

Estudio comparativo sobre herramientas de software para la enseñanza del diseño conceptual de bases de datos

Lorena Chavarría-Báez¹, José Antonio Hajar-Miranda², Dario Emmanuel Vázquez Ceballos³

Facultad de Ingeniería Mecánica y Eléctrica
Universidad Autónoma de Nuevo León

SEPI-ESCOM IPN, Av. Juan de Dios Bátiz s/n, esquina con Miguel Othón de Mendizábal, México, D.F.,
C.P. 07738

{lorena.chavarria¹, jahm1989², kienkane³}@gmail.com

Fecha de recepción: 5 de septiembre 2014

Fecha de aceptación: 2 de octubre 2014

Resumen

La enseñanza de bases de datos juega un rol importante en la formación de estudiantes de ingeniería en sistemas y áreas afines. Dentro del área de base de datos, el proceso de diseño es uno de los conceptos clave con los que debe contar un ingeniero en sistemas. El uso de herramientas de software que apoyen a la enseñanza es una práctica que ha tenido éxito en otras áreas, por lo que aplicar este concepto a las bases de datos podría ser factible. Actualmente existen herramientas de software que han sido usadas para fortalecer los conceptos de diseño de bases de datos. En este trabajo se hace una comparativa entre estas herramientas de software con el fin de identificar las características que deben tener las herramientas de software para la enseñanza de las bases de datos

Palabras Clave

Base de datos, diseño conceptual de base de datos, enseñanza de base de datos, herramientas de software para base de datos

1 Introducción

Los cursos de Base de Datos (BD) juegan un papel trascendental en la formación de alumnos a nivel licenciatura de las áreas de cómputo y sistemas. Su relevancia es tal, que organismos internacionales como la IEEE y la ACM [4], [6] han identificado un conjunto de temas básicos que en algún momento los educandos deben analizar y, sobretodo, dominar. Entre estos temas se encuentran los conceptos de sistemas de BD, el modelado de datos, las BD's relacionales, los lenguajes de consulta de BD, el diseño de BD relacionales, el procesamiento de transacciones, las BD's distribuidas, y el diseño físico de BD. Estos temas se encuentran organizados de forma que se parte de una solución conceptual del problema hasta la implementación y manipulación de la BD en un sistema computacional. Esta organización también se presenta en el ciclo de vida de una BD ya que, una vez hecha la definición del sistema, se debe realizar el diseño conceptual y físico de la misma, implementar el diseño en un Sistema Gestor de BD, hacer la carga/conversión de datos/aplicaciones, realizar las pruebas pertinentes, poner el sistema en operación y efectuar la supervisión y el mantenimiento. Por lo tanto, no es de extrañarse que en los cursos actuales de BD se siga una estructura similar para su enseñanza: primero, se revisa el diseño conceptual, dado por el modelo Entidad – Relación propuesto en 1976 por Peter P. Chen [14], y, posteriormente, se analizan los conceptos restantes.

El uso de herramientas de software es una opción que ha probado ser útil para la docencia en otras áreas, los resultados muestran que el aprendizaje de los alumnos ha sido favorable. Hoy día existen diferentes herramientas orientadas a apoyar en diferentes aspectos a la enseñanza de BD. En ese trabajo se abordan herramientas dedicadas al proceso de diseño de las BD's. También se presenta una comparativa de dichas herramientas, mostrando las ventajas y desventajas de cada una con respecto al diseño. Este estudio puede ser usado como base para la creación de una herramienta que combine lo mejor de las herramientas existentes. En particular, se detectó que es importante contar con un módulo para la verificación de errores en los diagramas y no sólo con un editor de los mismos. Adicionalmente, es deseable poder evaluar el progreso de los estudiantes y que la herramienta sea accesible desde Internet.

El resto del artículo está organizado de la siguiente manera: en la Sección 2, se describe el diseño conceptual de BD's y se introduce la forma en la que actualmente se enseña. En la sección 3 se presentan herramientas de software capaces de ayudar en la enseñanza del proceso de diseño de BD's, principalmente para la realización de los diagramas entidad – relación. En la Sección 4, se muestra una comparativa de las herramientas mostradas en la sección anterior, indicando las ventajas y desventajas de cada una en diferentes aspectos. Finalmente, la Sección 5 concluye este trabajo

2 Enseñanza del Diseño Conceptual de Base de Datos

2.1 El Diseño Conceptual de Base de Datos

El modelo Entidad – Relación es un modelo de datos de alto nivel que usa una especificación gráfica, es decir, es comprensible por varias personas sin necesidad de un entrenamiento exhaustivo. Consiste de elementos visuales que permiten representar entidades, conjuntos que agrupan objetos con características similares; sus propiedades o atributos; y relaciones, que definen asociaciones entre entidades y que también pueden tener atributos.

El éxito de este modelo se debe a su gran capacidad para representar un sistema en un nivel de abstracción alto y su fuerte e importante conexión con el modelo relacional. La traducción del modelo Entidad – Relación al modelo relacional puede lograrse usando un algoritmo que establece las equivalencias entre los elementos de ambos modelos. La Tabla 1 resume tales equivalencias.

La figura 1 muestra los elementos principales del modelo Entidad – Relación.

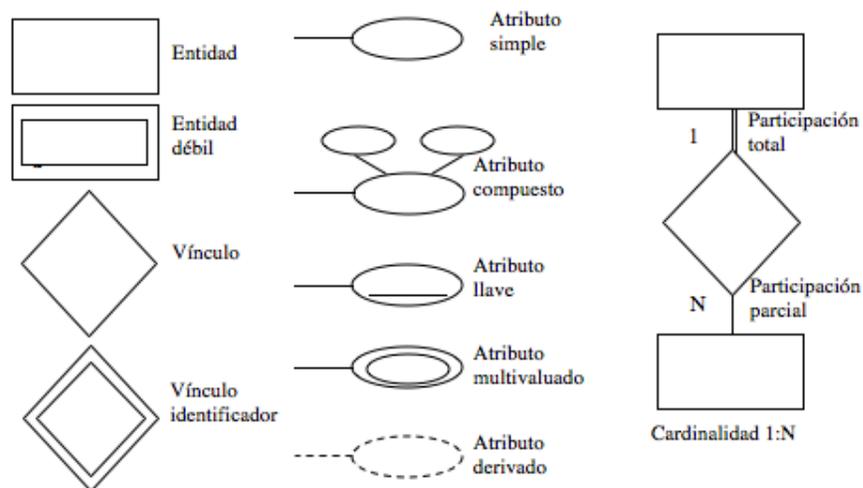


Figura 1. Elementos principales del modelo Entidad – Relación.

Modelo Entidad – Relación	Modelo Relacional
Entidad	Tabla
Atributo	Columna
Llave primaria	Restricción de llave primaria
Llave candidata	Restricción única
Atributo obligatorio	Restricción NOT NULL
Relación Muchos a Uno	Llave externa (restricción NOT NULL de una relación obligatoria)
Relación Uno a Uno	Restricción foránea y única (restricción NOT NULL para una relación obligatoria)
Relación Muchos a Muchos	Tabla nueva, llaves primarias, restricciones de llaves foráneas.
Entidad Débil	Restricción de llave externa, restricción de llave primaria

Tabla 1. Equivalencias entre el modelo Entidad – Relación y el modelo relacional.

La figura 2 muestra un diagrama entidad – relación y su correspondiente modelo relacional.

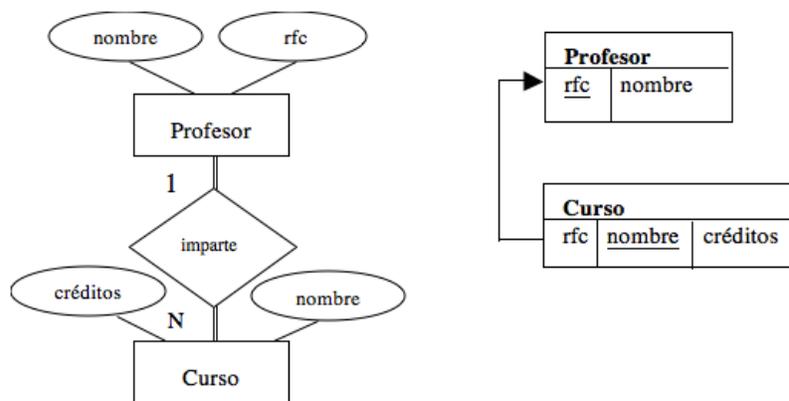


Figura 2. Diagrama entidad – relación y su respectivo modelo relacional.

2.2 Enseñanza del Diseño Conceptual de Base de Datos

En la literatura se han propuesto diferentes enfoques para la enseñanza del modelo •Entidad – Relación. El esquema más empleado expone primero la descripción de un problema que luego los estudiantes deben analizar para construir el diagrama entidad – relación correspondiente. Por cada ejercicio presentado en el curso, se obtiene un modelo canónico y una serie de soluciones distintas que difieren en algunos elementos del diagrama canónico pero que siguen representando la naturaleza del problema. Trabajos como el presentado en la referencia [15] han tratado de cambiar el paradigma de enseñanza usando un enfoque constructivista en el que el diseño se aprende con la práctica. Sin embargo, tal como lo indican los autores, el

empleo de este método implica una mayor carga de trabajo y requiere de estudiantes con un grado de madurez mayor, lo cual no siempre es posible.

Aunque los conceptos del modelo Entidad – Relación son relativamente fáciles de entender, ya que es un modelo de alto nivel, los estudiantes tienen dificultades cuando desarrollan los diagramas porque el texto de los problemas es ambiguo y/o incompleto, no existe una única solución y, si no está el profesor presente, no tienen forma de verificar si es correcta y/o no existen herramientas adecuadas que puedan auxiliarlos en la creación de los modelos. En un intento por apoyar a los educandos a asimilar la construcción de diagramas entidad –relación, se han hecho esfuerzos por integrar las nuevas tecnologías de información en el proceso de enseñanza. De esta forma, se han desarrollado herramientas de software que toman en cuenta las habilidades de los jóvenes; sin embargo, todavía queda mucho por hacer en este rubro ya que las propuestas actuales se limitan a permitir el trazo de diagramas sin indicar si son correctos o no.

3 Herramientas de Software Orientadas a la Enseñanza del Diseño Conceptual de Bases de Datos

En este apartado se abordan algunas herramientas orientadas a la enseñanza del modelo Entidad – Relación, principalmente. Adicionalmente, se describen algunas asociadas con el modelo relacional y las consultas SQL.

ERM – VLE (Entity Relationship Modelling in a Virtual Learning Environment). Utiliza el paradigma Multi User Dimensions (MUD), usado en muchos juegos, como una forma de desarrollar un entorno virtual de aprendizaje basado en notaciones textuales para construir diagramas entidad – relación [1].

COLER (Collaborative Learning environment for Entity Relationship modeling) es un ambiente de aprendizaje colaborativo basado en Internet que facilita a participantes remotos el practicar sus habilidades colaborativas al resolver, en pequeños equipos, problemas de modelación conceptual de BD's [2].

CODASYS, está pensada para ayudar a realizar diagramas entidad – relación que puedan ser transformados en esquemas de BD's normalizados, libres de dependencias derivadas. Se basa en estudios previos de los mismos autores en donde analizan los errores más frecuentes cometidos por los diseñadores inexpertos y la forma de mejorar el diseño conceptual a través de comentarios. No soporta relaciones unarias, sólo permite una relación entre dos entidades [3].

KERMIT, es un sistema de tutoría que aborda el diseño conceptual de BD's usando el modelo de datos Entidad – Relación. Es un entorno de resolución de problemas en el que los estudiantes implementan esquemas entidad – relación que satisfacen un determinado conjunto de requerimientos. El sistema proporciona retroalimentación a los conocimientos de cada estudiante. Los principales componentes de KERMIT son su interfaz de usuario y su módulo pedagógico. Contiene un número de problemas de

BD's predefinidos junto con sus soluciones, especificadas por un humano experto en el tema [5].

DATsys, proporciona un entorno para la realización de diagramas a través de CourseMarker. Actualmente se han extendido sus capacidades de corrección automática. Han desarrollado comandos específicos para puntuar los diagramas entidad – relación y herramientas para su dibujo [7].

#EER es una herramienta diseñada para facilitar el diseño de BD's a los alumnos, automatizando algunos de los procesos. Está basada en el modelo Entidad – Relación Extendido y, por lo tanto, permite representar este tipo de diagramas. Una vez dibujado el diagrama, el sistema realiza de forma automática la generación del esquema de BD relacional equivalente y también puede generar los comandos de definición de datos para generar la BD. Está pensada para que los alumnos que sólo realizan un breve curso introductorio de BD's puedan generar los comandos necesarios para la creación de la BD diseñada. No es una herramienta de corrección automática, ni tampoco ofrece ningún tipo de retroalimentación al alumno [8].

GNU Sqltutor es un sistema Web de tutorial interactivo dirigido a la práctica de Structured Query Language (SQL). En el primer paso se selecciona un tutorial de la página inicial. Después, se resuelven una serie de preguntas. Cuando termina la ejecución, Ssqltutor muestra la evaluación final con la revisión de todas las preguntas formuladas durante la sesión de consultas SQL del usuario y se presentan las correcciones a las respuestas equivocadas. Ssqltutor presenta un sistema de BD relacional PostgreSQL. El proyecto consiste de un programa en C++ publicado bajo la Licencia Pública General GNU (GNU GPL). El programa de scripts CGI que selecciona preguntas SQL desde su BD verifica las respuestas y evalúa el puntaje final. La segunda parte del sistema es una colección preguntas y respuestas que representan a los tutoriales de SQL. Ssqltutor permite ejecutar varios tutoriales en diferentes idiomas a partir de una BD única [12].

Relational es una herramienta educativa que dispone de una interfaz de usuario que permite cargar y guardar relaciones, ejecutar consultas sobre ellas y mostrar los resultados. Está desarrollada en Python y tiene una Licencia Pública General de GNU (GNU GPL). Permite la optimización de consultas simples pero no la realiza en función de estadísticas provenientes de las relaciones. No existe traducción entre el álgebra relacional y SQL ni tampoco devuelve la definición de árboles de expresiones al álgebra relacional. Está disponible para Windows, Mac OS X y Debian. No presenta una documentación amplia y la ayuda al usuario es limitada [13].

LEAP RDBMS es una herramienta orientada a la educación desarrollada en Oxford Brookes University. Está desarrollada en C y tiene una Licencia Pública General de GNU (GNU GPL). Se distribuye el código fuente y el usuario requiere compilarlo. No cuenta con una interfaz gráfica de usuario y se basa en comandos de terminal.

Además, presenta una sección de ayuda al usuario limitada y no permite la optimización de consultas. Permite la obtención de árboles de expresiones en álgebra relacional, pero al carecer de interfaz gráfica son mostrados en modo texto. Presenta una amplia documentación adjunta [10].

WinRDBI es una herramienta educativa desarrollada en la Arizona State University. Su versión actual está implementada en Java y cuenta con una interfaz gráfica de usuario que permite cargar/guardar relaciones y consultas. No dispone de ayuda para la introducción de las consultas. Permite la carga de relaciones a través de diferentes formatos de archivo. No permite la optimización de consultas. La documentación es amplia y proponen algunos ejemplos incluyendo una BD y consultas. Dispone de instalador y es multiplataforma [4].

RelationalQuery fue desarrollada en la Universidad de Sevilla, es una herramienta educativa desarrollada en Java, que posee una interfaz gráfica de usuario que permite almacenar/recuperar consultas y elegir el lenguaje en el que se va a trabajar (álgebra relacional, TRC o SQL). Permite la realización de consultas sobre una BD SQL (trae incorporado un driver JDBC para Oracle aunque pueden utilizarse otros drivers). Se comporta como un traductor de consultas entre diferentes lenguajes [11].

Gradiance SQL. En la enseñanza de SQL utiliza una combinación de ejercicios de construcción de consultas escrituras y cuestionarios, para enfatizar el paradigma de aprender programando, junto a un conjunto de material pertinente. Los ejercicios de programación se encuentran en Internet y cuentan con retroalimentación constructiva acerca de la exactitud de las consultas enviadas. Los casos de prueba, comentarios y sugerencias se proporcionan para ilustrar los errores y ayudar a depurar las consultas on-line. El resultado es un enfoque de enseñanza automatizada a través de ejercicios de consultas y programación. Gradiance para SQL y otros temas son ampliamente utilizados en clases de BD's en Stanford [9].

4 Comparación de Herramientas de Software para el Diseño Conceptual de Bases de Datos

A continuación se presenta una tabla comparativa entre algunos aspectos de las herramientas presentadas en la sección anterior. Los elementos que se consideraron fueron: diseño conceptual, verificación, normalización, monitoreo del progreso y aprendizaje a distancia. Los tres primeros aspectos están relacionados directamente con el apoyo que ofrecen para realizar diagramas de BD's. Los últimos dos elementos tienen que ver con el soporte que brindan para evaluar a los estudiantes y la ayuda disponible para acceder de manera remota.

	Aspectos a comparar				
	Diseño conceptual	Verificación	Normalización	Aprendizaje a distancia	Monitoreo de progreso
ERM-VLE	Si	No	No	No	Si
COLER	Si	--	No	Si	Si
CODASYS	Si	No	Si	--	Si
KERMIT	Si	Si	No	Si	No
DATsys	Si	Si	No	--	Si
#EER	Si	No	Si	No	No
GNU Sqtutor	No	Si	No	Si	No
Relational	Si	Si	No	No	--
LEAP RDBMS	No	Si	No	No	No
WinRDBI	No	No	No	No	No
Relational Query	No	Si	No	No	Si
Gradiance SQL	No	Si	No	Si	Si

Tabla 2. Comparación entre las herramientas

Como puede verse, todas las herramientas que están directamente relacionadas con el diseño conceptual de BD's permiten dibujar diagramas, pero sólo dos cuentan con la verificación de los mismos, es decir, sólo esas dos le permiten al estudiante identificar sus errores. Es importante notar que esta carencia dificulta la asimilación de los conceptos por parte de los estudiantes porque es necesaria la presencia del profesor para resolver dudas y corregir errores. Por lo tanto, es necesario seguir trabajando en el desarrollo de nuevas alternativas

5 Conclusión

Las herramientas presentadas en este trabajo poseen características que pueden ser aprovechadas para fortalecer la enseñanza de algunos de los tópicos relevantes en BD's. Con base en la tabla 2, se puede observar que algunas herramientas poseen una interfaz gráfica para realizar el modelado conceptual de las BD's y un módulo de verificación. Otras herramientas, poseen módulos para dar seguimiento a las actividades del usuario y algunas funciones que actúan como guías para los alumnos. Y en algunos otros casos, el aprendizaje es a distancia con herramientas basadas en Web. Estos factores son factibles para la enseñanza del diseño de BD's; sin embargo, si una herramienta tuviera estas características de manera integral, ayudaría de una mejor manera a la enseñanza de BD's. Este estudio ha permitido identificar los requisitos que deben tener las herramientas de software para el aprendizaje de BD's.

Referencias

- [1] Hall, L., & Gordon, A. (1998, March). A virtual learning environment for entity relationship modelling. In *ACM SIGCSE Bulletin* (Vol. 30, No. 1, pp. 345-349). ACM.
- [2] de los Angeles Constantino-González, M., & Suthers, D. D. (2000, January). A coached collaborative learning environment for entity-relationship modeling. In *Intelligent Tutoring Systems* (pp. 324-333). Springer Berlin Heidelberg.
- [3] Antony, S. R., & Batra, D. (2002). CODASYS: a consulting tool for novice database designers. *ACM Sigmis Database*, 33(3), 54-68.
- [4] Cvetanovic, M., Radivojevic, Z., Blagojevic, V., & Bojovic, M. (2011). ADVICE— Educational System for Teaching Database Courses. *Education, IEEE Transactions on*, 54(3), 398-409.
- [5] Suraweera, P., & Mitrovic, A. (2002, January). KERMIT: a constraint-based tutor for database modeling. In *Intelligent Tutoring Systems* (pp. 377-387). Springer Berlin Heidelberg.
- [6] Enciso, M., & Soler, E. (2013, March). Teaching database design: A reverse engineering approach. In *Global Engineering Education Conference (EDUCON), 2013 IEEE* (pp. 474-480). IEEE.
- [7] Higgins, C. A., Bligh, B., Symeonidis, P., & Tsintsifas, A. (2009). Authoring diagram-based CBA with CourseMarker. *Computers & Education*, 52(4), 749-761.
- [8] Bogdanović, M., Stanimirović, A., Davidović, N., & Stoimenov, L. (2008). The development and usage of a relational database design tool for educational purposes. In *Informing Science & IT Education Conference (InSITE'08)* (pp. 251-258).
- [9] Jeffrey D. Ullman. 2005. Gradiance On-Line Accelerated Learning. In *Proceedings of the Twenty-eighth Australasian conference on Computer Science - Volume 38 (ACSC '05)*, Vladimir Estivill-Castro (Ed.), Vol. 38. Australian Computer Society, Inc., Darlinghurst, Australia, Australia, 3-6.
- [10] Leyton, R. (2010). LEAP RDBMS: An Educational Relational Database Management System. leap. sourceforge. net.

- [11] Gutiérrez, J. J., Escalona, M. J., Villadiego, D., & Mejías, M. (2005). Comparativa de herramientas para la enseñanza de lenguajes relacionales. Actas de las XI Jornadas de Enseñanza Universitaria de la Informática.
- [12] Landa, M., & Cepek, A. (2010). Learning PostGIS using Sqltutor.
- [13] Relational 1.1. <http://galileo.dmi.unict.it/wiki/relational/doku.php>. Última consulta 02/10/2014.
- [14] Peter Pin-Shan Chen. 1976. The entity-relationship model—toward a unified view of data. *ACM Trans. Database Syst.* 1, 1 (March 1976), 9-36. DOI=10.1145/320434.320440 <http://doi.acm.org/10.1145/320434.320440>
- [15] Connolly, T. M., & Begg, C. E. (2006). A constructivist-based approach to teaching database analysis and design. *Journal of Information Systems Education*, 17(1), 43.