERENCES HR.SABST030_6 (PLANTA);  ALTER TABLE HR.SABST030_1 ADD FOREIGN KEY (PROVINCIA) REFERENCES HR.SABST030_7 (PROVINCIA);</pre>

**Tabla 6.23:** Modelo de datos DDL entidad SABST030

<b>SAURT009</b>		
<pre>CREATE TABLE HR.SAURT009_1 (   AMBULANCIA NUMBER (5,0),   ALCOHOLEMIA NUMBER (5,0),   ANULADO NUMBER(5,0),   CARABINERO NUMBER (5,0), CIERRE DATE,   CODIGO NUMBER(10) PRIMARY KEY,   DIAGPOS1 NUMBER(10), DIAGPOS2   NUMBER(10),   FP NUMBER(10) NOT NULL, FCREA DATE,   FECHALLEG DATE, LUGAR NUMBER(10),   MEDIO NUMBER(10),   MOTIVO NUMBER(10) NOT NULL,   PERSTRATANTE NUMBER(10),   PRESION CHAR(10), PULSO NUMBER (5,0),   PROFEJE NUMBER(10) NOT NULL,   PROFSOL NUMBER(10) NOT NULL,   SERVDEST NUMBER(10),   SERVEJE NUMBER(10) NOT NULL,   SERVSOL NUMBER(10) NOT NULL,   TEMPAXILAR NUMBER (3,1),   TEMPRECTAL NUMBER (3,1));  CREATE TABLE HR.SAURT009_10 ( DPROFSOL   VARCHAR2(200) NOT NULL, PROFSOL   NUMBER(10) PRIMARY KEY, UNIQUE   (DPROFSOL));  CREATE TABLE HR.SAURT009_11 ( DSERVDEST   VARCHAR2(30) NOT NULL, SERVDEST   NUMBER(10) PRIMARY KEY, UNIQUE   (DSERVDEST));  CREATE TABLE HR.SAURT009_12 ( DSERVEJE   VARCHAR2(30) NOT NULL, SERVEJE   NUMBER(10) PRIMARY KEY, UNIQUE   (DSERVEJE));</pre>	<pre>CREATE TABLE HR.SAURT009_13 (   DSERVSOL VARCHAR2(30) NOT   NULL, SERVSOL NUMBER(10)   PRIMARY KEY , UNIQUE   (DSERVSOL));  CREATE TABLE HR.SAURT009_2 (   DDIAGPOS1 VARCHAR2(60) NOT   NULL, DIAGPOS1 NUMBER(10)   PRIMARY KEY, UNIQUE   (DDIAGPOS1));  CREATE TABLE HR.SAURT009_3 (   DDIAGPOS2 VARCHAR2(60) NOT   NULL, DIAGPOS2 NUMBER(10)   PRIMARY KEY, UNIQUE   (DDIAGPOS2));  CREATE TABLE HR.SAURT009_4 (   DFP VARCHAR2(30) NOT NULL NOT   NULL, FP NUMBER(10) PRIMARY   KEY, UNIQUE (DFP));  CREATE TABLE HR.SAURT009_5 (   DLUGAR VARCHAR2(30) NOT NULL,   LUGAR NUMBER(10) PRIMARY KEY,   UNIQUE (DLUGAR));  CREATE TABLE HR.SAURT009_6 (   DMEDIO VARCHAR2(30) NOT NULL,   MEDIO NUMBER(10) PRIMARY KEY,   UNIQUE (DMEDIO));  CREATE TABLE HR.SAURT009_7 (   DMOTIVO VARCHAR2(30) NOT   NULL, MOTIVO NUMBER(10)   PRIMARY KEY, UNIQUE   (DMOTIVO));  CREATE TABLE HR.SAURT009_8 (   DPERSTRATANTE VARCHAR2(200)   NOT NULL, PERSTRATANTE   NUMBER(10) PRIMARY KEY,   UNIQUE (DPERSTRATANTE));</pre>	<pre>CREATE TABLE HR.SAURT009_9 (   DPROFEJE VARCHAR2(200) NOT NULL,   PROFEJE NUMBER(10) PRIMARY KEY,   UNIQUE (DPROFEJE));  ALTER TABLE HR. SAURT009_1 ADD FOREIGN   KEY (PROFSOL) REFERENCES HR.SAURT009_10   (PROFSOL);  ALTER TABLE HR. SAURT009_1 ADD FOREIGN   KEY (SERVDEST) REFERENCES HR.SAURT009_11   (SERVDEST);  ALTER TABLE HR. SAURT009_1 ADD FOREIGN   KEY (SERVEJE) REFERENCES HR.SAURT009_12   (SERVEJE);  ALTER TABLE HR. SAURT009_1 ADD FOREIGN   KEY (SERVSOL) REFERENCES HR.SAURT009_13   (SERVSOL);  ALTER TABLE HR. SAURT009_1 ADD FOREIGN   KEY (DIAGPOS1) REFERENCES HR.SAURT009_2   (DIAGPOS1);  ALTER TABLE HR. SAURT009_1 ADD FOREIGN   KEY (DIAGPOS2) REFERENCES HR.SAURT009_3   (DIAGPOS2);  ALTER TABLE HR. SAURT009_1 ADD FOREIGN   KEY (FP) REFERENCES HR.SAURT009_4 (FP);  ALTER TABLE HR. SAURT009_1 ADD FOREIGN   KEY (LUGAR) REFERENCES HR.SAURT009_5   (LUGAR);  ALTER TABLE HR. SAURT009_1 ADD FOREIGN   KEY (MEDIO) REFERENCES HR.SAURT009_6   (MEDIO);  ALTER TABLE HR. SAURT009_1 ADD FOREIGN   KEY (MOTIVO) REFERENCES HR.SAURT009_7   (MOTIVO);  ALTER TABLE HR. SAURT009_1 ADD FOREIGN   KEY (PERSTRATANTE) REFERENCES   HR.SAURT009_8 (PERSTRATANTE);  ALTER TABLE HR. SAURT009_1 ADD FOREIGN   KEY (PROFEJE) REFERENCES HR.SAURT009_9   (PROFEJE);</pre>

**Tabla 6.24:** Modelo de datos DDL entidad SAURT009

Para finalizar, las instrucciones SQL son ejecutadas en Oracle 10G EE lo que entrega como resultado un modelo de datos relacional, tal como se muestra en la Figura 6.4. Dicho modelo contempla la creación de entidades y la relación entre las mismas.

El modelo propuesto para éste caso de estudio requirió de algunas modificaciones en las DFs, debido a que se observaron atributos y entidades que no se correspondían con la realidad. Dichos cambios se efectuaron en la misma herramienta y se evaluaron sus efectos en el modelo propuesto de inmediato.

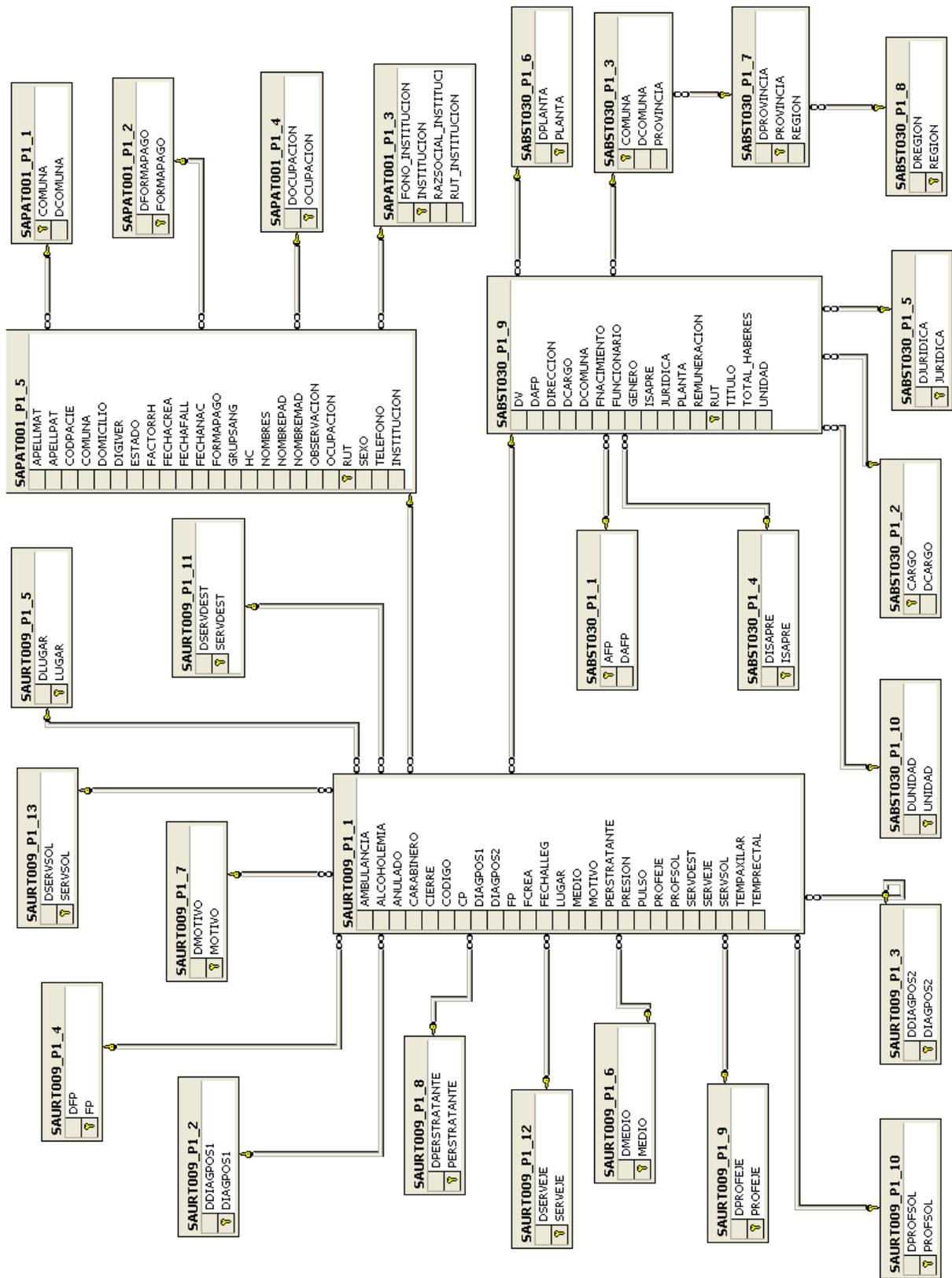


Figura 6.4: Diagrama de datos relacional propuesto

## 6.4 Conclusiones del Capítulo

Durante este último capítulo se ha aplicado un caso de estudio basado en la realidad de un hospital público en Chile, el que tuvo como resultado la generación de un nuevo sistema de información más basado en un modelo de datos normalizado, en una nueva plataforma y conforme a los requerimientos actuales. No obstante, lo anterior por razones de rendimiento y tiempos de respuesta es necesario mantener un modelo o parte de este desnormalizado.

El hecho de actualizar un SIH pareciera una situación normal y cotidiana. No obstante, su aplicación es mucho más compleja, debido a la heterogeneidad de fuentes de datos, escaso acceso a fuentes de información y errores en los datos producto de una baja mantención de los mismos. Además, de la falta de guías de acción comprobadas que faciliten la tarea.

La aplicación de EOG en un caso real, generó de manera natural a realizar ajustes en las actividades planteadas inicialmente en el marco de trabajo. Esta situación permitió evolucionar e interiorizarnos mayormente sobre detalles propios de aplicaciones heredadas que utilizan bases de datos no relacionales. Dichos ajustes tienen relación con la creación y unión de actividades en cada una de las etapas del marco de trabajo.

EOG facilitó la obtención de un nuevo modelo de datos relacional en 3FN en un tiempo inferior al estimado inicialmente en el proyecto de desarrollo de la nueva aplicación, debido a que propuso un nuevo modelo a partir de la fuente de datos del SIH y no requirió mayores recursos para su obtención. Los recursos utilizados fueron sólo Horas Hombre (HH) de expertos, equipo de trabajo y un computador personal que permitió analizar los datos. El tiempo total del equipo de trabajo en la aplicación de EOG consumió 160HH, aproximadamente, es decir, un 20% menor que las 200HH planificadas inicialmente en el proyecto. Por otro lado, el tiempo de computación consumido por máquina para la obtención de DFs de las tres entidades no superó 5 minutos, gracias a la poda de atributos inocuos. Del mismo modo, la generación del modelo de datos en la aplicación resultó muy sencilla y con un consumo de tiempo computacional que no superó los dos minutos<sup>5</sup>.

Para concluir, el modelo de datos planteado por EOG fue una base importante para continuar con el desarrollo del nuevo sistema, por lo que su generación se transformó en un gran aporte en la etapa de diseño.

---

<sup>5</sup> Utilizando un computador con procesador Pentium Dual Core T4300 2.10Ghz, 4GB Ram y sistema operativo Windows 7 32 bits.

## **Capítulo 7**

### **Conclusiones**

---

Sin duda alguna, los SIH son una problemática para organizaciones de diversas áreas y, en particular, para aquellas cuya evolución natural los ha llevado a evaluar el cambio de éstos debido, en general, a la falta de adaptabilidad del sistema a los nuevos requisitos.

Conforme a lo anterior, y tal como se ha descrito en esta investigación, las organizaciones han recurrido a diferentes alternativas para hacer frente a los SIH, entre ellas reingeniería, abandono, o soluciones híbridas. Las investigaciones estudiadas han permitido tener una clara concepción sobre los métodos que facilitan hacer frente a la problemática. Sin embargo, y a lo largo de éste estudio, se ha detectado la falta de guías de acción o marcos de trabajo que indiquen claramente a través de etapas, procedimientos y herramientas comprobadas mediante un caso de estudio, el cómo obtener un modelo de datos relacional a partir desde SIH.

Considerando los antecedentes, esta investigación se centró esencialmente en generar una contribución hacia la comunidad científica mediante el desarrollo de un marco de trabajo validado mediante un caso de estudio aplicado en el HCHM. Específicamente, en la actualización del módulo de emergencia del Sistema de Información Hospitalario.

El camino recorrido durante esta investigación ha permitido obtener la experiencia necesaria para concluir qué aspectos tienen una preponderancia sobre otros para obtener resultados óptimos.

Tomando en cuenta lo anterior, es posible indicar que la primera etapa es meramente mecánica y no representa mayor complejidad, puesto que, en rigor, se realiza la extracción de la fuente de datos del SIH hacia una estructura de datos intermedia que opera como almacén de datos, en la cual se realizan operaciones, que facilitan la obtención de DFs. Este último punto es de suma relevancia, debido al grado de desarrollo de ciertos SIH y la disponibilidad de fuentes de información, particularmente expertos, quienes son un pilar fundamental para el éxito en la ejecución de las dos últimas etapas.

La etapa de obtención de DFs no sólo facilita la generación de las mismas de manera automática, sino que permite establecer las bases sobre el dominio del problema. Es decir, la obtención de DFs preexistentes es un análisis semántico realizado por los expertos y grupo de trabajo tendiente a establecer de una manera clara el ámbito de los datos. Dicho proceso termina siendo más relevante aún, puesto que dichas DFs deben ser comprobadas o satisfechas en los datos contenidos en el ADI, y es aquí donde es necesario aplicar conocimientos específicos de limpieza de datos que facilitarán la satisfacción de las DFs encontradas en los datos.

Más aún, la eliminación de atributos considerados inocuos facilita enormemente la labor de obtención de DFs de manera automática, las pruebas realizadas indican que, para grandes entidades, es imposible obtener resultados manejables en la cantidad de DFs encontradas y en un tiempo razonable, de este modo la poda se transforma en un elemento clave para lograr este objetivo.

Concretamente, la obtención de DFs sin considerar la limpieza de datos y la poda de atributos es compleja, debido a que, tal como se ha mencionado durante esta investigación, los datos en los SIH generalmente presentan errores, por lo que si se evalúan las entidades originalmente sin procesarlas previamente, los resultados son absolutamente alejados de la realidad en tiempos extremadamente extensos.

Por otro lado, la obtención de DFs de modo automático genera una gran cantidad de DFs que son propias de los datos y no del dominio de los atributos y entidades. Por esta razón, se hace necesaria la depuración de éstas por parte de expertos y del equipo de trabajo.

Esta visión se contrapone a algunos estudios revisados previamente quienes proponen un método mecánico con mínima intervención humana donde los algoritmos tomen decisiones de manera automatizada. En la generalidad de los casos, la realidad dista de lo planteado, por lo que los datos deben ser tratados previamente.

La generación del modelo de datos es resultado de procesos iterativos, actividades que se pueden ejecutar varias veces hasta obtener un resultado óptimo, es decir, más cercano a la realidad. Por otro lado, las herramientas han jugado un rol fundamental en la ejecución del marco de trabajo.

El modelo obtenido como resultado posee una mayor cantidad de entidades que el SIH original. De este modo, el nuevo modelo de datos satisface reglas relacionales. Sin embargo se ha planteado la inquietud sobre el cómo migrar desde el SIH hacia el nuevo modelo. Esta es, quizás, una motivación de trabajo que permitiría complementar esta investigación como un trabajo futuro.

Por otro lado, surge la necesidad de obtener herramientas ad-hoc mayormente integradas que faciliten la ejecución del marco de trabajo, proveer de una herramienta que contemple todas las etapas del marco de trabajo en que se puedan considerar las contribuciones de expertos.

En último lugar, la aplicación de este marco de trabajo en un caso de estudio reportó beneficios hacia el HCHM, puesto que hoy se encuentra en funcionamiento una nueva aplicación de la unidad de emergencia en que se utilizó el marco de trabajo para obtener un modelo de datos relacional.

## **Contraste de Resultados**

En el transcurso de nuestra investigación se desarrolló un artículo denominado “Definición De Un Marco De Trabajo Para La Obtención De Un Modelo De Base De Datos Relacional Desde Sistemas Heredados” con el que se participó en el Encuentro Chileno de Tesistas de Magister y Doctorado en el marco de las Jornadas Chilenas de Computación en la ciudad de Antofagasta en el año 2010.

## **Trabajo Futuro**

Durante la investigación se han planteado diversas inquietudes que podrían derivar en uno o varias líneas de investigación futura. En particular, el modelo de datos relacional en 3FN obtenido a través de EOG posee una mayor cantidad de entidades que las del SIH original, las que se deben poblar con datos provenientes desde el propio SIH. Por otro lado, emerge la necesidad de obtener herramientas ad-hoc mayormente integradas que faciliten la ejecución del marco de trabajo, esto es, proveer de una herramienta que contemple todas las etapas de EOG, incluyendo las contribuciones de expertos.

## **Capítulo 8**

### **Bibliografía**

---

## Referencias

- ABDELSALAM, M., A. AKHTAR, et al. (2008). Relational Database Migration: A Perspective. Proceedings of the 19th international conference on Database and Expert Systems Applications. Turin, Italy, Springer-Verlag.
- ALHAJJ, R. (2003). "Extracting the extended entity-relationship model from a legacy relational database\* 1." Information Systems **28**(6): 597-618.
- ALKHATIB, G. (1992). "The maintenance problem of application software: An empirical analysis." Journal of Software Maintenance: Research and Practice **1**(2): 83-104.
- ANDERSSON, M. (1994). "Extracting an entity relationship schema from a relational database through reverse engineering." Entity-Relationship Approach—ER'94 Business Modelling and Re-Engineering: 403-419.
- AVERSANO, L., G. CANFORA, et al. (2001). Migrating legacy systems to the web: an experience report, Published by the IEEE Computer Society.
- BAIXERIES, J. (2004). A formal concept analysis framework to mine functional dependencies. Workshop on mathematical mehtods for learning. Italy.
- BELL, S. and P. BROCKHAUSEN (1995). Discovery of data dependencies in relational databases, Universitat Dortmund.
- BERGEY, J., L. NORTHROP, et al. (1997). Enterprise Framework for the Disciplined Evolution of Legacy Systems. Pittsburgh, Software Engineering Institute, Carnegie Mellon University.
- BISBAL, J., D. LAWLESS, et al. (1999). "Legacy information system migration: A brief review of problems, solutions and research issues." Computer Sciences Department, Trinity College, Dublin, Ireland.
- BISBAL, J., D. LAWLESS, et al. (1999). "Legacy information systems: Issues and directions." IEEE software **16**(5): 103-111.
- BISBAL, J., D. LAWLESS, et al. (1997). An overview of legacy information system migration. Software Engineering Conference, Australian.
- BITTON, D., J. MILLMAN, et al. (1989). A feasibility and performance study of dependency inference [database design], IEEE.
- BODHUIN, T., E. GUARDABASCIO, et al. (2003). Migrating COBOL Systems to the WEB by using the MVC design pattern, IEEE.
- BORONAT, A., J. PÉREZ, et al. (2004). "Two experiences in software dynamics." Journal of Universal Computer Science **10**(4): 428–453.
- BRODIE, M. and M. STONEBRAKER (1998). Migrating legacy systems: gateways, interfaces & the incremental approach.
- CASTELLANOS, M. and F. SALTOR (1993). "Extraction of data dependencies." Jaakkola, Kangassalo, Kitahashi, and Markus, editors, Information Modelling and Knowledge Bases **3**: 400-420.
- CERVIÑO, M. and G. MORO (2009). Análisis Crítico De Metodologías Usadas En El Proceso De Reingeniería De Aplicaciones Orientadas A Objetos. Facultad de Ingeniería, Universidad Austral: 71.
- CIMITILE, A., A. FASOLINO, et al. (1997). Legacy Systems Assessment to Support Decision Making, Citeseer.
- CODD, E. F. (1970). "A relational model of data for large shared data banks." Communications of the ACM **13**(6): 377-387.
- COLOSIMO, M., A. DE LUCIA, et al. (2007). "Introducing Legacy System Migration Technologies in a Software Company: a Controlled Experiment."

- COLOSIMO, M., A. D. LUCIA, et al. (2009). "Evaluating legacy system migration technologies through empirical studies." Information and Software Technology **51**(2): 433-447.
- COREY, S. M. (1988). Action research to improve school practices, Columbia University, Teachers college.
- CHIANG, R., T. BARRON, et al. (1993). "Performance evaluation of reverse engineering relational databases into extended entity-relationship models." Entity-Relationship Approach - ER'93: 352-363.
- CHIANG, R., T. BARRON, et al. (1994). "Reverse engineering of relational databases: Extraction of an EER model from a relational database\* 1." Data & Knowledge Engineering **12**(2): 107-142.
- CHIANG, R., T. BARRON, et al. (1996). "A framework for the design and evaluation of reverse engineering methods for relational databases\* 1." Data & Knowledge Engineering **21**(1): 57-77.
- DE LUCIA, A., R. FRANCESE, et al. (2008). "Developing legacy system migration methods and tools for technology transfer." Software: Practice and Experience **38**(13): 1333-1364.
- DE MIGUEL, A. and M. PIATTINI (1999). "Fundamentos y modelos de bases de datos." Ediciones Ra-Ma, Segunda edición. Madrid, España.
- DE MIGUEL, A., M. PIATTINI, et al. (2000). Diseño de Base de Datos Relacionales.
- FESENMEYER, D. (2008). Design und prototypische Implementierung eines Software-Tools zur Normalisierung relationaler Datenbanken. Applied Sciences, Konstanz University.
- FLACH, P. and I. SAVNIK (1999). "Database dependency discovery: a machine learning approach." AI communications **12**(3): 139-160.
- FOSTER, J. (1993). Cost factors in software maintenance. School of Engineering and Computer Science. Durham, University of Durham.
- FUENTES, J., P. SÁEZ, et al. (2011). Propuesta de un método para la obtención de Dependencias Funcionales basado en Dualidad de Hipergrafos. XVI Jornadas de Ingeniería del Software y Bases de Datos. Coruña.
- GARCÍA-RODRIGUEZ, I., M. POLO, et al. (2004). "Estado del arte de la reingeniería y la ingeniería inversa: 2001-2003." 4as Jornadas Iberoamericanas de Ingeniería del Software e Ingeniería del Conocimiento.
- GARCÍA, G. (2008). "Evolución e Integración de Aplicaciones Legadas: Comenzar de Nuevo o Actualizar?" Congreso Colombiano de Computación.
- HENRARD, J., A. CLEVE, et al. (2004). "Inverse Wrappers for Legacy Information Systems Migration." ISSN 0926-4515 All rights reserved editors: prof. dr. PME De Bra prof. dr. ir. JJ van Wijk **4**: 30.
- HENRARD, J., D. ROLAND, et al. (2007). An Industrial Experience Report on Legacy Data-Intensive System Migration Software Maintenance, 2007. ICSM 2007. IEEE International Conference on **473 - 476**
- HUHTALA, Y., J. KÄRKKÄINEN, et al. (1999). "Tane: An Efficient Algorithm for Discovering Functional and Approximate Dependencies." The Computer Journal **42**.
- JOHANNESSON, P. (1994). A Method for Transforming Relational Schemas Into Conceptual Schemas. Proceedings of the Tenth International Conference on Data Engineering, IEEE Computer Society.
- KITCHENHAM, B. and S. CHARTERS (2007) "Guidelines for performing systematic literature reviews in software engineering." **Volume**, DOI:
- LEWIN, K. (1946). "Action research and minority problems." Journal of social issues **2**(4): 34-46.
- LIN, C. (2008). "Migrating to relational systems: Problems, methods, and strategies." Contemporary Management Research **4**(4): 369-380.
- LOPES, S., J. PETIT, et al. (2002). "Functional and approximate dependency mining: database and FCA points of view." Journal of Experimental & Theoretical Artificial Intelligence **14**(2): 93-114.

- LOPES, S., J. PETIT, et al. (2000). Efficient Discovery of Functional Dependencies and Armstrong Relations. *Advances in Database Technology — EDBT 2000*, Springer Berlin / Heidelberg. **1777**: 350-364.
- LUBYTE, L. and S. TESSARIS (2007). "Extracting ontologies from relational databases." *Proc. of DL 7*: 387-395.
- MAIRA, M., P. NAVARRO, et al. (2008). "Rediseño del Sistema de Información de una Red Integrada de Salud de cobertura nacional en Chile, Proyecto SIRIS."
- MALDONADO, A., G. GUTIÉRREZ, et al. (2009) "Hacia la obtención de un modelo de base de datos relacional desde sistemas heredados: Un caso de estudio " **Volume**, DOI:
- MANNILA, H. and K. RÄIHÄ (1986). "Design by example: An application of Armstrong relations." *Journal of computer and system sciences* **33**(2): 126-141.
- MANNILA, H. and K. RÄIHÄ (1994). "Algorithms for inferring functional dependencies from relations." *Data & Knowledge Engineering* **12**(1): 83-99.
- MCTAGGART, R. (1991). "Principles for participatory action research." *Adult Education Quarterly* **41**(3): 168.
- MOORE, M. and L. MOSHKINA (2002). *Migrating legacy user interfaces to the Internet: Shifting dialogue initiative*, IEEE.
- NOVELLI, N., R. CICCHETTI, et al. (2001). FUN: An Efficient Algorithm for Mining Functional and Embedded Dependencies Springer Berlin / Heidelberg. **1973**: 189-203.
- O'CALLAGHAN, A. (2009). "Focus issue on legacy information systems and business process change: migrating large scale legacy systems to component- based and object technology: The Evolution of a Pattern Language." *Communications of the association for information systems* **2**.
- PEDERSEN, D. and L. GOLDBERG (2009) "System Transformation: Be Careful or You Will Not Get What You Asked For." **Volume**, DOI:
- PETIT, J., J. KOULOUMDJIAN, et al. (1994). "Using queries to improve database reverse engineering." *Entity-Relationship Approachâ€”ER'94 Business Modelling and Re-Engineering*: 369-386.
- PRATHER, J., D. LOBACH, et al. (1995). *Converting a legacy system database into relational format to enhance query efficiency*, American Medical Informatics Association.
- RAFREIDER, T. (2008). Data-Mining Verfahren und Tools zur Unterstützung der Datenbankanalyse und Schemaoptimierung. *Applied Sciences*, Konstanz University.
- REICHERTZ, P. (2006). "Hospital information systems-past, present, future." *International Journal of Medical Informatics*
- RODRÍGUEZ, A., A. MÁRQUEZ, et al. (2001). Gestión de la evolución del software. El eterno problema de los legacy systems, JISBD2001. VI Jornadas de Ingeniería del Software. Universidad de Granada. España.
- SAVNIK, I. and P. FLACH (1993). *Bottom-up induction of functional dependencies from relations*. AAAI'93 Workshop on Knowledge Discovery in Databases, AAAI Press.
- SCHLIMMER, J. (1993). *Efficiently inducing determinations: A Complete and Systematic Search Algorithm that Uses Optimal Pruning*. In Proceedings of the Tenth International Conference on Machine Morgan Kaufmann.
- SILBERSCHATZ, A., H. KORTH, et al. (1991). *Fundamentos de bases de datos*, McGraw-Hill.
- SNEED, H. (1995). "Planning the reengineering of legacy systems." *IEEE software* **12**(1): 24-34.
- SNEED, H. (2002). *Estimating the costs of software maintenance tasks*, IEEE.
- SOMMERVILLE, I. (2005). *Ingeniería del software*, Pearson Educación.
- SULTAN, O. and I. PETROUNIAS (2009). Database Reverse Engineering.
- THIRAN, P. and W. VAN DEN HEUVEL (2004). Inverse Wrappers for Legacy Information Systems Migration. *First International Workshop on Wrapper Techniques for Legacy Systems In connection with the 11th Working Conference on Reverse Engineering*.
- VASSILIADIS, P., A. SIMITSIS, et al. (2002). *Conceptual modeling for ETL processes*, ACM.

- WADSWORTH, Y. (1998). "What is Participatory Action Research?" Action Research International.
- Wang, C., A. Lo, et al. (2004). Converting legacy relational database into XML database through reverse engineering, Citeseer.
- WYSS, C., C. GIANNELLA, et al. (2001). FastFDs: A Heuristic-Driven, Depth-First Algorithm for Mining Functional Dependencies from Relation Instances Extended Abstract Data Warehousing and Knowledge Discovery, Springer Berlin / Heidelberg. **2114**: 101-110.
- YEH, D., Y. LI, et al. (2008). "Extracting entity-relationship diagram from a table-based legacy database." Journal of Systems and Software **81**(5): 764-771.
- YIN, R. (2009). Case study research: Design and methods, Sage publications, INC.
- ZOU, Y. and K. KONTOGIANNIS (2001). Towards a web-centric legacy system migration framework, Citeseer.

## **Anexos**

---

## **Anexo A**

### **Revisión Sistemática de la Literatura de la Investigación**

#### **A.1 Desarrollo de la RSL**

Tal como se ha señalado una RSL es una técnica que permite conocer el estado del arte que existe sobre una materia a partir de una revisión exhaustiva y ordenada sobre la literatura existente en distintos medios, ya sean estos electrónicos o impresos.

Dicha labor se llevó a cabo mediante un proceso metódico y riguroso sobre el cual se realizó una planificación y posterior ejecución de la revisión. A continuación se describen detalladamente las actividades que componen cada una de estas etapas:

#### **Planificación de la Revisión**

Esta etapa tiene como propósito identificar las necesidades de la revisión indicando sus objetivos además de las fuentes desde donde se obtendrán los estudios, así mismo de las restricciones, criterios de inclusión y exclusión. En particular, cobran relevancia los criterios que se utilizarán para evaluar la calidad de los estudios y cómo se extraerán y sintetizarán los datos desde estos estudios.

#### **Identificación de la Necesidad de Revisión**

Existen diversas y poderosas razones para desarrollar esta investigación. Para comenzar, los SIH están inmersos en una cantidad considerable de organizaciones y muchas de éstas requieren de metodologías o marcos de trabajos que faciliten un proceso de migración hacia nuevas plataformas. Por lo cual, cobra relevancia la identificación de trabajos relacionados con técnicas de migración de SIH y generación automática de modelos de datos. En particular las nuevas necesidades de la organización y mejoramiento en el rendimiento de aplicaciones contribuyen inevitablemente hacia la evolución.

#### **Objetivos de la Investigación**

En esta etapa se establece el propósito y finalidad de la investigación, el que se organiza en objetivos generales y específicos, los que se describen a continuación.

## **Objetivo General**

Realizar una RSL tendiente a reunir y resumir la información relacionada con la migración de SIH hacia nuevas plataformas que permita generar un estado de arte y encontrar una brecha de investigación concreta.

## **Objetivos Específicos**

- Definido el objetivo general es esencial puntualizar qué es lo que se pretende lograr de modo más concreto.
- Búsqueda y selección de material bibliográfico que permita elaborar un estado del arte sobre migración de sistemas heredados.
- Análisis general de la literatura disponible para que facilite la generación de un estado del arte.

## **Interrogantes de la Investigación**

Las interrogantes se plantean con el objeto de identificar cual es el motor que mueve la investigación y su planteamiento debe ser resuelto por medio de la RSL.

- ¿Existen métodos que permitan obtener un modelo de datos normalizado a partir de un SIH?

## **Recursos disponibles para realizar la investigación**

De acuerdo al tema y la disponibilidad de material bibliográfico la mayor parte de las fuentes de información provendrán desde la red (internet), esto debido a que la universidad cuenta con convenios con revistas electrónicas que permiten extraer artículos y conferencias. Por otro lado, la literatura clásica proveniente de libros se encuentra disponible en la biblioteca de la misma universidad y en la propia red. A saber algunos sitios para realizar la búsqueda:

- Google Scholar
- ACM Digital Library
- ISI Web of Knowledge
- DBLP Computer Science Bibliography
- ScienceDirect
- Scirus
- Conferencias
- Citeseer

- Libros sobre el tema

### Definición de un protocolo de búsqueda

La definición del protocolo de búsqueda se basa en la idea de obtener e identificar los estudios significativos o relevantes, es decir, estudios primarios y secundarios, con el propósito de dar respuesta a las preguntas de investigación.

### Términos Clave

Los términos clave facilitan la identificación de los parámetros de búsqueda tendientes a seleccionar los conceptos principales y relacionados que posteriormente serán unidos estratégicamente por conectores lógicos para realizar la búsqueda en las fuentes indicadas en el punto anterior.

<u>Temática</u>	<u>Palabras Clave</u>
Definición y clasificación de SIH	Legacy System
Métodos de migración	Migrating
Obtención de Modelo de datos	Data Model

**Tabla A.1.** Breve descripción de términos clave.

### Protocolo de Búsqueda

Los términos o palabras claves definidas en la Tabla A.1 son concatenados por medio del conector lógico “and”. Esto permitirá filtrar las búsquedas con el objeto de encontrar estudios que cumplan en mayor parte con los términos buscados. La búsqueda se realizará solo en inglés. En la Tabla A2 se especifican algunas combinatorias posibles para realizar la búsqueda.

<u>Combinatoria de Palabras Clave</u>
Legacy System and Migrating
Legacy System and Data Mdel
Legacy System and migrating and Data Model

**Tabla A.2** Combinatoria de palabras clave.

La combinatoria permitió encontrar investigaciones que potencialmente pueden ser utilizadas toda vez que se alineen con los criterios de inclusión. El registro se debe realiza en la Tabla A4. Respecto de la búsqueda de libros será delimitado sólo a los recursos disponibles en la biblioteca de la Universidad del Bío Bío.

Una vez identificado el material es solicitado y revisado realizando una lectura minuciosa de los resúmenes para comprobar si cumple con los criterios de inclusión. A continuación se muestra la Tabla A.3 que permite el registro de las búsquedas realizadas en sitios electrónicos y biblioteca.

<b>Id</b>	<b>Fecha</b>	<b>Sitio</b>	<b>URL</b>	<b>Búsqueda</b>	<b>Filtros</b>	<b>Resultados</b>

**Tabla A.3** Permite el registro de búsquedas identificando datos relevantes de esta.

**Id** : Identificador autonumérico de la búsqueda

**Fecha** : Indicará la fecha en que se realizó la búsqueda.

**Sitio** : Permitirá identificar el sitio donde se realizó la búsqueda.

**URL** : Indicará la URL de respuesta, esto con el objeto de volver a utilizarla eventualmente.

**Búsqueda** : Permitirá indicar los parámetros de búsqueda utilizados, es decir las combinatorias de palabras claves.

**Filtros** : Tendrá por objetivo identificar cuáles fueron los filtros que el sitio posee utilizados en la búsqueda.

**Resultados** : Indica la cantidad de resultados obtenidos en la búsqueda.

En definitiva, cada uno de los estudios clasificados son registrados y al mismo tiempo ingresados en el software “Endnote” con sus datos bibliográficos obtenidos desde los propios portales, en caso que corresponda. Tal como muestra la Tabla A.4, permite el registro de los resultados obtenidos mediante las búsquedas consignadas en la Tabla A.3. Se establece una relación de cero a muchos entre la Tabla A.3 y A.4, es decir una búsqueda puede tener muchos estudios considerados como relevantes.

<u>Id. B.</u>	<u>Id. E.N.</u>	<u>T. Material</u>	<u>Material</u>	<u>P. Res.</u>	<u>Abstract</u>	<u>Obs.</u>	<u>Estado</u>

**Tabla A.4** Registro de Material

**Id. B.** : Identificador de la búsqueda realizada en la Tabla 3

**Id. E.N.** : Identificador otorgado por el software Endnote.

**T. Material** : Clasificación del estudio. (Artículo, libro)

**Material** : Nombre del estudio.

**P. Res.** : Posición del estudio al realizar la búsqueda.

**Abstract** : Porcentaje de asignado a resumen de acuerdo a los criterios de inclusión.

**Obs.** : Descripción general de la temática del estudio.

**Estado** : Estado de acuerdo si se pudo obtener por vía digital o impresión.

### **Definición del Protocolo de Revisión**

Etapa en que se definen las normas de revisión, criterios de inclusión y exclusión de material, así como la estrategia de extracción de los datos relevantes a la investigación.

### **Normas de Revisión**

Para realizar la revisión se utilizarán criterios tendientes a incluir términos relacionados directamente con “Legacy System”, así como también a “Migrating”, es decir se podrán realizar por separado inicialmente y posteriormente realizando combinatorias.

### **Criterios de Inclusión**

Los criterios de inclusión son aquellos que permitirán filtrar los resultados obtenidos en las búsquedas sobre las fuentes descritas previamente. Se utiliza un método que permite realizar búsquedas por medio de varios pasos que suponen repetición de estos en la medida que se necesario. Particularmente, la inclusión de una investigación considera la coincidencia del título, palabras clave y

posteriormente el resumen del estudio. De esta forma se podrá establecer la relación que existe entre el estudio los términos clave.

Este criterio permitirá eliminar aquellos estudios que no contribuyan a dar respuesta a las interrogantes de la investigación.

### **Criterios de Exclusión**

Serán excluidas todas aquellas investigaciones analizadas cuyo título o resumen no aporten y no tengan directa relación con la migración de sistemas heredados o métodos que faciliten la extracción de modelo de datos.

### **Estrategia de Extracción de Datos**

La extracción de datos se llevará a cabo de acuerdo al material recolectado en la etapa de búsqueda de material y conforme a los criterios mencionados en los puntos anteriores.

### **Estrategia de Síntesis de Datos**

Una vez que los datos han sido debidamente seleccionados se procede a analizar con mayor profundidad los estudios realizando un resumen por cada uno de ellos, el que deberá quedar registrado electrónicamente (típicamente en la aplicación que administra la bibliografía). Esto permitirá un ordenamiento de los datos extraídos revisando cada pregunta de investigación por separado y realizar una síntesis de cada uno artículos analizados.

### **Identificación de Palabras claves a utilizar como criterio de búsqueda**

Este proceso se realiza de una manera estructurada de manera que le es asignado una identificación a cada combinatoria de palabras claves, las que serán aplicadas a cada buscador. Dicho identificador permite establecer una relación con la tabla que registra las búsquedas, de esta forma se establece de una manera sencilla a qué combinatoria de palabra clave corresponde el artículo analizado. A continuación en la Tabla A.5 se muestra la organización utilizada para tal efecto.

<b>Número</b>	<b>Palabras clave</b>
1	Migrating legacy system
2	Legacy system database
3	Legacy system schema

**Tabla A.5** Palabras Clave

## A.2 Ejecución de la Revisión

Conforme a la planificación de la revisión se realizó la exploración de investigaciones relevantes en las fuentes indicadas en el protocolo de búsqueda basado en los criterios de inclusión, exclusión y palabras clave. Con el objeto de mantener un registro estructurado se documentaron los resultados obtenidos en tablas, tal como se muestra en la Tabla A.6.

## A.3 Resultados de la Revisión

Ajustado a la planificación de la revisión se realizaron las búsquedas conforme a los criterios establecidos, ante lo cual los resultados son los que se especifican detalladamente en la Tabla A.6. La tabla exhibe una muestra de los resultados encontrados durante nuestra investigación, sin embargo, las cantidades indicadas no representan el total de estudios analizados.

### Resultados encontrados:

<b>Búsqueda</b>	<b>ACM</b>	<b>Scholar</b>	<b>Google</b>	<b>DBLP</b>	<b>citeseer</b>	<b>SCiencieDirect</b>	<b>Scirus</b>
migrating legacy system	829	407	487.000	15	47.698	1262	37189
legacy system database	5.986	309	1.430.000	14	170.637	8698	311099
legacy system schema	1.921	291	490.000	0	56.301	2182	48699

**Tabla A.6** Registro de Material

Del universo de resultados encontrados solo se analizaron aquellos artículos que tuvieran relación directa con la investigación. Esto debido a que muchos estudios coinciden sintácticamente en el título con alguna de las palabras clave, sin embargo juzgando por el contexto del título de la investigación no coincide con el criterio de búsqueda.

## Anexo B

### Fuentes de Información de Sistema SAM

#### Script SQL

La documentación utilizada en nuestra investigación se basó en el estudio los script SQL para tablas, procedimientos almacenados, triggers, vistas y funciones. Estos documentos electrónicos poseen una estructura sencilla con un encabezado en el cual se identifica el nombre del archivo, la fecha de creación y modificaciones, el nombre del mismo y en último lugar la descripción. A continuación de cada atributo se registra un comentario indicando la función que este cumple en la tabla.

```
--* Archivo      : SAPAT001.SQL
--* Historia     : 05/03/96 Maria Luisa Diaz Creacion
--* Nombre       : Indice
--* Descripcion  : Tabla que contiene los antecedentes basicos de un
--*              : Paciente. Siempre que un paciente exista, debe haber
--*              : una fila en esta tabla.
--*              : En el Índice de pacientes pueden coexistir pacientes
--*              : reales (personas) o pacientes ficticios (Instituciones).
--*              : Si se trata la previsión de un Paciente ficticio, aún
--*              : siendo mandatoria, es irrelevante (toma el valor por
--*              : defecto que exista en el modelo)

--DROP TABLE SAPAT001;
CREATE TABLE SAPAT001
(
    CodPacie SERIAL, --Codigo del Paciente
    FechaCrea DATE DEFAULT TODAY NOT NULL, --Fecha de Creacion
    Estado SMALLINT DEFAULT 1 NOT NULL
        CHECK (Estado IN (1,2,3,4))
        CONSTRAINT SAPAT001_C_001, --Estado del Paciente.
        --Activo=1, Pasivo=2, Fallecido=3, Traslado=4
    RUT INTEGER, --Rol Unico Tributario
    DigiVer CHAR(1), --Digito verificador
    HC CHAR(10), --Codigo de Historia Clinica
    ApellPat VARCHAR(20,10) DEFAULT "NN" NOT NULL,
        --Apellido Paterno del Paciente
    ApellMat VARCHAR(15,10) DEFAULT "NN" NOT NULL,
        --Apellido Materno del Paciente
    Nombres VARCHAR(20,10) DEFAULT "NN" NOT NULL,
        --Nombres del Paciente
    FechaNac DATETIME YEAR TO MINUTE, --Fecha de Nacimiento
    FechaFall DATETIME YEAR TO MINUTE, --Fecha de fallecimiento
    Sexo SMALLINT DEFAULT 3 NOT NULL
        CHECK (Sexo IN (1,2,3,9))
        CONSTRAINT SAPAT001_C_002,
        --Masculino=1, Femenino=2, Indefinido=3, Ignorado=9
    FormaPago INTEGER NOT NULL, --Prevision del Paciente.
    Dformapago VARCHAR(40,20), --Nombre de Prevision
    Domicilio VARCHAR(60,30) NOT NULL, --Domicilio del Paciente
    Comuna INTEGER, --Comuna del domicilio del Paciente
    Dcomuna VARCHAR(40,20), --Nombre de la Comuna del domicilio del Paciente
    Telefono VARCHAR(20,9), --Telefono del Paciente
    GrupSang SMALLINT CHECK (GrupSang IN (1,2,3,4))
        CONSTRAINT SAPAT001_C_011,
        --Grupo Sanguineo. AB=1, A=2, B=3, O=4
    FactorRH SMALLINT CHECK (FactorRH IN (-1,1))
        CONSTRAINT SAPAT001_C_012,
        --FactorRH. Negativo=-1, Positivo=1
    Ocupacion INTEGER, --Ocupacion del paciente
    Docupacion VARCHAR(40,20), --Nombre de la ocupacion del paciente
    NombrePad VARCHAR(20,10), --Nombres del Padre del Paciente
    NombreMad VARCHAR(20,10), --Nombres de la Madre del Paciente
    Observacion VARCHAR(200,100), --Observacion del paciente
    Institucion INTEGER, --Para el paciente ficticio debe indicarse la
        --Institucion que este representa.
    Razon_institucion VARCHAR(40,20), -- Razon social de la Institucion
```

```

        Fono_institucion VARCHAR(20,10) -- Fono de la Institucion
        CHECK (FechaFall >= FechaNac)    CONSTRAINT SAPAT001_C_001
    )
--IN sadbs EXTENT SIZE 145958 NEXT SIZE 7279 LOCK MODE row;

--DROP INDEX SAPAT001_CL_001;
CREATE UNIQUE CLUSTER INDEX SAPAT001_CL_001
ON SAPAT001(CodPacie);

ALTER TABLE SAPAT001 ADD CONSTRAINT PRIMARY KEY (CodPacie) CONSTRAINT
SAPAT001_P_001;
--* Archivo      : SABST030.SQL
--* Historia     : 01/06/95  Andres Valdes  Creacion
--*             : 29/07/96  Malu. Modificacion (Agrega campo EmisorCred)
--*             : 02/12/96  Se dividio nombres en : nombrel y nombre2
--* Nombre      : Personas.
--* Descripcion : Tabla que contiene a los Profesionales del
--*             Establecimiento, Profesionales Externos al
--*             Establecimiento e Instituciones Externas
--*             (Empresas, Establecimientos hospitalarios u otros).
--DROP TABLE SABST030;
CREATE TABLE SABST030
(
    RUT                INTEGER, --Rol Unico Tributario
    Dv                 CHAR(1), --Digito Verificador del RUT
    Funcionario        VARCHAR(60,0), --Nombre del Funcionario
    Planta             INTEGER, --Codigo de la Planta
    D_Planta           VARCHAR(40,10), --Nombre de la Planta
    Juridica           INTEGER, --Codigo de Calidad Juridica
    D_Juridica         VARCHAR(15,10), --Nombre de Calidad Juridica
    Genero             CHAR(1), --Genero del Funcionario
    Direccion          VARCHAR(60,10), --Direccion del Funcionario
    Comuna             INTEGER, --Codigo de la Comuna
    D_Comuna           VARCHAR(40,10), --Nombre de la Comuna
    Provincia          INTEGER, --Codigo de la Provincia
    D_Provincia        VARCHAR(40,10), --Nombre de la Provincia
    Region            INTEGER, --Codigo de la Region
    D_Region           VARCHAR(40,10), --Nombre de la Provincia
    Isapre             INTEGER, --Codigo de la Isapre
    D_Isapre           VARCHAR(40,10), --Nombre de la Isapre
    Afp                INTEGER, --Codigo de la Afp
    D_Afp              VARCHAR(40,10), --Nombre de la Afp
    F_Nacimiento       DATE, --Fecha de nacimiento del Funcionario
    Unidad             INTEGER, --Codigo de la Unidad a la que pertenece el Funcionario
    D_Unidad           VARCHAR(40,10), --Nombre de la Unidad
    Cargo              INTEGER, --Codigo del Cargo que ocupa el Funcionario
    D_Cargo            VARCHAR(40,10), --Descripcion del Cargo
    Titulo             VARCHAR(60,10), --Titulo del Funcionario
    Total_Haber        INTEGER, --Total haberes del Funcionario
    Remuneracion       INTEGER --Remuneracion del Funcionario

    PRIMARY KEY (RUT)    CONSTRAINT SABST030_P_001
)
--IN sadbs EXTENT SIZE 375 LOCK MODE row;
--* Archivo      : SAURT009.SQL
--* Historia     : 16/04/96  Maria Luisa Diaz  Creacion
--* Nombre      : DAU's
--* Descripcion : Contiene los antecedentes propios (basicos relevantes)
--*             de la atencion de Urgencia, representados en el DAU.

--DROP TABLE SAURT009;
CREATE TABLE SAURT009
(
    Codigo             SERIAL, --Codigo de la atencion de urgencia
    CP                 INTEGER NOT NULL, --Codigo del paciente
    FP                 INTEGER NOT NULL, --Codigo de la Forma de Pago
    DFP                INTEGER NOT NULL, --Codigo de la Forma de Pago
    ServSol            INTEGER NOT NULL, --Servicio solicitante
    DservSol           INTEGER NOT NULL, --Servicio solicitante
    ProfSol            INTEGER NOT NULL, --Profesional solicitante
    DProfSol           INTEGER NOT NULL, --Profesional solicitante
    FechaLleg         DATETIME YEAR TO MINUTE DEFAULT
                    CURRENT YEAR TO MINUTE NOT NULL,
                    --Fecha de Ingreso a la Urgencia
    Motivo             INTEGER NOT NULL, --Motivo de la consulta
    Medio              INTEGER,
                    --Medio de Llegada del paciente
    DMedio             INTEGER,
                    --Medio de Llegada del paciente

    Lugar             INTEGER, --Lugar Origen del percance

```

```

DLugar          INTEGER, --Lugar Origen del percance
ServEje         INTEGER NOT NULL, --Servicio ejecutante
DServEje        INTEGER NOT NULL, --Servicio ejecutante
Comentario      VARCHAR(200,10), --Comentario sobre admision del paciente
Anulado         INTEGER, --Campo de anulaci3n de atencion
Cierre          DATE, -- fecha del cierre de atencion
Fcrea           DATE -- fecha de creacion de registro
Presion         DECIMAL (3,2), -- presion del paciente
Pulso           DECIMAL (3,2), -- pulso del paciente
Tempaxilar      DECIMAL (3,2), -- temperatura axilar del paciente
TempRectal      DECIMAL (3,2), -- temperatura recata del paciente
Alcoholemia     DECIMAL (3,2), -- grado de alcohol en la sangre
Ambulancia      INTEGER, -- codigo de ambulancia de traslado
Carabinero      VARCHAR(40,10), -- nombre de carabinero que adopto procedimiento
Diagpos1        INTEGER, -- codigo de diagnostico presunto 1
Ddiagpos1       VARCHAR(60,10), -- Descripcion del Diagnostico 1
Diagpos2        INTEGER, -- Codigo del diagnostico presunto 2
Ddiagpos2       VARCHAR(60,10),-- Descripcion del diagnostico presunto 2
Perstratante    INTEGER, -- Codigo del Profesional tratante
DPerstratante   VARCHAR(60,10), -- Nombre del Profesionista tratante
Profeje         INTEGER, -- Profesional que ejecuta la prestacion
Dprofeje        VARCHAR(60,10), -- Nombre de profesional
ServDest        INTEGER, -- Codigo de Servicio de destino
DservDest       VARCHAR(60,10), -- Nombre de servicio de destino

PRIMARY KEY(Codigo)                CONSTRAINT SAURT009_P_001,
) --IN sadsb EXTENT SIZE 59062 NEXT SIZE 2953 LOCK MODE row;

```

## Interfaz de Usuario Modulo Pacientes

Este tipo de fuente de informaci3n permite visualizar la interfaz de usuarios con la que interactúan los usuarios, esta misma potencia el uso del teclado con un fuerte uso de teclas de funci3n. Todo esto en un ambiente que carece de componentes visuales intuitivos, tal como se observa en la Figura B.1. El desarrollo de estas aplicaciones se realiz3 en lenguaje 4GL de IBM y en todas las interfaces de usuarios en el extremo superior derecho figura una combinaci3n de letras y números que identifica al nombre del formulario, de este modo es sencillo acudir a los c3digos fuentes para ubicar los script del mismo.

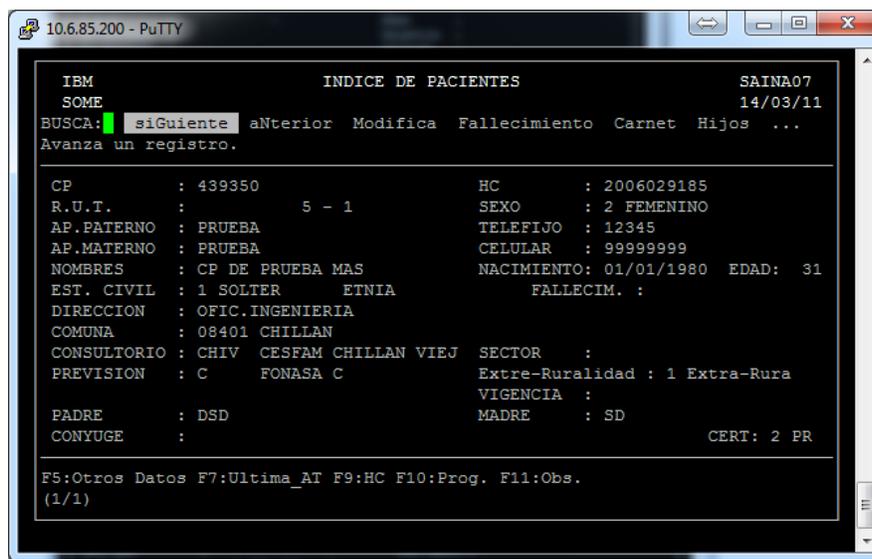


Figura B.1 Interfaz Grafica de Usuario

## Estructura de directorios en códigos fuentes

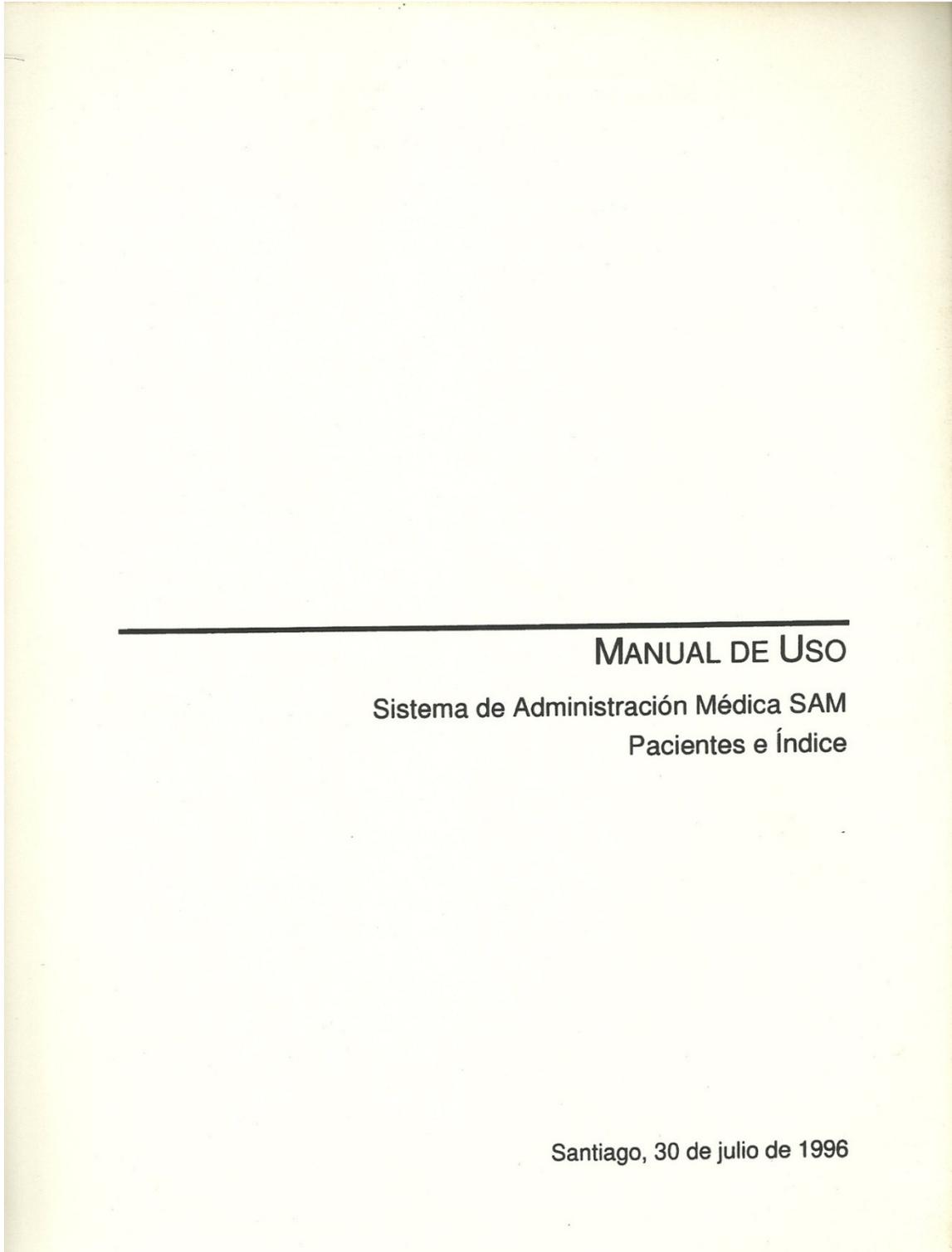
Los códigos fuentes del sistema SAM se encuentran ordenadas bajo una estructura que facilita su ubicación. Dicha estructura está basada en jerarquías, de este modo todos los sistemas se encuentran al interior del directorio SAM, tal como se exhibe en la Figura B.2.

<b>sam</b>	
<b>ag</b>	(Agenda)
<b>bn</b>	(Banco de Sangre)
<b>bs</b>	(Básicos)
<b>ca</b>	(Caja)
<b>di</b>	(Diagnósticos)
<b>fa</b>	(Farmacia)
<b>fu</b>	(Funciones)
<b>fz</b>	(Finanzas)
<b>gs</b>	(Gestión)
<b>hc</b>	(Historias Clínicas)
<b>hp</b>	(Hospitalización)
<b>in</b>	(Informaciones)
<b>pa</b>	(Pacientes)
<b>pr</b>	(Prestaciones)
<b>qu</b>	(Unidad Quirúrgica)
<b>ua</b>	(Unidades de Apoyo al Diagnostico)
<b>ur</b>	(Urgencia)
<b>ut</b>	(Unidades de Tratamiento)

**Figura B.2** Estructura de directorio en códigos fuentes

## **Manual de Uso de Sistema SAM**

Una de las fuentes de datos que se encontró disponible es el manual de usuario de SAM, el que se muestra a continuación en las próximas páginas.



Versiones	1.0 Santiago, 05 de Junio de 1996
Autores	Castro Barroilhet Alberto Chávez Reyes Luis Tapia Salinas Cecilia
Ubicación	docsalud@wagner/metadocumentos/SAM
<b>IBM - CLIENTES AUTORIZADOS</b>	

## Tabla de Contenido

<b>CAPÍTULO 1 INTRODUCCIÓN.....</b>	<b>1</b>
1.- DESCRIPCIÓN .....	1
<b>CAPÍTULO 2 PASO CERO .....</b>	<b>2</b>
1.- CODIGOS DE SEGURIDAD .....	2
2.- CÓDIGOS DE ERROR .....	3
3.- MANTENCIÓN .....	8
<b>CAPÍTULO 3 PROCEDIMIENTOS.....</b>	<b>10</b>
1.- OFICINA/VENTANILLA DE ÍNDICE DE PACIENTES.....	10
2.- GENERACIÓN DE NÓMINAS DE PACIENTES.....	10
3.- BÚSQUEDA DE PACIENTES (FICHA ÍNDICE).....	11
4.- INSTITUCIONES CON CARACTERÍSTICAS DE PACIENTE.....	11
5.- DEFINICIÓN DE INSTITUCIONES A FACTURAR.....	11
<b>CAPÍTULO 4 CASOS ESPECIALES .....</b>	<b>12</b>
1.- PACIENTES NN EN URGENCIA .....	12
2.- CREACIÓN DE ÍNDICE EN ADMISIÓN DE URGENCIA Y ADMISIÓN HOSPITALARIA .....	12
3.- NÓMINAS DE PACIENTES.....	12
<b>CAPÍTULO 5 PANTALLAS.....</b>	<b>13</b>
1.- FLUJO DE PANTALLAS.....	13
2.- DESCRIPCIÓN .....	14
a.- Índice de Pacientes. (SAINA07) H001 .....	14
b.- Fecha Fallecimiento. (SAINA07) H013 .....	16
c.- Auditoria del Paciente. (SAINA07) H014.....	16
d.- Otros Antecedentes del Paciente. (SAINA07) H002 .....	16
e.- Historia Clínica del Paciente. (SAINA07) H003 .....	17
f.- Programas. (SAINA07) H004 .....	17
g.- Ingresar en Programa. (SAINA07) H015 .....	18
h.- Fecha de Cierre. (SAINA07) H016.....	18
i.- Hijos del Paciente. (SAINA08) H017.....	18
j.- Registro Recien Nacido. (SAINA08) H018 .....	19
k.- Búsqueda Estandar de Pacientes LU28. (SAPAL28) H005.....	19
l.- Nómina de Pacientes. (SAINA06) H006 .....	20
m.- Paciente Completo. (SAINA03) H007.....	21
n.- Pacientes Masivos. (SAINA01) H008.....	21
o.- Entidades para Copago. (SAINA04) H009.....	22
p.- Prevision por Defecto. (SAPAA10) H010.....	23
q.- Registro de Paciente Mínimo. (SAINA00) H011 .....	23
r.- Búsqueda de Paciente Copago LU14. (SAPAL14) H012 .....	24
<b>CAPÍTULO 6 GLOSARIO.....</b>	<b>25</b>

---

# Capítulo 1 Introducción

## 1.- Descripción

Este Manual describe el uso de las aplicaciones relacionadas con la administración de los datos generales de un Paciente (independientes de la atención específica), lo que representa básicamente los módulos Paciente e Índice.

Los puntos a tratar, se refieren a los requerimientos de instalación (Paso Cero); los procedimientos administrativos y como se relacionan con el Sistema; las pantallas y su descripción y finalmente una lista de casos especiales y su tratamiento por el sistema.

## Capítulo 3 Procedimientos

### 1.- Oficina/Ventanilla de Índice de Pacientes.

La creación de su registro de índice es el proceso previo que debe efectuar todo Paciente del Establecimiento que desee una atención. La atención (cualquiera sea su tipo) de un Paciente, está supeditada a que exista un registro de éste en el índice del sistema, el cual contiene todos los antecedentes personales y previsionales de ellos.

Este proceso requiere de un funcionario Admisor de Índice y un funcionario Calificador de Régimen Previsional.

Es función de todos los estamentos del Hospital, ajenos a la Oficina/Ventanilla de Índice, incentivar a los Pacientes a que concurran a crear o actualizar el registro de índice.

El proceso requiere la presencia del Paciente o persona cercana (familiar, vecino, etc.) para entregar los datos que son requeridos por el funcionario Admisor de Índice, el cual los registra en la Pantalla "ÍNDICE DE PACIENTES" (H001 d.e.d.) previa búsqueda exhaustiva que asegure la no existencia previa del Paciente.

Éstos son los antecedentes personales más relevantes a registrar, dentro de los cuales destaca el Régimen Previsional, dato imprescindible para procesar el registro. El Paciente debe acreditar su previsión mediante la "credencial previsional" correspondiente (u otro documento oficial) y ser contrastada con la base de datos Fonasa/Isapre. Si el Paciente no posee la documentación previsional, debe ser enviado con el funcionario encargado de clasificarlo (p.e. el Calificador de Derecho). El ingreso y modificación del campo "PREVISIÓN" requieren códigos de seguridad adicionales (diferentes para cada una de las dos operaciones).

El Admisor de Índice debe enfatizar la presentación de documentos que acrediten los datos entregados (p.e. carné de identidad, credencial previsional, certificado de nacimiento, otros).

Un conjunto de datos secundarios pueden ser registrados en la Pantalla "OTROS ANTECEDENTES DEL PACIENTE" (H002 d.e.d.), donde ninguno de los campos editables es mandatorio. Alterar el Estado del Índice requiere seguridad adicional.

La Ficha Clínica (más otros antecedentes de ésta) del Paciente pueden registrarse en la Pantalla "HISTORIA CLÍNICA DEL PACIENTE" (H003 d.e.d.). El número de Ficha puede ser creado en forma manual (respetando antiguo número) o automática (asignado por el sistema de acuerdo al criterio del Establecimiento).

La asignación o retiro de programas de salud se realizan en la Pantalla "PROGRAMAS" (H004 d.e.d.).

La creación/consulta del registro recién nacido para el Paciente, se efectúa en la Pantalla "REGISTRO RECIÉN NACIDO" (H017).

La consulta de los hijos del Paciente se efectúa en la Pantalla "HIJOS DEL PACIENTE" (H018). Desde el Índice de Pacientes no es posible crear ni eliminar registros con calidad hijos.

Una vez creado el registro de Índice, el Admisor debe generar el Carné Hospitalario y entregarlo al Paciente exigiendo portarlo cada vez que concurra al Establecimiento.

El término exitosos de este proceso involucra crear el registro de índice (identificando al Paciente con su código o CP) y la habilitación de la Cuenta Corriente.

### 2.- Generación de Nóminas de Pacientes.

La generación de Nóminas de Pacientes se realizan en la Pantalla "NÓMINA DE PACIENTES" (H006 d.e.d.), utilizando dos formatos: La Nómina de Pacientes y La Nómina de Pacientes Resumida que se diferencian por la cantidad de columnas a desplegar.

El proceso exige la definición del título de la Nómina y del orden en que se desplegarán los datos. Posteriormente, el usuario debe especificar el criterio de búsqueda o selección, el cual definirá el listado de Pacientes a emitir (p.e. pacientes fallecidos, pacientes según fecha de nacimiento, pacientes según etnia, pacientes según previsión, etc.). Si no se ingresa un criterio de selección, el proceso listará el total de Pacientes existentes en el índice.  
Por la cantidad de registros involucrados, el proceso puede demorarse.

### 3.- Búsqueda de Pacientes (Ficha Índice).

La búsqueda de Pacientes que se realiza en la Pantalla "FICHA INDICE" (H007 d.e.d.) permite obtener un resumen de los datos más relevantes contenidos en el registro de índice.  
El proceso reemplaza el cardex de las fichas índice y requiere el ingreso del criterio de búsqueda para el paciente a seleccionar.  
El registro del Paciente seleccionado puede visualizarse en la pantalla o en el reporte impreso.  
La búsqueda de Pacientes también se efectúa en la Pantalla "INDICE DE PACIENTES" (H001 d.e.d.).

### 4.- Instituciones con características de Paciente.

El proceso está orientado a facilitar el ingreso de prestaciones asociadas a una nómina de Pacientes provenientes de la misma Institución (Regimiento, Municipio, otros). Para ello se genera un registro que clasifica a dicha Institución como "Paciente" del Hospital para luego cargar todas las prestaciones en una sola cuenta corriente evitando crear "n" índices con "n" cuentas corrientes.  
Para este proceso se utiliza la Pantalla "INDICE PACIENTES MASIVOS" (H008 d.e.d.), en la cual, sobre el campo "Rut" se define la Institución cuyos antecedentes ya se encuentran definidos en el registro de "personas" (es necesario crear previamente el registro de la Institución en el mantenedor de "personas" de la BASE).  
Posteriormente debe especificarse el convenio que la Institución a suscrito con el Establecimiento para efecto del cobro de las prestaciones.  
Si corresponde cobro de las prestaciones mediante una liquidación (factura) implica generar un registro de Institución a Facturar (siguiente Proceso número 5.) para la Institución.

### 5.- Definición de Instituciones a Facturar.

Toda Institución que tenga convenio con el Establecimiento, y a la cual se requiera emitir algún tipo de documento para cobro de prestaciones, debe tener un registro de Institución a Facturar.  
La calidad de Institución a Facturar se otorga en la Pantalla "ENTIDADES PARA COPAGO" (H009 d.e.d.). Para definir la Institución se edita el campo "Rut" recuperando los restantes antecedentes de ésta definidos en el registro de "personas" (toda Institución a Facturar debe encontrarse ingresada previamente en el mantenedor de "personas" de la BASE).

## Capítulo 4 Casos Especiales

### 1.- Pacientes NN en Urgencia

La operatoria relacionada con la creación de un registro de Paciente "NN" en la Urgencia se analiza en dicho Módulo. El único alcance es con el módulo Paciente, donde se genera el registro del Paciente (CP y cuenta corriente).

### 2.- Creación de Índice en Admisión de Urgencia y Admisión Hospitalaria

Cuando eventualmente la Atención de Urgencia, Atención Abierta, así como la Atención Cerrada, requieren generar registros de Índice de Pacientes, cuentan con tres opciones: La primera consiste en enviar al Paciente a la Oficina de Índice y luego, una vez ingresado al sistema, proceder a la admisión en el servicio pertinente; La segunda es que el Admisor cuente con acceso a las aplicaciones de Índice para efectuar esta tarea; La tercera es acceder a la Pantalla "REGISTRO DE PACIENTE MÍNIMO" (H011 d.e.d.) donde, sin necesidad de abandonar su aplicación particular, pueden generar registros de Pacientes ingresando un set reducido de datos de éstos al sistema.

Esta aplicación permite Ingresar un Paciente al sistema (al Índice), efectuar búsquedas de Pacientes existentes en el sistema y generar el Carné del Establecimiento. El Admisor debe retornar a su anterior aplicación mediante la opción "Selecciona". Con esta opción obtendrá el nuevo registro sin necesidad de efectuar otra búsqueda en "BÚSQUEDA DE PACIENTES" (H005 d.e.d.). Si retorna con la opción "Termina", no retendrá el código del Paciente debiendo buscar el registro recién generado.

Es obligación de los Admisores efectuar una nueva búsqueda del Paciente en esta Pantalla (extendiendo el criterio de selección), antes de insertar un nuevo registro, para asegurar la no existencia previa. En adición, es necesario pesquisar aquellos Pacientes generados a través de esta aplicación para, posteriormente, complementar los antecedentes en la Pantalla de "ÍNDICE DE PACIENTES" (H001 d.e.d.)

### 3.- Nóminas de Pacientes

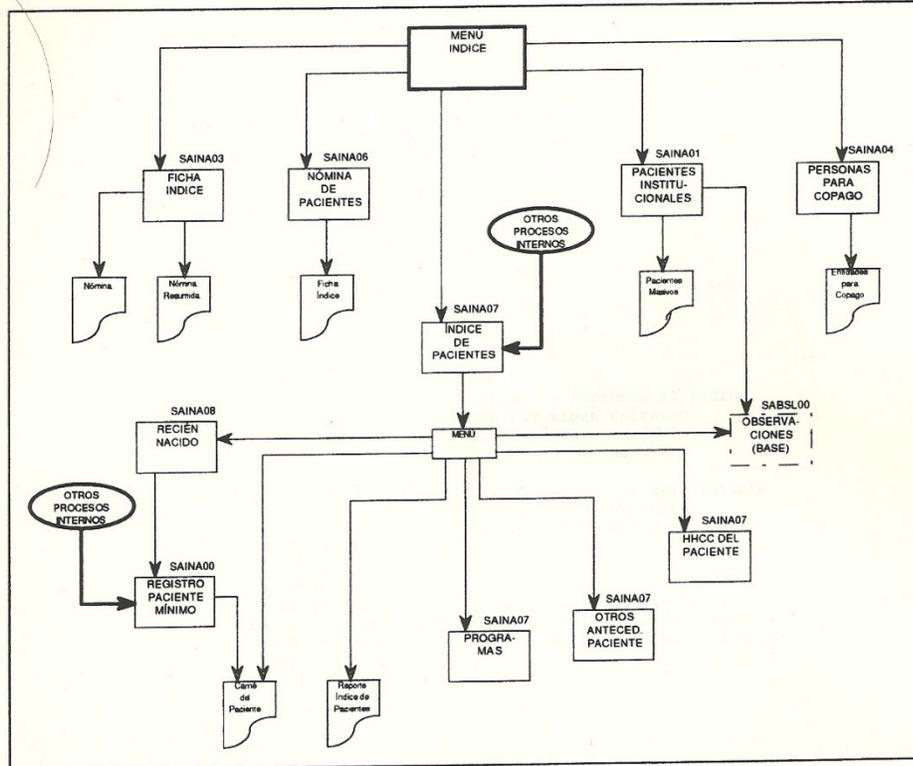
Cuando un Establecimiento Dependiente del Servicio u otra Organización (municipalidad, empresa, regimiento, escuela, otro), envía nóminas de Pacientes con sus respectivas muestras no es necesario generar registros de Índice para cada Paciente de la Nómina. Basta con generar un registro de Paciente-Institución (Paciente ficticio) dándole a la Institución el carácter de Paciente del Hospital al cual deberá cargar todas las prestaciones que involucren el contenido de la nómina. De esta forma se consigue efectuar un sólo cobro por el total de prestaciones y evitar la creación innecesaria de Índices de Pacientes.

La creación del Paciente-Institución debe efectuarse en la Pantalla "PACIENTES INSTITUCIONALES" (H0008 d.e.d.) definiendo especialmente (campo mandatorio) el Convenio con el cual dicha Institución cancelará las prestaciones asociadas.

Es necesario que la Institución, y todos los datos pertinentes, estén ingresados en el Mantenedor de Personas del módulo BASE.

# Capítulo 5 Pantallas

## 1.- Flujo de Pantallas



## 2.- Descripción

### a.- ÍNDICE DE PACIENTES. (SAINA07) H001

```

.1
+-----+
| I B M           INDICE DE PACIENTES           SAINA07 |
| S O M E                04/06/96             |
|ADMINISTRACION:  Ingresa Busca Carnet Hijos rEcién-nacido ... |
|Ingresa paciente. |
+-----+
|CP               :                               HC       : |
|R.U.T.           :                               SEXO      : |
|AP. PATERNO     :                               TELEFONO   : |
|AP. MATERNO     :                               NACIMIENTO  :   EDAD: |
|NOMBRES         :                               FALLECIMIE. : |
|EST. CIVIL      :                               ETNIA      : |
|DIRECCION       :                               |
|COMUNA          :                               |
|LOCALIZACION    :                               SECTOR     : |
|PREVISION       :                               CARACTER   : |
|CREDENCIAL      :                               VIGENCIA   : |
|PADRE           :                               MADRE     : |
|CONTACTO        :                               CERT      : |
+-----+

```

#### Descripción de la Barra de Acciones

**Ingresar** : Ingresar datos de un nuevo Paciente al Indica.  
**Busca** : Busca a un Paciente por algún criterio.  
**Carnet** : Imprime Carnet del Paciente.  
           Ver Reporte "Carnet del Paciente".  
**Hijos** :  
**recién Nacido** : Muestra datos de recién nacido(s) del Paciente.  
**Reporte** : Imprime datos de una lista de Pacientes.  
           Ver Reporte "Índice de Pacientes"  
**Termina** : Salir.

#### Teclas de Funciones

**F5** : Permite ingresar a ventana Otros antecedentes del Paciente.  
**F9** : Permite ingresar a ventana Historia Clínica del Paciente.  
**F10** : Permite consultar datos de Programas asociados al Paciente.  
**F11** : Permite editar Observaciones.  
**F5, F9 y F10** corresponden a Pantallas.

#### Ingresar

##### Validaciones de Campos

**NOMBRE, AP.PATERNO, AP.MATERNO, DIRECCION**: NOT NULL  
**NACIMIENTO** : NOT NULL, debe ser menor o igual a la actual.  
**EDAD** : (Estos campos son alternativos, puede ingresarse la fecha y se calcula internamente la edad, o puede ingresarse edad y calcula la Fecha de nacimiento)  
**PREVISION** : NOT NULL, por defecto debe tener lo que se ingresó en Módulo Paciente (Previsión por defecto), Ayuda F6 (LU61).  
**EST.CIVIL** : NOT NULL, default =0 (SIN INFORMACION)  
**SEXO** : NOT NULL  
**ETNIA** : NOT NULL, default =1 (CHILENO)  
**CARACTER** : NOT NULL, default =1 (PROVISORIA)  
**COMUNA** : NULL, Ayuda F6 (LU52-COMUNAS)  
**LOCALIZACION** : NULL, si existe implica que existe COMUNA.  
           Ayuda F6 (LU92-LOCALIZACIONES COMUNAS)  
**SECTOR** : NULL, si existe implica que existe LOCALIZACION.  
           Ayuda F6 (LU93-SECTORES LOCALIZACION)

**NOTA:** El resto de los datos de esta pantalla, pueden quedar nulos.

#### Busca

Permite buscar por los siguientes criterios: CP, RUT, EST.CIVIL, SEXO, AP.PATERNO, AP.MATERNO, NOMBRES, CREDENCIAL. Incluye la opción F6 Ayuda, la cual llama al LU28 de Pacientes (El Estandar)

del Sistema).

Al modificar a un Paciente, presenta las siguientes opciones :

Descripción de la Barra de Acciones

Modifica : Permite actualizar datos del Paciente.

Fallecimiento: Permite ingresar fecha de Fallecimiento del Paciente. Debe ser menor o igual a la actual y Mayor o igual a Fecha de Nacimiento.  
(Puede modificarse)

Carnet : Imprime Carnet del Paciente.

recien Nacido: Muestra datos de Recien Nacido(s) del Paciente.

Auditoria : Permite consultar, datos de últimas modificaciones del Índice y Previsión.

Termina : Salir.

**Reportes**

- 1.-Desde la Opción reporte se da un criterio de búsqueda, generándose una lista con los pacientes seleccionados. Se incluyen casi todos los datos entregados en Índice.
- 2.-Carnet, entrega la información del Índice en un formato de 1/3 de página, listo para ser doblado y entregarlo al paciente.

**b.- FECHA FALLECIMIENTO. (SAINA07) H013**

```
.13
+-----+
|           Fecha Fallecimiento           |
+-----+
| FECHA (dd/mm/aa) :                     |
| HORA (hh:mm)    :                     |
+-----+
| ESC : Termina  F8 : Procesa.         |
| Fecha fallecimiento paciente.         |
+-----+
```

**c.- AUDITORIA DEL PACIENTE. (SAINA07) H014**

```
.14
+-----+
| I B M           AUDITORIA DEL PACIENTE           SAINA07 |
| S O M E                                     04/06/96 |
+-----+
| ULTIMO MODIFICADOR INDICE      :           |
| FECHA ULTIMA MODIFICACION INDICE :           |
| USUARIO REGISTRADOR PREVISION :           |
| FECHA REGISTRO PREVISION       :           |
| ULTIMO MODIFICADOR PREVISION   :           |
| FECHA ULTIMA MODIFICACION PREVISION :       |
+-----+
| Presione una tecla para continuar. |
+-----+
```

**d.- OTROS ANTECEDENTES DEL PACIENTE. (SAINA07) H002**

```
.2
+-----+
|           OTROS ANTECEDENTES DEL PACIENTE           |
| ESCOLARIDAD : 1 BASICA INCOMPLETA           ANALFABETO : 0 NO |
| GRUPO       : 4 0                               FACTOR RH : + |
| DEUDOR SOME : 0                               DEUDOR FINANZAS : 0 |
| EMISOR CRED. PREV. : 61108000 CAJA PREVISION DE EJERCITO |
| OCUPACION    :                               |
| EMPRESA      :                               |
| ESTADO DEL INDICE DEL PACIENTE |
| ESTADO : 1 ULTIMO CAMBIO ESTADO : 03/02/1996 ELIMINAR : 0 |
+-----+
| 0 No - 1 BasInc - 2 BasCom - 3 MedInc - 4 MedCom. - 5 supInc - 6 supCom. |
+-----+
Validaciones de Campos
ESCOLARIDAD : NULL, si es mayor o igual 2, campo ANALFABETO=0
GRUPO       : NULL, si existe debe ingresar FACTOR RH.
DEUDOR SOME, DEUDOR FINANZAS, ESTADO, ULTIMO CAMBIO ESTADO,
ELIMINAR    : Campos informativos. (No modificables)
```

**e.- HISTORIA CLÍNICA DEL PACIENTE. (SAINA07) H003**

```

.3
-----
HISTORIA CLINICA DEL PACIENTE
ANTIGUO No HC      :
HC                 : 93006875   FECHA GENERACION : 04/03/1993
PERDIDA           : 0           ULTIMO MOVIMIENTO :
PERDIDAS HISTORICAS : 0       ULTIMA PERDIDA   :
LUGAR ARCHIVO    : ARCH ARCHIVO HISTORIAS CLINICAS
LUGAR ACTUAL     : ARCH ARCHIVO HISTORIAS CLINICAS
FICHA ODONTOLOGICA : 0 UBICACION :
-----
    
```

Descripción de la Barra de Acciones  
**Manual** : Genera HC manual, si esta no existe.  
**Automática** : Genera HC automática, puede existir previamente.  
**Modifica** : Permite cambiar datos (modificables) de HC.  
**Termina** : Salir.

Validaciones de Campos  
**ANTIGUO N°HC** : NULL, registra el antiguo número de HC, si existe.  
**HC** : NULL, F5 permite generar HC en forma automática (De acuerdo a criterio definido en Biblioteca HHCC)  
**FECHA GENERACION** : NOT NULL, debe ser mayor o igual a la actual.  
**ULTIMO MOVIMIENTO** : Campo informativo, tiene valor si fue Prestada. (Actualización interna, No modificable)  
**PERDIDA** : NOT NULL, default =0  
**PERDIDAS HISTORICAS**: Default =0, Campo informativo, tiene valor si Perdida cambia de 0 a 1. (No modificable)  
**ULTIMA PERDIDA** : Campo informativo, tiene valor si Perdida cambia de 0 a 1. (No modificable)  
**LUGAR ARCHIVO** : NOT NULL, default =Lugar definido en Biblioteca HHCC (Base)  
**LUGAR ACTUAL** : NOT NULL, default =Lugar definido en Biblioteca HHCC (Base)  
**FICHA ODONTOLOGICA**: NOT NULL, default =0 (NO)  
 Si es 0, campo UBICACION es NULL (No modif)  
 Si es 1, campo UBICACION is not null (Lugar)

**f.- PROGRAMAS. (SAINA07) H004**

```

.4
-----
PROGRAMAS DEL PACIENTE
INGRESO   CIERRE  CODIGO   DENOMINACION
-----
|ESC : Termina F5 : Abrir F7 : Cerrar.
|(0/0)
-----
    
```

Permite definir Programas a Pacientes NO FALLECIDOS.  
 No es posible borrar Programas definidos al Paciente.

Teclas de Funciones  
**F5** : Abrir Programa.  
**F7** : Cerrar Programa.

**g.- INGRESAR EN PROGRAMA. (SAINA07) H015**

```
.15
+-----+
| INGRESAR EN PROGRAMA |
| PROGRAMAS :          |
| ESC : Termina F6 : Ayuda |
+-----+
Teclas de Funciones
F6 : LU17-PROGRAMAS (SABSL17).
```

**h.- FECHA DE CIERRE. (SAINA07) H016**

```
.16
+-----+
| Fecha de Cierre |
| Fecha (dd/mm/aa) : |
| ESC Cancela. |
+-----+
```

**i.- HIJOS DEL PACIENTE. (SAINA08) H017**

```
.17
+-----+
| I B M                HIJOS DEL PACIENTE                SAINA08 |
| S O M E                05/06/96 |
| BUSCA: Modifica Elimina Asocia Indice Primero Ultimo ... |
| Modifica el registro actual. |
+-----+
| MADRE : AGUAYO BRITO HIPOLITA DEL C.                CP : 3000 |
| PACIENTE : ALVARES ALVARES LUIS                CP : 10000 |
+-----+
| NOMBRE : XXX |
| SEXO : 1 Masculino |
| APGAR 5 : |
| PESO : 2000 (Gr.) |
| BCG : |
| EGRESO : |
| PRE-NATAL : |
| CONDICION AL NACER : 1 Vivo |
| FECHA PARTO : 28/05/1996 18:00 |
| TALLA : (Cm.) |
| POSICION : 1 DE 1 |
| GEMELAR : 1 |
+-----+
| (1/5) |
+-----+
```

Descripción de la Barra de Acciones

Ingresar : Permite ingresar nuevo registro.  
 Modifica : Permite modificar datos de los hijos del Paciente.  
 Elimina : Permite eliminar un registro.  
 Asocia : Permite asociar un paciente del indice al registro RN.  
 Indice : Permite generar un registro de Indice de Pacientes.  
 Termina : Salir.

Validaciones de Campos

NOMBRE : NOT NULL, Nombre del Recien Nacido.  
 SEXO : NOT NULL, 1:MASCULINO, 2:FEMENINO, 3:INDEFINIDO.  
 CONDICION AL NACER: NOT NULL default =1 (1:VIVO, 2:MUERTO)  
 FECHA PARTO : Fecha de Parto o Nacimiento.

*j.- REGISTRO RECIEN NACIDO. (SAINA08) H018*

```

.18
-----
| I B M          REGISTRO RECIEN NACIDO          SAINA08 |
| S O M E          05/06/96                      |
|-----|
|BUSCA: Ingresa Modifica Elimina Termina      |
|Ingresa recién nacido.                        |
|-----|
| MADRE :                                           CP : |
| PACIENTE : AGUAYO BRITO HIPOLITA DEL C.        CP : 3000 |
|-----|
| NOMBRE :                                           |
| SEXO :                                           CONDICION AL NACER : |
| APGAR 5 :                                           FECHA PARTO : |
| PESO : (Gr.) TALLA : (cm.)                        |
| BCG :                                           POSICION : DE |
| EGRESO :                                           GEMELAR : |
| PRE-NATAL :                                           |
|-----|

```

Validaciones de Campos  
 NOMBRE : Debe copiar el campo PACIENTE.  
 SEXO, CONDICION AL NACER, FECHA PARTO: Retorna la informacion  
 contenida en INDICE.

*k.- BUSQUEDA ESTANDAR DE PACIENTES LU28. (SAPAL28) H005*

```

.5
-----
| I B M          BUSQUEDA DE PACIENTES          SAPAL28 |
| S O M E          30/05/96                      |
|-----|
| CP      R.U.T.   HC      AP.PATERNO  AP.MATERNO  NOMBRES  CER |
|-----|
|-----|
| ESC : Cancela F8 : Busca                       |
|-----|

```

Esta es la manera general de buscar pacientes en todo el Sistema.  
 Se pueden especificar distintos criterios, contestando con la lista  
 de pacientes que cumplen con estas.  
 Esta lista está restringida a un número máximo de pacientes. En  
 caso que este máximo sea sobrepasado, no es mostrado ningún  
 paciente, indicando que se ingrese un criterio más restringido.

I.- NÓMINA DE PACIENTES. (SAINA06) H006

```

.6
+-----+
| I B M                NOMINA DE PACIENTES                SAINA06 |
| S O M E                30/05/96                        |
|ADMINISTRACION:  Reporte resumen  reporta Completo  Termina |
|Imprime reporte resumido segun una condicion.                |
+-----+
| TITULO :                                                    |
| ORDEN :   1=RUT  2=HC  3=Nombre  4=Fecha Ingreso          |
|DEFINIR CRITERIO DE SELECCION :                              |
| AP.PATERNO :                AP.MATERNO :                  |
| NOMBRES :                    SEXO :                        |
| FECHA INGRESO :                                                    |
| FECHA NACIMIENTO :                                                |
| FECHA FALLECIMIENTO :                                            |
| CERTEZA :                                                    |
| ETNIA :          ESTADO :                ELIMINAR :        |
| PREVISION :                CARACTER :                          |
| DEUDOR SOME :    DEUDOR FINANZAS :                              |
|                   ESTE PROCESO PUEDE TOMAR DEMASIADO TIEMPO |
+-----+

```

Generación de un listado de Pacientes  
 Descripción de la Barra de Acciones  
 Reporte resumen : Permite imprimir lista simple de Pacientes.  
 reporte Completo : Permite imprimir lista más completa de Pacientes.  
 Termina : Salir.

Validaciones de Campos  
 TITULO : NOT NULL, título para encabezado de la Nómina.  
 ORDEN : NOT NULL, selecciona orden de impresión de los datos.

NOTA : El criterio de búsqueda lo puede conformar cualquier campo de la pantalla y sus combinaciones, excepto campo CERTEZA.

Reportes  
 1.-Reporte Resumen, entrega en formato de 80 caracteres, un paciente por línea (los que cumplen con el criterio definido).  
 Entrega : CP, RUT, HHCC, Nombre Completo, Fecha Ingreso  
 2.-Reporte Completo, entrega en 132 caracteres, un paciente por línea (los que cumplen con el criterio).  
 Entrega : CP, RUT, HHCC, Nombre Completo, fecha Ingreso, Previsión, Caracter, Sexo, Fecha Nacimiento, Fecha Fallecimiento, Estado, Deudor SOME, Deudor Fiannzas, Étnia, Por Eliminar.

**m.- PACIENTE COMPLETO. (SAINA03) H007**

```

.7
+-----+
| I B M                FICHA INDICE                SAINA03 |
| S O M E                03/06/96                |
| ADMINISTRACION:  Seleccionar Reporte Termina |
| Busca datos segun un criterio.                |
+-----+
| PACIENTE : |
| CREA.INDICE :      ESTADO :      ELIMINAR : |
| CP :              HC :          R.U.T. : |
| EST.CIVIL :      ETNIA :      GRUPO SANG.:  RH: |
| ESCOLARIDAD :    SEXO :      ANALFABETO : |
| NACIMIENTO :     FALLEC.:     DEUDA SO/PZ:  / |
| CREDENCIAL :     EMISOR :     VIGENCIA : |
| PREVISION : |
| DIRECCION : |
| COMUNA :              TELEFONO : |
| EMPRESA :            OCUPACION: |
| CONTACTO : |
| PADRE/MADRE : |
+-----+

```

Búsqueda Rápida de todos los Datos de un Paciente  
 Descripción de la Barra de Acciones  
**Seleccionar** : Permite retornar datos del Paciente, ingresado un criterio de búsqueda (LU28)  
**Reporte** : Imprime los datos del Paciente, igual a la Pantalla.  
**Termina** : Salir.

**n.- PACIENTES MASIVOS. (SAINA01) H008**

```

.8
+-----+
| I B M                PACIENTES INSTITUCIONALES    SAINA01 |
| S O M E                04/06/96                |
| ADMINISTRACION:  Ingresar Elimina Busca Observacion Reporte Termina |
| Ingresar paciente masivo.                |
+-----+
| R.U.T. : |
| NOMBRE : |
| DIRECCION : |
| COMUNA : |
| TELEFONO :              FAX : |
| OBSERVACION : |
| CONVENIO : |
+-----+

```

Operación sobre Pacientes Institucionales. Instituciones que deben Ser tratadas como pacientes, para procesar series de exámenes sin necesitar ingresarlos cada uno al Índice.  
 Descripción de la Barra de Acciones  
**Ingresar** : Permite ingresar Pacientes Masivos.  
**Elimina** : Permite eliminar la persona de la lista.  
**Busca** : Permite buscar según criterio.  
**Observacion** : Permite definir Obsevacion para el Paciente Masivo.  
**Reporte** : Imprime lista de Pacientes Masivos.  
**Termina** : Salir.

Validaciones de Campos al Ingresar  
**RUT** : Rut de la Persona para agregar a la lista.  
 El Paciente Masivo, debe existir en Mantenedor de

Personas (BASE). F6 (LU59 Módulo BASE)  
CONVENIO: Convenio Previsional. F6 (LU61-PREVISION)  
Reportes  
1.-Dado un criterio, imprime la lista de Pacientes másivos.  
Entrega : Codigo Usuario de la Institución, RUT y Razón Social.

0.- ENTIDADES PARA COPAGO. (SAINA04) H009

.9

I B M	ENTIDADES PARA COPAGO	SAINA04
S O M E		30/05/96
ADMINISTRACION:	Ingresar Elimina Busca Observacion Reporte Termina	
Ingresar entidad para copago.		
R.U.T.	:	
NOMBRE	:	
DIRECCION	:	
COMUNA	:	
TELEFONO	:	FAX :
OBSERVACION	:	

Definición de las instituciones con las que se tiene convenio, es necesario transformarlas en Pacientes, para facturarles.

Descripción de la Barra de Acciones

Ingresar : Permite ingresar Entidades para Copago.  
Elimina : Permite eliminar la persona de la lista.  
Busca : Permite buscar según criterio.  
Observacion : Permite definir Obsevacion para la Entidad para Copago.  
Reporte : Imprime lista de Pacientes Masivos.  
Termina : Salir.

Validaciones de Campos

RUT : Rut de la Entidad para Copago a insertar en la lista.  
La Entidad para Copago debe existir en Personas SABST030 (BASE) y debe tener las siguientes características:  
(ESTADO=1 y TIPO=2) o  
(ESTADO=1 y TIPO=1 y (ATENPAC=1 o DIAGNOST=1))  
Permite utilizar ayuda:  
F6 (LU71 personas Naturales (Módulo BASE))  
F7 (LU70 personas Juridicas (Módulo BASE))

Reportes

1.-Dado un criterio, imprime la lista de Entidades para Copago.  
Entrega : Codigo Usuario de la Institución, RUT y Razón Social.



IBM - CLIENTES AUTORIZADOS

Termina : Salir.

Validaciones de Campos

PREVISION : NOT NULL, por defecto debe tener lo que se ingresó  
en Módulo Paciente (Previsión por defecto)  
Ayuda F6 (LU61).

SEXO : NOT NULL, default =3 (INDEFINIDO)

CHARACTER : NOT NULL, default =1 (PROVISORIA)

r.- BUSQUEDA DE PACIENTE COPAGO LU14. (SAPAL14) H012

.12

I B M	BUSQUEDA DE PACIENTE COPAGO	SAPAL14
S O M E		03/06/96
CP	R.U.T.	NOMBRES

Descripción de la Barra de Acciones

Criterio : Ajusta criterio de seleccion.

siGiente : Avanza un registro.

aNterior : Va al registro anterior.

-Pag . : Despliega la pagina anterior.

+Pag . : Despliega proxima pagina.

salTa : Avanza a una posicion.

Selecciona : Elige el registro actual.

Termina : Salir.

## Capítulo 6 Glosario

Sistema de Seguridad

Sistema externo a SAM, el cual almacena todas las validaciones de Seguridad para todos los sistemas. En especial es relevante la conexión entre los usuarios definidos en el, con los profesionales en SABST030 en SAM, a través de un RUT igual.

D.e.d.

De este Documento

## Anexo C

### Dependencias Funcionales Obtenidas Automáticamente

En las tablas C1,C2,C3 se exhiben las DFs encontradas automáticamente por medio de TANE JAVA.

**Tabla SAPAT001**

N°	Determinante	->	Determinado	N°	Determinante	->	Determinado
1	APELLMAT	->	FONO_INSTITUCION	134	DFORMAPAGO	->	RUT_INSTITUCION
2	APELLMAT	->	INSTITUCION	135	DFORMAPAGO, DOCUPACION, FACTORRH	->	GRUPSANG
3	APELLMAT	->	RAZSOCIAL_INSTITUCION	136	DFORMAPAGO, FACTORRH, OCUPACION	->	GRUPSANG
4	APELLMAT	->	RUT_INSTITUCION	137	DFORMAPAGO, FACTORRH, SEXO	->	GRUPSANG
5	APELLMAT, APELLPAT, COMUNA, RUT, SEXO	->	FECHAFALL	138	DFORMAPAGO, FECHANAC	->	FACTORRH
6	APELLMAT, APELLPAT, DCOMUNA, RUT, SEXO	->	FECHAFALL	139	DFORMAPAGO, FECHANAC	->	GRUPSANG
7	APELLMAT, APELLPAT, DOCUPACION	->	FACTORRH	140	DFORMAPAGO, FECHANAC, HC	->	DOCUPACION
8	APELLMAT, APELLPAT, DOCUPACION	->	GRUPSANG	141	DFORMAPAGO, FECHANAC, HC	->	FECHAFALL
9	APELLMAT, APELLPAT, HC, NOMBRES	->	SEXO	142	DFORMAPAGO, FECHANAC, HC	->	OCUPACION
10	APELLMAT, APELLPAT, NOMBRES, RUT	->	FECHAFALL	143	DFORMAPAGO, GRUPSANG	->	FACTORRH
11	APELLMAT, APELLPAT, OCUPACION	->	FACTORRH	144	DFORMAPAGO, HC, NOMBRES	->	DOCUPACION
12	APELLMAT, APELLPAT, OCUPACION	->	GRUPSANG	145	DFORMAPAGO, HC, NOMBRES	->	FECHAFALL
13	APELLMAT, COMUNA, FECHANAC, RUT	->	FECHAFALL	146	DFORMAPAGO, HC, NOMBRES	->	OCUPACION
14	APELLMAT, COMUNA, HC	->	DOCUPACION	147	DOCUPACION	->	FONO_INSTITUCION
15	APELLMAT, COMUNA, HC	->	OCUPACION	148	DOCUPACION	->	INSTITUCION
16	APELLMAT, COMUNA, HC, NOMBRES	->	SEXO	149	DOCUPACION	->	OCUPACION
17	APELLMAT, COMUNA, HC, SEXO	->	FECHAFALL	150	DOCUPACION	->	RAZSOCIAL_INSTITUCION
18	APELLMAT, DCOMUNA, FECHANAC, RUT	->	FECHAFALL	151	DOCUPACION	->	RUT_INSTITUCION
19	APELLMAT, DCOMUNA, HC	->	DOCUPACION	152	DOCUPACION, FACTORRH, FORMAPAGO	->	GRUPSANG
20	APELLMAT, DCOMUNA, HC	->	OCUPACION	153	DOCUPACION, FACTORRH, SEXO	->	GRUPSANG
21	APELLMAT, DCOMUNA, HC, NOMBRES	->	SEXO	154	DOCUPACION, GRUPSANG	->	FACTORRH
22	APELLMAT, DCOMUNA, HC, SEXO	->	FECHAFALL	155	FACTORRH	->	FONO_INSTITUCION
23	APELLMAT, DFORMAPAGO, HC	->	DOCUPACION	156	FACTORRH	->	INSTITUCION
24	APELLMAT, DFORMAPAGO, HC	->	FECHAFALL	157	FACTORRH	->	RAZSOCIAL_INSTITUCION
25	APELLMAT, DFORMAPAGO, HC	->	OCUPACION	158	FACTORRH	->	RUT_INSTITUCION
26	APELLMAT, FACTORRH	->	GRUPSANG	159	FACTORRH, FECHANAC	->	GRUPSANG
27	APELLMAT, FECHANAC	->	FACTORRH	160	FACTORRH, FORMAPAGO, OCUPACION	->	GRUPSANG
28	APELLMAT, FECHANAC	->	GRUPSANG	161	FACTORRH, FORMAPAGO, SEXO	->	GRUPSANG
29	APELLMAT, FECHANAC, HC	->	DFORMAPAGO	162	FACTORRH, NOMBRES	->	GRUPSANG
30	APELLMAT, FECHANAC, HC	->	DOCUPACION	163	FACTORRH, OCUPACION, SEXO	->	GRUPSANG
31	APELLMAT, FECHANAC, HC	->	FECHAFALL	164	FECHAFALL	->	FONO_INSTITUCION
32	APELLMAT, FECHANAC, HC	->	FORMAPAGO	165	FECHAFALL	->	INSTITUCION
33	APELLMAT, FECHANAC, HC	->	OCUPACION	166	FECHAFALL	->	RAZSOCIAL_INSTITUCION
34	APELLMAT, FECHANAC, HC, NOMBRES	->	APELLPAT	167	FECHAFALL	->	RUT_INSTITUCION
35	APELLMAT, FECHANAC, NOMBRES, RUT	->	FECHAFALL	168	FECHANAC	->	FONO_INSTITUCION
36	APELLMAT, FORMAPAGO, HC	->	DOCUPACION	169	FECHANAC	->	INSTITUCION
37	APELLMAT, FORMAPAGO, HC	->	FECHAFALL	170	FECHANAC	->	RAZSOCIAL_INSTITUCION
38	APELLMAT, FORMAPAGO, HC	->	OCUPACION	171	FECHANAC	->	RUT_INSTITUCION
39	APELLMAT, GRUPSANG	->	FACTORRH	172	FECHANAC, FORMAPAGO	->	FACTORRH
40	APELLMAT, HC, NOMBRES	->	DOCUPACION	173	FECHANAC, FORMAPAGO	->	GRUPSANG
41	APELLMAT, HC, NOMBRES	->	FECHAFALL	174	FECHANAC, FORMAPAGO, HC	->	DOCUPACION
42	APELLMAT, HC, NOMBRES	->	OCUPACION	175	FECHANAC, FORMAPAGO, HC	->	FECHAFALL
43	APELLMAT, NOMBRES	->	FACTORRH	176	FECHANAC, FORMAPAGO, HC	->	OCUPACION
44	APELLMAT, NOMBRES	->	GRUPSANG	177	FECHANAC, GRUPSANG	->	FACTORRH
45	APELLPAT	->	FONO_INSTITUCION	178	FECHANAC, HC, NOMBRES	->	DOCUPACION
46	APELLPAT	->	INSTITUCION	179	FECHANAC, HC, NOMBRES	->	OCUPACION

47	APELLPAT	->	RAZSOCIAL_INSTITUCION	180	FECHANAC, HC, NOMBRES	->	SEXO
48	APELLPAT	->	RUT_INSTITUCION	181	FECHANAC, HC, SEXO	->	DOCUPACION
49	APELLPAT, COMUNA, DFORMAPAGO, HC	->	FECHAFALL	182	FECHANAC, HC, SEXO	->	OCUPACION
50	APELLPAT, COMUNA, FECHAFALL, FECHANAC, HC	->	DFORMAPAGO	183	FECHANAC, NOMBRES	->	FACTORRH
51	APELLPAT, COMUNA, FECHAFALL, FECHANAC, HC	->	FORMAPAGO	184	FECHANAC, NOMBRES	->	GRUPSANG
52	APELLPAT, COMUNA, FORMAPAGO, HC	->	FECHAFALL	185	FONO_INSTITUCION	->	INSTITUCION
53	APELLPAT, DCOMUNA, DFORMAPAGO, HC	->	FECHAFALL	186	FONO_INSTITUCION	->	RAZSOCIAL_INSTITUCION
54	APELLPAT, DCOMUNA, FECHAFALL, FECHANAC, HC	->	DFORMAPAGO	187	FONO_INSTITUCION	->	RUT_INSTITUCION
55	APELLPAT, DCOMUNA, FECHAFALL, FECHANAC, HC	->	FORMAPAGO	188	FORMAPAGO	->	DFORMAPAGO
56	APELLPAT, DCOMUNA, FORMAPAGO, HC	->	FECHAFALL	189	FORMAPAGO	->	FONO_INSTITUCION
57	APELLPAT, DFORMAPAGO, DOCUPACION, SEXO	->	FACTORRH	190	FORMAPAGO	->	INSTITUCION
58	APELLPAT, DFORMAPAGO, DOCUPACION, SEXO	->	GRUPSANG	191	FORMAPAGO	->	RAZSOCIAL_INSTITUCION
59	APELLPAT, DFORMAPAGO, OCUPACION, SEXO	->	FACTORRH	192	FORMAPAGO	->	RUT_INSTITUCION
60	APELLPAT, DFORMAPAGO, OCUPACION, SEXO	->	GRUPSANG	193	FORMAPAGO, GRUPSANG	->	FACTORRH
61	APELLPAT, DOCUPACION, FORMAPAGO, SEXO	->	FACTORRH	194	FORMAPAGO, HC, NOMBRES	->	DOCUPACION
62	APELLPAT, DOCUPACION, FORMAPAGO, SEXO	->	GRUPSANG	195	FORMAPAGO, HC, NOMBRES	->	FECHAFALL
63	APELLPAT, FACTORRH	->	GRUPSANG	196	FORMAPAGO, HC, NOMBRES	->	OCUPACION
64	APELLPAT, FECHAFALL, FECHANAC, HC, NOMBRES	->	DFORMAPAGO	197	GRUPSANG	->	FONO_INSTITUCION
65	APELLPAT, FECHAFALL, FECHANAC, HC, NOMBRES	->	FORMAPAGO	198	GRUPSANG	->	INSTITUCION
66	APELLPAT, FECHANAC	->	FACTORRH	199	GRUPSANG	->	RAZSOCIAL_INSTITUCION
67	APELLPAT, FECHANAC	->	GRUPSANG	200	GRUPSANG	->	RUT_INSTITUCION
68	APELLPAT, FECHANAC, HC	->	DOCUPACION	201	GRUPSANG, NOMBRES	->	FACTORRH
69	APELLPAT, FECHANAC, HC	->	OCUPACION	202	GRUPSANG, OCUPACION	->	FACTORRH
70	APELLPAT, FECHANAC, HC, RUT	->	DFORMAPAGO	203	GRUPSANG, SEXO	->	FACTORRH
71	APELLPAT, FECHANAC, HC, RUT	->	FORMAPAGO	204	HC	->	APELLMAT
72	APELLPAT, FECHANAC, RUT	->	FECHAFALL	205	HC	->	APELLPAT
73	APELLPAT, FORMAPAGO, OCUPACION, SEXO	->	FACTORRH	206	HC	->	CODPACIE
74	APELLPAT, FORMAPAGO, OCUPACION, SEXO	->	GRUPSANG	207	HC	->	COMUNA
75	APELLPAT, GRUPSANG	->	FACTORRH	208	HC	->	DCOMUNA
76	APELLPAT, HC, NOMBRES	->	DOCUPACION	209	HC	->	DFORMAPAGO
77	APELLPAT, HC, NOMBRES	->	OCUPACION	210	HC	->	DOCUPACION
78	APELLPAT, NOMBRES	->	FACTORRH	211	HC	->	FACTORRH
79	APELLPAT, NOMBRES	->	GRUPSANG	212	HC	->	FECHAFALL
80	CODPACIE	->	APELLMAT	213	HC	->	FECHANAC
81	CODPACIE	->	APELLPAT	214	HC	->	FONO_INSTITUCION
82	CODPACIE	->	COMUNA	215	HC	->	FORMAPAGO
83	CODPACIE	->	DCOMUNA	216	HC	->	GRUPSANG
84	CODPACIE	->	DFORMAPAGO	217	HC	->	INSTITUCION
85	CODPACIE	->	DOCUPACION	218	HC	->	NOMBRES
86	CODPACIE	->	FACTORRH	219	HC	->	OCUPACION
87	CODPACIE	->	FECHAFALL	220	HC	->	RAZSOCIAL_INSTITUCION
88	CODPACIE	->	FECHANAC	221	HC	->	RUT
89	CODPACIE	->	FONO_INSTITUCION	222	HC	->	RUT_INSTITUCION
90	CODPACIE	->	FORMAPAGO	223	HC	->	SEXO
91	CODPACIE	->	GRUPSANG	224	HC, RUT	->	FECHAFALL
92	CODPACIE	->	HC	225	INSTITUCION	->	FONO_INSTITUCION
93	CODPACIE	->	INSTITUCION	226	INSTITUCION	->	RAZSOCIAL_INSTITUCION
94	CODPACIE	->	NOMBRES	227	INSTITUCION	->	RUT_INSTITUCION
95	CODPACIE	->	OCUPACION	228	NOMBRES	->	FONO_INSTITUCION
96	CODPACIE	->	RAZSOCIAL_INSTITUCION	229	NOMBRES	->	INSTITUCION
97	CODPACIE	->	RUT	230	NOMBRES	->	RAZSOCIAL_INSTITUCION
98	CODPACIE	->	RUT_INSTITUCION	231	NOMBRES	->	RUT_INSTITUCION
99	CODPACIE	->	SEXO	232	OCUPACION	->	DOCUPACION
100	COMUNA	->	DCOMUNA	233	OCUPACION	->	FONO_INSTITUCION
101	COMUNA	->	FONO_INSTITUCION	234	OCUPACION	->	INSTITUCION

102	COMUNA	->	INSTITUCION	235	OCUPACION	->	RAZSOCIAL_INSTITUCION
103	COMUNA	->	RAZSOCIAL_INSTITUCION	236	OCUPACION	->	RUT_INSTITUCION
104	COMUNA	->	RUT_INSTITUCION	237	RAZSOCIAL_INSTITUCION	->	FONO_INSTITUCION
105	COMUNA, DFORMAPAGO, FECHANAC, HC, NOMBRES	->	APELLPAT	238	RAZSOCIAL_INSTITUCION	->	INSTITUCION
106	COMUNA, DFORMAPAGO, HC, SEXO	->	FECHAFALL	239	RAZSOCIAL_INSTITUCION	->	RUT_INSTITUCION
107	COMUNA, DOCUPACION, FACTORRH	->	GRUPSANG	240	RUT	->	APELLMAT
108	COMUNA, FACTORRH, OCUPACION	->	GRUPSANG	241	RUT	->	APELLPAT
109	COMUNA, FECHANAC, FORMAPAGO, HC, NOMBRES	->	APELLPAT	242	RUT	->	CODPACIE
110	COMUNA, FECHANAC, HC	->	DOCUPACION	243	RUT	->	COMUNA
111	COMUNA, FECHANAC, HC	->	OCUPACION	244	RUT	->	DCOMUNA
112	COMUNA, FECHANAC, NOMBRES, RUT	->	FECHAFALL	245	RUT	->	DFORMAPAGO
113	COMUNA, FORMAPAGO, HC, SEXO	->	FECHAFALL	246	RUT	->	DOCUPACION
114	COMUNA, GRUPSANG	->	FACTORRH	247	RUT	->	FACTORRH
115	DCOMUNA	->	COMUNA	248	RUT	->	FECHAFALL
116	DCOMUNA	->	FONO_INSTITUCION	249	RUT	->	FECHANAC
117	DCOMUNA	->	INSTITUCION	250	RUT	->	FONO_INSTITUCION
118	DCOMUNA	->	RAZSOCIAL_INSTITUCION	251	RUT	->	FORMAPAGO
119	DCOMUNA	->	RUT_INSTITUCION	252	RUT	->	GRUPSANG
120	DCOMUNA, DFORMAPAGO, FECHANAC, HC, NOMBRES	->	APELLPAT	253	RUT	->	HC
121	DCOMUNA, DFORMAPAGO, HC, SEXO	->	FECHAFALL	254	RUT	->	INSTITUCION
122	DCOMUNA, DOCUPACION, FACTORRH	->	GRUPSANG	255	RUT	->	NOMBRES
123	DCOMUNA, FACTORRH, OCUPACION	->	GRUPSANG	256	RUT	->	OCUPACION
124	DCOMUNA, FECHANAC, FORMAPAGO, HC, NOMBRES	->	APELLPAT	257	RUT	->	RAZSOCIAL_INSTITUCION
125	DCOMUNA, FECHANAC, HC	->	DOCUPACION	258	RUT	->	RUT_INSTITUCION
126	DCOMUNA, FECHANAC, HC	->	OCUPACION	259	RUT	->	SEXO
127	DCOMUNA, FECHANAC, NOMBRES, RUT	->	FECHAFALL	260	RUT_INSTITUCION	->	FONO_INSTITUCION
128	DCOMUNA, FORMAPAGO, HC, SEXO	->	FECHAFALL	261	RUT_INSTITUCION	->	INSTITUCION
129	DCOMUNA, GRUPSANG	->	FACTORRH	262	RUT_INSTITUCION	->	RAZSOCIAL_INSTITUCION
130	DFORMAPAGO	->	FONO_INSTITUCION	263	SEXO	->	FONO_INSTITUCION
131	DFORMAPAGO	->	FORMAPAGO	264	SEXO	->	INSTITUCION
132	DFORMAPAGO	->	INSTITUCION	265	SEXO	->	RAZSOCIAL_INSTITUCION
133	DFORMAPAGO	->	RAZSOCIAL_INSTITUCION	266	SEXO	->	RUT_INSTITUCION

Tabla C.1 DFs automáticas tabla SAPAT001

Tabla SABST030

N°	Determinante	->	Determinado	N°	Determinante	->	Determinado
1	AFP	->	DAFP	148	DCARGO	->	CARGO
2	AFP, CARGO, DISAPRE, TOTAL_HABERES	->	DREGION	149	DCARGO, DISAPRE, DUNIDAD, TOTAL_HABERES	->	DPROVINCIA
3	AFP, CARGO, DISAPRE, TOTAL_HABERES	->	REGION	150	DCARGO, DISAPRE, DUNIDAD, TOTAL_HABERES	->	PROVINCIA
4	AFP, CARGO, DJURIDICA, TOTAL_HABERES	->	DREGION	151	DCARGO, DISAPRE, TOTAL_HABERES, UNIDAD	->	DPROVINCIA
5	AFP, CARGO, DJURIDICA, TOTAL_HABERES	->	REGION	152	DCARGO, DISAPRE, TOTAL_HABERES, UNIDAD	->	PROVINCIA
6	AFP, CARGO, DUNIDAD	->	DPLANTA	153	DCARGO, DJURIDICA	->	DPLANTA
7	AFP, CARGO, DUNIDAD	->	PLANTA	154	DCARGO, DJURIDICA	->	PLANTA
8	AFP, CARGO, ISAPRE, TOTAL_HABERES	->	DREGION	155	DCARGO, DJURIDICA, DUNIDAD, TOTAL_HABERES	->	DPROVINCIA
9	AFP, CARGO, ISAPRE, TOTAL_HABERES	->	REGION	156	DCARGO, DJURIDICA, DUNIDAD, TOTAL_HABERES	->	PROVINCIA
10	AFP, CARGO, JURIDICA, TOTAL_HABERES	->	DREGION	157	DCARGO, DJURIDICA, TOTAL_HABERES, UNIDAD	->	DPROVINCIA
11	AFP, CARGO, JURIDICA, TOTAL_HABERES	->	REGION	158	DCARGO, DJURIDICA, TOTAL_HABERES, UNIDAD	->	PROVINCIA
12	AFP, CARGO, REMUNERACION	->	DPROVINCIA	159	DCARGO, DUNIDAD, ISAPRE, TOTAL_HABERES	->	DPROVINCIA
13	AFP, CARGO, REMUNERACION	->	PROVINCIA	160	DCARGO, DUNIDAD, ISAPRE, TOTAL_HABERES	->	PROVINCIA
14	AFP, CARGO, UNIDAD	->	DPLANTA	161	DCARGO, DUNIDAD, JURIDICA, TOTAL_HABERES	->	DPROVINCIA
15	AFP, CARGO, UNIDAD	->	PLANTA	162	DCARGO, DUNIDAD, JURIDICA, TOTAL_HABERES	->	PROVINCIA
16	AFP, COMUNA, DPLANTA, REMUNERACION	->	CARGO	163	DCARGO, DUNIDAD, TOTAL_HABERES	->	DREGION
17	AFP, COMUNA, DPLANTA, REMUNERACION	->	DCARGO	164	DCARGO, DUNIDAD, TOTAL_HABERES	->	REGION
18	AFP, COMUNA, PLANTA, REMUNERACION	->	CARGO	165	DCARGO, ISAPRE, TOTAL_HABERES, UNIDAD	->	DPROVINCIA
19	AFP, COMUNA, PLANTA, REMUNERACION	->	DCARGO	166	DCARGO, ISAPRE, TOTAL_HABERES, UNIDAD	->	PROVINCIA
20	AFP, DCARGO, DISAPRE, TOTAL_HABERES	->	DREGION	167	DCARGO, JURIDICA	->	DPLANTA
21	AFP, DCARGO, DISAPRE, TOTAL_HABERES	->	REGION	168	DCARGO, JURIDICA	->	PLANTA

22	AFP, DCARGO, DJURIDICA, TOTAL_HABERES	->	DREGION	169	DCARGO, JURIDICA, TOTAL_HABERES, UNIDAD	->	DPROVINCIA
23	AFP, DCARGO, DJURIDICA, TOTAL_HABERES	->	REGION	170	DCARGO, JURIDICA, TOTAL_HABERES, UNIDAD	->	PROVINCIA
24	AFP, DCARGO, DUNIDAD	->	DPLANTA	171	DCARGO, REMUNERACION	->	DISAPRE
25	AFP, DCARGO, DUNIDAD	->	PLANTA	172	DCARGO, REMUNERACION	->	DJURIDICA
26	AFP, DCARGO, ISAPRE, TOTAL_HABERES	->	DREGION	173	DCARGO, REMUNERACION	->	DPLANTA
27	AFP, DCARGO, ISAPRE, TOTAL_HABERES	->	REGION	174	DCARGO, REMUNERACION	->	ISAPRE
28	AFP, DCARGO, JURIDICA, TOTAL_HABERES	->	DREGION	175	DCARGO, REMUNERACION	->	JURIDICA
29	AFP, DCARGO, JURIDICA, TOTAL_HABERES	->	REGION	176	DCARGO, REMUNERACION	->	PLANTA
30	AFP, DCARGO, REMUNERACION	->	DPROVINCIA	177	DCARGO, REMUNERACION	->	TOTAL_HABERES
31	AFP, DCARGO, REMUNERACION	->	PROVINCIA	178	DCARGO, TOTAL_HABERES	->	DPLANTA
32	AFP, DCARGO, UNIDAD	->	DPLANTA	179	DCARGO, TOTAL_HABERES	->	PLANTA
33	AFP, DCARGO, UNIDAD	->	PLANTA	180	DCARGO, TOTAL_HABERES, UNIDAD	->	DREGION
34	AFP, DCOMUNA, DPLANTA, REMUNERACION	->	CARGO	181	DCARGO, TOTAL_HABERES, UNIDAD	->	REGION
35	AFP, DCOMUNA, DPLANTA, REMUNERACION	->	DCARGO	182	DCOMUNA	->	COMUNA
36	AFP, DCOMUNA, PLANTA, REMUNERACION	->	CARGO	183	DCOMUNA	->	DPROVINCIA
37	AFP, DCOMUNA, PLANTA, REMUNERACION	->	DCARGO	184	DCOMUNA	->	DREGION
38	AFP, DPLANTA, DPROVINCIA, REMUNERACION	->	CARGO	185	DCOMUNA	->	PROVINCIA
39	AFP, DPLANTA, DPROVINCIA, REMUNERACION	->	DCARGO	186	DCOMUNA	->	REGION
40	AFP, DPLANTA, PROVINCIA, REMUNERACION	->	CARGO	187	DISAPRE	->	ISAPRE
41	AFP, DPLANTA, PROVINCIA, REMUNERACION	->	DCARGO	188	DISAPRE, DREGION, DUNIDAD, TOTAL_HABERES	->	DPROVINCIA
42	AFP, DPROVINCIA, PLANTA, REMUNERACION	->	CARGO	189	DISAPRE, DREGION, DUNIDAD, TOTAL_HABERES	->	PROVINCIA
43	AFP, DPROVINCIA, PLANTA, REMUNERACION	->	DCARGO	190	DISAPRE, DREGION, TOTAL_HABERES, UNIDAD	->	DPROVINCIA
44	AFP, DUNIDAD, TOTAL_HABERES	->	DPROVINCIA	191	DISAPRE, DREGION, TOTAL_HABERES, UNIDAD	->	PROVINCIA
45	AFP, DUNIDAD, TOTAL_HABERES	->	DREGION	192	DISAPRE, DUNIDAD, REGION, TOTAL_HABERES	->	DPROVINCIA
46	AFP, DUNIDAD, TOTAL_HABERES	->	PROVINCIA	193	DISAPRE, DUNIDAD, REGION, TOTAL_HABERES	->	PROVINCIA
47	AFP, DUNIDAD, TOTAL_HABERES	->	REGION	194	DISAPRE, REGION, TOTAL_HABERES, UNIDAD	->	DPROVINCIA
48	AFP, PLANTA, PROVINCIA, REMUNERACION	->	CARGO	195	DISAPRE, REGION, TOTAL_HABERES, UNIDAD	->	PROVINCIA
49	AFP, PLANTA, PROVINCIA, REMUNERACION	->	DCARGO	196	DJURIDICA	->	JURIDICA
50	AFP, REMUNERACION	->	TOTAL_HABERES	197	DJURIDICA, DREGION, DUNIDAD, TOTAL_HABERES	->	DPROVINCIA
51	AFP, TOTAL_HABERES, UNIDAD	->	DPROVINCIA	198	DJURIDICA, DREGION, DUNIDAD, TOTAL_HABERES	->	PROVINCIA
52	AFP, TOTAL_HABERES, UNIDAD	->	DREGION	199	DJURIDICA, DREGION, TOTAL_HABERES, UNIDAD	->	DPROVINCIA
53	AFP, TOTAL_HABERES, UNIDAD	->	PROVINCIA	200	DJURIDICA, DREGION, TOTAL_HABERES, UNIDAD	->	PROVINCIA
54	AFP, TOTAL_HABERES, UNIDAD	->	REGION	201	DJURIDICA, DUNIDAD, REGION, TOTAL_HABERES	->	DPROVINCIA
55	CARGO	->	DCARGO	202	DJURIDICA, DUNIDAD, REGION, TOTAL_HABERES	->	PROVINCIA
56	CARGO, DAFP, DISAPRE, TOTAL_HABERES	->	DREGION	203	DJURIDICA, REGION, TOTAL_HABERES, UNIDAD	->	DPROVINCIA
57	CARGO, DAFP, DISAPRE, TOTAL_HABERES	->	REGION	204	DJURIDICA, REGION, TOTAL_HABERES, UNIDAD	->	PROVINCIA
58	CARGO, DAFP, DJURIDICA, TOTAL_HABERES	->	DREGION	205	DJURIDICA, REMUNERACION	->	DISAPRE
59	CARGO, DAFP, DJURIDICA, TOTAL_HABERES	->	REGION	206	DJURIDICA, REMUNERACION	->	ISAPRE
60	CARGO, DAFP, DUNIDAD	->	DPLANTA	207	DJURIDICA, REMUNERACION	->	TOTAL_HABERES
61	CARGO, DAFP, DUNIDAD	->	PLANTA	208	DPLANTA	->	PLANTA
62	CARGO, DAFP, ISAPRE, TOTAL_HABERES	->	DREGION	209	DPLANTA, DUNIDAD, REMUNERACION	->	CARGO
63	CARGO, DAFP, ISAPRE, TOTAL_HABERES	->	REGION	210	DPLANTA, DUNIDAD, REMUNERACION	->	DCARGO
64	CARGO, DAFP, JURIDICA, TOTAL_HABERES	->	DREGION	211	DPLANTA, REMUNERACION	->	DISAPRE
65	CARGO, DAFP, JURIDICA, TOTAL_HABERES	->	REGION	212	DPLANTA, REMUNERACION	->	DJURIDICA
66	CARGO, DAFP, REMUNERACION	->	DPROVINCIA	213	DPLANTA, REMUNERACION	->	ISAPRE
67	CARGO, DAFP, REMUNERACION	->	PROVINCIA	214	DPLANTA, REMUNERACION	->	JURIDICA
68	CARGO, DAFP, UNIDAD	->	DPLANTA	215	DPLANTA, REMUNERACION	->	TOTAL_HABERES
69	CARGO, DAFP, UNIDAD	->	PLANTA	216	DPLANTA, REMUNERACION, UNIDAD	->	CARGO
70	CARGO, DISAPRE, DUNIDAD, TOTAL_HABERES	->	DPROVINCIA	217	DPLANTA, REMUNERACION, UNIDAD	->	DCARGO
71	CARGO, DISAPRE, DUNIDAD, TOTAL_HABERES	->	PROVINCIA	218	DPROVINCIA	->	DREGION
72	CARGO, DISAPRE, TOTAL_HABERES, UNIDAD	->	DPROVINCIA	219	DPROVINCIA	->	PROVINCIA
73	CARGO, DISAPRE, TOTAL_HABERES, UNIDAD	->	PROVINCIA	220	DPROVINCIA	->	REGION
74	CARGO, DJURIDICA	->	DPLANTA	221	DREGION	->	REGION
75	CARGO, DJURIDICA	->	PLANTA	222	DREGION, DUNIDAD, ISAPRE, TOTAL_HABERES	->	DPROVINCIA
76	CARGO, DJURIDICA, DUNIDAD, TOTAL_HABERES	->	DPROVINCIA	223	DREGION, DUNIDAD, ISAPRE, TOTAL_HABERES	->	PROVINCIA
77	CARGO, DJURIDICA, DUNIDAD, TOTAL_HABERES	->	PROVINCIA	224	DREGION, DUNIDAD, JURIDICA, TOTAL_HABERES	->	DPROVINCIA
78	CARGO, DJURIDICA, TOTAL_HABERES, UNIDAD	->	DPROVINCIA	225	DREGION, DUNIDAD, JURIDICA, TOTAL_HABERES	->	PROVINCIA

79	CARGO, DJURIDICA, TOTAL_HABERES, UNIDAD	->	PROVINCIA	226	DREGION, ISAPRE, TOTAL_HABERES, UNIDAD	->	DPROVINCIA
80	CARGO, DUNIDAD, ISAPRE, TOTAL_HABERES	->	DPROVINCIA	227	DREGION, ISAPRE, TOTAL_HABERES, UNIDAD	->	PROVINCIA
81	CARGO, DUNIDAD, ISAPRE, TOTAL_HABERES	->	PROVINCIA	228	DREGION, JURIDICA, TOTAL_HABERES, UNIDAD	->	DPROVINCIA
82	CARGO, DUNIDAD, JURIDICA, TOTAL_HABERES	->	DPROVINCIA	229	DREGION, JURIDICA, TOTAL_HABERES, UNIDAD	->	PROVINCIA
83	CARGO, DUNIDAD, JURIDICA, TOTAL_HABERES	->	PROVINCIA	230	DUNIDAD	->	UNIDAD
84	CARGO, DUNIDAD, TOTAL_HABERES	->	DREGION	231	DUNIDAD, ISAPRE, REGION, TOTAL_HABERES	->	DPROVINCIA
85	CARGO, DUNIDAD, TOTAL_HABERES	->	REGION	232	DUNIDAD, ISAPRE, REGION, TOTAL_HABERES	->	PROVINCIA
86	CARGO, ISAPRE, TOTAL_HABERES, UNIDAD	->	DPROVINCIA	233	DUNIDAD, JURIDICA, REGION, TOTAL_HABERES	->	DPROVINCIA
87	CARGO, ISAPRE, TOTAL_HABERES, UNIDAD	->	PROVINCIA	234	DUNIDAD, JURIDICA, REGION, TOTAL_HABERES	->	PROVINCIA
88	CARGO, JURIDICA	->	DPLANTA	235	DUNIDAD, PLANTA, REMUNERACION	->	CARGO
89	CARGO, JURIDICA	->	PLANTA	236	DUNIDAD, PLANTA, REMUNERACION	->	DCARGO
90	CARGO, JURIDICA, TOTAL_HABERES, UNIDAD	->	DPROVINCIA	237	DUNIDAD, REMUNERACION	->	DISAPRE
91	CARGO, JURIDICA, TOTAL_HABERES, UNIDAD	->	PROVINCIA	238	DUNIDAD, REMUNERACION	->	DJURIDICA
92	CARGO, REMUNERACION	->	DISAPRE	239	DUNIDAD, REMUNERACION	->	DPROVINCIA
93	CARGO, REMUNERACION	->	DJURIDICA	240	DUNIDAD, REMUNERACION	->	ISAPRE
94	CARGO, REMUNERACION	->	DPLANTA	241	DUNIDAD, REMUNERACION	->	JURIDICA
95	CARGO, REMUNERACION	->	ISAPRE	242	DUNIDAD, REMUNERACION	->	PROVINCIA
96	CARGO, REMUNERACION	->	JURIDICA	243	DUNIDAD, REMUNERACION	->	TOTAL_HABERES
97	CARGO, REMUNERACION	->	PLANTA	244	ISAPRE	->	ES
98	CARGO, REMUNERACION	->	TOTAL_HABERES	245	ISAPRE, REGION, TOTAL_HABERES, UNIDAD	->	DPROVINCIA
99	CARGO, TOTAL_HABERES	->	DPLANTA	246	ISAPRE, REGION, TOTAL_HABERES, UNIDAD	->	PROVINCIA
100	CARGO, TOTAL_HABERES	->	PLANTA	247	JURIDICA	->	DJURIDICA
101	CARGO, TOTAL_HABERES, UNIDAD	->	DREGION	248	JURIDICA, REGION, TOTAL_HABERES, UNIDAD	->	DPROVINCIA
102	CARGO, TOTAL_HABERES, UNIDAD	->	REGION	249	JURIDICA, REGION, TOTAL_HABERES, UNIDAD	->	PROVINCIA
103	COMUNA	->	DCOMUNA	250	JURIDICA, REMUNERACION	->	DISAPRE
104	COMUNA	->	DPROVINCIA	251	JURIDICA, REMUNERACION	->	ISAPRE
105	COMUNA	->	DREGION	252	JURIDICA, REMUNERACION	->	TOTAL_HABERES
106	COMUNA	->	PROVINCIA	253	PLANTA	->	DPLANTA
107	COMUNA	->	REGION	254	PLANTA, REMUNERACION	->	DISAPRE
108	COMUNA, DAFP, DPLANTA, REMUNERACION	->	CARGO	255	PLANTA, REMUNERACION	->	DJURIDICA
109	COMUNA, DAFP, DPLANTA, REMUNERACION	->	DCARGO	256	PLANTA, REMUNERACION	->	ISAPRE
110	COMUNA, DAFP, PLANTA, REMUNERACION	->	CARGO	257	PLANTA, REMUNERACION	->	JURIDICA
111	COMUNA, DAFP, PLANTA, REMUNERACION	->	DCARGO	258	PLANTA, REMUNERACION	->	TOTAL_HABERES
112	DAFP	->	AFP	259	PLANTA, REMUNERACION, UNIDAD	->	CARGO
113	DAFP, DCARGO, DISAPRE, TOTAL_HABERES	->	DREGION	260	PLANTA, REMUNERACION, UNIDAD	->	DCARGO
114	DAFP, DCARGO, DISAPRE, TOTAL_HABERES	->	REGION	261	PROVINCIA	->	DPROVINCIA
115	DAFP, DCARGO, DJURIDICA, TOTAL_HABERES	->	DREGION	262	PROVINCIA	->	DREGION
116	DAFP, DCARGO, DJURIDICA, TOTAL_HABERES	->	REGION	263	PROVINCIA	->	REGION
117	DAFP, DCARGO, DUNIDAD	->	DPLANTA	264	REGION	->	DREGION
118	DAFP, DCARGO, DUNIDAD	->	PLANTA	265	REMUNERACION	->	DREGION
119	DAFP, DCARGO, ISAPRE, TOTAL_HABERES	->	DREGION	266	REMUNERACION	->	REGION
120	DAFP, DCARGO, ISAPRE, TOTAL_HABERES	->	REGION	267	REMUNERACION, UNIDAD	->	DISAPRE
121	DAFP, DCARGO, JURIDICA, TOTAL_HABERES	->	DREGION	268	REMUNERACION, UNIDAD	->	DJURIDICA
122	DAFP, DCARGO, JURIDICA, TOTAL_HABERES	->	REGION	269	REMUNERACION, UNIDAD	->	DPROVINCIA
123	DAFP, DCARGO, REMUNERACION	->	DPROVINCIA	270	REMUNERACION, UNIDAD	->	ISAPRE

3							
12	4	DAFP, DCARGO, REMUNERACION	->	PROVINCIA	271 REMUNERACION, UNIDAD	->	JURIDICA
12	5	DAFP, DCARGO, UNIDAD	->	DPLANTA	272 REMUNERACION, UNIDAD	->	PROVINCIA
12	6	DAFP, DCARGO, UNIDAD	->	PLANTA	273 REMUNERACION, UNIDAD	->	TOTAL_HABERES
12	7	DAFP, DCOMUNA, DPLANTA, REMUNERACION	->	CARGO	274 RUT	->	AFP
12	8	DAFP, DCOMUNA, DPLANTA, REMUNERACION	->	DCARGO	275 RUT	->	CARGO
12	9	DAFP, DCOMUNA, PLANTA, REMUNERACION	->	CARGO	276 RUT	->	COMUNA
13	0	DAFP, DCOMUNA, PLANTA, REMUNERACION	->	DCARGO	277 RUT	->	DAFP
13	1	DAFP, DPLANTA, DPROVINCIA, REMUNERACION	->	CARGO	278 RUT	->	DCARGO
13	2	DAFP, DPLANTA, DPROVINCIA, REMUNERACION	->	DCARGO	279 RUT	->	DCOMUNA
13	3	DAFP, DPLANTA, PROVINCIA, REMUNERACION	->	CARGO	280 RUT	->	DISAPRE
13	4	DAFP, DPLANTA, PROVINCIA, REMUNERACION	->	DCARGO	281 RUT	->	DJURIDICA
13	5	DAFP, DPROVINCIA, PLANTA, REMUNERACION	->	CARGO	282 RUT	->	DPLANTA
13	6	DAFP, DPROVINCIA, PLANTA, REMUNERACION	->	DCARGO	283 RUT	->	DPROVINCIA
13	7	DAFP, DUNIDAD, TOTAL_HABERES	->	DPROVINCIA	284 RUT	->	DREGION
13	8	DAFP, DUNIDAD, TOTAL_HABERES	->	DREGION	285 RUT	->	DUNIDAD
13	9	DAFP, DUNIDAD, TOTAL_HABERES	->	PROVINCIA	286 RUT	->	ISAPRE
14	0	DAFP, DUNIDAD, TOTAL_HABERES	->	REGION	287 RUT	->	JURIDICA
14	1	DAFP, PLANTA, PROVINCIA, REMUNERACION	->	CARGO	288 RUT	->	PLANTA
14	2	DAFP, PLANTA, PROVINCIA, REMUNERACION	->	DCARGO	289 RUT	->	PROVINCIA
14	3	DAFP, REMUNERACION	->	TOTAL_HABERES	290 RUT	->	REGION
14	4	DAFP, TOTAL_HABERES, UNIDAD	->	DPROVINCIA	291 RUT	->	REMUNERACION
14	5	DAFP, TOTAL_HABERES, UNIDAD	->	DREGION	292 RUT	->	TOTAL_HABERES
14	6	DAFP, TOTAL_HABERES, UNIDAD	->	PROVINCIA	293 RUT	->	UNIDAD
14	7	DAFP, TOTAL_HABERES, UNIDAD	->	REGION	294 UNIDAD	->	DUNIDAD

**Tabla C.2** DFs automáticas tabla SABST030

**Tabla SAURT009**

N°	Determinante	->	Determinado	N°	Determinante	->	Determinado
1	CODIGO	->	CP	17	DDIAGPOS2, DPERSTRATANTE, DSERVSOL, FP, 3 MEDIO	->	SERVDEST
2	CODIGO	->	DDIAGPOS1	17	DDIAGPOS2, DPERSTRATANTE, DSERVSOL, FP, 4 SERVDEST	->	DMOTIVO
3	CODIGO	->	DDIAGPOS2	17	DDIAGPOS2, DPERSTRATANTE, DSERVSOL, FP, 5 SERVDEST	->	MOTIVO
4	CODIGO	->	DFP	17	DDIAGPOS2, DPERSTRATANTE, DSERVSOL, 6 SERVDEST	->	DPROFEJE
5	CODIGO	->	DIAGPOS1	17	DDIAGPOS2, DPERSTRATANTE, DSERVSOL, 7 SERVDEST	->	PROFEJE
6	CODIGO	->	DIAGPOS2	17	DDIAGPOS2, DPERSTRATANTE, FP, 8	->	DPROFEJE

7	CODIGO	->	DLUGAR	17	9	DDIAGPOS2, DPERSTRATANTE, FP	->	PROFEJE
8	CODIGO	->	DMEDIO	18	0	DDIAGPOS2, DPERSTRATANTE, FP, LUGAR	->	DMOTIVO
9	CODIGO	->	DMOTIVO	18	1	DDIAGPOS2, DPERSTRATANTE, FP, LUGAR	->	MOTIVO
10	CODIGO	->	DPERSTRATANTE	18	2	DDIAGPOS2, DPERSTRATANTE, FP, LUGAR, MEDIO	->	DSERVDEST
11	CODIGO	->	DPROFEJE	18	3	DDIAGPOS2, DPERSTRATANTE, FP, LUGAR, MEDIO	->	SERVDEST
12	CODIGO	->	DPROFSOL	18	4	DDIAGPOS2, DPERSTRATANTE, FP, MEDIO	->	DMOTIVO
13	CODIGO	->	DSERVDEST	18	5	DDIAGPOS2, DPERSTRATANTE, FP, MEDIO	->	MOTIVO
14	CODIGO	->	DSERVEJE	18	6	DDIAGPOS2, DPERSTRATANTE, FP, MEDIO, SERV SOL	->	DSERVDEST
15	CODIGO	->	DSERVSOL	18	7	DDIAGPOS2, DPERSTRATANTE, FP, MEDIO, SERV SOL	->	SERVDEST
16	CODIGO	->	FP	18	8	DDIAGPOS2, DPERSTRATANTE, FP, SERVDEST, SERV SOL	->	DMOTIVO
17	CODIGO	->	GRUPO	18	9	DDIAGPOS2, DPERSTRATANTE, FP, SERVDEST, SERV SOL	->	MOTIVO
18	CODIGO	->	LUGAR	19	0	DDIAGPOS2, DPERSTRATANTE, LUGAR, SERVDEST	->	DPROFEJE
19	CODIGO	->	MEDIO	19	1	DDIAGPOS2, DPERSTRATANTE, LUGAR, SERVDEST	->	PROFEJE
20	CODIGO	->	MOTIVO	19	2	DDIAGPOS2, DPERSTRATANTE, MEDIO	->	DPROFEJE
21	CODIGO	->	DPERSTRATANTE	19	3	DDIAGPOS2, DPERSTRATANTE, MEDIO	->	PROFEJE
22	CODIGO	->	PROFEJE	19	4	DDIAGPOS2, DPERSTRATANTE, SERVDEST, SERV SOL	->	DPROFEJE
23	CODIGO	->	PROFSOL	19	5	DDIAGPOS2, DPERSTRATANTE, SERVDEST, SERV SOL	->	PROFEJE
24	CODIGO	->	SERVDEST	19	6	DDIAGPOS2, DPROFEJE	->	DPROFSOL
25	CODIGO	->	SERVEJE	19	7	DDIAGPOS2, DPROFEJE	->	PROFSOL
26	CODIGO	->	SERVSOL	19	8	DDIAGPOS2, DPROFEJE, DSERVDEST, DSERVSOL, FP	->	DMOTIVO
27	DDIAGPOS1	->	DSERVEJE	19	9	DDIAGPOS2, DPROFEJE, DSERVDEST, DSERVSOL, FP	->	MOTIVO
28	DDIAGPOS1	->	SERVEJE	20	0	DDIAGPOS2, DPROFEJE, DSERVDEST, FP, SERV SOL	->	DMOTIVO
29	DDIAGPOS1, DDIAGPOS2	->	DPROFSOL	20	1	DDIAGPOS2, DPROFEJE, DSERVDEST, FP, SERV SOL	->	MOTIVO
30	DDIAGPOS1, DDIAGPOS2	->	PROFSOL	20	2	DDIAGPOS2, DPROFEJE, DSERV SOL, FP, MEDIO	->	DSERVDEST
31	DDIAGPOS1, DDIAGPOS2, DFP, DLUGAR, DPERSTRATANTE	->	DSERVDEST	20	3	DDIAGPOS2, DPROFEJE, DSERV SOL, FP, MEDIO	->	SERVDEST
32	DDIAGPOS1, DDIAGPOS2, DFP, DLUGAR, DPERSTRATANTE	->	SERVDEST	20	4	DDIAGPOS2, DPROFEJE, DSERV SOL, FP, SERVDEST	->	DMOTIVO
33	DDIAGPOS1, DDIAGPOS2, DFP, DLUGAR, DPERSTRATANTE	->	DSERVDEST	20	5	DDIAGPOS2, DPROFEJE, DSERV SOL, FP, SERVDEST	->	MOTIVO
34	DDIAGPOS1, DDIAGPOS2, DFP, DLUGAR, DPERSTRATANTE	->	SERVDEST	20	6	DDIAGPOS2, DPROFEJE, FP	->	DPERSTRATANTE
35	DDIAGPOS1, DDIAGPOS2, DFP, DPERSTRATANTE	->	DMOTIVO	20	7	DDIAGPOS2, DPROFEJE, FP	->	DPERSTRATANTE
36	DDIAGPOS1, DDIAGPOS2, DFP, DPERSTRATANTE	->	MOTIVO	20	8	DDIAGPOS2, DPROFEJE, FP, LUGAR	->	DMOTIVO
37	DDIAGPOS1, DDIAGPOS2, DFP, DPERSTRATANTE, DSERVSOL	->	DSERVDEST	20	9	DDIAGPOS2, DPROFEJE, FP, LUGAR	->	MOTIVO
38	DDIAGPOS1, DDIAGPOS2, DFP, DPERSTRATANTE, DSERVSOL	->	SERVDEST	21	0	DDIAGPOS2, DPROFEJE, FP, LUGAR, MEDIO	->	DSERVDEST
39	DDIAGPOS1, DDIAGPOS2, DFP, DPERSTRATANTE, LUGAR	->	DSERVDEST	21	1	DDIAGPOS2, DPROFEJE, FP, LUGAR, MEDIO	->	SERVDEST
40	DDIAGPOS1, DDIAGPOS2, DFP, DPERSTRATANTE, LUGAR	->	SERVDEST	21	2	DDIAGPOS2, DPROFEJE, FP, MEDIO	->	DMOTIVO
41	DDIAGPOS1, DDIAGPOS2, DFP, DPERSTRATANTE, SERVSOL	->	DSERVDEST	21	3	DDIAGPOS2, DPROFEJE, FP, MEDIO	->	MOTIVO
42	DDIAGPOS1, DDIAGPOS2, DFP, DPERSTRATANTE, SERVSOL	->	SERVDEST	21	4	DDIAGPOS2, DPROFEJE, FP, MEDIO, SERVSOL	->	DSERVDEST

SERVSOL		4	
43	DDIAGPOS1, DDIAGPOS2, DFP, DPROFEJE	->	DMOTIVO
		21	5 DDIAGPOS2, DPROFEJE, FP, MEDIO, SERVSOL -> SERVDEST
44	DDIAGPOS1, DDIAGPOS2, DFP, DPROFEJE	->	MOTIVO
		21	6 DDIAGPOS2, DPROFEJE, FP, SERVDEST, SERVSOL -> DMOTIVO
45	DDIAGPOS1, DDIAGPOS2, DFP, PROFEJE	->	DMOTIVO
		21	7 DDIAGPOS2, DPROFEJE, FP, SERVDEST, SERVSOL -> MOTIVO
46	DDIAGPOS1, DDIAGPOS2, DFP, PROFEJE	->	MOTIVO
		21	8 DDIAGPOS2, DPROFEJE, LUGAR -> DPERSTRATANTE
47	DDIAGPOS1, DDIAGPOS2, DFP, PROFEJE, SERVSOL	->	DSERVDEST
		21	9 DDIAGPOS2, DPROFEJE, LUGAR -> PERSTRATANTE
48	DDIAGPOS1, DDIAGPOS2, DFP, PROFEJE, SERVSOL	->	SERVDEST
		22	0 DDIAGPOS2, DPROFEJE, MEDIO -> DPERSTRATANTE
49	DDIAGPOS1, DDIAGPOS2, DLUGAR, DMOTIVO, DPERSTRATANTE	->	DSERVDEST
		22	1 DDIAGPOS2, DPROFEJE, MEDIO -> PERSTRATANTE
50	DDIAGPOS1, DDIAGPOS2, DLUGAR, DMOTIVO, DPERSTRATANTE	->	SERVDEST
		22	2 DDIAGPOS2, DSERVDEST, DSERVOL, FP, PERSTRATANTE -> DMOTIVO
51	DDIAGPOS1, DDIAGPOS2, DLUGAR, DMOTIVO, PERSTRATANTE	->	DSERVDEST
		22	3 DDIAGPOS2, DSERVDEST, DSERVOL, FP, PERSTRATANTE -> MOTIVO
52	DDIAGPOS1, DDIAGPOS2, DLUGAR, DMOTIVO, PERSTRATANTE	->	SERVDEST
		22	4 DDIAGPOS2, DSERVDEST, DSERVOL, FP, PROFEJE -> DMOTIVO
53	DDIAGPOS1, DDIAGPOS2, DLUGAR, DPERSTRATANTE, DSERVOL	->	DMOTIVO
		22	5 DDIAGPOS2, DSERVDEST, DSERVOL, PROFEJE -> MOTIVO
54	DDIAGPOS1, DDIAGPOS2, DLUGAR, DPERSTRATANTE, DSERVOL	->	DSERVDEST
		22	6 DDIAGPOS2, DSERVDEST, DSERVOL, PERSTRATANTE -> DPROFEJE
55	DDIAGPOS1, DDIAGPOS2, DLUGAR, DPERSTRATANTE, DSERVOL	->	MOTIVO
		22	7 DDIAGPOS2, DSERVDEST, DSERVOL, PERSTRATANTE -> PROFEJE
56	DDIAGPOS1, DDIAGPOS2, DLUGAR, DPERSTRATANTE, DSERVOL	->	SERVDEST
		22	8 DDIAGPOS2, DSERVDEST, FP, PERSTRATANTE, SERVSOL -> DMOTIVO
57	DDIAGPOS1, DDIAGPOS2, DLUGAR, DPERSTRATANTE, FP	->	DSERVDEST
		22	9 DDIAGPOS2, DSERVDEST, FP, PERSTRATANTE, SERVSOL -> MOTIVO
58	DDIAGPOS1, DDIAGPOS2, DLUGAR, DPERSTRATANTE, FP	->	SERVDEST
		23	0 DDIAGPOS2, DSERVDEST, FP, PROFEJE, SERVSOL -> DMOTIVO
59	DDIAGPOS1, DDIAGPOS2, DLUGAR, DPERSTRATANTE, MOTIVO	->	DSERVDEST
		23	1 DDIAGPOS2, DSERVDEST, FP, PROFEJE, SERVSOL -> MOTIVO
60	DDIAGPOS1, DDIAGPOS2, DLUGAR, DPERSTRATANTE, MOTIVO	->	SERVDEST
		23	2 DDIAGPOS2, DSERVDEST, LUGAR, PERSTRATANTE -> DPROFEJE
61	DDIAGPOS1, DDIAGPOS2, DLUGAR, DPERSTRATANTE, SERVSOL	->	DMOTIVO
		23	3 DDIAGPOS2, DSERVDEST, LUGAR, PERSTRATANTE -> PROFEJE
62	DDIAGPOS1, DDIAGPOS2, DLUGAR, DPERSTRATANTE, SERVSOL	->	DSERVDEST
		23	4 DDIAGPOS2, DSERVDEST, PERSTRATANTE, SERVSOL -> DPROFEJE
63	DDIAGPOS1, DDIAGPOS2, DLUGAR, DPERSTRATANTE, SERVSOL	->	MOTIVO
		23	5 DDIAGPOS2, DSERVDEST, PERSTRATANTE, SERVSOL -> PROFEJE
64	DDIAGPOS1, DDIAGPOS2, DLUGAR, MOTIVO, PERSTRATANTE	->	SERVDEST
		23	6 DDIAGPOS2, DSERVOL -> DPROFSOL
65	DDIAGPOS1, DDIAGPOS2, DLUGAR, PERSTRATANTE, SERVSOL	->	DMOTIVO
		23	7 DDIAGPOS2, DSERVOL -> PROFSOL
66	DDIAGPOS1, DDIAGPOS2, DLUGAR, PERSTRATANTE, SERVSOL	->	DSERVDEST
		23	8 DDIAGPOS2, DSERVOL, FP, MEDIO, PERSTRATANTE -> DSERVDEST
67	DDIAGPOS1, DDIAGPOS2, DLUGAR, PERSTRATANTE, SERVSOL	->	MOTIVO
		23	9 DDIAGPOS2, DSERVOL, FP, MEDIO, PERSTRATANTE -> SERVDEST
68	DDIAGPOS1, DDIAGPOS2, DLUGAR, PERSTRATANTE, SERVSOL	->	SERVDEST
		24	0 DDIAGPOS2, DSERVOL, FP, MEDIO, PROFEJE -> DSERVDEST
69	DDIAGPOS1, DDIAGPOS2, DLUGAR, PROFEJE	->	DSERVDEST
		24	1 DDIAGPOS2, DSERVOL, FP, MEDIO, PROFEJE -> SERVDEST
70	DDIAGPOS1, DDIAGPOS2, DLUGAR, PROFEJE	->	SERVDEST
		24	2 DDIAGPOS2, FP, MEDIO, PERSTRATANTE -> DMOTIVO
71	DDIAGPOS1, DDIAGPOS2, DLUGAR, PROFEJE, SERVSOL	->	DMOTIVO
		24	3 DDIAGPOS2, FP, MEDIO, PERSTRATANTE -> MOTIVO
72	DDIAGPOS1, DDIAGPOS2, DLUGAR, PROFEJE, SERVSOL	->	MOTIVO
		24	4 DDIAGPOS2, FP, MEDIO, PERSTRATANTE, SERVSOL -> DSERVDEST
73	DDIAGPOS1, DDIAGPOS2, DMEDIO, DPERSTRATANTE	->	DSERVDEST
		24	5 DDIAGPOS2, FP, MEDIO, PERSTRATANTE, SERVSOL -> SERVDEST
74	DDIAGPOS1, DDIAGPOS2, DMEDIO, DPERSTRATANTE	->	SERVDEST
		24	6 DDIAGPOS2, FP, MEDIO, PROFEJE -> DMOTIVO
75	DDIAGPOS1, DDIAGPOS2, DMEDIO, DPROFEJE	->	DSERVDEST
		24	7 DDIAGPOS2, FP, MEDIO, PROFEJE -> MOTIVO
76	DDIAGPOS1, DDIAGPOS2, DMEDIO, DPROFEJE	->	SERVDEST
		24	8 DDIAGPOS2, FP, MEDIO, PROFEJE, SERVSOL -> DSERVDEST
77	DDIAGPOS1, DDIAGPOS2, DMEDIO, PERSTRATANTE	->	DSERVDEST
		24	9 DDIAGPOS2, FP, MEDIO, PROFEJE, SERVSOL -> SERVDEST

78	DDIAGPOS1, DDIAGPOS2, DMEDIO, PERSTRATANTE	->	SERVDEST	25	0	DDIAGPOS2, FP, PERSTRATANTE	->	DPROFEJE
79	DDIAGPOS1, DDIAGPOS2, DMEDIO, PROFEJE	->	DSERVDEST	25	1	DDIAGPOS2, FP, PERSTRATANTE	->	PROFEJE
80	DDIAGPOS1, DDIAGPOS2, DMEDIO, PROFEJE	->	SERVDEST	25	2	DDIAGPOS2, FP, PERSTRATANTE, SERVDEST, SERV SOL	->	DMOTIVO
81	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DPERSTRATANTE	->	DPROFEJE	25	3	DDIAGPOS2, FP, PERSTRATANTE, SERVDEST, SERV SOL	->	MOTIVO
82	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DPERSTRATANTE	->	PROFEJE	25	4	DDIAGPOS2, FP, PROFEJE	->	DPERSTRATANTE
83	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DPERSTRATANTE, DSERV SOL	->	DSERVDEST	25	5	DDIAGPOS2, FP, PROFEJE	->	PERSTRATANTE
84	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DPERSTRATANTE, DSERV SOL	->	SERVDEST	25	6	DDIAGPOS2, FP, PROFEJE, SERVDEST, SERV SOL	->	DMOTIVO
85	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DPERSTRATANTE, LUGAR	->	DSERVDEST	25	7	DDIAGPOS2, FP, PROFEJE, SERVDEST, SERV SOL	->	MOTIVO
86	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DPERSTRATANTE, LUGAR	->	SERVDEST	25	8	DDIAGPOS2, LUGAR, PERSTRATANTE, SERVDEST	->	DPROFEJE
87	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DPERSTRATANTE, SERV SOL	->	DSERVDEST	25	9	DDIAGPOS2, LUGAR, PERSTRATANTE, SERVDEST	->	PROFEJE
88	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DPERSTRATANTE, SERV SOL	->	SERVDEST	26	0	DDIAGPOS2, LUGAR, PROFEJE	->	DPERSTRATANTE
89	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DPROFEJE, DSERV SOL	->	DSERVDEST	26	1	DDIAGPOS2, LUGAR, PROFEJE	->	PERSTRATANTE
90	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DPROFEJE, DSERV SOL	->	SERVDEST	26	2	DDIAGPOS2, MEDIO, PERSTRATANTE	->	DPROFEJE
91	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DPROFEJE, SERV SOL	->	DSERVDEST	26	3	DDIAGPOS2, MEDIO, PERSTRATANTE	->	PROFEJE
92	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DPROFEJE, SERV SOL	->	SERVDEST	26	4	DDIAGPOS2, MEDIO, PROFEJE	->	DPERSTRATANTE
93	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DSERV SOL, PERSTRATANTE	->	DSERVDEST	26	5	DDIAGPOS2, MEDIO, PROFEJE	->	PERSTRATANTE
94	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DSERV SOL, PERSTRATANTE	->	SERVDEST	26	6	DDIAGPOS2, PERSTRATANTE	->	DPROFSOL
95	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DSERV SOL, PROFEJE	->	DSERVDEST	26	7	DDIAGPOS2, PERSTRATANTE	->	PROFSOL
96	DDIAGPOS1, DDIAGPOS2, DMOTIVO, DSERV SOL, PROFEJE	->	SERVDEST	26	8	DDIAGPOS2, PERSTRATANTE, SERVDEST, SERV SOL	->	DPROFEJE
97	DDIAGPOS1, DDIAGPOS2, DMOTIVO, LUGAR, PERSTRATANTE	->	DSERVDEST	26	9	DDIAGPOS2, PERSTRATANTE, SERVDEST, SERV SOL	->	PROFEJE
98	DDIAGPOS1, DDIAGPOS2, DMOTIVO, LUGAR, PERSTRATANTE	->	SERVDEST	27	0	DDIAGPOS2, PROFEJE	->	DPROFSOL
99	DDIAGPOS1, DDIAGPOS2, DMOTIVO, PERSTRATANTE	->	DPROFEJE	27	1	DDIAGPOS2, PROFEJE	->	PROFSOL
100	DDIAGPOS1, DDIAGPOS2, DMOTIVO, PERSTRATANTE	->	PROFEJE	27	2	DDIAGPOS2, SERV SOL	->	DPROFSOL
101	DDIAGPOS1, DDIAGPOS2, DMOTIVO, PERSTRATANTE, SERV SOL	->	DSERVDEST	27	3	DDIAGPOS2, SERV SOL	->	PROFSOL
102	DDIAGPOS1, DDIAGPOS2, DMOTIVO, PERSTRATANTE, SERV SOL	->	SERVDEST	27	4	DDIAGPOS2	->	DIAGPOS2
103	DDIAGPOS1, DDIAGPOS2, DMOTIVO, PROFEJE, SERV SOL	->	DSERVDEST	27	5	DFP	->	FP
104	DDIAGPOS1, DDIAGPOS2, DMOTIVO, PROFEJE, SERV SOL	->	SERVDEST	27	6	DFP	->	DSERVEJE
105	DDIAGPOS1, DDIAGPOS2, DPERSTRATANTE, DSERVDEST	->	DPROFEJE	27	7	DFP	->	SERVEJE
106	DDIAGPOS1, DDIAGPOS2, DPERSTRATANTE, DSERVDEST	->	PROFEJE	27	8	DFP, DIAGPOS1	->	DPROFSOL
107	DDIAGPOS1, DDIAGPOS2, DPERSTRATANTE, DSERV SOL	->	DPROFEJE	27	9	DFP, DIAGPOS1	->	PROFSOL
108	DDIAGPOS1, DDIAGPOS2, LUGAR, PROFEJE, SERV SOL	->	MOTIVO	28	0	DFP, DPROFEJE	->	PROFSOL
109	DDIAGPOS1, DDIAGPOS2, MEDIO, PERSTRATANTE	->	DSERVDEST	28	1	DFP, PERSTRATANTE	->	DPROFSOL
110	DDIAGPOS1, DDIAGPOS2, MEDIO, PERSTRATANTE	->	SERVDEST	28	2	DFP, PERSTRATANTE	->	PROFSOL
111	DDIAGPOS1, DDIAGPOS2, MEDIO, PROFEJE	->	DSERVDEST	28	3	DFP, PROFEJE	->	DPROFSOL
112	DDIAGPOS1, DDIAGPOS2, MEDIO, PROFEJE	->	SERVDEST	28	4	DFP, PROFEJE	->	PROFSOL
113	DDIAGPOS1, DDIAGPOS2, MOTIVO, PERSTRATANTE	->	DPROFEJE	28	DIAGPOS1	->	DDIAGPOS1	

3		5	
11		28	
4	DDIAGPOS1, DDIAGPOS2, MOTIVO, PERSTRATANTE -> PROFEJE	6	DIAGPOS1 -> DSERVEJE
11	DDIAGPOS1, DDIAGPOS2, MOTIVO, PERSTRATANTE, SERVSOL -> DSERVDEST	28	7 DIAGPOS1 -> SERVEJE
11	DDIAGPOS1, DDIAGPOS2, MOTIVO, PERSTRATANTE, SERVSOL -> SERVDEST	28	8 DIAGPOS1, DIAGPOS2 -> DPROFSOL
11	DDIAGPOS1, DDIAGPOS2, MOTIVO, PROFEJE, SERVSOL -> DSERVDEST	28	9 DIAGPOS1, DIAGPOS2 -> PROFSOL
11	DDIAGPOS1, DDIAGPOS2, MOTIVO, PROFEJE, SERVSOL -> SERVDEST	29	0 DIAGPOS1, PROFEJE -> DPROFSOL
11	DDIAGPOS1, DDIAGPOS2, PERSTRATANTE, SERVDEST -> DPROFEJE	29	1 DIAGPOS1, PROFEJE -> PROFSOL
12	DDIAGPOS1, DDIAGPOS2, PERSTRATANTE, SERVDEST -> PROFEJE	29	2 DIAGPOS2 -> DDIAGPOS2
12	DDIAGPOS1, DDIAGPOS2, PERSTRATANTE, SERVSOL -> DPROFEJE	29	3 DIAGPOS2 -> DDIAGPOS2
12	DDIAGPOS1, DDIAGPOS2, PERSTRATANTE, SERVSOL -> PROFEJE	29	4 DIAGPOS2 -> DSERVEJE
12	DDIAGPOS1, DDIAGPOS2, PROFEJE -> DPERSTRATANTE	29	5 DIAGPOS2 -> SERVEJE
12	DDIAGPOS1, DDIAGPOS2, PROFEJE -> PERSTRATANTE	29	6 DLUGAR -> DSERVEJE
12	DDIAGPOS1, DFP -> DPROFSOL	29	7 DLUGAR -> LUGAR
12	DDIAGPOS1, DFP -> PROFSOL	29	8 DLUGAR -> SERVEJE
12	DDIAGPOS1, DFP, DIAGPOS2, DLUGAR, DPERSTRATANTE -> DSERVDEST	29	9 DMEDIO -> DSERVEJE
12	DDIAGPOS1, DFP, DIAGPOS2, DLUGAR, DPERSTRATANTE -> SERVDEST	30	0 DMEDIO -> MEDIO
12	DDIAGPOS1, DFP, DIAGPOS2, DLUGAR, PERSTRATANTE -> DSERVDEST	30	1 DMEDIO -> SERVEJE
13	DDIAGPOS1, DFP, DIAGPOS2, DPROFEJE -> DMOTIVO	30	2 DMOTIVO -> DSERVEJE
13	DDIAGPOS1, DFP, DIAGPOS2, DPROFEJE -> MOTIVO	30	3 DMOTIVO -> MOTIVO
13	DDIAGPOS1, DFP, DIAGPOS2, DPROFEJE, DSERVOL -> DSERVDEST	30	4 DMOTIVO -> SERVEJE
13	DDIAGPOS1, DFP, DIAGPOS2, DPROFEJE, DSERVOL -> SERVDEST	30	5 DPERSTRATANTE -> DSERVEJE
13	DDIAGPOS1, DFP, DIAGPOS2, DPROFEJE, SERVSOL -> DSERVDEST	30	6 DPERSTRATANTE -> PERSTRATANTE
13	DDIAGPOS1, DFP, DIAGPOS2, DPROFEJE, SERVSOL -> SERVDEST	30	7 DPERSTRATANTE -> SERVEJE
13	DDIAGPOS1, DFP, DIAGPOS2, DSERVOL, PERSTRATANTE -> DSERVDEST	30	8 DPERSTRATANTE, FP -> DPROFSOL
13	DDIAGPOS1, DFP, DIAGPOS2, DSERVOL, PERSTRATANTE -> SERVDEST	30	9 DPROFEJE -> DSERVEJE
13	DDIAGPOS1, DFP, DIAGPOS2, DSERVOL, PROFEJE -> DSERVDEST	31	0 DPROFEJE -> PROFEJE
13	DDIAGPOS1, DFP, DIAGPOS2, DSERVOL, PROFEJE -> SERVDEST	31	1 DPROFEJE -> SERVEJE
14	DDIAGPOS1, DFP, DIAGPOS2, LUGAR, PERSTRATANTE -> DSERVDEST	31	2 DPROFEJE, FP -> DPROFSOL
14	DDIAGPOS1, DFP, DIAGPOS2, LUGAR, PERSTRATANTE -> SERVDEST	31	3 DPROFEJE, FP -> PROFSOL
14	DDIAGPOS1, DFP, DIAGPOS2, PERSTRATANTE -> DMOTIVO	31	4 DPROFSOL -> DSERVEJE
14	DDIAGPOS1, DFP, DIAGPOS2, PERSTRATANTE -> MOTIVO	31	5 DPROFSOL -> PROFSOL
14	DDIAGPOS1, DFP, DIAGPOS2, PERSTRATANTE, SERVSOL -> DSERVDEST	31	6 DPROFSOL -> SERVEJE
14	DDIAGPOS1, DFP, DIAGPOS2, PERSTRATANTE, SERVSOL -> SERVDEST	31	7 DSERVDEST -> DSERVEJE
14	DDIAGPOS1, DFP, DIAGPOS2, PROFEJE -> DMOTIVO	31	8 DSERVDEST -> SERVDEST
14	DDIAGPOS1, DFP, DIAGPOS2, PROFEJE -> MOTIVO	31	9 DSERVDEST -> SERVEJE
14	DDIAGPOS1, DFP, DMEDIO, DSERVOL, PERSTRATANTE -> MOTIVO	32	0 DSERVEJE -> SERVEJE

14	9	DDIAGPOS1, DFP, MEDIO, PROFEJE, SERVSOL	->	DMOTIVO	32	1	DSERVSOL	->	DSERVEJE
15	0	DDIAGPOS1, DFP, MEDIO, PROFEJE, SERVSOL	->	MOTIVO	32	2	DSERVSOL	->	SERVSOL
15	1	DDIAGPOS1, DIAGPOS2	->	DPROFSOL	32	3	FP	->	DFP
15	2	DDIAGPOS1, DIAGPOS2	->	PROFSOL	32	4	FP	->	DSERVEJE
15	3	DDIAGPOS1, FP	->	DPROFSOL	32	5	FP	->	SERVEJE
15	4	DDIAGPOS1, FP	->	PROFSOL	32	6	FP, PERSTRATANTE	->	DPROFSOL
15	5	DDIAGPOS1, FP, LUGAR, PROFEJE, SERVDEST, SERVSOL	->	DMOTIVO	32	7	FP, PERSTRATANTE	->	PROFSOL
15	6	DDIAGPOS1, FP, LUGAR, PROFEJE, SERVDEST, SERVSOL	->	MOTIVO	32	8	FP, PROFEJE	->	DPROFSOL
15	7	DDIAGPOS1, FP, MEDIO, PERSTRATANTE, SERVSOL	->	DMOTIVO	32	9	FP, PROFEJE	->	PROFSOL
15	8	DDIAGPOS1, FP, MEDIO, PERSTRATANTE, SERVSOL	->	MOTIVO	33	0	LUGAR	->	DLUGAR
15	9	DDIAGPOS1, FP, MEDIO, PROFEJE, SERVSOL	->	DMOTIVO	33	1	LUGAR	->	SERVEJE
16	0	DDIAGPOS1, FP, MEDIO, PROFEJE, SERVSOL	->	MOTIVO	33	2	MEDIO	->	SERVEJE
16	1	DDIAGPOS1, PERSTRATANTE	->	DPROFSOL	33	3	MEDIO	->	DMEDIO
16	2	DDIAGPOS1, PERSTRATANTE	->	PROFSOL	33	4	MOTIVO	->	SERVEJE
16	3	DDIAGPOS1, PROFEJE	->	DPROFSOL	33	5	MOTIVO	->	DMOTIVO
16	4	DDIAGPOS1, PROFEJE	->	PROFSOL	33	6	PERSTRATANTE	->	DPERSTRATANTE
16	5	DDIAGPOS1	->	DIAGPOS1	33	7	PERSTRATANTE	->	SERVEJE
16	6	DDIAGPOS2	->	DSERVEJE	33	8	PROFEJE	->	DPROFEJE
16	7	DDIAGPOS2	->	SERVEJE	33	9	PROFEJE	->	SERVEJE
16	8	DDIAGPOS2, DPERSTRATANTE, DSERVDEST, LUGAR	->	DPROFEJE	34	0	PROFSOL	->	DPROFSOL
16	9	DDIAGPOS2, DPERSTRATANTE, DSERVDEST, LUGAR	->	PROFEJE	34	1	PROFSOL	->	SERVEJE
17	0	DDIAGPOS2, DPERSTRATANTE, DSERVDEST, SERVSOL	->	DPROFEJE	34	2	SERVDEST	->	DSERVDEST
17	1	DDIAGPOS2, DPERSTRATANTE, DSERVDEST, SERVSOL	->	PROFEJE	34	3	SERVDEST	->	SERVEJE
17	2	DDIAGPOS2, DPERSTRATANTE, DSERVDEST, FP, MEDIO	->	DSERVDEST	34	4	SERVEJE	->	DSERVEJE
					34	5	SERVSOL	->	DSERVSOL

**Tabla C.3** DFs automáticas tabla SAURT009

## Glosario de Términos

ADI	Almacén de Datos Intermedio
ASP	Active Server Pages
CORBA	Common Object Request Broker Architecture
CPU	Central Processing Unit (Unidad Central de Proceso)
DDL	Data Definition Language (Lenguaje de Definición de Datos)
DFD	Diagrama de Flujo de Datos
DFs	Dependencias Funcionales
DTD	Document Type Definition
DTS	Data Transformation System
EOG	Extracción, Obtención y Generación
ETL	Extract, Transform and Load
FD	File Description
FDF	Formato de Definición de Campo
FNBC	Forma Normal de Boyce-Codd
HC	Historia Clínica
HCHM	Hospital Clínico Herminda Martín
HTTP	Hypertext Transfer Protocol
IGU	Interfaz Gráfica de Usuario
IPA	interfaz de programación de aplicaciones
JDBC	Java Database Connectivity
M-FF2RDB	Method-Flat_File to Relational_Data_Base

MVC	Modelo Vista Controlador
NN	Del latín Nomen nescio (nombre desconocido) se utiliza para referirse a pacientes sin identidad, es decir se desconoce su nombre real
ODBC	Open Database Connectivity
RAM	Random Access Memory
RSL	Revisión Sistemática de la Literatura
SAM	Sistema de Admisión Medica
SGBD	Sistema de Gestión de Base de Datos
SI	Sistema de Información
SIH	Sistema de Información Heredado
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
1FN	Primera Forma Normal
2FN	Segunda Forma Normal
3FN	Tercera Forma Normal