# Reingeniería de Bases de Datos: Arquitectura de una Herramienta Abierta Basada en Modelo Semántico

**Marcelo Colman - Gustavo Larriera - Fabiana Piotti - Raúl Ruggia**

*siLAB Laboratorio de Sistemas de Información*
Universitario Autónomo del Sur – Montevideo, Uruguay
*e-mail: silab@ei.edu.uy – Web: http://www.silab.ei.edu.uy/*

***Febrero, 1999***[1]

**Resumen**. La Reingeniería de Bases de Datos (DBRE) es el conjunto de técnicas que permite la obtención de una representación conceptual de un esquema de base de datos a partir de su codificación. Sus aplicaciones son múltiples, desde la re-documentación de bases de datos que evolucionaron en el ambiente operativo hasta la reutilización de esquemas de bases de datos, pasando por el apoyo a la migración y la construcción de metabases. El proceso de DBRE consiste en revertir las dos últimas fases comúnmente aplicadas en el proceso de "ingeniería hacia adelante". Específicamente, deben revertirse secuencialmente la fase lógica, donde a partir de un esquema conceptual se elabora un esquema lógico, y la fase física, donde el esquema lógico es optimizado para un DBMS en particular, generandose el esquema físico de la base de datos. Se denomina a la primera fase de reversión, fase de extracción; a la segunda fase de reversión se la denomina fase de conceptualización.

Este artículo presenta el estado actual de una herramienta de DBRE en pleno desarrollo, cuyas características más relevantes son: (a) Captura la semántica de la base de datos usando un Modelo Semántico independiente del uso que se dará a la especificación semántica, lo cual permite derivar otras en una variedad amplia de modelos, por ejemplo Modelo Entidad-Relación y Modelos Multidimensionales. (b) Está orientada, no solo a re-documentar bases de datos, sino también a servir como base para herramientas de exploración de bases de datos. (c) Finalmente, integra los resultados de algoritmos basados en diferentes técnicas.

***Keywords****: Database Reverse Engineering, Databases, Reverse Engineering, Semantic Models, Semantic Discovery, Relational Model, Conceptual Design, Entity Relationship Model.*

---

## 1. Introducción

La Reingeniería de Bases de Datos (DBRE) consiste en un conjunto de técnicas y herramientas que permiten construír una descripción conceptual (e.g. un modelo de entidades y relacionamientos) a partir de una base de datos en producción. El uso de la DBRE permite, entre otras cosas, reconstruír y/o actualizar documentación perdida, incompleta o inexistente de bases de datos, facilitar el proceso de migración de datos y colaborar en la exploración y extracción de datos en bases poco documentadas. En nuestro trabajo asumimos que la base a ser reingenierizada es una base de datos relacional [Cod70, Cod79].

Durante el proceso de reingeniería de una base de datos -denominada *base de datos fuente*- se distinguen dos fases principales [HCT*93]: (i) La *fase de extracción*, durante la cual se recuperan las estructuras de datos implementadas en el esquema físico (e.g. tablas, atributos, claves primarias, claves foráneas); tales objetos de interés se almacenan en una estructura de datos denominada *base de conocimiento (DBRE-KB),* y (ii) La *fase de conceptualización*, durante la cual se explicitan las estructuras conceptuales que derivaron en las estructuras de datos implementadas. La fase de conceptualización produce como salida un esquema conceptual

[1] Este artículo fue presentado en las *V Jornadas de Informática e Investigación Operativa – VIII Encuentro del Laboratorio de Ciencia de la Computación.* Facultad de Ingeniería, Universidad de la República (Montevideo, Uruguay). Marzo 1-3 de 1999.

utilizando algún modelo semántico [TL82, HK87, PM88] (e.g. un modelo de entidades y relacionamientos [Che76]), que se almacena en una base de datos denominada *base de datos semántica (DBRE-SBD).*

En este artículo se detalla la arquitectura en módulos revisada de una herramienta de DBRE propuesta originalmente en [CLR97] y el diseño de los repositorios de datos DBRE-KB y DBRE-SDB utilizados por los módulos de la herramienta. Las principales contribuciones de este trabajo son: (a) La descripción general de las componentes arquitectónicas de la herramienta, en su visión actual (b) La descripción de los mencionados repositorios de datos.

El resto del artículo se estructura de la siguiente forma. La Sección 2 presenta en forma general a las fases metodológicas que se encuentran en el proceso de reingeniería de bases de datos. En la Sección 3 se describen a los módulos que componen la herramienta. En la Sección 4 se detallan los repositorios de datos utilizados. Finalmente, la Sección 5 presenta algunas conclusiones y trabajo futuro.

## 2. Fases de la reingeniería de bases de datos

El diseño arquitectónico de nuestra herramienta DBRE está directamente influido por las fases metodológicas de la reingeniería de bases de datos. Para comprender a los procesos de reingeniería de bases de datos, resulta de interés conocer los diseños de proceso "hacia adelante" utilizados por los diseñadores cuando diseñan su base de datos. En forma simplificada, se puede ver al “proceso de diseño relacional hacia adelante” como formado por dos fases que se realizan en secuencia [BCN92]. La *fase lógica* utiliza como entrada a un esquema conceptual (e.g. un modelo de entidades y relacionamientos) y produce como salida un esquema lógico (e.g. un conjunto de relaciones y restricciones de integridad). La *fase física* acepta como entrada al esquema lógico y produce un esquema físico optimizado para un DBMS específico. Entonces, durante el proceso de reingeniería de una base de datos se distinguen a su vez dos fases, denominadas *fase de extracción* y *fase de conceptualización*, que revierten respectivamente a la fase física y a la fase lógica [HCT*93].

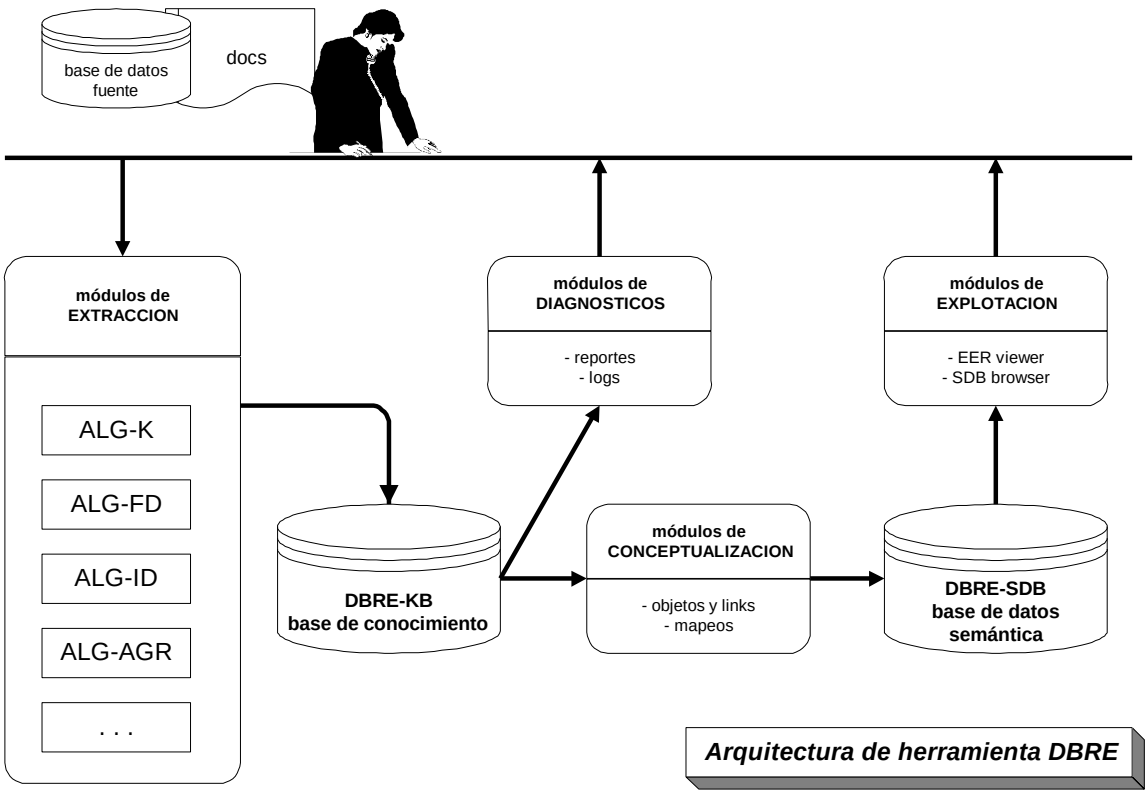
Durante la reingeniería de una base de datos, en su *fase de extracción* los procesos acceden a la base de datos fuente para recuperar información de las estructuras de datos implementadas en el esquema físico. Los principales objetos de interés son, por ejemplo las tablas, columnas, claves primarias y claves foráneas. Cuando la base de datos fuente está implementada en un DBMS relacional, la información puede obtenerse directamente del diccionario de datos o catálogo [CA97]. Toda la información extraída se almacena en *aserciones de trabajo* de una base de conocimiento (DBRE-KB) [Lar98]. La DBRE-KB almacena al esquema lógico extraído del esquema físico de la base de datos fuente, y es utilizada como entrada para los procesos de la fase de conceptualización.

Posteriormente, sobre el esquema lógico extraído en la fase de extracción se aplican diferentes procesos que permiten generar un esquema conceptual. Estos procesos ocurren durante la *fase de conceptualización*, en la cual se explicitan las estructuras conceptuales que derivaron en las estructuras de datos implementadas en la base de datos fuente. Esta fase produce como salida un esquema conceptual utilizando algún modelo semántico. Este modelo semántico se almacena en una estructura a la que genéricamente se denomina *base de datos semántica.* En la base de datos semántica se almacenarán los objetos conceptuales y los vínculos existentes entre ellos.

## 3. Módulos de la herramienta

La herramienta DBRE originalmente propuesta en [CLR97] es un software abierto para reingeniería de bases de datos. La herramienta se considera "abierta" en el sentido de que está diseñada para soportar diferentes familias de algoritmos de reingeniería, en forma intercambiable. En forma general, la herramienta se compone de un conjunto de algoritmos de

reingeniería, un par de repositorios de datos y módulos auxiliares para conceptualización, diagnóstico, explotación y gestión de la interfaz al usuario [ver fig. siguiente].




Los *módulos de reingeniería* comprenden a los algoritmos que realizan las tareas típicas de la fase de extracción. Los algoritmos se agrupan en *familias* según el tipo de objetos que detectan: *detectores de claves* (algoritmos-K) [PTB*96], *detectores de dependencias* funcionales y de inclusión (algoritmos-FD y algoritmos-ID) [PTB*96] [Chi95] y *detectores de agrupamientos de atributos* (algoritmos-AGR). Los algoritmos de reingeniería obtienen información de diversos orígenes: el esquema de la base de datos fuente, la extensión de la misma y las operaciones SQL existentes en las aplicaciones. La intervención del usuario experto es requerida por algunos algoritmos. La información obtenida, asi también como el grado de confiabilidad de la misma - dependiente del origen de donde se obuvo dicha información- se almacena como resultado intermedio, en el repositorio denominado *base de conocimiento* que se describe más adelante.

Los *módulos de conceptualización* comprenden a los algoritmos utilizados en la fase de conceptualización. Estos algoritmos acceden al repositorio de la base de conocimiento y mapean sus objetos en objetos semánticos, que se almacenan en el repositorio de la base semántica. Los algoritmos de conceptualización se encargan de detectar los objetos, links entre objetos y tipos de links [Lar98b].

Como módulos adicionales que no forman parte de alguna fase específica de la reingeniería, disponemos de los *módulos de diagnósticos* (encargados de la generación de informes y documentación de apoyo al usuario experto, y logs de las actividades realizadas durante el proceso de reingeniería) y de los *módulos de explotación* (encargados de realizar las representaciones conceptuales de los objetos semánticos almacenados en la base de datos semántica). Finalmente, los módulos de gestión de la interfaz al usuario permiten interactuar con todas las componentes.

# 4. Repositorios de datos

Los algoritmos utilizan repositorios de datos a los efectos de almacenar la información que va siendo descubierta durante la reingeniería. El primer repositorio de datos utilizado se denomina *base de conocimiento* y almacena información del esquema físico, que es descubierta durante la fase de extracción. El segundo repositorio se denomina *base de datos semántica* y almacena el modelo conceptual semántico descubierto durante la fase de conceptualización. En las siguientes secciones se describen ambos repositorios.

## 4.1. La base de conocimiento

La *base de conocimiento (DBRE-KB)* es el repositorio que almacena toda la información generada durante el proceso global de reingeniería, realizado por los módulos que implementan a los algoritmos de las diversas familias. La información almacenada en la DBRE-KB consiste en los datos básicos del esquema físico de la base de datos reingenierizada y sus restricciones de integridad. Estos datos básicos se implementan en la forma de *aserciones del esquema*, que contienen la información del esquema físico de la base de datos, dependencias funcionales, dependencias de inclusión, condiciones de join, y atributos de agrupamiento. Adicionalmente se registra un valor de confiabilidad, que permite establecer un control de calidad de la información obtenida, que puede depender de los algoritmos de reingeniería que se utilicen.

Podemos distinguir dos grupos de aserciones del esquema: (i) Las *aserciones del esquema básicas*, obtenidas directamente del esquema físico y de las expresiones SQL. En esta categoría tenemos las aserciones que representan información estructural de la base de datos reingenierizada (información sobre tablas y sus atributos), aserciones que representan vínculos entre las tablas (e.g. obtenidas de expresiones de join SQL) y aserciones que representan agrupamientos de atributos (e.g. obtenidas de expresiones GROUP BY de SQL). (ii) Las *aserciones del esquema derivadas*, obtenidas mediante aplicación de algoritmos. En esta categoría tenemos a las aserciones construidas por los algoritmos detectores de las diversas familias.

## 4.2. La base de datos semántica

Cuando la base de datos fuente es sometida al proceso de reingeniería, la información en bruto que debe considerarse consiste en los objetos existentes y los vínculos (links) que pudieran haber entre los objetos. En forma general, denominamos *especificación semántica* a la colección de información de los objetos y los vínculos entre ellos, y denominamos *base de datos semántica (DBRE-SDB),* al repositorio que almacena la información de objetos y sus vínculos.

A su vez, al conjunto O de objetos los clasificamos en (i) Objetos atómicos o *átomos*, son aquellos objetos simples que no incluyen a otros objetos; y (ii) Objetos compuestos o *moléculares*, son aquellos objetos que están formados por otros objetos. Notaremos a las clases de átomos y moléculas, respectivamente como A y M. Se cumple que $A \cup M \equiv O$ y $A \cap M = \varnothing$ .

Los *links* representan asociaciones definidas entre los objetos. Los links entre objetos pueden representar conceptualmente asociaciones diferenciadas según sean átomos o moléculas los objetos vinculados. En forma más general, podemos definir al conjunto de links L y luego a las clases de links, según cuáles sean los tipos de objetos vinculados. Por ejemplo, en el EER son moléculas los *entity types* y los *relationship types*; los links son los *role-links* y los *attribute-links*, que respectivamente vinculan a un *entity tipe* con un *relationship type*, y un atributo con *un entity type* (o con un *relationship type*). Así entonces, clasificamos los links en *links intermoleculares* (i.e. vinculan dos nodos que son moléculas), *links interatómicos* (i.e. vinculan dos nodos que son

átomos) y *links mixtos* (i.e. vinculan una molécula con un átomo), y los denotamos respectivamente $L_{IM}$, $L_{IA}$ y $L_{MX}$. Los conjuntos son disjuntos entre sí.

Como instancias de $L_{IM}$, tenemos a la *generalización* y a la *agregación*, con la semántica habitual que se les da en los modelos de datos [TL82]. Como instancias de $L_{IA}$, tenemos a la relación *subparte*, que permite especificar una jerarquía de atributos compuestos. El link *dimensión* establece una jerarquía entre atributos que puede utilizarse para representar aspectos multidimensionales [GMR98]. Entre los links de $L_{MX}$ vamos a considerar al que usualmente se denomina *attribute link*, que vincula a una molécula con cada atributo de ella.

En forma abstracta representamos a los objetos y links como un grafo dirigido cuyos nodos son los objetos y cuyos aristas representan los links entre los objetos. Los links pueden representar aspectos semánticos que pueden derivarse mediante el uso de *algoritmos de conceptualización* [Lar98b]. Por ejemplo, un link $L \in L_{IM}$ podría representar una jerarquía de herencia [RH97] [And94], entidades débiles [And94] o relacionamientos binarios entre entidades [PTB*96]. Adicionalmente se almacenan mapeos que permiten asociar a cada objeto conceptual con su objeto fuente de la base de conocimiento. Estos mapeos permiten, entre otras cosas, que una interfaz de navegación del grafo semántico pueda acceder a los objetos correspondientes en la base de datos fuente.

# 5. Conclusiones y trabajo futuro

Se ha presentado en forma general las componentes de una herramienta de reingeniería de bases de datos, formada por módulos que se corresponden directamente con los procesos de las fases de extracción y conceptualización. Estas fases metodológicas de la reingeniería de bases de datos revierten respectivamente a las fases física y lógica de las metodologías habituales de diseño “hacia adelante”. La arquitectura de la herramienta consta además de dos repositorios de datos, la base de conocimiento y la base de datos semántica que se utilizan respectivamente para almacenar información generada en las fases de extracción y de conceptualización. Adicionalmente se han mostrado someramente algunos módulos no vinculados a los procesos de reingeniería pero que son de uso práctico: los módulos de diagnósticos y los módulos de explotación.

El presente proyecto está siendo implementado. Los repositorios y algunos algoritmos han sido inicialmente prototipados utilizando PROLOG [Amb87, CM94, DEC96]. Posteriormente se han implementado las diversas componentes y los repositorios utilizando lenguajes procedurales y sistemas de bases de datos relacionales. Específicamente, se han desarrollado algoritmos de extracción y un browser semántico gráfico, en Visual Basic. Los repositorios se han implementado en bases de datos relacionales accesibles a través de ODBC, a saber Access y SQL Server.

Como trabajo futuro inmediato se pretende: (a) Integrar en forma más sólida las diversas componentes que ya han sido implementadas. (b) Estudiar la representación conceptual de jerarquías dimensionales. Ello implica el desarrollo de algoritmos de extracción nuevos, posiblemente basados en GROUP BY de SQL, asi también como formas alternativas de representar conceptualmente dicha información, en base a ideas de la propuesta de [GMR98]. (c) Adicionalmente se pretende agregar todo lo referente a los grados de confiabilidad [CLR97] que permitan aplicar un criterio de control de calidad sobre la información obtenida durante la reingeniería.

El desarrollo de este proyecto se realiza dentro del contexto del Laboratorio de Sistemas de Informacion (siLAB), del Universitario Autonomo del Sur (Montevideo, Uruguay).

## Referencias


[Amb87] T. Amble. *Logic Programming and Knowledge Engineering.* Addison-Wesley, 1987.

[And94] M. Andersson. Extracting an Entity Relationship Schema from a Relational Database Through Reverse Engineering, *Proc. 13th Int. Conf. on ER Approach,* Manchester UK. Dec. 1994.

[CM94] W. Clocksin, C. Melish. *Programming in Prolog,* 4th Edition. Springer, 1994.

[Cod70] E. Codd. A Relational Model of Data for Large Shared Data Banks. *Comm. ACM 13, 6.* Jun. 1970.

[Cod79] E. Codd. Extending the Database Relational Model to Capture More Meaning. *ACM 0362-5915.* 1979.

[BCN92] C. Batini, S. Ceri, S. Navathe. *Conceptual Database Design: An Entity-Relationship Approach.* Addison-Wesley. 1992.

[CA97] I. Comyn-Wattiau, J. Akoka. Reverse Engineering of Relational Database Physical Schemas. 1997.

[Che76] P. Chen. The Entity Relationship Model - Toward a Unified View of Data. *ACM TODS, 1(1),* 1976.

[CLR97] M. Colman, G. Larriera, R. Ruggia. Database Reverse Engineering: Proposal of an Open Tool Based on a Semantic Model, *1er. Congreso Uruguayo de Informática,* 1997.

[DEC96] P. Deransart, A. Ed-Dbali, L. Cervoni. *Prolog: The Standard.* Springer, 1996.

[GMR98] M. Golfarelli, D. Maio, S. Rizzi. Conceptual Design of Data Warehouses from E/R Schemes. *Proc. of the Hawaii Int. Conf. On System Sciences.* Jan. 1998, Kona, Hawaii.

[HCT*93] J. Hainaut, M. Chandelon, C. Tonneau, M. Joris. Contribution to a Theory of Database Reverse Engineering, *Working Conferenc. on Reverse Engineering.* Baltimore, May. 1993.

[HK87] R. Hull, R. King. Semantic Database Modeling: Survey, Applications, and Research Issues. *ACM Computing Surveys, Vol. 19, No. 3,* Sep. 1987.

[Joh94] P. Johannesson. A Method for Transforming Relational Schemas into Conceptual Schemas. *Proc. of the 10th Int. Conf. on Data Engineering,* págs. 190-201, Houston, Texas, Feb. 1994. IEEE Computer Society.

[Lar98] G. Larriera. *Descripción de una Base de Conocimiento para una Herramienta de Reingeniería de Bases de Datos.* Mayo 1998.

[Lar98b] G. Larriera. *Representación Semántica de un Esquema Conceptual Obtenida Mediante Reingeniería de Bases de Datos*. Octubre, 1998.

[PKB*94] J. Petit, J. Kouloumdjian, J. Boulicaut, F. Toumani. Using Queries to Improve Database Reverse Engineering. *Proc. of the 13th Int. Conf. on ER Approach*. Manchester, UK. Dec. 1994.

[PM88] J. Peckham, F. Maryanski. Semantic Data Models. *ACM Computing Surveys, Vol. 20, No. 3,* Sep. 1988.

[PTB*96] J. Petit, F. Toumani, J. Boulicaut, J. Kouloumdjian. Towards the Reverse Engineering of Denormalized Relational Databases, *Proc. of 12th Int. Conf. on Data Engineering,* 1996.

[RH97] S. Ramanathan, J. Hodges. Extraction of Object-oriented Structures from Existing Relational Databases, *ACM SIGMOD Record, Vol. 26 No. 1,* 1997.

[TL82] D. Tsichritzis, F. Lochovsky. *Data Models.* Prentice-Hall, 1982.