

# Herramienta para la enseñanza de Modelado Conceptual de Bases de Datos

Mg. R Bertone, Mg. P Thomas, A.C. S Antonietti, A.C. C Miglio

Instituto de Investigaciones en Informática LIDI. Facultad de Informática. UNLP  
{pbertone,pthomas}@lidi.info.unlp.edu.ar, {soledadantonietti,arielmiglio}@gmail.com

**Abstract.** Este trabajo presenta una herramienta desarrollada para el Modelado Conceptual de Bases de Datos, denominada CASER. Esta herramienta fue concebida para utilizarse con fines didácticos por la cátedra Introducción a las Bases de Datos de la Facultad de Informática de la UNLP. En general, las herramientas disponibles en el mercado, ya sea de uso libre como aquellas comerciales, están orientadas a usuarios experimentados que generan modelos de datos, a partir de la toma gradual de decisiones vinculadas con el conocimiento propio. De esta forma, los alumnos en la fase de aprendizaje de Diseño Conceptual, carecen de un software para generar esquemas conceptuales de datos en forma simple, asistida y automatizada. La herramienta aquí propuesta respeta el desarrollo gradual de un esquema conceptual, para lo cual, incorpora dos funcionalidades básicas: un editor de texto para trabajar con la especificación de un problema, y un editor gráfico para visualizar el esquema conceptual que se genera en forma asistida a partir de la especificación anterior.

**Keywords:** Modelado Conceptual. Modelo de Entidades y Relaciones. Diseño Conceptual. Herramienta Educativa.

## 1 Introducción

El diseño de una Base de Datos (BD) es un proceso complejo que abarca decisiones a distintos niveles. La complejidad de un problema se aborda mejor si se descompone dicho problema en subproblemas y se resuelve cada uno de éstos independientemente, utilizando métodos y técnicas específicas. El diseño de BD se descompone en diseño conceptual (en el que se centra este trabajo), diseño lógico y diseño físico [1].

El diseño conceptual parte de la especificación de requerimientos y su resultado es el esquema conceptual de la BD. Un esquema conceptual es una descripción de alto nivel de la estructura de la BD, independientemente del software Gestor de Bases de Datos (DBMS) que se utilice para administrarla. Un modelo conceptual es un lenguaje que se usa para describir esquemas conceptuales. El propósito del diseño conceptual es describir el contenido de información de la BD, en lugar de las estructuras de almacenamiento que se necesitarán para administrar esta información.

La tarea de diseño conceptual no es sencilla y consiste en un proceso de naturaleza iterativa, que comienza con una versión inicial y sufre transformaciones sucesivas, hasta arribar a una versión estabilizada. Finalmente, el esquema obtenido representará la información del problema a resolver en concordancia con las necesidades del usuario.

El diseño conceptual tiene diferentes enfoques. El utilizado por la Cátedra de Introducción a las Bases de Datos de la Facultad de Informática de la Universidad Nacional de La Plata (UNLP), es el propuesto por los autores Batini, Navathe y Cieri [1]. Este modelo no se condice directamente con ninguna herramienta del mercado (libre o paga) que facilite y asista esta tarea de modelar los tres niveles: conceptual, lógico y físico. Herramientas tales como DBDesigner o Erwin solamente apuntan a resolver el modelo lógico y físico, con metodologías que no respetan las bases del modelo de Entidades y Relaciones propuesto por Chen.

Sin herramientas, el proceso de definición de un esquema conceptual se transforma en una tarea manual muy tediosa, para el cual se utiliza solo papel y lápiz, o herramientas de edición gráfica que no están pensadas y diseñadas para trabajar con esquemas conceptuales de datos.

A continuación se resumen los conceptos más importantes de Modelado Conceptual; posteriormente se presenta a la herramienta creada para asistir con el modelado (CASER). Por último se muestran los resultados obtenidos a partir de la aplicación piloto durante 2008 en uno de los cursos de la asignatura, que llevaron a decidir utilizar la herramienta para los 500 alumnos de la cátedra durante el segundo semestre de 2009.

## **2 Modelo conceptual de datos**

Los elementos básicos previstos por el modelo Entidad Relación (ER) son entidades, interrelaciones y atributos. Los términos “entidad” e “interrelación” denotan clases de objetos. Las entidades representan clases de objetos de la realidad y se representan gráficamente por medio de rectángulos. Las interrelaciones, que representan agregaciones de dos o más entidades, son graficadas con rombos.

Las interrelaciones que conectan una entidad consigo misma son denominadas “anillos”. También son conocidas como interrelaciones recursivas. Para distinguir entre los roles de la entidad en la interrelación, se asocian dos rótulos con la entidad.

Cada interrelación tiene un significado específico; por ello, deben definirse nombres nemotécnicos para éstas. Se caracterizan además por la cardinalidad, que expresa cuántas entidades de un extremo de la relación están relacionadas con las entidades del otro extremo. La cardinalidad se expresa con un valor mínimo y un valor máximo. En [1] se propone las cardinalidades como pares de valores indicando el valor mínimo y máximo, representándolas cerca de la conexión entre la entidad (rectángulo) y la relación (rombo).

El tercer elemento primordial para la construcción del esquema lo representan los atributos. Estos definen las propiedades básicas de las entidades o interrelaciones. Al igual que las interrelaciones, los atributos se caracterizan por su cardinalidad mínima y máxima. La cardinalidad mínima indica el número mínimo de valores de un atributo. La cardinalidad máxima indica el número máximo de valores de un atributo asociado con cada entidad o interrelación. En función de la cardinalidad, se definen dos tipos de atributos: monovalentes y polivalentes. Que un atributo pertenezca a uno u otro, tipo depende exclusivamente de su cardinalidad máxima; si equivale a 1, el atributo es monovalente, en caso de que sea mayor a 1 es polivalente.

Estos tres elementos básicos fueron propuestos por Chen en 1976, a partir de esa definición inicial se han incorporado más elementos que dotan al modelo conceptual de mayor expresividad en el momento de representar los datos de un problema.

Distintos autores [1][3][4][5][6][9] plantean diferentes enfoques respecto de los elementos que enriquecen conceptualmente el esquema. La cátedra considera como elementos adicionales: atributos compuestos, jerarquías de generalización e identificadores.

Los atributos compuestos son grupos de atributos que tienen afinidad en cuanto a su significado o a su uso.

Las jerarquías de generalización permiten establecer generalización entre las entidades. Una entidad E es una generalización de un grupo de entidades E1, E2,...En, si cada objeto de las clases E1, E2,...En, es también un objeto de la clase E. Se cumple en este caso la relación “*es un*” entre las entidades E1..En (consideradas hijos) y la entidad genérica E (considerada su padre).

Por último, un identificador de una entidad E es un grupo de atributos o de entidades relacionadas con E, que tiene la propiedad de determinar en forma única todos los casos de E. Los identificadores pueden ser simples o compuestos, e internos o externos. Los identificadores simples están definidos a partir de un solo atributo, los compuestos de más de uno. Una entidad tiene identificadores internos, si solamente utiliza atributos propios, en caso de necesitar un atributo de otra entidad el identificador para a ser externo.

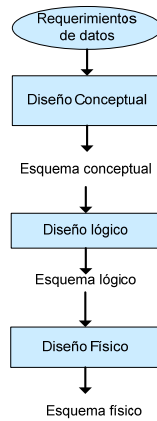
### **3 Marco Teórico Utilizado para el Modelado de Datos**

El diseño de base de datos se descompone en diseño conceptual, diseño lógico y diseño físico, en una secuencia que los representa en la figura 1. La sección anterior describió los principales conceptos del diseño conceptual. El diseño lógico parte del esquema conceptual y genera como resultado un esquema lógico. Un esquema lógico es una descripción de la estructura de la base de datos que puede ser procesado por un DBMS.

El diseño físico parte del esquema lógico y obtiene como resultado un esquema físico; el cual es una descripción de la implantación de una Base de Datos en memoria secundaria; además, describe las estructuras de almacenamiento y los métodos utilizados para tener un acceso efectivo a los datos.

Las herramientas existentes en el mercado que sirven de apoyo en el diseño de BD, se orientan a usuarios con experiencia en esa tarea, y toman como punto de partida al diseño lógico en el proceso de construcción.

A partir de lo expuesto, resulta imprescindible disponer de una herramienta que asista a las tareas de diseño conceptual, ya que esta fase de construcción permite la incorporación de conocimiento más naturalmente en el diseño de base de datos. Un alumno en la fase de aprendizaje no puede obviar o soslayar esta etapa.

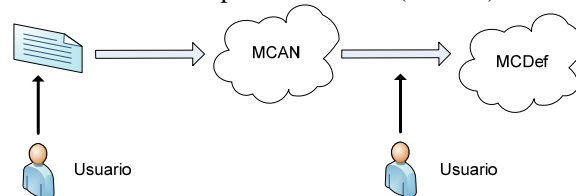


**Fig. 1.** Etapas de diseño.

La herramienta propuesta en este trabajo provee la generación asistida de un esquema conceptual. Se parte de una especificación detallada del problema, y en forma semi-automática, se genera el esquema conceptual, con el fin de facilitar el proceso de enseñanza y aprendizaje.

Por razones prácticas, y a partir de la discusión generada en la Cátedra de Introducción a las Bases de Datos, se hace referencia a Modelo o Esquema Conceptual, indistintamente.

La figura 2 resume el comportamiento propuesto para CASER, el comienzo de la actividad se inicia desde la especificación general de un problema, se trabaja con el documento de requerimientos disponible, y se genera un Modelo Conceptual de Alto Nivel (MCAN). Luego, a partir de un proceso de depuración y refinado a cargo del usuario, se obtiene el Modelo Conceptual Definitivo (MCDef).



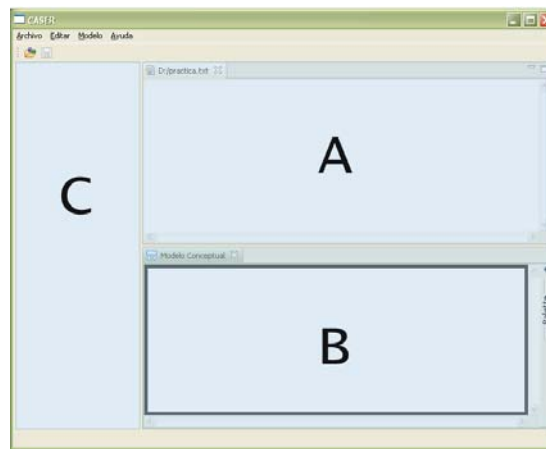
**Fig. 2.** generación asistida con la herramienta e incidencia de los usuarios.

#### 4. CASER: Herramienta para Modelado Conceptual

La herramienta propone un entorno, que permite definir el modelo de datos conceptual a partir de la especificación detallada de un problema. CASER está concebida para utilizarse en el ámbito académico, aunque también podría utilizarse con propósitos comerciales.

El proceso de construcción de un modelo conceptual comienza con la disponibilidad de una especificación del problema (especificación de requerimientos); continúa con la generación preliminar del modelo (MCAN) en forma asistida; finalizando con la construcción detallada del modelo conceptual (MCDef). Cabe aclarar que se excluye explícitamente la tarea de construcción de la Especificación de Requerimientos, dado que esa labor no es pertinente para el propósito de la Cátedra.

En la figura 3 se presenta la interfaz principal de CASER, la cual se divide en tres partes: un editor de texto (área A), un editor gráfico (área B) y una jerarquía representada en un árbol (área C).



**Fig. 3.** pantalla inicial de CASER

El área A corresponde a un editor de texto simple donde se debe incluir la especificación de requerimientos del problema a resolver. La funcionalidad esencial de este editor radica en la posibilidad de realizar marcas específicas sobre distintas palabras o frases, que tengan impacto directo sobre los datos del problema. Es así como estas marcas identifican dentro del texto, potenciales entidades, relaciones y atributos, que como se mencionó previamente, constituyen los tres elementos básicos del modelado conceptual.

El usuario tiene la posibilidad de importar un archivo de texto con la especificación de requerimientos y editarlo, o escribirlo directamente. El rasgo esencial de este editor es poder realizar marcas, lo cual enriquece el contenido de la especificación, en términos de resaltar los conceptos a considerar en la tarea de modelado. Estas marcas se realizan utilizando distintos colores, según representen entidades, relaciones o atributos.

Paralelamente a la selección y marcado de palabras y/o frases, se genera automáticamente y en forma incremental, el modelo conceptual preliminar, de acuerdo a las marcas realizadas sobre el texto. Este modelo conceptual preliminar se presenta en el área B, y en el área C se lo representa como una jerarquía de objetos. La figura 4 resume la descripción realizada.

Esta generación automática e incremental del modelo que ofrece CASER, permite tener una percepción real y rápida, para determinar si la construcción del modelo se orienta a lo que el usuario desea o espera.

En el área de trabajo que corresponde a la representación del modelo con forma de árbol, se representan los objetos del modelo como una estructura jerárquica, la cual, si bien no es estrictamente necesario en términos de la construcción del modelo, ofrece otra visión que beneficia al usuario en su toma de decisiones de diseño.

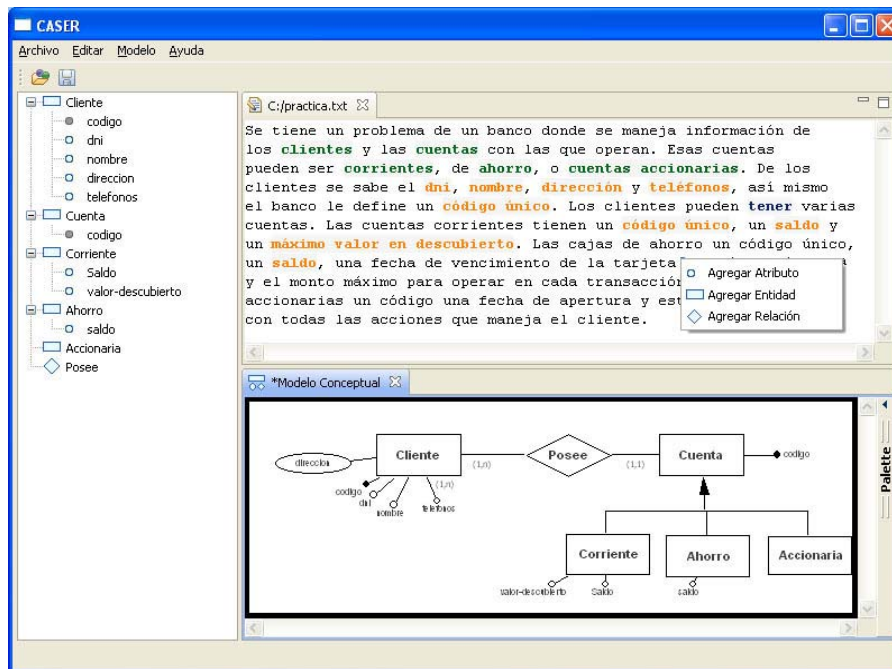


Fig. 4. CASER con un documento en edición

En la figura 5 se presenta la selección de objetos en el árbol. Esta selección permite visualizar las propiedades de estos objetos. La finalidad es permitir al usuario de CASER realizar operaciones de cambio de nombres, determinar si una entidad es subconjunto de otra entidad, especificar cardinalidades en atributos y relaciones, o directamente, eliminar algún objeto. Es de destacar que, cualquier cambio realizado, tiene impacto en la representación grafica, en forma automática.

La figura 6 presenta la interfaz utilizada para la edición de los elementos del diagrama de ER, donde se visualizan sus propiedades y es posible especificar las características de cada una de ellas.

Respecto a las entidades, se puede especificar si se trata de subconjunto de otra entidad genérica (en este caso, especificando de cual depende), o si se trata de una entidad padre (especificando necesariamente el tipo de cobertura).

Si bien desde el punto de vista teórico es factible tener relaciones n-arias, es aconsejable que éstas sean binarias o a lo sumo ternarias. CASER asume esta conveniencia y restringe la representación de relaciones como binarias o ternarias. En este punto, se puede observar que el modelo propuesto carece del concepto de Agregación de Entidades. Este no se encuentra incluido en la propuesta de modelado

utilizada y, por ende, no disponible en la herramienta. Esto hace necesario la posibilidad de disponer la utilización de relaciones ternarias.

En cuanto al tercer grupo de objetos, los atributos, la herramienta brinda la posibilidad de definir si son simples o compuestos; además, se debe especificar si son monovalentes o polivalentes, y las cardinalidades mínima y máxima. Asimismo, los atributos pueden ser o formar parte del identificador de las entidades, para lo cual se permite definir identificadores simples o compuestos, así como internos, externos, o mixtos.

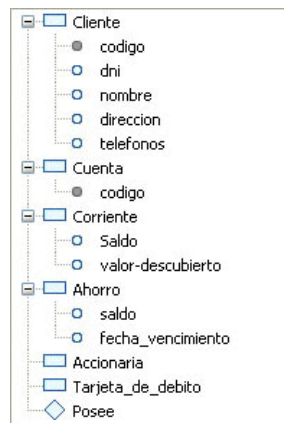


Fig. 5. árbol de composición de objetos.

En el editor gráfico, los objetos se presentan según las pautas de representación utilizadas por la cátedra. Allí, cada objeto puede ser seleccionado y se pueden ejecutar las mismas operaciones que permite el menú del árbol.

El usuario debe finalizar la operación de marcado de objetos en la especificación del problema en forma explícita. Al realizar dicha operación, se habilitan las propiedades de edición del modo gráfico, que hasta el momento estaban deshabilitadas. La única habilitación, hasta ese momento en el editor gráfico, corresponde a la posibilidad de mover entidades y relaciones, con el propósito de disponerlas en forma ordenada dentro del área que ocupan (área B).

Es importante destacar que la transformación de las marcas realizadas en la especificación (área A) en una representación gráfica preliminar del modelo, es unidireccional.

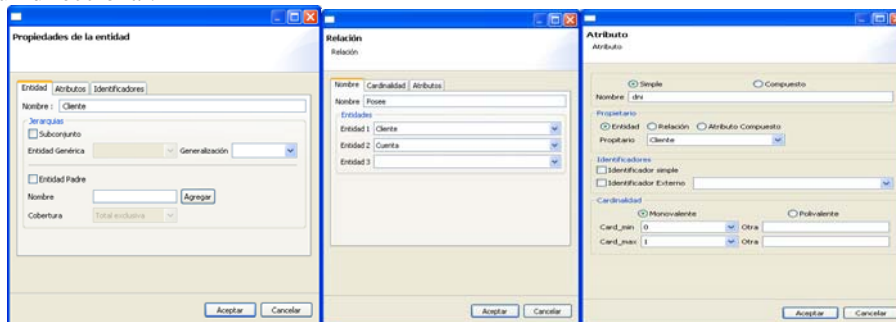


Fig. 6. Interfaz de alta de objetos.

Una vez finalizada la edición y realizadas las marcas correspondientes, no se puede continuar trabajando con el área de edición del documento de especificación de requerimientos. Solo puede continuar el trabajo desde la sección B y C de la figura 3.

Esta restricción da robustez y hace más consistente a la herramienta, con lo cual se evita redundancia. De lo contrario, podría ocurrir que el usuario marque en el editor de texto nuevas entidades, atributos o relaciones que ya agregó en el modo gráfico, generando de esa manera incompatibilidad.

No obstante, siempre existe la posibilidad de generar nuevamente un diagrama que se corresponda con los objetos marcados en la especificación. En este último caso, la representación gráfica será solamente a partir de los objetos marcados, perdiéndose cualquier agregado realizado sobre el editor gráfico con posterioridad. Las modificaciones que se realicen en el editor gráfico se verán reflejadas inmediatamente en el árbol y viceversa.

En la figura 7 se muestra la representación gráfica de un modelo conceptual definitivo, producto final obtenido utilizando de la herramienta. Para finalizar, se resumen los pasos a seguir utilizando CASER:

- Editar y/o importar una especificación de un problema (área A).
- Marcar las entidades, relaciones y atributos potenciales, expresados en la especificación (área A).
- Paralelamente a la realización de marcas, se genera en forma automática un modelo conceptual preliminar (área B)
- Refinar el modelo generado, ya que no necesariamente todos los conceptos del problema están expresados en la especificación: Esta tarea se lleva a cabo directamente en el editor gráfico (área B), en la representación jerarquía de objetos (área C), o en ambas.

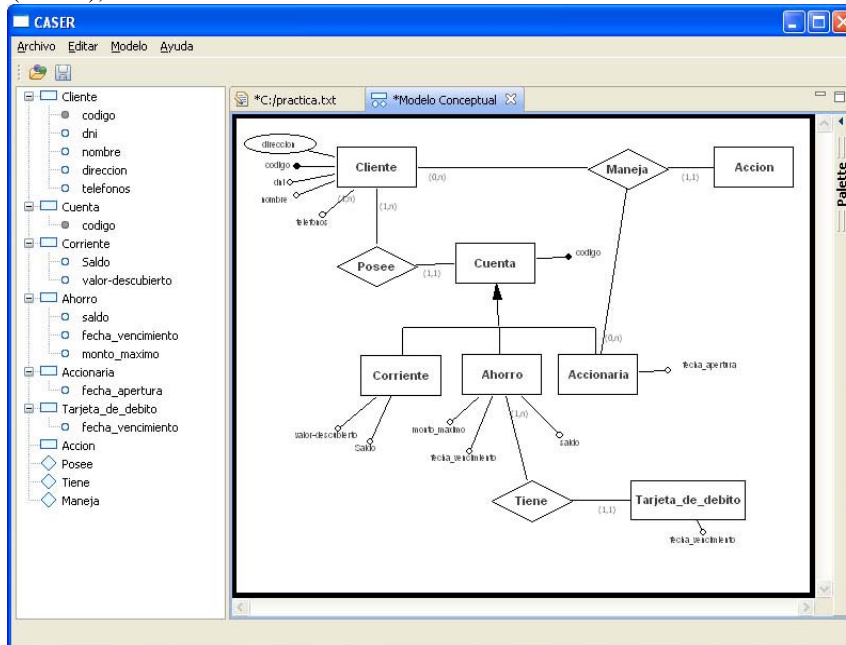


Fig. 7. modelo conceptual definitivo alcanzado.



## **5. Características de Implementación**

La herramienta fue desarrollada con RCP (Rich Client Platform). Esta plataforma permite la creación de aplicaciones Java multiplataforma según [2] y [8]. Sigue el paradigma de programación orientada a objetos y MVC (Model View Controller). De dicha plataforma se utilizó GEF (Graph Editing Framework), librería creada con el objeto de facilitar la labor de construcción de aplicaciones para dibujar diagramas.

Para la implementación de las partes restantes se utilizaron otras herramientas que componen RCP, como SWT (Standard Widget Toolkit) y JFace. En [7] se explica en detalle la funcionalidad y modo de implementación de cada una de estas últimas.

## **6. Resultados Obtenidos**

Durante 2008, la herramienta fue utilizada como prueba piloto en el curso de IBD que se dictó en el Centro Regional Las Flores. El grupo estaba compuesto por 6 alumnos. La herramienta fue presentada por los docentes de la cátedra y, en general, la utilización y refinamiento de conocimiento fue realizado en forma intuitiva por los alumnos. Esto permitió asegurar la compatibilidad entre CASER y la forma de presentar el tema de modelado en las teorías de la asignatura. Solamente fueron reportados bugs menores.

Los resultados fueron positivos. En primer lugar, los alumnos dispusieron de un entorno que les permitió trabajar con el modelado de datos sin la utilización del papel y lápiz, permitiendo movilizar los objetos definidos con facilidad y en forma legible.

En segundo lugar, las consultas realizadas a la cátedra fueron más efectivas. A partir del envío del archivo generado por CASER, la corrección y puesta a punto de los modelos conceptuales de datos se realizó de manera rápida y eficiente.

Por último, la comprensión que los alumnos lograron del tema se reflejó en una aprobación del 100% en las evaluaciones de Modelado de Datos.

Por otra parte, durante el primer semestre del 2009, la herramienta fue utilizada por un grupo de 30 alumnos correspondientes a la asignatura Bases de Datos 1 en la Universidad Nacional del Noroeste (UNNOBA). Si bien el uso de la herramienta no fue obligatorio, los resultados fueron nuevamente positivos.

El objetivo para el presente ciclo lectivo, es su utilización en el segundo semestre por parte de los alumnos de la asignatura de IBD de la Facultad de Informática de la UNLP. Se pretende que este CASE permita salvar los inconvenientes que típicamente surgen en la etapa de modelado, donde hasta ahora el alumno básicamente debía utilizar papel y lápiz para generar esquemas conceptuales. En este último caso, la experiencia se realizará con un grupo de 500 alumnos durante los meses de Agosto y Septiembre del año 2009.

## 7. Conclusiones

Abstraer conceptos de la realidad y representarlos en un modelo gráfico puede parecer sencillo, pero definitivamente no lo es. El desarrollo de un modelo de datos implica utilizar este mecanismo de la forma más “objetiva” posible. Para ello, la asistencia, en la construcción de modelos conceptuales, es indispensable para mejorar esta tarea.

La herramienta presentada cumple el rol de asistir al usuario en etapas tempranas de diseño de una Base de Datos. Con ella, se pueden vincular directamente conceptos inmersos en una especificación, con potenciales entidades, relaciones y/o atributos de un modelo conceptual. Esto garantiza un avance significativo en la ayuda para el proceso de aprendizaje de un alumno.

Por último, el mercado no ofrece un Software con estas características, dado que todos los asistentes disponibles comienzan desde el diseño lógico, obviando el diseño conceptual, tan necesario para quienes intentan construir una Base de Datos sin experiencia previa, y tan aconsejable para quienes ya tienen experiencia.

## 8. Trabajos futuros

A partir de esta versión inicial de CASER, la línea de trabajo actual y futura es ambiciosa. En primer lugar, la utilización de esta herramienta por parte de 500 alumnos en el año 2009, ofrecerá una retroalimentación concreta en cuanto a mejoras de interfase y usabilidad.

Luego, se prevé incorporar el pasaje del MCDef a un Modelo Lógico, y posteriormente al Modelo Físico, considerando en este último caso el DBMS utilizado.

Por último, el propósito es la integración de la herramienta con el modelado de objetos, para generar este tipo de modelos por medio de un asistente a partir del modelo conceptual.

## Bibliografía

- [1] Batini, Navathe, Cieri. Diseño Conceptual de Bases de Datos: un enfoque de entidades-interrelaciones. Addison Wesley 1991.
- [2] Berthold Daum. Professional Eclipse 3 for Java™ Developers. John Wiley & Sons Ltd. 2004.
- [3] Date. Introducción a los sistemas de Bases de Datos. Addison Wesley. 1994.
- [4] Elmasri, Navathe. Fundamentos de Sistemas de Bases de Datos. Pearson-Addison Wesley. 2002.
- [5] Korth-Silberchatz. Fundamentos de Bases de Datos. McGraw Hill. 1998.
- [6] Kroenke. Procesamiento de Bases de Datos. Prentice Hall. 1996.
- [7] Matthew Scarpino, Stephen Holder, Stanford Ng, Laurent Mihalkovic. SWT/JFace in action. Manning. 2005.
- [8] Miguel Angel Abián. El archipiélago Eclipse (parte 3 de 4). 2005.
- [9] Miguel Piattini. Fundamentos y modelos de Bases de Datos. Rama. 1998.