

Sistema para la integración del proceso de normalización de bases de datos relacionales con gestores de bases de datos (SINORGES)

Integration system standardization process relational databases with database managers (SINORGES)

Yanet Espinal Martín, Lic., & Manuel Enrique Puebla Martínez, M. Sc.

Universidad de las Ciencias Informáticas, Ciudad de la Habana, Cuba

yanete@uci.cu; mpuebla@uci.cu

Recibido para revisión 05 de febrero de 2010, aceptado 28 de octubre de 2010, versión final 20 de noviembre de 2010

Resumen— Dado el crecimiento vertiginoso de los volúmenes de información en la sociedad actual y por consiguiente, el aumento de la información manipulada por los Sistemas Informáticos (SI), se hace necesario optimizar el tratamiento de los grandes volúmenes de información que genera nuestra sociedad, así como, integrar dicha optimización con los Sistemas Gestores de Bases de Datos (SGBD) existentes; facilitando el desarrollo de los SI.

Reconocida la necesidad explicada anteriormente, se propuso el desarrollo de un SI que integrara el proceso de normalización de Bases de Datos Relacionales (BDR) con los SGBD; con el objetivo de elevar la calidad de los modelos de datos y favorecer la interrelación de la etapa de diseño con la etapa de implementación de las diferentes soluciones informáticas que se realizan. Además, se incorporó un módulo con objetivos docentes que permite la utilización del proceso de normalización de BDR en el proceso de enseñanza, beneficiando la enseñanza de esta temática a través de la visualización de algoritmos básicos del proceso.

El principal aporte de la investigación es la concepción y obtención de un SI capaz de automatizar e integrar el proceso de normalización de BDR y su aplicación en la enseñanza, con SGBD existentes.

Palabras Clave— Diseño de Bases de Datos; Normalización de Bases de Datos Relacionales; Sistema Informático.

Abstract— Given the explosive growth in the volumes of information in today's society and therefore increasing the information manipulated by computer systems (CS), it is necessary to optimize the processing of large volumes of information generated by our society and integrate this optimization with the Database Management System (DBMS) in place by facilitating the development of CS.

Recognized the need explained above, we propose the development of an CS that integrates the process of normalization of Relational Database (RDB) with the DBMS, with the aim of

improving the quality of data models and encourage the interrelationship of the stage design to the stage of implementation of different solutions that are conducted. Also included is a module with teaching objectives, which allows use of the standardization process of RDB in the teaching process, benefiting the teaching of this subject through the display of basic algorithms of the process.

The main contribution of this research is the design and procurement of an CS capable of automating and integrating the process of normalization of RDB and their application in teaching, with existing DBMS.

Keywords— Database Design; Normalization of Relational Databases; Computer System.

I. INTRODUCCIÓN

El crecimiento considerable del volumen de información que se maneja en la sociedad actual, está influyendo de manera proporcional en el aumento del volumen de información que utilizan los sistemas de gestión de la información (SGI); lo que provoca una mayor dificultad en la manipulación de la información y hace completamente engoroso la aplicación, de forma manual, de los mecanismos existentes para lograr un diseño correcto de la información que se almacena; aspecto de vital importancia que determina la eficiencia de los SGI.

En el proceso y construcción de toda aplicación informática, el diseño de las bases de datos ocupa un lugar importante dentro de su desarrollo; a tal punto que ésta puede verse como un proceso relativamente independiente dentro de la confección de un sistema, compuesto por una serie de etapas a realizar.

En el diseño de una base de datos relacional pueden encontrarse anomalías de inserción, eliminación, actualización

de los datos, así como restricciones artificiales en la estructura de los mismos y dependencia entre ellos, implicando grandes problemas en la gestión y obtención de su información [5].

En la actualidad existen varias herramientas de software a nivel mundial que comprueban y dan solución a estas anomalías en la etapa del diseño de forma automática para modelos de BDR como son: Web-Based Tool , Normit , DBDesign Tool , LDBN , JMathNorm . Estas herramientas fueron estudiadas y de ellas se analizaron las ventajas y desventajas que brindan para su uso. (Ver detalles en el epígrafe IV.I). Este estudio arrojó como resultado que en ellas están presentes uno o varios inconvenientes que van desde la interfaz visual, el idioma, los mecanismos para obtener los resultados, la integración con las próximas etapas del diseño que intervienen en el proceso de desarrollo de la aplicación, la falta de profundización en el proceso de normalización de BDR, hasta su ambiente de desarrollo propietario. Además, ninguna proporciona mecanismos que garanticen la integración de dicho resultado con los sistemas gestores de bases de datos.

Por tal motivo se plantea el siguiente problema a resolver:

La falta de integración del proceso de normalización de bases de datos relacionales con los sistemas gestores de bases de datos, provoca pérdida en la calidad de los modelos de datos y dificulta la interrelación de la etapa de diseño con la etapa de implementación de dichos modelos.

Basado en los resultados del análisis anterior, en este trabajo se presenta un Sistema Informático, nombrado SINORGES, que permite integrar el proceso de normalización de bases de datos relacionales con los sistemas gestores de bases de datos; con el objetivo de elevar la calidad de los modelos de datos y favorecer la interrelación de la etapa de diseño con la etapa de implementación de las diferentes soluciones informáticas.

II. MATERIALES Y MÉTODOS

"SINORGES" versión 1.0, es un sistema Informático que garantiza la integración del proceso de Normalización de Bases de Datos Relacionales con los Sistemas Gestores de Bases de Datos: PostgreSQL, Microsoft SQL Server y MySQL. Para la implementación del sistema se utilizó como lenguaje de programación C Sharp (C#), bajo el Entorno de Desarrollo Integrado (IDE) de software para su realización SharpDevelop 3.0 y como plataforma de desarrollo Microsoft. NET Framework 3.5, SINORGES está compuesto por 5 módulos:

- Entrada del Modelo de datos Relacional.
- Proceso de Normalización del Modelo de datos.
- Generación de scripts.
- Generación del Modelo Físico de Base de Datos.
- Visualización de los algoritmos de Normalización.

Entrada del Modelo de datos Relacional: Permite crear relaciones para constituir el modelo relacional a través de la inserción de atributos y dependencias funcionales, como se puede apreciar en la Figura 1. Manipula las funciones generales para el diseño de la entrada del modelo de datos Relacional como son: inserción, eliminación, modificación de los datos que corresponden al modelo y del modelo en general, funciones gráficas específicas para cada relación como: mover, redimensionar, ocultar componentes de la relación. Este módulo también tiene como función cargar scripts de Bases de datos (BD) realizados en los SGBD: PostgreSQL, Microsoft SQL Server y MySQL , o en herramientas CASE de ayuda a la modelación de BD como: Embarcadero ER/ Studio y ERwin.

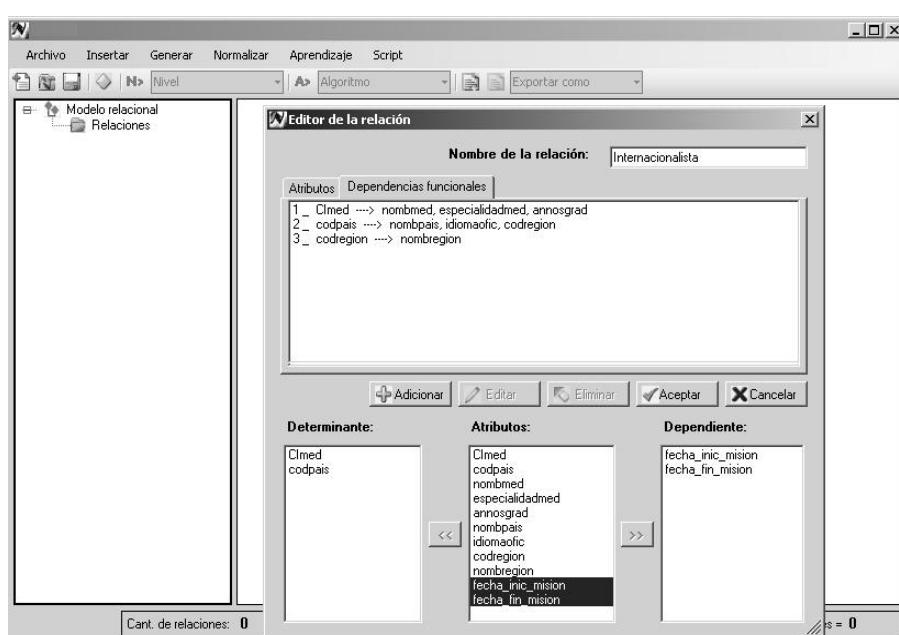


Figura 1. Representación de la entrada de una Relación con el Sistema "SINORGES"

Proceso de Normalización del Modelo de datos: Módulo de validación de la estructura lógica de los datos, detecta el nivel de normalización del modelo de datos, desde la segunda forma normal ($2FN$) hasta la forma normal de Boyce-Codd ($FNBC$), notificándole al usuario la localización exacta donde se encuentra el problema para llegar a niveles superiores de normalización. Este módulo permite además la corrección de la estructura lógica del modelo aplicando el proceso de normalización hasta $FNBC$ utilizando las dos técnicas que existen: Descomposición y Síntesis. Las funcionalidades en el sistema que sustentan todo este proceso son:

- Cálculo de las Claves Primarias de cada relación.
- Cálculo del Recubrimiento Minimal del conjunto de dependencias funcionales de cada relación.

- Verificación del nivel de Normalización en el que se encuentra el modelo de datos.
- Verificación del nivel de Normalización en el que se encuentra cada relación.
- Normalizar a $2FN$.
- Normalizar a $3FN$ por Descomposición. Ver detalles en la *Figura 2*.
- Normalizar a $3FN$ por Síntesis.
- Normalizar a $FNBC$.
- Verificación de la propiedad sin pérdida de información (Lossless Join, sus siglas en inglés *LLJ*).
- Verificación de la propiedad de conservación de las dependencias funcionales.

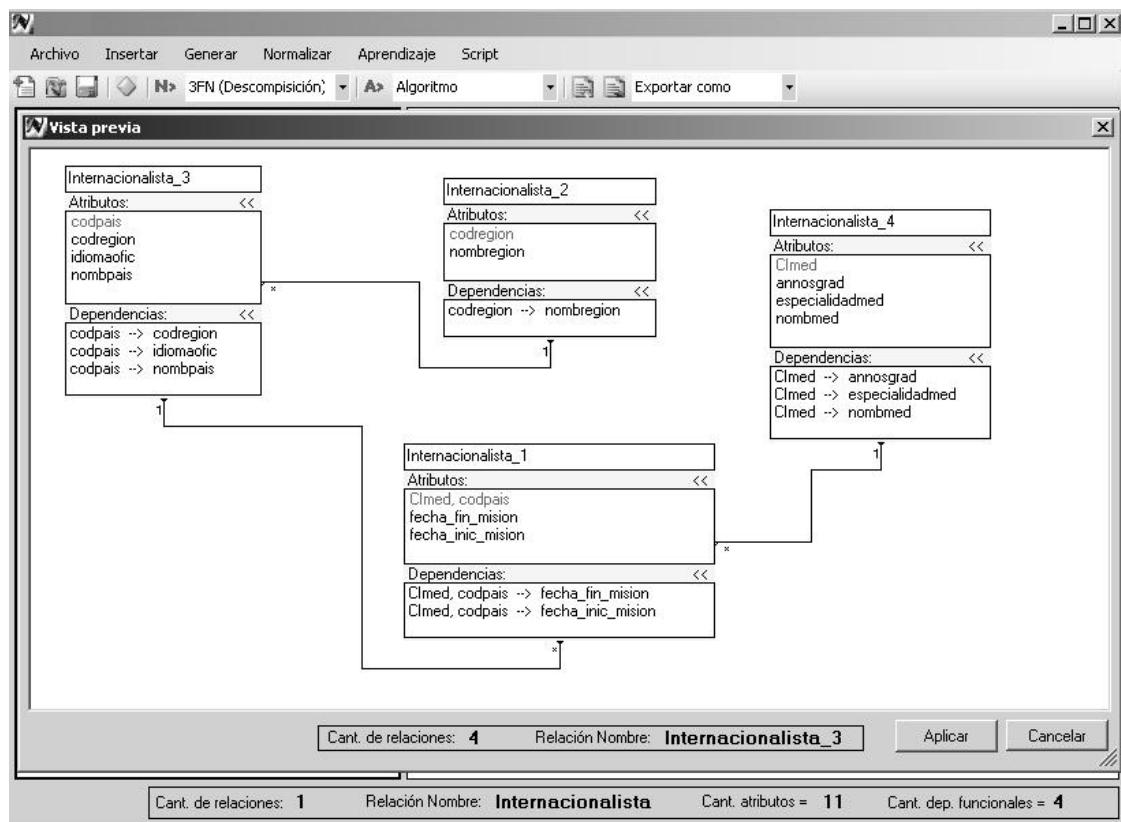


Figura 2. Representación del proceso de Normalización a $3FN$ por Descomposición con el Sistema “*SINORGES*”

Generación de scripts: A partir del esquema o modelo de datos normalizado genera las sentencias de código *SQL* correspondientes, almacenándolas en un fichero con extensión *.sql* para los *SGBD*. *PostgreSQL*, *Microsoft SQL Server* y *MySQL*, permitiendo visualizar este fichero en modo lectura, como se puede observar en la *Figura 3*. Para realizar estas operaciones se necesita la entrada de los diferentes tipos de

datos asociados a los atributos introducidos en la confección del esquema relacional, junto con otras características como tamaño, escala, precisión, tipos de valores. Este módulo realiza validaciones con respecto a la integridad de los datos introducidos por el usuario, dígase restricciones de dominios, reglas de integridad de entidades y regla de integridad referencial.

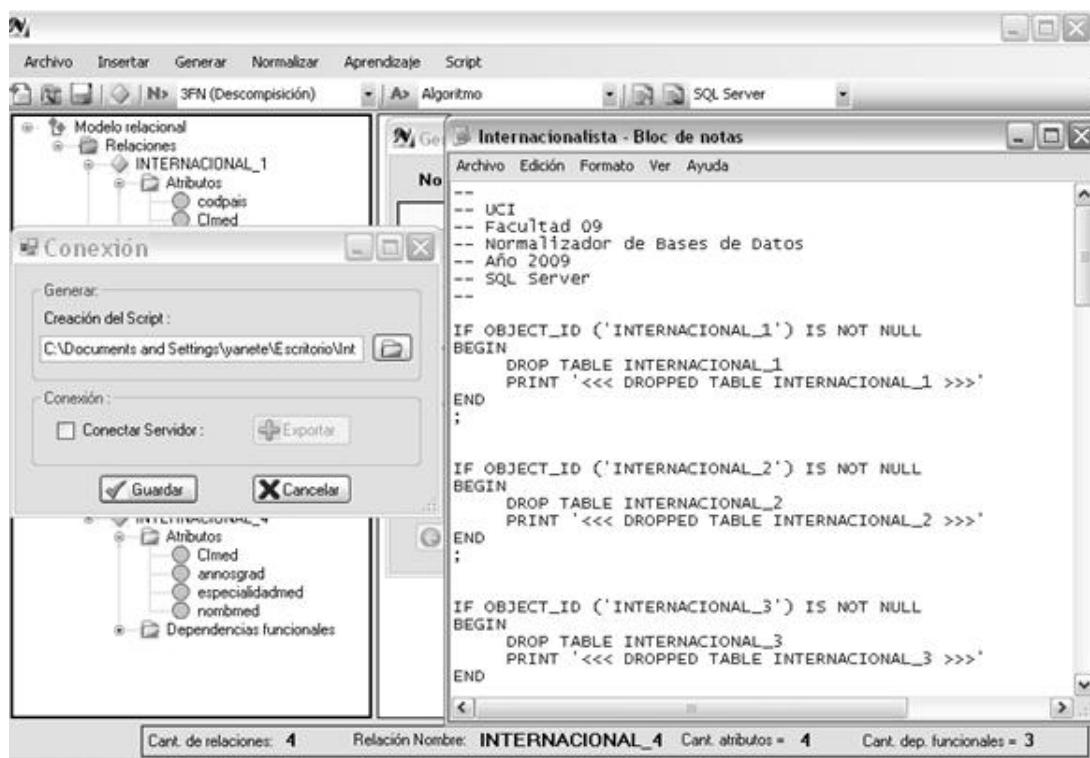


Figura 3. Generación de scripts para Microsoft SQL Server en el Sistema “SINORGES”

Generación del Modelo Físico de Base de Datos: Cuenta con las funcionalidades necesarias para transformar el modelo lógico de datos a su correspondiente esquema físico de *BD*

para los *SGBD PostgreSQL, Microsoft SQL Server y MySQL* a partir del modelo de datos normalizado, como se puede apreciar en la Figura 4

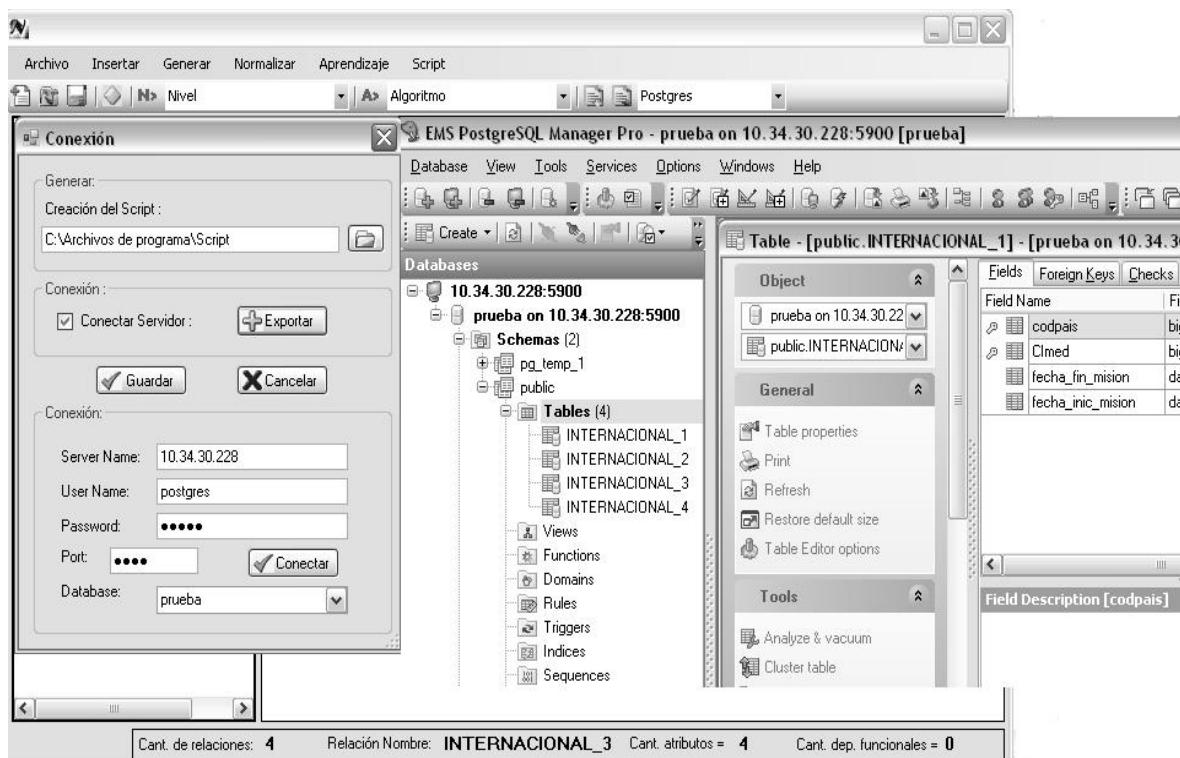


Figura 4. Representación de la generación del modelo físico de BD en el Sistema "SINORGES"

Visualización de los algoritmos de Normalización: Visualiza los algoritmos básicos para comenzar con el proceso de normalización en forma de seudo-código, brindando una descripción detallada de cada paso al ejecutar el código, así como los valores específicos para cada iteración que va adquiriendo el conjunto solución, estas funcionalidades de “*SINORGES*” permiten que los estudiantes de carreras afines al perfil informático adquieran nuevos conocimientos referente a esta teoría de normalización. Las funcionalidades con las que cuenta el sistema actualmente dentro de este módulo son:

- Visualización del Cálculo de Cierre de Descriptores de relaciones. Ver en la *Figura 5*.
- Visualización del Cálculo del Recubrimiento Minimal del conjunto de dependencias funcionales de relaciones.
- Visualización de la Equivalencia entre dos conjuntos de dependencias funcionales.
- Visualización del Cálculo de las Claves Primarias de relaciones.

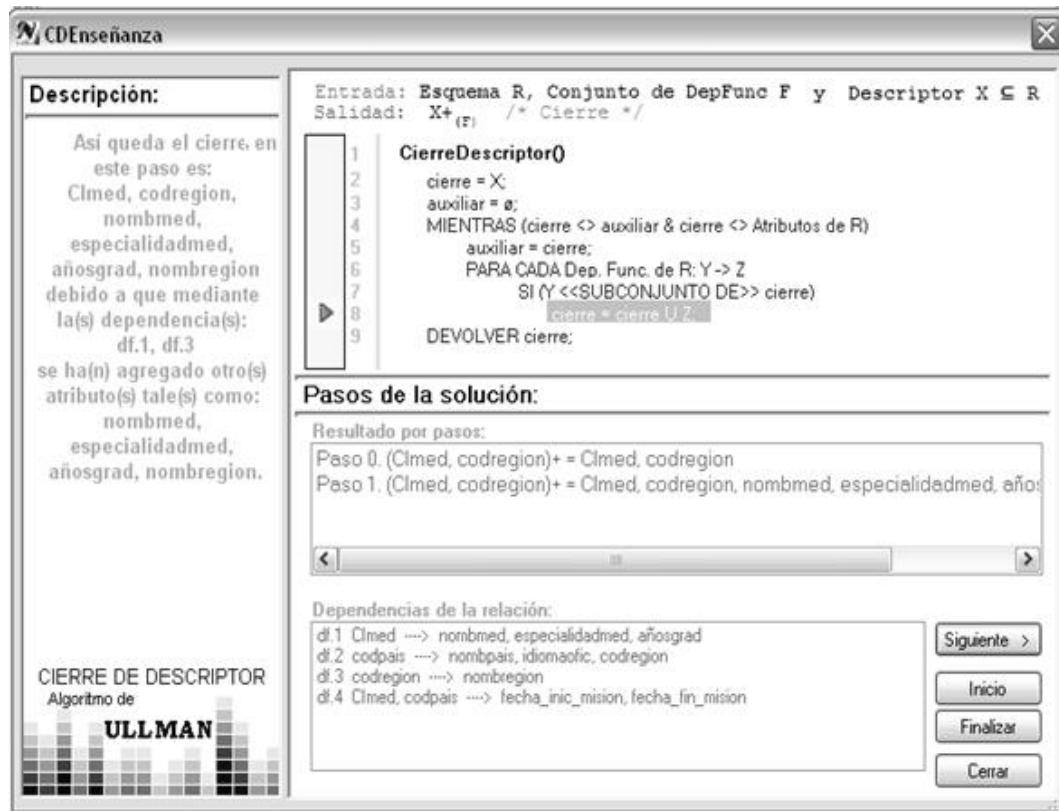


Figura 5. Visualización del Cálculo del Cierre de un Descriptor a través del Sistema “*SINORGES*”

III. RESULTADOS Y DISCUSIÓN

3.1. Proceso de normalización de BDR

Para llevar a cabo el proceso de normalización de BDR, es necesario transitar por un conjunto de algoritmos previos que permiten descomponer el modelo original en un conjunto de relaciones más sencillas y simples, en término de las operaciones que se ejecutan sobre ellas.

El proceso de normalización de BDR parte del modelo relacional, con el objetivo de representar el diseño lógico de la realidad que describe el modelo, es decir, conocer la semántica de los atributos de la relación reflejada a través del conjunto de dependencias funcionales; término principal que se utiliza para poder realizar dicho proceso, [6].

El conjunto de dependencias funcionales que se establecen en cada una de las relaciones del modelo relacional, no constituyen las únicas dependencias funcionales que se cumplen y existen en el esquema relacional. ¿Cómo determinarlas? Para darles respuesta a esta interrogante se pensó en el algoritmo para el cálculo del Cierre de un Conjunto de Dependencias Funcionales (DF+) aplicando los Axiomas de Armstrong, que permiten derivar dependencias a partir de un conjunto inicial DF. El inconveniente del método es su excesiva complejidad computacional (es del tipo NP-completo), aún cuando el número inicial de DF sea pequeño, el número total de DF+ sería muy grande. Por otro lado, no todas las dependencias en DF+ son útiles, por lo tanto, se debe buscar métodos que no estén basados en el cálculo de DF+. El algoritmo para el cálculo del cierre de un descriptor es el recomendado en la

mayoría de las bibliografías relacionada con el diseño de BD en la teoría relacional, propuesto en [17], con una complejidad temporal para el caso peor (O-grande) de O($n*m$), donde n es el número de atributos del esquema relacional y m es el número de DF de la relación. El mismo es utilizado por casi todos los algoritmos que intervienen en el cálculo de DF+, constituyendo el eje central en la implementación de algoritmos para la determinación del conjunto de claves o llaves primarias, primer concepto importante dentro de los esquemas de relaciones en un modelo de datos.

Una de las variantes más triviales que se pudiera pensar para desarrollar dicho algoritmo, sería calcular el cierre para cada conjunto de descriptores del esquema de relación, (representado en este caso por R) y seleccionar aquellos que coincidan con el conjunto de atributos de R . Esta variante a pesar de ser la más trivial, suponía hacer, $\sum_{i=1}^n C_n^i$ (*Combinaciones de cálculo de cierre de descriptores*), en el caso peor el algoritmo estaría acotado por una función exponencial. Esta variante la utiliza la herramienta (*DBDesignTools (Librería en C++)*) para la determinación de las claves primarias. “SINORGES” usa un algoritmo más eficiente, con respecto a la complejidad temporal, para darle solución a esta funcionalidad y es precisamente el algoritmo descrito en [4].

La **complejidad temporal**, de los algoritmos comentados anteriormente, es proporcional al conjunto de atributos y dependencias funcionales que pertenecen al modelo. Por lo tanto, lograr reducir dichos conjuntos sin afectar ninguna de las restricciones del negocio que el modelo de datos describe, contribuye proporcionalmente a disminuir la complejidad temporal; no sólo de estos métodos, sino también de los algoritmos de normalización que los usan para su funcionamiento. **Por tal motivo se hace necesario sintetizar dichos conjunto a través de algoritmos para la determinación de coberturas minimales** de esquemas relacionales encontrado en [8].

Para la obtención de esquemas normalizados en 3FN existen dos grandes grupos de algoritmos:

- *Algoritmo de síntesis.*
- *Algoritmo de descomposición.*

El *Algoritmo de Síntesis* basado en las definiciones de esquemas sintetizados propuesto por el Dr. P.Bernstein, constituye el primer intento satisfactorio para instrumentar el proceso de normalización planteado por el Dr.Codd en 1970. Su descripción formal se encuentra en [4].

El *Algoritmo de Descomposición* basado en la teoría de descomposición de relaciones, es una de las técnicas más nombradas en las diferentes fuentes consultadas por los autores, utilizadas para la normalización de esquemas, no sólo para 3FN sino también para formas normales superiores como FNBC, 4FN y 5FN. Es el método que utilizan la mayoría de las herramientas existentes mencionadas. Su inclinación hacia este método está

proporcionada por la capacidad de permitir modelos normalizados sin pérdida de información o **descomposición sin pérdida de información (Lossless Join) (sus siglas en inglés LLJ)** y modelos normalizados con **preservación de dependencias funcionales**, aspectos importantes que deben tenerse siempre muy presentes si se pretende obtener verdaderos modelos de datos con elevada calidad en su diseño. Su demostración formal se centra en la teoría relacional encontrada en [4], [8].

La propuesta del sistema “SINORGES” que se presenta en esta investigación, está diseñada para brindar los dos grupos de algoritmos para realizar el proceso de normalización de esquemas relacionales hasta FNBC, y dejar en manos de los diseñadores o especialistas de BD su elección.

El **Aporte** de la presente investigación se expresa en la obtención de un Sistema Informático capaz de automatizar e integrar el proceso de normalización de bases de datos relacionales y su aplicación en la enseñanza, con *SGBD* existentes. A continuación se abordarán algunos criterios que justifican la investigación:

1. Conveniencia y utilidad metodológica:

La herramienta será de gran utilidad **para los estudiantes** de Informática o de carreras afines al perfil informático. Les servirá para apropiarse de nuevos conocimientos en la teoría del proceso de normalización de *BDR*, a través de la visualización de algoritmos básicos dentro del proceso; permitiéndole observar en cada paso de la ejecución los resultados que se van obteniendo hasta llegar a la solución.

A **los diseñadores** les facilitará la construcción de modelos de *BDR* normalizados de forma automática, brindándole su integración directa con los *SGBD* como: *PostgreSQL*, *Microsoft SQL Server* y *MySQL*; obteniendo mayor nivel de confiabilidad pues se utilizan algoritmos demostrados y probados matemáticamente por expertos en la materia que garantizan un alto grado de seguridad en la obtención de la solución. Actualmente la mayoría de los diseñadores de *BD* realizan el diseño de la *BD* e integración con los *SGBD* de forma manual o independiente, permitiendo la inserción del error humano.

2. Relevancia social:

El sistema contribuirá a que los futuros estudiantes Informáticos, emergan con un mayor nivel de conocimientos de la asignatura de “Sistemas de Bases de Datos” en la teoría de normalización de *BDR*, lo cual repercutirá de forma positiva en el desarrollo de soluciones informáticas realizadas y con ello brindará un mayor beneficio para nuestra sociedad.

3.2. Comparación con las herramientas informáticas que automatizan el proceso de normalización de *BDR*

JMathNorm

Es una herramienta interactiva de escritorio, con una interfaz

gráfica descrita en Java. Permite obtener modelos normalizados hasta la FNBC, incluyendo también como parte del sistema la 2FN y 3FN. Arroja además, operaciones que forman parte del proceso de normalización como: *el cálculo del cierre de descriptores, cobertura minimal del conjunto de dependencias funcionales, eliminación de atributos extraños en la relación, eliminación de dependencias funcionales redundantes*. Su interface desarrollada en Java hace uso de la componente J/Link¹, para lograr una integración directa con el Mathematica². [11].

Otras características importantes de la herramienta **JMathNorm** son [10]:

- Permite adicionar atributos al modelo de manera interactiva por el usuario.
- Permite crear el conjunto de dependencias funcionales que representan las reglas del negocio, con la notación usada en la mayoría de las bibliografías existentes de BD.
- Permite mostrar el resultado final de cada uno de los procesos particulares, que no son más que resultados de los diferentes pasos que forman parte de la técnica de normalización, para cualquiera de los niveles que se encuentre analizando.

A pesar de encontrarnos con un gran número de características significativas es importante mencionar que esta herramienta, en la actualidad, no cuenta con algunas funcionalidades necesarias para su mejor

aprovechamiento, por ejemplo: la capacidad de integrarse con los SGBD; no muestra el proceso de obtención de las claves candidatas, ni lo tiene agregado como una prestación del sistema; no permite *cargar* un modelo de BD almacenado, así como tampoco permite *salvar* los modelos para que sean utilizados nuevamente.

NORMIT

Sistema tutorial web basado en restricciones para la enseñanza, con una arquitectura centralizada, creado para resolver problemas de normalización de bases de datos, cuyo modelo de dominio es representado por un conjunto de 81 problemas de restricciones independientes. Maneja una Base de Conocimiento Basada en Restricciones (CBM), la cual se manipula a través del lenguaje de programación *Lisp*³ y como servidor de aplicaciones web utiliza el *AllegroServe* [15]. Sistema capaz de analizar y ofrecerles una retroalimentación

-
1. Conjunto de herramientas que integra Mathematica y Java. Permite llamar a Java desde Mathematica de una manera completamente transparente.
 2. Programa utilizado en áreas científicas, de ingeniería, matemáticas y áreas computacionales, herramienta poderosa de cómputo que permite realizar cualquier tipo de cálculo algebraico o numérico y generar gráficos y análisis profundos.

a los estudiantes del tema, como objetivo de complementar y ejercitarse los conocimientos adquiridos en el aula [14]. Para la resolución de los problemas se basa en la estrategia metodológica de la *auto-explicación*⁴, exigiéndole por cada acción que realice por primera vez los estudiantes una explicación de las respuestas seleccionadas [15].

A pesar de estos elementos significativos, presenta algunos inconvenientes tales como: el sistema no proporciona ayuda sobre los conceptos básicos del dominio abarcado, o sea, sólo es provechoso para estudiantes que posean conocimientos del la normalización; no admite introducir nuevos problemas de estudio por los estudiantes, sólo le permite la ejercitación a través de los ya existentes en la base de conocimiento; además, no tiene la capacidad de integrarse con los SGBD.

WEB-BASED TOOL

Sistema web desarrollada con *applet de Java*⁵ para mejorar la enseñanza y el aprendizaje del proceso de normalización de bases de datos; fácil de utilizar para los estudiantes, permite obtener modelos normalizados hasta la 3FN a partir de un conjunto de dependencias funcionales, asociadas a una relación, introducidas por el usuario sin exceder las 10 unidades.[9]:

Web-Based Tool, es el software que menos prestaciones brinda (de los analizados) para facilitar el proceso de enseñanza de la normalización de BDR para los estudiantes. Como aspecto positivo vale la pena resaltar su interfaz de fácil entendimiento y la opinión favorable que presentan la mayoría de los estudiantes encuestados según Hsiang-Jui Kung en su artículo: “A Web-Based Tool to enhance teaching/learning database normalization”. Como elemento negativo podemos citar los siguientes: no refleja el procedimiento de la obtención de los resultados; no incluye dentro de sus funcionalidades la obtención de las claves candidatas, cierre de un descriptor y el cálculo de la cobertura minimal del modelo; normaliza sólo hasta la 3FN; restringe el modelo a diez unidades, como máxima cantidad de dependencias funcionales que se pueden introducir; no permite cargar un modelo de BD almacenado; así como también, no permite salvar los modelos para que sean utilizados nuevamente, estas últimas características pueden influir de manera negativa en la utilización de la herramienta por los estudiantes; no está creada para identificar el nivel de normalización que poseen los modelos de datos introducidos

-
3. Es el segundo lenguaje de programación más antiguo (*después de Fortran*) de alto nivel. Lenguaje multi-paradigma, o sea, de tipo orientado a objeto y declarativo.
 4. Es un tipo de explicación desarrollada para uno mismo con el fin de aclararse una circunstancia, proceso que incluye tanto recordar principios ya conocidos como sustituir principios antiguos por otros nuevos.
 5. Es un código JAVA que carece de un método principal, se utiliza principalmente para el trabajo de páginas web, es un pequeño programa que es utilizado en una página HTML y representado por una pequeña pantalla gráfica dentro de esta.

por el usuario; restringe a crear modelo de datos de una sola relación; además no incluye la posibilidad de integrarse con los SGBD.

DBDesignTools

Es una librería en construcción (versión 1.0) escrita en el lenguaje de programación C++, con el fin de resolver algunos de los problemas de la estructura lógica del modelo de datos que se muestra en cualquier diseño de BDR. Esta librería implementa algunos algoritmos relacionados con la normalización de BDR como son : el cálculo del cierre de un conjunto de dependencias funcionales, la proyección de un conjunto de atributos con respecto a un conjunto de dependencias funcionales, equivalencia entre conjuntos de dependencias funcionales, cierre de un conjunto de atributos, cálculo de claves candidatas, cobertura minimal de un conjunto de dependencias funcionales y la comprobación de los diferentes niveles de normalización hasta FNBC [13].

A pesar de constar con una herramienta que ponga en manos de nuestra comunidad la implementación de un gran número de algoritmos importantísimos, que son usados casi en su totalidad como complemento para darle solución a los diferentes problemas del diseño lógico de las BDR, aún no permite transformar el modelo actual en modelos normalizados que cumplan con los diferentes niveles de normalización, digase 2FN, 3FN y FNBC. Además, si se analiza profundamente la implementación de cada una de estas funcionalidades, se percibe que algunos de los algoritmos implementados en *DBDesignTools* no son eficientes, atendiendo a su complejidad temporal; aspecto que el autor de esta herramienta reconoce y pone en manos de la comunidad de desarrolladores para su optimización.

LDBN - Learning Database Normalization

Sistema web creado por Nikolay Georgiev en el 2008, con el fin de brindarle al estudiante una forma fácil y suficiente de comprobar sus conocimientos de la teoría de normalización de bases de datos relacionales, desarrollado por la parte del

cliente en *JavaScript* usando la tecnología *Ajax* para lograr un interfaz del sistema más interactiva al usuario. **LDBN** permite al estudiante a partir de un esquema relacional-universal (URF)⁶, ya sea creado por él o cargado de la aplicación, realizar todo el proceso de normalización del esquema hasta FNBC, incluyendo 2FN y 3FN usando la teoría de la descomposición, dirigidos por un conjunto de pasos que les son brindados como guía para la obtención de la solución. Brinda la posibilidad de obtener la solución correcta del URF en caso que sea solicitado por el usuario [7].

LDBN es, de los software analizados anteriormente con enfoque docente, el que mejor interfaz de usuario y prestaciones brinda para representar todo el proceso de normalización de BDR, esta característica lo puede convertir en uno de los favoritos en esta esfera; como aspectos positivos a resaltar, desde el punto de vista docente, tenemos: la posibilidad de mostrar la evaluación con exactitud de los pasos que componen todo el proceso de normalización, permite cargar un URF lo que asegura un uso muy fácil y rápido de la herramienta sin necesidad de realizar la entrada completamente por teclado.

A pesar de todas sus bondades, presenta al igual que el resto algunos inconvenientes, los más significativos están relacionados con la no representación del proceso de obtención de las claves candidatas, así como el cálculo de cierre de descriptores, los cuales constituyen dos pasos de vital importancia para comenzar a aplicar el proceso de normalización, y que le resultaría de mucho apoyo al estudiante para alcanzar la solución correcta, además de no tener la capacidad de integrarse con los SGBD.

A continuación se mostrará en resumen el comportamiento de estas herramientas con relación a la nueva propuesta.

(1): Web-Based Tool	(4): LDBN
(2): Normit	(5): JMathNorm
(3): DBDesignTool	(6): SINORGES

Tabla 1. Comportamiento de las herramientas existentes con relación a la propuesta

Software	(1)	(2)	(3)	(4)	(5)	(6)
Interfaz amigable	No	Sí	-	Sí	No	Sí
Algoritmos Disponibles			Algunos		Todos	
Representación del proceso	No	Sí	-	Sí	No	Sí
Cargar en SGBD.			No		Sí	
Salvar Modelo Relacional	No	No	-	Sí	No	Sí
Cargar script			No		Sí	
Idioma en la interfaz			Inglés			Español

6. Una relación con todos los atributos y el conjunto de dependencias funcionales de la DB.

Este sistema se encuentra en fase de prueba por la entidad de la Universidad de las Ciencias Informáticas (UCI) y va a ser utilizado próximamente en la asignatura de Sistemas de Bases de Datos en la carrera de Ingeniería Informática, así como también, servirá de apoyo para el desarrollo de sistemas informáticos realizados en la entidad. Con grandes perspectivas de ponerlo a la luz para toda la comunidad de desarrolladores en el mundo.

IV. CONCLUSIONES

- La integración del proceso de normalización con los *SGBD* permite desarrollar modelos robustos, eficientes, con mayor nivel de flexibilidad ante cambios introducidos en el negocio y favorece la interrelación de la etapa de diseño con la etapa de implementación del modelo de datos, demostrándose la importancia de la normalización en el entorno productivo.
- La integración del proceso de normalización con los *SGBD* contribuye a la disminución del tiempo de desarrollo de software en soluciones informáticas que manipulan un gran volumen de información.
- La visualización de los algoritmos utilizados en el proceso de normalización de bases de datos relacionales, a través del sistema informático “*SINORGES*”, contribuye a mejorar la asimilación de los contenidos teóricos de la asignatura “Sistemas de Bases de Datos” que se imparte en las carreras del perfil informático.
- El sistema “*SINORGES*” que se desarrolló puede utilizarse en cualquier empresa productora de software y en universidades donde se estudien carreras con un perfil informático.

V. BIBLIOGRAFÍA

- [1] Batini C. 2004. Diseño conceptual de bases de datos, pp 3-4.
- [2] Cabrera N., García C., González L., y Rodríguez A., 2007. Diseño y validación estructural de esquemas conceptuales utilizando una herramienta CASE. Revista Cubana de Ciencias Informáticas, Vol. 1, No.4, pp. 72-81.
- [3] Coronel C., y Rob P. 2003. Sistemas de bases de datos: diseño, implementación y administración. 5ta edición. pp. 7, 21-22, 30-31, 196-197.
- [4] Covadonga M., 1987. El modelo racional de datos: De los fundamentos a los modelos deductivos. Ediciones Díaz de Santos, 11 P.
- [5] Date C. J., 2001. Introducción a los Sistemas de Bases de Datos. Séptima Edición, pp 9-10, 845, 331-333, 350-351.
- [6] Domínguez C., Pérez T., y Elizondo A., 8-10 julio 2009. ¿Todavía interesa normalizar bases de datos? XV JENUI. Barcelona. En: <http://upcommons.upc.edu/revistes/bitstream/2099/7884/6/p122.pdf>
- [7] Georgiev N., September 2008. A Web-Based Environment for

Learning Normalization of Relational Database Schemata.
Department of Computing Science, Umea University, pp 2-19.

- [8] Herrera N., 2006. Organización de archivos y Bases de Datos I. Argentina. Universidad Nacional de San Luis. En: <http://www.dirinfo.unsl.edu.ar/~oaybd1/>
- [9] Jui-Kung H., y Tung H-L. 2006. A WEB-BASED TOOL to enhance teaching/learning database normalization. Southern Association for Information Systems Conference.
- [10] Karakaya Z., y Yazici A. 2006. Normalizing Relational Database Schemas Using Mathematica. LNCS. Springer-Verlag Berlin Heidelberg, Vol.3992, pp 375-382.
- [11] Karakaya Z., y Yazici A. ICCS 2007. JMathNorm: A Database Normalization Tool Using Mathematica. Springer-Verlag Berlin Heidelberg. Vol. Part II, pp. 186-193.
- [12] Kroenke D. M. 2003. Procesamiento de bases de datos: fundamentos, diseño e implementación. Octava Edición.
- [13] Martín O. 2006. Documentación de DBDesign Tools. En: <http://sourceforge.net/projects/dbdesigntools/>
- [14] MITROVIC A. 2005. The Effect of Explaining on Learning: a Case Study with a Data Normalization Tutor. University of Canterbury, New Zealand. Intelligent Computer Tutoring Group Department of Computer Science and Software Engineering. Retrieved from. En: <http://www.cosc.canterbury.ac.nz/tanja.mitrovic/NORMIT-AIED05.pdf>
- [15] MITROVIC A. 2002. Supporting Self-Explanation in a Data Normalization Tutor. University of Canterbury. Private Bag 4800, Christchurch, New Zealand. Intelligent Computer Tutoring Group. En:http://www.cs.usyd.edu.au/~aied/vol9/vol9_roger_Mitrovic.pdf
- [16] Quiroz J. 2003. El modelo relacional de bases de datos. Boletín de Política Informática, No. 6.
- [17] Ullman J.D. 1988. Principles of Database and Knowledge Base Systems. Computers Science Press. Vol. I.

Universidad Nacional de Colombia Sede Medellín

Facultad de Minas



Escuela de Ingeniería de Sistemas

Pregrado

- ❖ Ingeniería de Sistemas e Informática.



Posgrado

- ❖ Doctorado en Ingeniería-Sistemas.
- ❖ Maestría en Ingeniería de Sistemas.
- ❖ Especialización en Sistemas con énfasis en:
 - Ingeniería de Software.
 - Investigación de Operaciones.
 - Inteligencia Artificial.
- ❖ Especialización en Mercados de Energía.

Escuela de Ingeniería de Sistemas
Dirección Postal:
Carrera 80 No. 65 - 223 Bloque M8A
Facultad de Minas. Medellín - Colombia
Tel: (574) 4255350 Fax: (574) 4255365
Email: esistema@unalmed.edu.co
<http://esisis.unalmed.edu.co/>

