

# CARACTERIZACIÓN DE LA DEPENDENCIA DE EXISTENCIA EN INTERRELACIONES BINARIAS

Carlos García<sup>1</sup>, Abel Rodríguez<sup>2</sup>, Norma Cabrera<sup>3</sup>, Luisa González<sup>4</sup>

Departamento de Ciencia de la Computación, Universidad Central de Las Villas, Cuba

## RESUMEN

En la infraestructura de todo sistema de información, la exactitud semántica de los datos es vital para garantizar la calidad de las aplicaciones que los utilizan. La especificación de cuándo y cuáles datos son semánticamente correctos, constituye una de las tareas más importantes en la modelación de datos, donde la lógica del negocio y los requisitos que definen los usuarios son traducidas en restricciones de integridad semánticas. Los modelos de datos proporcionan diferentes medios para representar las restricciones de integridad, en la modelación conceptual la dependencia de existencia es un recurso útil para asegurar la integridad semántica entre las propiedades estáticas y dinámicas de los datos. En general, las restricciones de integridad son captadas a nivel conceptual y deben ser transformadas adecuadamente para su soporte en el nivel lógico, particularmente la dependencia de existencia queda modelada en los esquemas relacionales a través de las llaves foráneas y su comportamiento. En este trabajo se hace un análisis de la dependencia de existencia en asociaciones y se proponen reglas para su transformación al modelo relacional, las cuales fueron implementadas en una herramienta de diseño de bases de datos.

**Palabras clave:** dependencia de existencia, diseño de bases de datos, entidades débiles, interrelaciones débiles, modelación de datos.

## ABSTRACT

In the infrastructure of an information system, the semantic accuracy of the data is vital to guarantee the quality of the applications that use the data. The specification of when and which data they are semantically correct, one of the most important tasks constitutes in the data modeling, where the business rules and the requirements defined by the users are translated in semantic integrity constraints. The models of data provide different means to represent the restrictions of integrity, in the conceptual modeling; the existence dependence is a useful resource to assure the semantic integrity among the static and dynamic properties of the data. In general, the integrity constraints are captured at conceptual level and they should be transformed appropriately for their support in the logical level, particularly the existence dependence is modeled in the relational schema using the foreign keys approach. In this work, an analysis of the existence dependence in binary relationships is done and the corresponding transformation rules are proposed. The transformation rules were implemented in a database design tool.

**Key words:** data modeling, database design, existence dependence, weak entities, weak relationships.

## 1. INTRODUCCIÓN

La exactitud semántica de los datos, básicamente se refiere al grado en que éstos reflejan correctamente la información del universo de discurso. La especificación de cuándo y cuáles datos son semánticamente correctos, constituye una de las tareas más importantes en el proceso de modelación y diseño de una base de datos [1, 2]. En este proceso, la lógica del negocio y los requisitos que definen los usuarios son traducidas en restricciones de integridad semánticas o simplemente restricciones de integridad.

Una restricción de integridad es la representación de una propiedad de los datos en las aplicaciones e imponen exigencias que deben ser cumplidas para garantizar la consistencia de los datos, los cuales, no sólo deben ser almacenados y recuperados sino que los sistemas deben garantizar su integridad semántica. La importancia de especificar y definir las restricciones de integridad fue reconocida desde los años 1970 por varios autores [3-7], a partir de entonces han sido estudiadas desde diferentes puntos de vistas por [8-11] y al tema se le han dedicado capítulos en libros de texto de base de datos [12-14].

Email: <sup>1</sup>cgarcia@uclv.edu.cu

<sup>2</sup>arm@uclv.edu.cu

<sup>3</sup>normacg@uclv.edu.cu

<sup>4</sup>luisagon@uclv.edu.cu

Otros libros de texto como [1, 15, 16] se concentran principalmente en los fundamentos teóricos de las restricciones de integridad. Los modelos de datos proporcionan diferentes medios para representar las restricciones e incluso, éstas pueden ser inherentes al modelo [17].

El modelo de datos semántico más utilizado es el Entidad Relación Extendido (ERE), el cual tiene sus bases en el modelo Entidad Relación (ER) propuesto por Chen [18], y es el resultado de muchas investigaciones que han incorporado nuevos conceptos, con el objetivo de enriquecer la representación de los esquemas conceptuales. Los conceptos básicos de este modelo son las entidades y las interrelaciones entre entidades; a partir de este núcleo básico se han desarrollado numerosas extensiones, no siempre tratadas de manera consistente en la literatura sobre el tema, como es el concepto de dependencia de existencia, usado frecuentemente de manera informal. Es muy frecuente encontrar que en un universo de discurso, la presencia de ciertas entidades dependa de la existencia de otras en un esquema, lo que se puede comprender intuitivamente, sin embargo este hecho y sus diferentes manifestaciones tienen implicaciones en los posibles estados válidos de una base de datos. El concepto de dependencia de existencia [19-22] se asocia intuitivamente a un comportamiento; se considera básicamente que existe dependencia de existencia entre dos entidades, cuando la eliminación de una, nombrada entidad dominante, conlleva también la eliminación de la otra, nombrada entidad subordinada. Explícitamente conceptos como entidades débiles, interrelaciones de identificación y jerarquías [23-25] expresan dependencia de existencia entre entidades, en otros conceptos aparece implícitamente y el comportamiento a que se hace referencia no siempre es el mencionado y hay que profundizar en cada construcción o concepto.

Aparentemente la interrelación de asociaciones no es un concepto que refleja de forma evidente dependencia de existencia, sin embargo sus restricciones estructurales, dadas por las propiedades de cardinalidad y las restricciones de participación, conocidas también como cardinalidades máxima y mínima, son de interés en este sentido. Las restricciones de participación especifican si la existencia de una entidad depende de que esté relacionada con otra. Muchos autores están de acuerdo en que la restricción de participación total en las asociaciones implica un cierto tipo de dependencia de existencia [14]. En ocasiones a simple vista no es tan fácil determinar cuando una asociación entre entidades expresa dependencia de existencia y descuidarlo puede acarrear inconsistencia semántica. Por estas razones se considera que el concepto de dependencia de existencia es un recurso útil para asegurar la integridad semántica entre las propiedades estáticas y dinámicas de los datos en la modelación conceptual.

El objetivo de este artículo es hacer un análisis de la dependencia de existencia en asociaciones, demostrar la conveniencia de uso desde el punto de vista semántico y proponer reglas para su transformación al modelo relacional, las que pueden ser implementadas en una herramienta de diseño de bases de datos.

## 2. CARACTERÍSTICAS ESTRUCTURALES DE LA DEPENDENCIA DE EXISTENCIA EN ASOCIACIONES

Existen varias definiciones de dependencia de existencia enunciadas por distintos autores, entre las que se pueden citar:

**Definición 1.** Sea el predicado  $\epsilon$  que denota dependencia de existencia. Se tiene que una instancia  $x$  es dependiente de existencia de otra instancia  $y$  (representada como  $ed(x,y)$ ), y debe existir siempre que  $x$  exista, o formalmente:  $ed(x,y) =_{\text{def}} \neg(\epsilon(x) \rightarrow \epsilon(y))$  [19].

**Definición 1.** Dados  $E1$  y  $E2$  conjuntos de entidades. Se dice que  $E1$  tiene dependencia de existencia de  $E2$ , se denota  $(E1 \leftarrow E2)$ , si y solo si una entidad  $e1$  de  $E1$  está asociada con solo una y siempre la misma entidad  $e2$  de  $E2$ . Se llamará a  $e1$  entidad dependiente o subordinada y  $e2$  entidad dominante, donde  $E1$  es el conjunto de entidades dependientes y  $E2$  es el conjunto de entidades dominantes [21].

**Definición 3.** Si cada entidad del conjunto de entidades  $E1$  siempre está asociada como mínimo con una, a lo sumo con una y siempre la misma entidad del conjunto de entidades  $E2$ , entonces  $E1$  tiene dependencia de existencia de  $E2$  [22].

Como consecuencia, si  $E_1$  tiene dependencia de existencia de  $E_2$  ( $E_1 \leftarrow E_2$ ), no puede insertarse una entidad  $e_1$  en el conjunto de entidades  $E_1$  a menos que se asocie con una entidad  $e_2$  del conjunto de entidades  $E_2$ . Análogamente al eliminar una entidad dominante de  $E_2$ , deben eliminarse todas las entidades de  $E_1$  asociadas a ella.

La dependencia de existencia es similar al concepto de dependencia de asociación descrita en [26]. La principal diferencia es cuando una entidad  $e_1$  depende de la existencia de dos entidades  $e_2$  y  $e_3$ , estas entidades no tienen que pertenecer necesariamente al mismo conjunto. Por ejemplo, el control de los préstamos en la biblioteca (Figura 1), donde la existencia de una entidad préstamo, donde se controla la información de las planillas de préstamos en la biblioteca, depende de la existencia de un ejemplar disponible y el miembro que lo solicita.



**Figura 1.** Diagrama ER para el control de los préstamos en una biblioteca.

## 2.1 Representación para la dependencia de existencia en asociaciones

La representación de la dependencia de existencia a través de un grafo es sintácticamente correcta. Un Grafo de Dependencia de Existencia (GDE), se define siguiendo la idea expresada en [21] como:

Sea  $G(V, A)$  un grafo dirigido donde  $V$  es el conjunto de los vértices que representan los conjuntos de entidades en el esquema conceptual;  $A$  las asociaciones que definen dependencia de existencia y  $G$  debe satisfacer las siguientes restricciones:

- No hay presencia de bucles, lo que significa que una entidad no tiene dependencia de existencia de sí misma.
- No hay presencia de ciclos.

La primera de las restricciones se justifica por el hecho de: si un conjunto de entidades  $E_1$  tiene dependencia de existencia de sí mismo, implicaría que una entidad  $e_1$  del conjunto  $E_1$  depende de la existencia de otra entidad  $e_2$  del mismo conjunto; en otras palabras no puede insertarse o crearse  $e_1$  si no se asocia o vincula a  $e_2$ , quien a su vez depende de la existencia de otra entidad del mismo conjunto. Obviamente ante tal situación  $E_1$  será un conjunto de entidades vacío. La segunda restricción, responde a un hecho similar, pero en este caso cuando hay más de un conjunto de entidad involucrado.

Retomando el ejemplo anterior, para los conjuntos de entidades que guardan la información sobre los miembros y préstamos de la biblioteca, la representación del grafo de dependencia de existencia que se muestra en la Figura 2 sería:  $G(\{MIEMBRO, PRESTAMO\}, \{(MIEMBRO, PRESTAMO)\})$ .



**Figura 2.** Grafo de dependencia de existencia entre los conjuntos de entidades MIEMBRO y PRESTAMO.

En el grafo de dependencia de existencia también se puede expresar la cardinalidad de la asociación, lo cual es definido en [21] como la cantidad de entidades subordinadas asociadas a una entidad dominante en un momento dado.

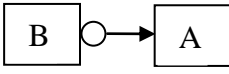
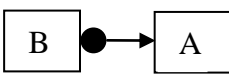
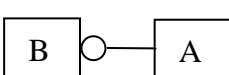
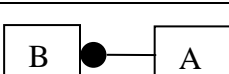
Notación para la cardinalidad de la asociación con dependencia de existencia:

- $E_1(1) \leftarrow E_2$ , si  $E_1 \leftarrow E_2$  y una entidad  $e_2$  de  $E_2$  tiene asociado a ella a lo sumo una entidad  $e_1$  de  $E_1$  en un momento dado
- $E_1(N) \leftarrow E_2$ , si  $E_1 \leftarrow E_2$  y una entidad  $e_2$  de  $E_2$  puede tener asociado a ella varias entidades  $e_1$  de  $E_1$  en un momento dado

Debe notarse que las asociaciones con dependencia de existencia son siempre obligatorias para la entidad subordinada, lo cual implica que para ella la cardinalidad siempre es como mínimo uno y a lo

sumo uno. Por otro lado, la participación de la entidad dominante puede ser opcional o estar involucrada con más de una entidad subordinada.

En [22] se propone una representación gráfica para las asociaciones que expresan dependencia de existencia con el objetivo de evitar confusiones entre estas y la representación clásica de las asociaciones. En la Figura 3 el círculo blanco indica la participación opcional en la asociación, una entidad dominante puede o no tener entidades subordinadas asociadas a ella. Mientras que el círculo negro nos indica que la participación es obligatoria, cada entidad dominante esta asociada como mínimo con una entidad subordinada en un momento determinado. La flecha indica cardinalidad máxima mucho, o sea que una entidad dominante tiene asociada a ella en un momento dado varias subordinadas y la línea por su lado denota cardinalidad máxima uno.

	Las entidades dominantes del conjunto de entidades B se asocian con cero, una o varias entidades subordinadas del conjunto de entidades A en un momento dado
	Las entidades dominantes del conjunto de entidades B se asocian con una o varias entidades subordinadas del conjunto de entidades A en un momento dado
	Las entidades dominantes del conjunto de entidades B se asocian con cero o una entidad subordinada del conjunto de entidades A en un momento dado
	Las entidades dominantes del conjunto de entidades B se asocian con una entidad subordinada del conjunto de entidades A en un momento dado
En cada caso las entidades de A están asociadas solamente con una y siempre la misma entidad dominante de B.	

**Figura 3.** Notación gráfica para la dependencia de existencia.

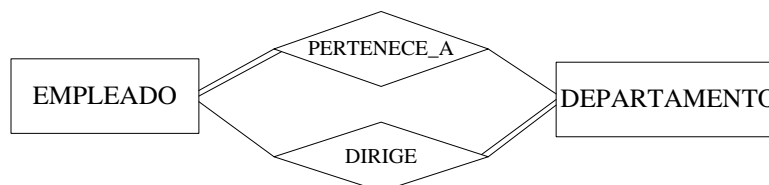
Para el ejemplo de la biblioteca, la representación según esta notación, quedaría como se muestra en la Figura 4. En un momento dado, cada miembro tiene cero, uno o varios préstamos y a su vez cada préstamo se asocia con solo uno y siempre el mismo miembro.



**Figura 4.** Asociación con dependencia de existencia entre los conjuntos de entidades MIEMBRO y PRESTAMO.

### 3. LA DEPENDENCIA DE EXISTENCIA EN ASOCIACIONES

Muchos autores están de acuerdo en que la restricción de participación total en las asociaciones implica un cierto tipo de dependencia de existencia [14], sin embargo para [22, 27, 28] una diferencia importante entre la restricción de participación total y dependencia de existencia radica en las reglas de actualización. Por ejemplo en una empresa (Figura 5), si la política establecida es que todo empleado debe pertenecer a un departamento, cada entidad de EMPLEADO sólo participa en una asociación PERTENECE\_A, donde la participación de EMPLEADO en PERTENECE\_A es total. Por otro lado, en la empresa no todos los empleados dirigen un departamento, así que la participación de EMPLEADO en DIRIGE es parcial.



**Figura 5.** Diagrama ER para controlar los empleados y departamentos en una empresa.

En el ejemplo anterior, según la definición, la asociación entre EMPLEADO y DEPARTAMENTO no expresa dependencia de existencia. En un departamento pueden existir varios empleados y cada empleado está asignado a lo sumo a un departamento. Sin embargo, la existencia de un empleado no depende de un departamento y a su vez la existencia de un departamento no depende de los empleados. En el modelo, sencillamente se refleja que el vínculo entre EMPLEADO y DEPARTAMENTO es mandatorio para EMPLEADO, ya que cada empleado siempre deberá estar vinculado a un departamento. Por otro lado, un empleado puede ser asignado a otro departamento, incluso regresar a un departamento al que perteneció en otro momento; en este caso debido a la unicidad de los identificadores, podría perderse la información inicial o sencillamente no ser claro en la semántica del problema. Generalmente se utiliza otro atributo, como la fecha, para modelar este comportamiento. Este requerimiento es importante en los sistemas de negocios, donde “qué ha sucedido” es tan importante como “qué está sucediendo”.

Considerando que la empresa tiene clientes a los que le brinda distintos servicios, entre ellos la realización de proyectos de software (Figura 6), donde cada proyecto es solicitado exactamente por un cliente, cuya solicitud es almacenada hasta que se le asigne un equipo de trabajo. Los empleados que se dedican a la realización de estos proyectos en un momento determinado solo pueden estar trabajando en uno.



**Figura 6.** Diagrama ER que modela los datos sobre los clientes, sus proyectos y empleados que trabajan en ellos.

La representación de ambas asociaciones es idéntica, aparentemente tiene el mismo comportamiento, sin embargo existe una sustancial diferencia en la semántica de las mismas. Cada empleado, en un momento dado puede estar trabajando en solo un proyecto, pero en el tiempo puede tener otras asignaciones, en otras palabras, el vínculo TRABAJA\_EN es modificable, mientras que la asociación SOLICITA no lo es. Los proyectos son pedidos por los clientes y pertenecen a ellos todo el tiempo. Por tanto el diagrama de la Figura 6 puede considerarse semánticamente incompleto, ya que hay un comportamiento de los datos que no queda reflejado.

### 3.1 Modelación de asociaciones con dependencia de existencia

Cuando se está en presencia de asociaciones con dependencia de existencia, se puede considerar que hay un comportamiento en el universo de discurso que deriva una modelación más precisa que solo la consideración de una llave foránea. Para que esta situación quede reflejada se propone modelarlo como un hecho independiente. En lugar de generar los esquemas:

Cliente (idCliente, otros\_atributos\_cliente)  
 Proyecto (idProyecto, otros\_atributos\_proyecto, idCliente)

Donde idCliente en el esquema Proyecto es un atributo no actualizable, debido a que PROYECTO tiene una fuerte asociación con dependencia de existencia de CLIENTE. Se sugiere agregar una llave artificial para sutilmente reflejar mediante la semántica de la misma que cambios en los atributos que se refieren a CLIENTE o PROYECTO originan un nuevo hecho.

Cliente (idCliente, otros\_atributos\_cliente)  
 Proyecto (idProyecto, otros\_atributos\_proyecto)  
 Contrato (idContrato, idProyecto, idCliente)

El código SQL que se genera a partir de los esquemas de relaciones anteriores es:

```

CREATE TABLE Cliente (
    idCliente,
    Lista_otros_atributos,
    PRIMARY KEY (idCliente));
    
```

```

CREATE TABLE Proyecto (
    
```

```

idProyecto,
Lista_otros_atributos,
PRIMARY KEY (idProyecto));

```

```

CREATE TABLE Contrato (
    idContrato,
    idCliente,
    idProyecto,
    PRIMARY KEY (idContrato),
    FOREIGN KEY (idCliente)
REFERENCES Cliente(idCliente)
ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (idProyecto)
REFERENCES Proyecto(idProyecto)
ON DELETE CASCADE ON UPDATE CASCADE));

```

De manera general, la regla que se propone para transformar asociaciones binarias (1:1, 1:N ó N:M) con dependencia de existencia hace lo siguiente: crear un nuevo esquema de relación y agregar un atributo subrogado como llave primaria, agregar como llaves extranjeras las llaves primarias de los conjuntos de entidades que participan en la asociación, definiendo como acciones referenciales la eliminación y actualización en cascada para cada una de las llaves extranjeras.

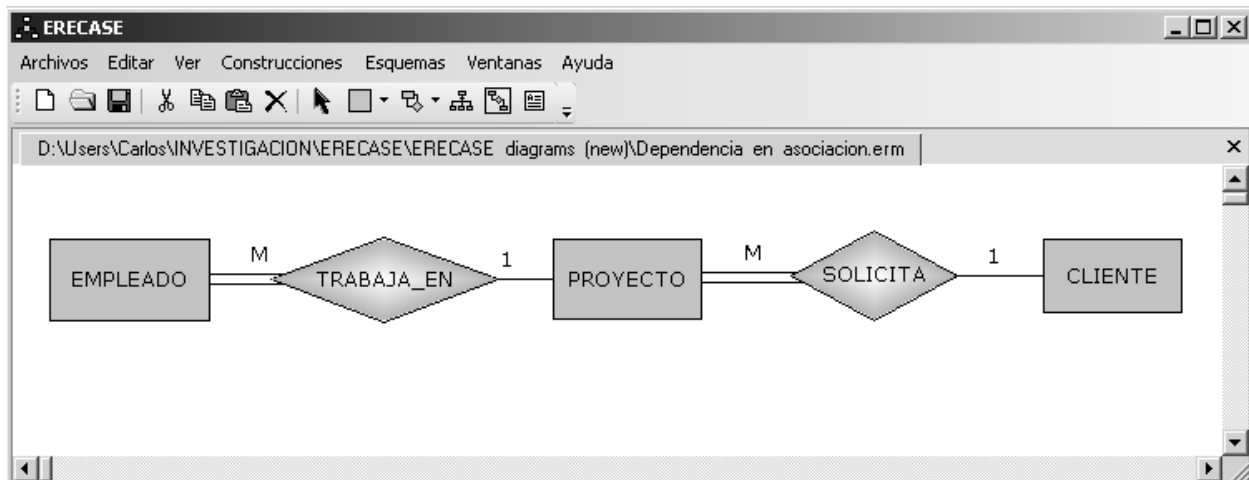
### 3.2 Aplicación de los resultados

Los resultados de este artículo han sido implementados en la herramienta ERECASE [29], la cual es una herramienta gráfica que apoya la creación de diagramas Entidad-Relación, y la transformación automática del esquema conceptual a los esquemas lógico y físico de la base de datos. Dentro de las facilidades que incluye la herramienta a diferencia de otras similares están la validación estructural del esquema conceptual y la detección y corrección de inconsistencias en el esquema lógico de la base de datos. Para la creación del modelo conceptual el diseñador dispone de las siguientes construcciones:

- conjunto de entidades fuertes y débiles.
- interrelaciones de asociación recursiva, binaria y ternaria.
- interrelaciones débiles.
- jerarquías de generalización/especialización.
- agregación.
- categorización.
- dependencia de existencia en asociaciones binarias.

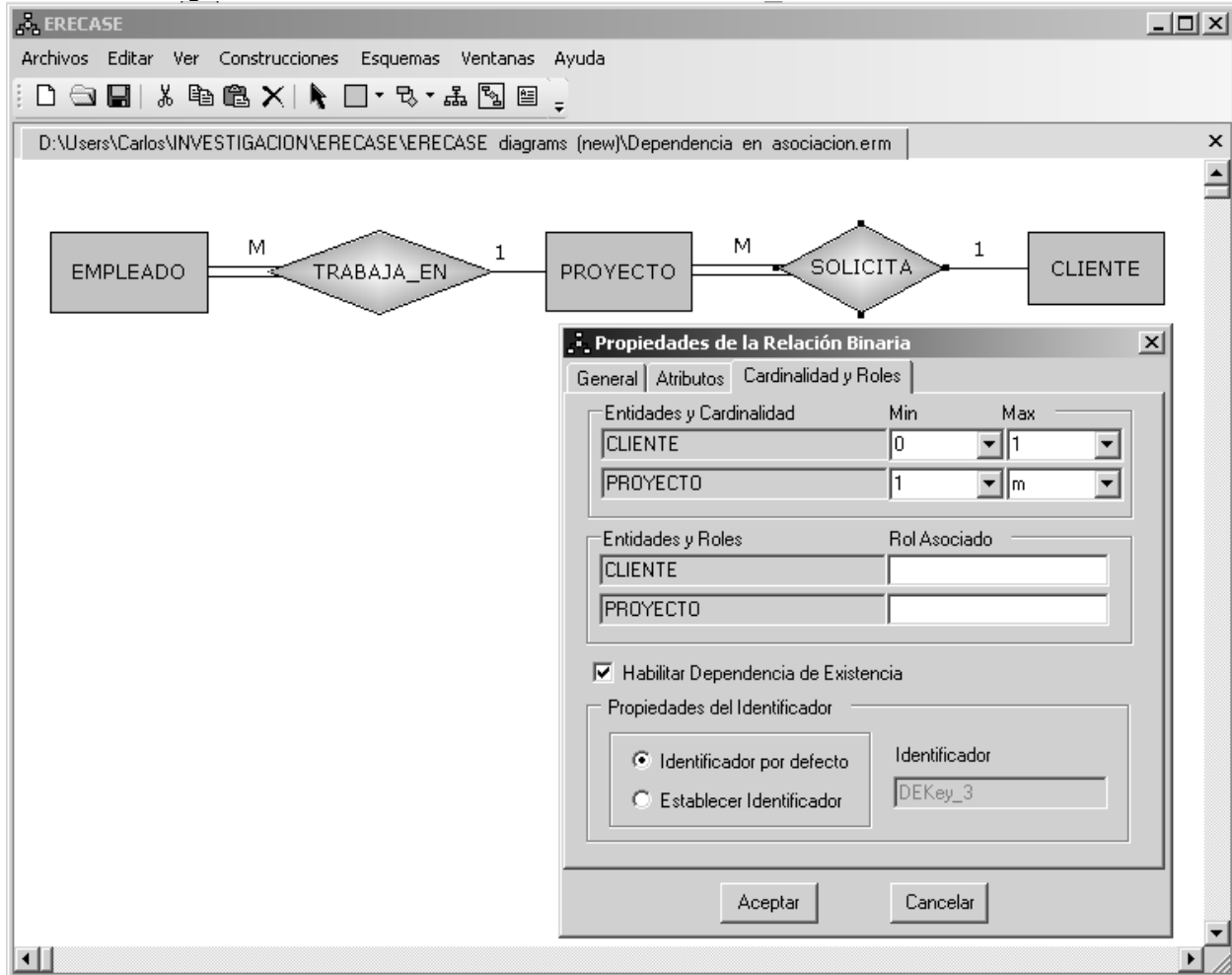
Para mostrar cómo se modela una interrelación binaria con dependencia de existencia en la herramienta se ha tomado el ejemplo analizado anteriormente, en el cual el conjunto entidad PROYECTO tiene una fuerte dependencia de existencia del conjunto entidad CLIENTE. A continuación se explicarán los pasos para modelar una interrelación binaria con dependencia de existencia.

1. Inicialmente el diseñador puede establecer una interrelación binaria normal entre dos conjuntos de entidades, en este caso entre los conjuntos de entidades PROYECTO y CLIENTE, tal y como se muestra en la Figura 7.



**Figura 7. La interrelación binaria SOLICITA.**

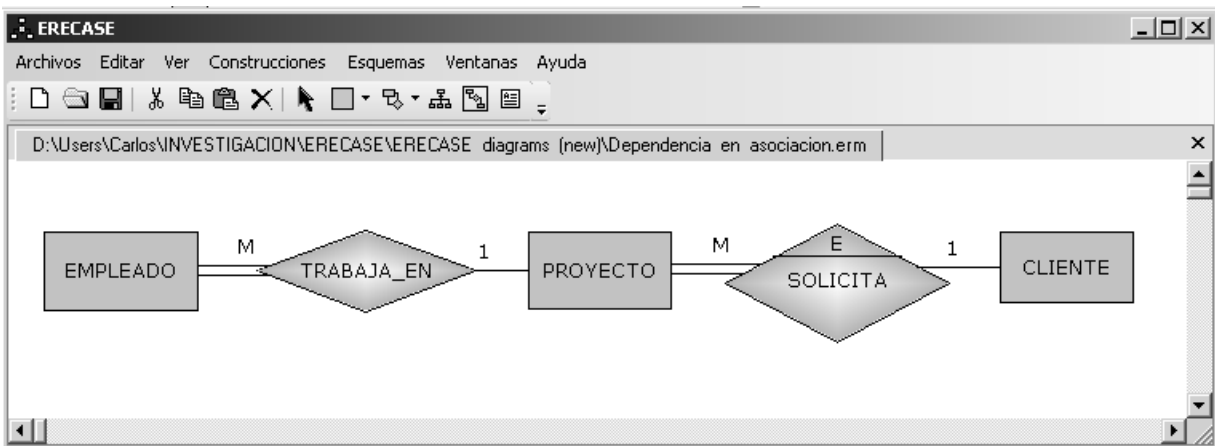
2. Para establecer una interrelación binaria con dependencia de existencia entre los conjuntos de entidades PROYECTO y CLIENTE, el diseñador debe hacer doble clic en el rombo que representa el conjunto de interrelación SOLICITA, con lo que aparece el diálogo que se muestra en la Figura 8.



**Figura 8. Propiedades de la interrelación binaria SOLICITA.**

3. Para hacer que la interrelación exprese dependencia de existencia entre los conjuntos de entidades que interrelaciona se debe marcar la opción de Habilitar Dependencia de Existencia, automáticamente se crea un atributo identificador que será utilizado en la transformación al

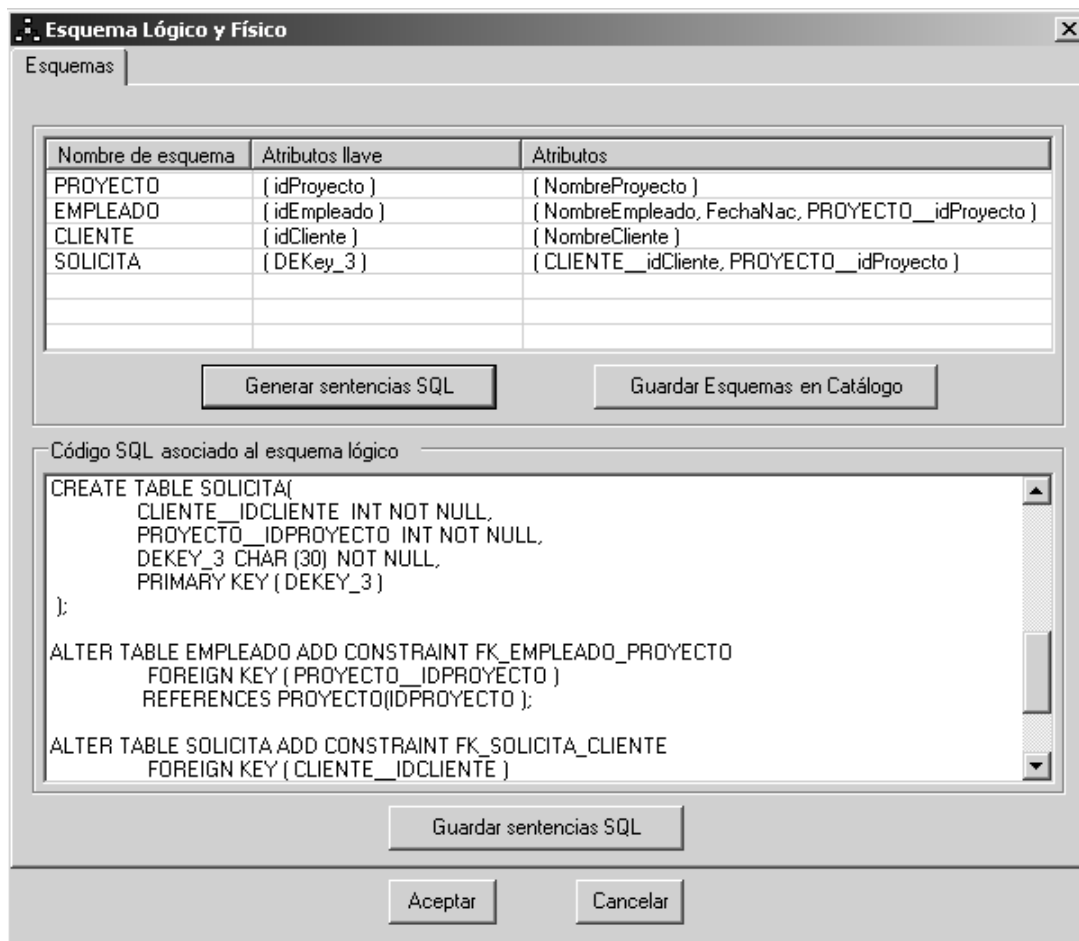
esquema lógico. Adicionalmente el diseñador puede proponer el nombre del atributo identificador. Por último se oprime el botón Aceptar.



**Figura 9.** La interrelación binaria SOLICITA con dependencia de existencia.

Tal y como se observa en la Figura 9, ahora la interrelación SOLICITA muestra otra simbología para expresar la dependencia de existencia entre los conjuntos de entidades PROYECTO y CLIENTE.

En la figura 10 se puede observar el esquema lógico obtenido para la interrelación SOLICITA aplicando la regla de transformación definida en la Sección 3.1.



**Figura 10.** Transformación de una interrelación binaria con dependencia de existencia.



### 3.3 Otras consideraciones acerca la dependencia de existencia en asociaciones

En el esquema conceptual de una base de datos se describen aspectos estáticos de los datos pero consideraciones semánticas o aspectos dinámicos, como puede ser el comportamiento en el tiempo pueden escapar en la modelación. Sobre la base de lo propuesto en [21], todas las entidades deben vincularse a través de asociaciones que expresen dependencia de existencia. A simple vista no parece ser obvia la asociación de todas las entidades con dependencia de existencia, sin embargo esto siempre será posible.

En un esquema conceptual la asociación entre dos conjuntos de entidades expresa dependencia de existencia o no. En la Figura 6 la asociación "SOLICITA" expresa dependencia de existencia, cada proyecto sólo puede existir dentro del contexto de un cliente y se refiere exactamente a uno y siempre el mismo todo el tiempo. Un cliente puede existir independientemente de si ha solicitado algún proyecto o no y puede tener varios proyectos en curso. Mientras que la asociación "TRABAJA\_EN" no expresa dependencia de existencia. Un empleado puede existir fuera del contexto de un proyecto y un proyecto puede existir fuera del contexto de un empleado.

Cuando una asociación no expresa dependencia de existencia, puede ser transformada en un conjunto de entidades que tiene dependencia de existencia de los conjuntos de entidades que participan en la asociación tal y como se muestra en la Figura 7. En este ejemplo de la empresa, la asociación "TRABAJA\_EN" con cardinalidad 1:N, es convertida en un conjunto de entidades ASIGNACION, que es dependiente de existencia de PROYECTO y EMPLEADO, tal y como se muestra en la Figura 11.



**Figura 11.** Diagrama ER que modela la asignación de los empleados a los proyectos.

De esta manera se modela lo que puede suceder durante el período de tiempo en que un proyecto y un empleado están relacionados el uno con el otro. Si en un proyecto pueden estar trabajando cero o varios empleados, entonces cada proyecto tiene cero a muchas asignaciones. Si cada empleado es asignado a exactamente un proyecto a la vez, entonces tiene exactamente una asignación por vez. El mismo razonamiento se aplica a asociaciones con cardinalidad máxima 1:1 y N:M.

## 4. CONCLUSIONES

En este artículo se ha hecho una caracterización de la dependencia de existencia en asociaciones destacando la conveniencia de uso en la modelación conceptual y las diferencias en cuanto a su semántica y comportamiento en relación con otras construcciones del modelo Entidad-Relación que tienen algún tipo de dependencia de existencia. También se propone una regla para transformar al modelo relacional las asociaciones binarias con dependencia de existencia. Los resultados de este trabajo fueron implementados en la herramienta de diseño de base de datos ERECASE, la cual fue desarrollada por los autores.

## REFERENCIAS

- [1] Maier, D. (1983), *The Theory of Relational Databases*: Computer Science Press.
- [2] Biskup, J. (1995), "Achievements of Relational Database Schema Design Theory Revisited," *Springer-Verlag, Berlin*, pp. 29-54,
- [3] Hammer, M. and McLeod, D. (1975), "Semantic Integrity in a Relational Data Base System," in *Proceedings of the International Conference on Very Large Data Bases, September 22-24, 1975, Framingham, Massachusetts, USA*, pp. 25-47.
- [4] Stonebraker, M. (1975), "Implementation of Integrity Constraints and Views by Query Modification," in *SIGMOD International Conference on Management of Data*, San Jose, California, pp. 65-78.
- [5] Eswaran, K. P. and Chamberlin, D. D. (1975), "Functional Specific Specifications of a Subsystem for Data Base Integrity," in *1st Int. Conf. on Very Large Data Bases*, Framingham, Mass., USA, pp. 48-68.

- [6] Brodie, M. L. (1978), "Specification and Verification of Data Base Semantic Integrity," in *Technical Report CSRG-91* University of Toronto.
- [7] Codd, E. F. (1979), "Extending the Data Base Relational Model to Capture More Meaning," *ACM Trans. on Database Systems*, vol. 4, pp. 397-434,
- [8] Grefen, P. W. P. J. (1992), "Integrity Control in Parallel Database systems," The Netherlands: University of Twente.
- [9] Desloch, S. (1993), "Semantic Integrity in Advanced Database Management Systems," in *Fachbereich Informatik: Universität Kaiserslautern*.
- [10] Gertz, M. (1996), *iagnosis and Repair of Constraint Violations in Database Systems*: Infix Verlag, St. Augustin, Germany.
- [11] Türker, C. (1999), *Semantic Integrity Constraints in Federated Database Schemata* vol. 63: Infix Verlag, St. Augustin, Germany.
- [12] Ullman, J. D. and Widom, J. (1999), *Introducción a los sistemas de Bases de Datos*. Mexico: Prentice Hall Hispanoamericana.
- [13] Date, C. J. (2000), *An Introduction to Database Systems*, 7th Ed ed. Boston, MA: Addison-Wesley.
- [14] Elmasri, R. and Navathe, S. B. (2003), *Fundamentals of Database Systems, fourth edition*, 4th Ed. ed.: Addison-Wesley.
- [15] Abiteboul, S., Hull, R., and Vianu, V. (1995), *Foundations of Databases*: Addison-Wesley.
- [16] Olivé, A. (2007), *Conceptual Modeling of Information Systems*: Springer Berlin Heidelberg.
- [17] Brodie, M. L., Mylopoulos, J., and Schmidt, J. W. (1984), *Perspectives from Artificial Intelligence, Databases, and Programming Languages*: New York: Springen-Verlag.
- [18] Chen, P. (1976), "The entity-relationship model: Toward a unified view of data.," *ACM Transactions on Database Systems*, vol. 1, pp. 9-36,
- [19] Guizzardi, G. and Wagner, G. (2008), "What's in a Relationship: An Ontological Analysis," in *ER 2008, LNCS 5231*, pp. 83-97.
- [20] Schneider, L. (2003), "Designing Foundational Ontologies," in *ER 2003, LNCS 2813*, pp. 91-104.
- [21] Snoeck, M. and Dedene, G. (1998), "Existence Dependency: The key to semantic integrity between structural and behavioural aspects of object types," *IEEE Transactions on Software Engineering*, vol. 24, April 1998.
- [22] Snoeck, M. and Dedene, G. (2000), "Core Modelling Concepts to define Agregation," *L'Objet*, vol. 7, pp. 281-306,
- [23] De Miguel, A. and Piattini, M. (1993), *Concepción y diseño de bases de datos: Del Modelo E/R al modelo relacional*. Madrid: RA-MA.
- [24] Ferg, S. (1985), "Modelling the Time Dimension in an Entity-Relationship Diagram," in *Fourth International Conference on Entity-Relationship Approach*, Chicago, Illinois, USA, pp. 280-286.
- [25] Smith, J. and Smith, D. (1977), "Database abstractions: Aggregation and generalization," *ACM Transactions On Database Systems*, vol. 2, pp. 105-133,
- [26] Kilov, H. and Ross, J. (1994), *Information Modeling: An Object Oriented Approach*: Cliffs N.J.: Prentice Hall.
- [27] Put, F. (1988), "Introducing dynamic and temporal aspects in a conceptual (database) schema," in *doctoral dissertation* Faculteit der Economische en Toegepaste Economische Wetenschappen: K.U.Leuven, p. 415.
- [28] Dogac, A., Ozkarahan, E., and Chen, P. (1990), "An integrity system for a relational database architecture," in *Eight International Conference on Entity-Relationship Approach*, Toronto, Canada, pp. 287 - 301.
- [29] Alvarez, W., Rodríguez, A., and García, C. (2006), "ERECASE v.2.0 Una herramienta para el diseño conceptual de bases de datos con validación estructural.," in *Departamento de Computación* Santa Clara: Universidad Central de Las Villas, p. 80.