

CasER 2.0: Herramienta para la enseñanza de Modelado de Bases de Datos

Alejandra Durán, María F. Rius, Rodolfo Bertone, Pablo Thomas.

Instituto de Investigación en Informática LIDI. Facultad de Informática. UNLP

aleduran04@hotmail.com, riusflor@gmail.com, {pbertone,pthomas}@lidi.info.unlp.edu.ar

Resumen. Este trabajo presenta la segunda versión de la herramienta para Diseño de Bases de Datos 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 Universidad Nacional de La Plata (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. De esta forma, alumnos en la fase de aprendizaje de Diseño Conceptual, Lógico y Físico de Bases de Datos, carecen de un software para generar esquemas de datos en forma simple, asistida y automatizada. La herramienta aquí propuesta respeta el desarrollo gradual de un Esquema Conceptual, Lógico y Físico, para lo cual, incorpora a las funcionalidades básicas, un editor de texto para trabajar con la especificación de un problema, y un editor gráfico para visualizar los esquemas generados. La versión 2.0 de la herramienta está siendo probada por los alumnos de la asignatura Introducción a las Bases de Datos durante el curso 2011.

Keywords: Modelado Conceptual, Modelado Lógico, Modelado Físico, Modelo de Entidades y Relaciones. Diseño Conceptual. Herramienta Educativa.

1 Introducción

Los modelos se utilizan en todo tipo de ciencias. Su finalidad es simbolizar una parte del mundo real de forma que sea manipulable más fácilmente. En definitiva es un esquema mental (conceptual) en el que se intenta reproducir las características de una realidad específica. En el caso de los modelos de datos, lo que intentan reproducir es información que se desea almacenar en una Base de Datos (BD).

El diseño de una 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. Una alternativa de diseño de BD consiste en descomponer esta actividad en tres etapas: diseño conceptual, 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 vez de las estructuras de almacenamiento que se necesitarán para administrar esta información.

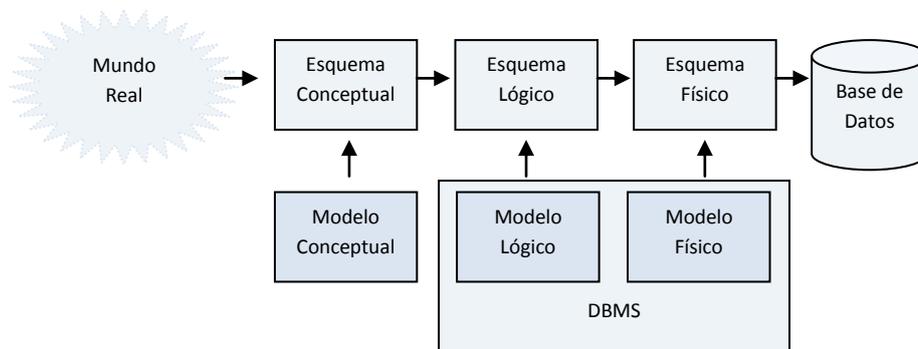


Fig. 1. Modelado de datos.

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

Un modelo lógico tiene como objetivo convertir el esquema conceptual en un modelo (representación) que pueda ser interpretado por el DBMS. El producto del diseño lógico es un esquema equivalente pero más eficiente para su uso.

Al comenzar con el modelado lógico se necesita definir qué tipo de DBMS se utilizará luego para su implantación física. La cadena de pasos de conversión está relacionada con el tipo de DBMS (relacional, orientado a objetos, jerárquico o de red) a utilizar.

La etapa del diseño lógico es independiente de los detalles de implantación y depende del tipo de DBMS que se vaya a utilizar. El resultado obtenido es el esquema lógico.

El diseño físico parte del esquema lógico global obtenido y produce una descripción de la implantación de la BD. Esta descripción es completamente dependiente del DBMS específico que se vaya a utilizar.

El diseño conceptual, lógico y físico tiene diferentes enfoques. El utilizado por la asignatura de Introducción a las Bases de Datos (IBBDD) de la Facultad de

Informática de la UNLP, es el propuesto en [1] y [2]. Este modelo no se condice directamente con ninguna herramienta existente en el mercado (libre o propietaria) que facilite y asista esta tarea de modelar los tres niveles: conceptual, lógico y físico.

La herramienta propuesta en este trabajo permite la generación asistida de los esquemas conceptual, lógico y físico. Para esto, es necesario disponer de una especificación detallada del problema, y en forma semi-automática, se genera el esquema conceptual, luego el esquema lógico y finalmente el esquema físico. Por razones prácticas se hace referencia a modelo o esquema (conceptual, lógico y físico), indistintamente.

La figura 2 resume el comportamiento de CasER. El comienzo de la actividad se inicia desde una especificación de requerimientos a partir de la cual se genera un Modelo Conceptual de Alto Nivel (MCAN). Luego, como parte de un proceso de depuración y refinado a cargo del usuario, se obtiene el Modelo Conceptual Definitivo (MCDef). A partir de este modelo definitivo se realiza el proceso de pasaje a Modelo Lógico de Alto Nivel (MLAN). Finalmente la última parte del modelado consiste en realizar el pasaje a Modelo Físico de Alto Nivel (MFAN).

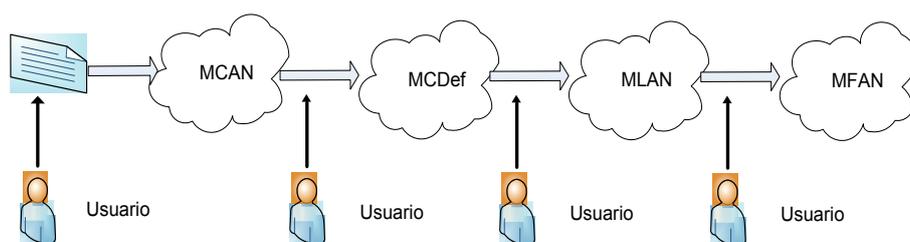


Fig. 2. Generación asistida de modelos de datos.

A continuación se resumen los conceptos más importantes de Modelado de Datos; posteriormente se presenta la versión 2.0 de la herramienta CasER. Finalmente se detallan los resultados obtenidos, se presentan las conclusiones y trabajos futuros.

2 Modelo de datos ER

El modelo de datos Entidad Relación (ER) está basado en una percepción del mundo real que consta de una colección de elementos básicos, llamados entidades, interrelaciones (entre entidades) y atributos.

El modelo ER se considera un modelo conceptual que permite representar con un nivel alto de abstracción, la información utilizada para la resolución de problema.

Los términos “entidad” e “interrelación” denotan clases de objetos. Las entidades representan clases de objetos del mundo real y se representan gráficamente con rectángulos. Las interrelaciones representan agregaciones de dos o más entidades y son graficadas con rombos.

Existen interrelaciones que conectan una entidad consigo misma y son conocidas como interrelaciones recursivas, es necesario en estos casos distinguir dos roles de la entidad en la interrelación.

Cada una de las interrelaciones de un esquema tiene un significado determinado; 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. En [1] se expresa las cardinalidades como pares de valores indicando el valor mínimo y máximo.

Otro elemento necesario para la construcción del esquema lo representan los atributos. Estos definen las propiedades básicas de las entidades o interrelaciones. También 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 y 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.

Diversos autores [1][2][3][4][5][6][9] plantean enfoques distintos respecto de los elementos que enriquecen conceptualmente el esquema. La cátedra Introducción a las Bases de Datos se basa en lo expuesto en [2] y 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 o especializaciones 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.

Por último, un identificador de una entidad E es un grupo de atributos o de entidades relacionadas con E, con 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.

El objetivo del diseño lógico es convertir el esquema conceptual en un esquema lógico que se ajuste al modelo de DBMS y obtener una representación eficiente en el uso de recursos. En el proceso se convertirá el modelo conceptual a un modelo ER mejorado. El objetivo será llegar al modelo lógico usado en los DBMS.

Según [2] en el proceso se van a tomar decisiones de conversión al modelo lógico sobre:

- atributos derivados, donde la información se puede calcular. Se mantendrán los más usados y se eliminarán los que se recalculan con frecuencia.

- ciclos de relaciones, se deberían quitar relaciones que generen información redundante, para que el modelo sea mínimo.
- atributos polivalentes, se eliminarán estos atributos generando una nueva entidad y relación, que vincule la entidad o relación que contenía el atributo polivalente, la cardinalidad será muchos a muchos.
- atributos compuestos, en este caso las opciones son tres: generar un único atributo igual a la concatenación de todos los atributos simples; definir en la entidad cada uno de los atributos simples que pertenecen al compuesto; o bien generar una nueva entidad que represente al atributo compuesto.
- jerarquías, existen tres formas de eliminarlas del modelo: dejar solo la generalización la cual incorpora los atributos de las especializaciones (estos son opcionales, no obligatorios); dejar las especializaciones que incluyen los atributos de la generalización; o dejar todas las entidades que son parte de la jerarquía agregar relaciones “es un” entre la generalización y las especializaciones.

Como resultado se obtiene: un esquema lógico ER, el cual será convertido en un esquema relacional. Los pasos necesarios que permiten generar un modelo físico de datos (sobre el modelo relacional) a partir del esquema lógico generado son:

- eliminación de identificadores externos, es decir, el modelado de identificadores externos como internos de las entidades,
- selección de claves, definido de una manera muy similar al concepto de identificador en el modelo ER ; una clave es un atributo o un conjunto de atributos que identifica de manera única una entidad del conjunto de entidades. Cuando se genera el esquema físico sobre el modelo relacional, se debe resolver la clave primaria CP a partir de los identificadores de cada entidad.
- transformación de entidades, en general, cada una de las entidades definidas se convierte en una tabla del modelo físico.
- transformación de relaciones, cada una de ellas se convertirá a tabla o no, dependiendo de la cardinalidad.

La herramienta CasER versión 1.0 [17] consistía en una herramienta para modelado conceptual de BD. La versión 2.0 de la herramienta extendió sus funcionalidades para modelado lógicos y físicos de alto nivel. Su característica principal es la de asistir en la creación de un esquema conceptual de alto nivel, y su posterior derivación al esquema lógico y físico. Existen diferentes productos de software que permiten crear modelos de datos, entre otros, Erwin [14], POWERDESIGNER [15], Workbench [16]; sin embargo, ninguno de ellos permite la creación de un modelo conceptual, lógico y físico con los conceptos referidos en [1] y [2].

CasER permite vincular partes de la especificación de requerimientos con elementos del esquema conceptual. De esta forma se obtiene trazabilidad entre la especificación del requerimientos sobre la solución del problema, y el modelo de datos que lo soporta.

Es importante destacar que, aunque existen otras notaciones para representar modelos de datos, la herramienta CasER está basada en la que utiliza la asignatura de

IBBDD [1] [2]. El marco teórico propuesto en [3], [4], [5], [6] y [9] es similar, sólo difieren en la representación gráfica de algunos elementos.

3 CasER 2.0. Herramienta para Modelado de datos.

En la figura 3 se presentan las partes que componen las áreas con las que el usuario/alumno puede interactuar. Dichas partes se corresponden con la especificación (A), el diagrama de modelo conceptual (B), el árbol que representa la jerarquía (C) de los objetos creados en (B) y marcados en (A), el diagrama de modelo lógico (D) y las tablas del modelo físico (E).

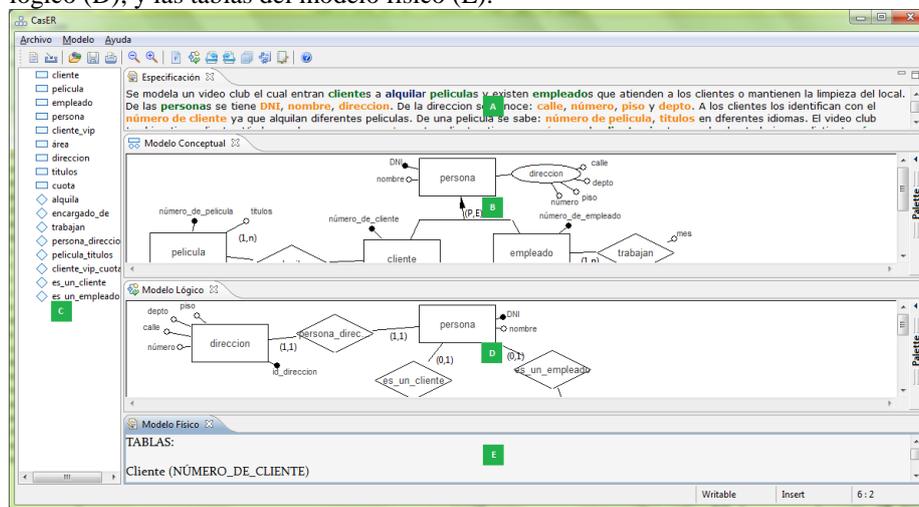


Fig. 3. CasER. Pantalla inicio.

El área A corresponde a un editor de texto simple que contiene 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. De esta manera, las marcas identifican dentro del texto potenciales entidades, relaciones y atributos, que constituyen los tres elementos básicos del modelado conceptual. Se realizan las marcas para ver los conceptos resaltados con determinados colores, según se represente una entidad, una relación o un atributo. Paralelamente a la selección y marcado de palabras y/o frases, se genera automáticamente y en forma incremental el modelo conceptual preliminar. Estos elementos generados se reflejan en el área B como representación gráfica del modelo, y en el área C como una jerarquía de objetos. Todos los objetos que se agregan desde la especificación, se mantienen sincronizados con los de las restantes áreas.

En el área D refleja la estructura de los objetos luego de aplicar las reglas de transformación a modelo lógico del modelo conceptual (de área B). Por último, en el área E presenta las tablas que resultantes luego de aplicar las reglas de transformación del modelo lógico en físico.

3.1 Componentes de la Herramienta.

La figura 4 presenta un modelo conceptual. En [17] se presentaron en detalle las características que permiten la generación del modelo conceptual. CasER 2.0 comienza desde la versión anterior incorporando las funcionalidades necesarias para obtener el modelo lógico y, posteriormente, el modelo físico.

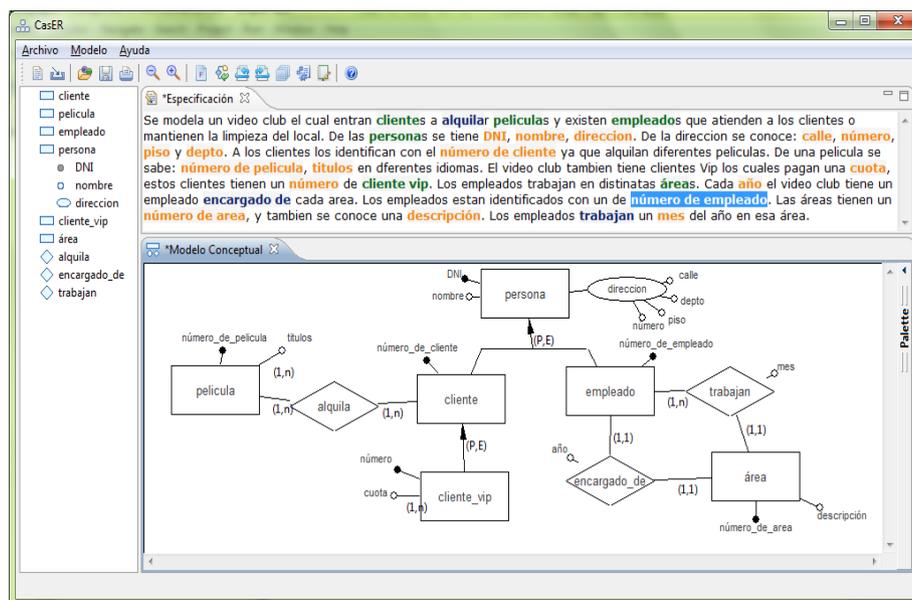


Fig. 4. Modelado Conceptual.

3.2 Generación del Modelo Lógico

La generación del modelo lógico consiste de una serie de pasos estructurales. En primer lugar se deben eliminar los atributos compuestos. El usuario podrá optar por diferentes alternativas. Estas opciones están en concordancia con los conceptos teóricos vertidos en [2].

En el segundo paso del modelado lógico se deben tomar decisiones respecto de atributos polivalentes. El modelo relacional no soporta atributos con cardinalidad mayor que 1 de una forma fácil, eficiente y efectiva. Por este motivo, en este segundo paso es menester tomar decisiones respecto de este tipo de atributos. La forma de conversión de atributos polivalentes, nuevamente, va en concordancia con los conceptos teóricos de la asignatura. El último paso del modelado lógico trata las jerarquías existentes en el modelo conceptual. El modelo relacional no tiene forma de representar el concepto de herencia. Por tal motivo, es necesario reemplazar dicho concepto con alguna alternativa viable. Se presentan entonces aquí tres opciones de trabajo. Dejar solo al padre, borrando a los hijos; dejar solo los hijos; o mantener toda

la estructura generando una relación ES-UN entre padre e hijos. El usuario deberá optar en este punto por la alternativa que considere más adecuada de acuerdo al problema. Es importante destacar que la decisión tomada conlleva una serie de aspectos estructurales que deberán ser considerados por el usuario. Esto significa que la herramienta no guía en detalle al alumno en este aspecto. La finalidad es permitir que el alumno/usuario tome sus propias decisiones sin ser guiado firmemente por la herramienta. La figura 5 muestra parcialmente la transformación del modelo conceptual en modelo lógico.

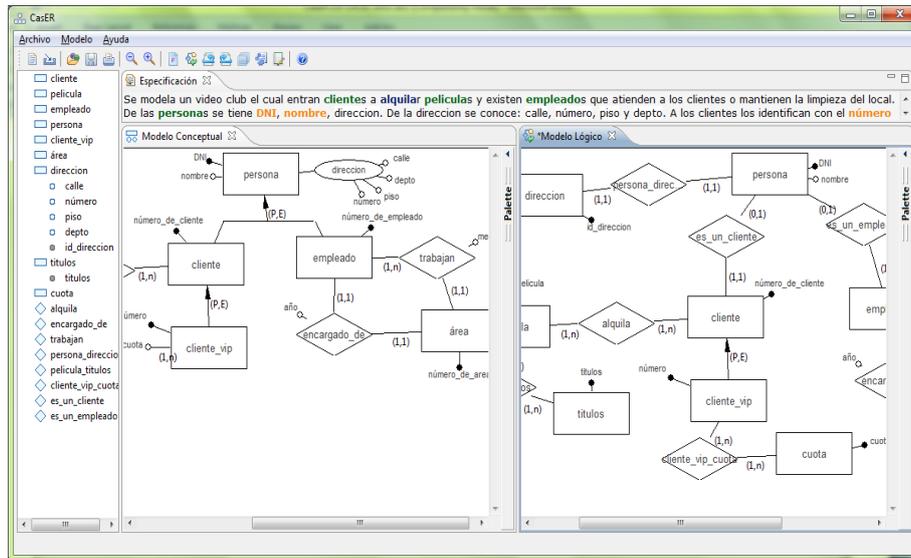


Fig. 5. Modelado Lógico

3.3 Generación del Modelo Físico

Por último, la generación del modelo físico es semiautomática, requiriéndose la participación del usuario, en caso que sea necesario. Se generan tablas con claves primarias indicadas en mayúsculas, tal como se presenta en la figura 6.

4 Conclusiones

CasER es utilizado desde el año 2008 en el curso regular de IBBDD con resultados exitosos en cuanto a usabilidad, legibilidad y asistencia en el aprendizaje de Diseño Conceptual de Bases de Datos.

La versión extendida presentada en este trabajo permite completar el proceso de Diseño de Bases de Datos. Por ende, a partir de una especificación de requerimientos, el alumno puede generar en forma asistida los tres esquemas (Conceptual, Lógico y

Físico). De esta forma CasER permite potenciar el proceso de aprendizaje de los alumnos.

Finalmente, es importante destacar que en el mercado no se encuentra un producto de Software (libre o comercial) con las características de CasER, dado que todos los asistentes disponibles comienzan desde la etapa de 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.

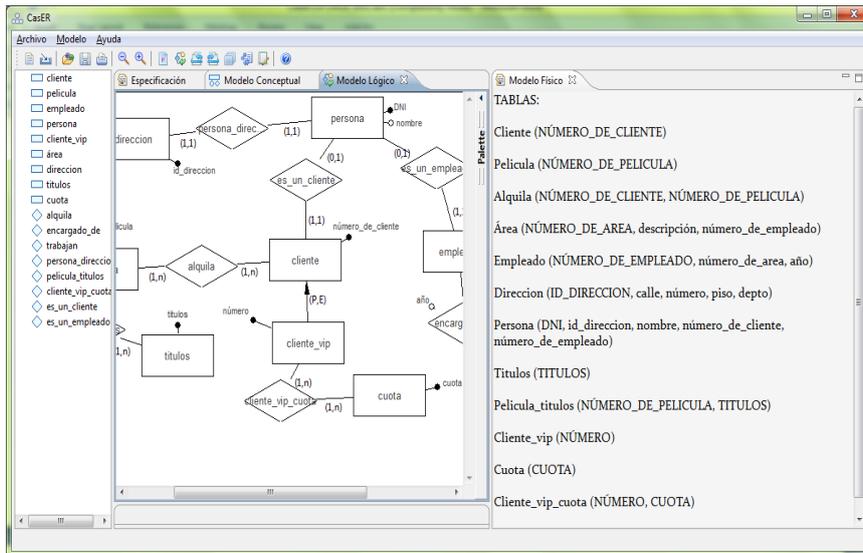


Fig. 6. Modelo Físico.

5 Trabajos futuros

Con CasER 2.0 se asiste la tarea de diseño de una BD. No obstante, se prevé poder generar tablas directamente sobre algún DBMS particular.

Bibliografía

- [1] Batini, Navathe, Cieri. Diseño Conceptual de Bases de Datos: un enfoque de entidades-interrelaciones. Addison Wesley 1991.
- [2] R Bertone, P Thomas, Introducción a las Bases de Datos, Fundamentos y Diseño. Editorial Pearson, ISBN 978-987615136-8. Mayo 2011
- [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.
- [10] Leite, J.C.S.P., Application Languages: A Product of Requirements Analysis, Departamento de Informática, PUC- /RJ, January 1989.
- [11] Jacobson I. et al. Object Oriented Software Engineering: AUse-Case Driven Approach. Addison Wesley, 1992
- [12] G. Booch, J. Rumbaugh, y I. Jacobson. The Unified Modeling Language User Guide. Addison–Wesley, 1999.
- [13] Eclipse - <http://www.eclipse.org>.
- [14] Erwin - <http://www.ca.com/us/database-design.aspx>
- [15] POWERDESIGNER-
<http://www.sybase.com/products/modelingdevelopment/powerdesigner>.
- [16] Workbench - <http://www.sql-workbench.net>
- [17] R. Bertone, P. Thomas, S. Antonetti, A. Miglio, “Herramienta para la enseñanza de Modelado Conceptual de Bases de Datos”, Congreso Argentina de Ciencias de la Computación. Octubre 2009, Jujuy