



## Introducción a la unidad

El tema que presentamos a continuación tiene gran relevancia para tu formación como informático, principalmente en el aspecto del desarrollo de sistemas de información. Así, una fase del proceso de desarrollo de sistemas es el diseño de la base de datos.

Con la información obtenida en la etapa de análisis se desarrolla una solución de almacenamiento de datos mediante un modelo (relacional, orientado a objetos, etc.). Por lo tanto, revisaremos en esta unidad la herramienta de modelado o diseño de datos más utilizada en la vida real: el modelo entidad-relación.

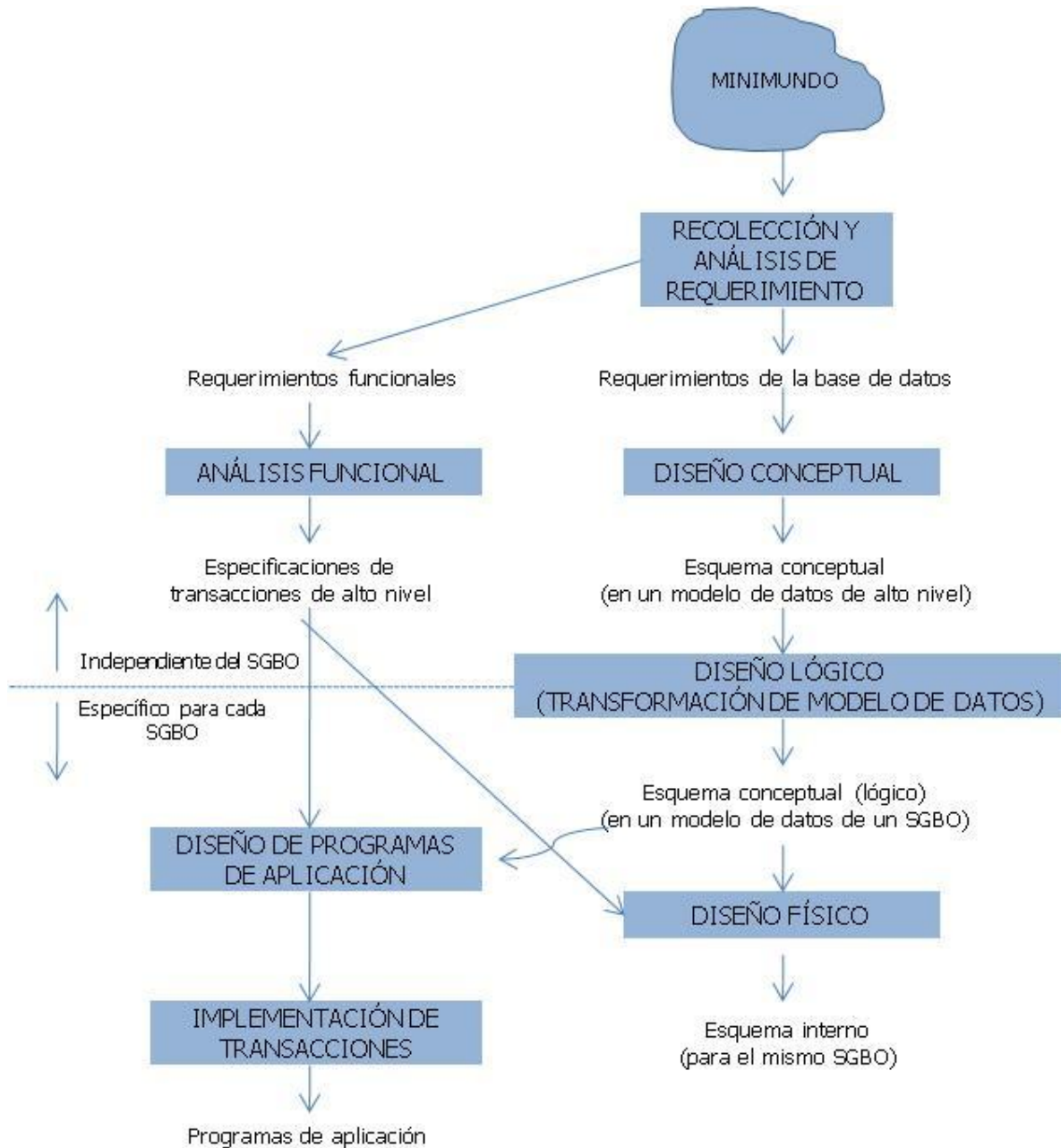
Aquí estudiarás los fundamentos teóricos de este modelo y los procedimientos para llevarlo a la práctica. Su principal ventaja es que resulta independiente del modelo de base de datos en el que será implementado.

El diseño es fundamental para la buena construcción de nuestra base de datos. Este proceso nos permitirá definir las restricciones de integridad del mundo real y llevarlas a la base de datos, además nos brindará la posibilidad de determinar cómo serán las estructuras de almacenamiento de datos y sus relaciones.

El éxito de la base de datos y del sistema en general depende (entre otras cosas) de un buen diseño. Planearlo bien nos garantiza eliminar problemas de redundancia y actualización de datos, además de ahorrar de recursos de cómputo y facilidad de consultas de datos.



## Unidad IV. Diseño





## Unidad IV. Diseño



### Objetivo particular de la unidad

Identificar los elementos del modelo entidad-relación para el diseño de bases de datos, así como reconocer los pasos fundamentales para transformar este modelo a un modelo relacional de tablas.

#### Lo que sé:

De la siguiente lectura Procedimientos de Transact-SQL que se encuentra en <http://msdn.microsoft.com/es-es/library/ms189626.aspx> (consulta 24 de marzo 2009), realiza un resumen de dos cuartillas y súbela al sitio en formato pdf.

Realiza tu actividad en un procesador de textos, guárdala en tu computadora en formato .pdf y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.



## Unidad IV. Diseño



### Temas de la unidad IV

1. Introducción al diseño
2. Modelo semántico
3. Modelo lógico
  - 3.1 E/R
    - a) Entidad
    - b) Atributo
      - Atributos compuestos
      - Atributos clave
      - Atributos multivaluados
      - Atributos derivados
    - c) Interrelación
      - Grado
      - Tipo
      - Papel (rol)
      - Cardinalidad
  - 3.2 E/R extendido
4. Modelo físico
  - 4.1 Implementación de un E/R al modelo relacional
5. Modelo de clases (UML)



## Unidad IV. Diseño



### Resumen de la unidad

En esta Unidad se vieron otros modelos para obtener una aplicación más representativa de la realidad y que se ajuste a las necesidades del usuario.

Asimismo, pudimos tener en cuenta que el diseño de bases de datos es más que un arte, refiriéndose más a un conjunto de actividades con las cuales se representa una solución para alguna aplicación. El proceso de diseño de bases de datos está integrado por tres etapas a saber: el modelo conceptual, el modelo lógico y el modelo físico, y sus dos refinaciones los modelos E/R y el Relacional.

También pudimos ver que los RDBMS (Sistema Administrador de Bases de Datos Relacionales) son los más utilizados hoy en día; es común que el proceso de diseño de una base de datos se realice inicialmente con el modelo E/R a través de un DER y después se realice un mapeo o transformación a relaciones o tablas de un modelo relacional.

De igual forma, el Modelo de Clases UML para sistemas orientados a objetos. Muestra la Estructura Estática del sistema con clases pero no todas las clases pueden terminar almacenadas en la base de datos. Entre los estándares de representación de este tipo de modelos de clases más conocido es el Lenguaje de Modelado Unificado (UML). Se trata de una norma de modelado mediante aspectos gráficos auspiciada por el Grupo de Administración de Objetos (*Object Management Group*, OMG) dedicado al desarrollo de especificaciones y estándares para crear componentes de software.



## Unidad IV. Diseño



### Tema 1. Introducción al diseño

#### Objetivo del tema

Reconocer la forma en que se realiza un diseño de base de datos.

#### Desarrollo

El diseño de bases de datos consiste en traducir un conjunto de datos inmersos en una realidad a un modelo manejable en una base de datos. Esta traducción debe darnos la estructura lógica de las estructuras para almacenar los datos y las restricciones sobre estos.

Diseñar es más un arte que una ciencia, o al menos no es posible encuadrarlo en rigurosos principios científicos.<sup>1</sup> En las siguientes secciones conocerás una metodología de diseño. Esta metodología estará basada en el desarrollo de un modelado semántico a través de un modelo entidad-relación y su posterior transformación en un modelo relacional.

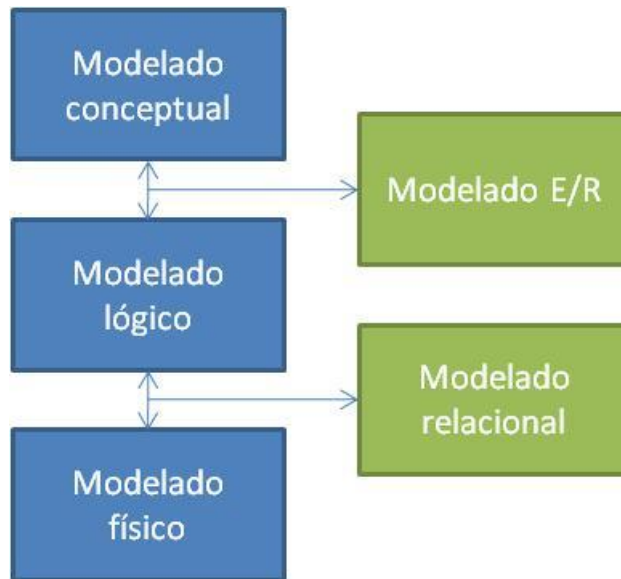
El proceso de diseño de base de datos puede verse como una interacción de tres etapas generales:

---

<sup>1</sup> Una manera de diseño de bases de datos relacionales con cierto rigor científico es el proceso de normalización que revisamos en la unidad 2.



## Unidad IV. Diseño



**Proceso del diseño de base de datos**



## Unidad IV. Diseño



### ACTIVIDAD 1

Define qué es:

- a) Metodología
- b) Modelo
- c) Modelo relacional
- d) Modelo conceptual
- e) Modelo lógico
- f) Modelo físico

Para enviar tu respuesta, pulsa el botón **Editar mi envío**; se mostrará un editor de texto en el cual puedes redactar tu información; una vez que hayas concluido, salva tu actividad pulsando el botón **Guardar cambios**.





## Unidad IV. Diseño



### Autoevaluación

Indica si las oraciones son verdaderas (V) o falsas (F). Al finalizar obtendrás tu calificación de manera automática.

	Verdadera	Falsa
1. Un modelo es un sistema.	( )	( )
2. El modelo relacional se basa en objetos de la realidad.	( )	( )
3. Un modelo lógico muestra relaciones.	( )	( )
4. Un modelo relaciones muestra jerarquías.	( )	( )
5. Un modelo físico opera sobre objetos.	( )	( )

### Bibliografía básica

Autor	Capítulo	Páginas

### Sitios electrónicos

Sitio	Descripción



## Unidad IV. Diseño



### Tema 2 Modelo semántico / Tema 3 Modelo lógico

#### Objetivo del tema

Identificar las relaciones entre los objetos y su definición semántica en el modelo relacional, así como las características del modelo conceptual y el relacional.

#### Desarrollo

El modelo relacional ha demostrado ser un modelo muy útil para el desarrollo de sistemas de información organizacionales y en algunas otras áreas de la actividad humana. Sin embargo, desde sus inicios ha sido criticado porque no representa mucha de la semántica de la realidad. El hecho de estar basado en relaciones y nada más que relaciones, le impide captar el significado de las interacciones entre éstas, su jerarquía y las restricciones asociadas a estas interacciones.

Por esta razón, y entre otras, fue propuesta una novedosa manera de modelar la realidad. Su principal característica era que intentaba representar en buena medida la semántica de la realidad. La propuesta se conoce como “Modelado Semántico”. En palabras de C. J. Date (2001: 419), lo que se trató de resolver es que “por lo regular los sistemas de bases de datos sólo tienen una comprensión muy limitada de lo que significa la información de la base de datos”.

El modelado semántico tuvo su principal desarrollo en un modelo que veremos adelante: el modelo Entidad-Relación (E/R). Este modelo materializó el objetivo del modelado semántico y tuvo las ventajas necesarias para convertirse en el más utilizado en nuestro tiempo. Claro que resulta importante aclarar que tampoco capta todos los sentidos de la realidad que representa, pero sí la gran mayoría.



## Unidad IV. Diseño



### MODELO LÓGICO

Mediante un modelado semántico de la realidad es posible obtener un modelo de datos en el nivel conceptual que nos permitirá representar la realidad en una forma dirigida al almacenamiento de datos. Como ya lo habíamos mencionado, es el modelo Entidad-Relación la manera más utilizada para hacerlo. Los elementos y características de este modelo se verán en la siguiente sección.

A partir del modelo conceptual se deriva un modelo lógico específico para un tipo de base de datos: orientado a objetos, relacional, jerárquico, etc. En nuestro caso, el modelo lógico será basado en el modelo relacional.

### E/R

El modelo Entidad-Relación (E/R) ayuda a realizar un diseño de bases de datos sin atender a un modelo en especial (jerárquico, relacional, orientado a objetos).



Fue propuesto por Peter Chen en el artículo “The entity-relationship model - Toward a unified view of data” (1976). En éste, Chen propone utilizar un enfoque más natural del mundo real basado en entidades e interrelaciones. Hoy en día,



## Unidad IV. Diseño



podríamos decir que más bien existe una familia de modelos, ya que muchos autores han realizado propuestas que han enriquecido al modelo E/R.

El modelo E/R, permite visualizar la base de datos desde un alto nivel de abstracción. Los elementos interesantes de la realidad que queremos modelar son las entidades, además modelamos sus atributos y las interacciones entre ellas.

Una ventaja del modelo E/R es que utiliza una representación gráfica conocida como Diagrama Entidad-Relación (DER). Es importante mencionar que existen distintas representaciones de un DER en las que cambian los aspectos gráficos, pero se modelan los mismos elementos. A continuación detallo los elementos del modelo E/R e incluyo su representación gráfica en el DER.

### Elementos del Modelo E-R

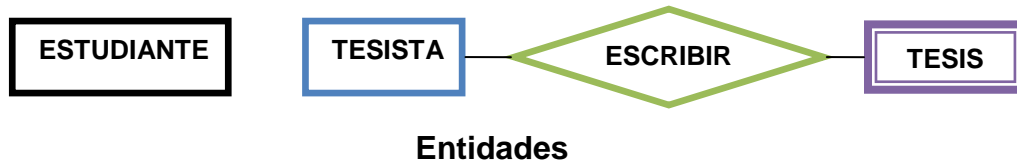
#### ENTIDAD

Una entidad es cualquier “objeto (real o abstracto) que existe en la realidad y acerca del cual queremos almacenar información en la base de datos” (De Miguel 2001: 49). Las entidades se agrupan en tipos de entidades, de los cuales podemos identificar ejemplares, por ejemplo, el Sr. Ruiz sería un ejemplar del tipo de entidad PERSONA. Los tipos de entidades se representan con un rectángulo con el nombre del tipo de entidad en su interior. Existen entidades, llamadas débiles, cuya existencia depende de que exista otra entidad, denominada fuerte o regular; las entidades débiles se representan con doble rectángulo.

Algunos ejemplos son los siguientes. El primer ejemplo corresponde a la entidad ESTUDIANTE. El segundo, se refiere a una interrelación (ESCRIBIR) entre una entidad fuerte, TESISTA, y una entidad débil, TESIS. Ésta última es débil porque su existencia depende de la entidad TESISTA.



## Unidad IV. Diseño

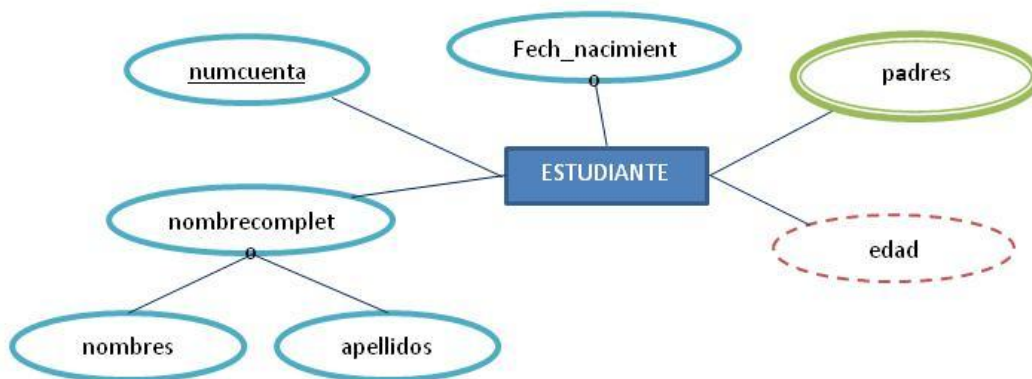


### ATRIBUTO

Un atributo es una característica o propiedad de un tipo de entidad o interrelación, que puede tomar distintos valores. Al conjunto de valores se le distingue como dominio. Los atributos se representan con elipseHos que incluyen el nombre en su interior.

Entre los atributos tenemos a los:

- Atributos compuestos
- Atributos clave
- Atributos multivaluados
- Atributos derivados



### Tipos de atributos en una entidad



## Unidad IV. Diseño



### INTERRELACIÓN

Una interrelación es “una asociación, vinculación o correspondencia entre entidades” (De Miguel 2001: 51). Igual que las entidades, las interrelaciones se agrupan en diferentes tipos. Por ejemplo, el tipo de interrelación IMPARTE es la vinculación entre las entidades PROFESOR y CURSO. Un ejemplar de esta interrelación sería la vinculación entre el profesor Sr. Ruiz y el curso Bases de datos. Las interrelaciones se representan gráficamente con un rombo que incluye el nombre del tipo de interrelación en su interior.

En algunos casos de diseño, será posible ver una interrelación como un tipo especial de entidad débil. Para decidir si lo modelaremos como entidad o como interrelación dependerá de la conveniencia en el modelo y de la realidad que analicemos.

Las interrelaciones cuentan con varias características:

- a) Grado
- b) Tipo
- c) Papel (rol)
- d) Cardinalidad

Para que conozcas a qué se refiere cada uno de los atributos y de las interrelaciones, por favor descarga el archivo **modelo E-R (ANEXO 1)**.



## Unidad IV. Diseño



### E/R extendido

El modelo E/R puede captar la mayoría de las características semánticas para la base de datos, pero es posible extenderlo para captar otros aspectos. Esta extensión es conocida como modelo Entidad-Relación extendido. Las características de modelo extendido son:

- \* Especialización
- \* Generalización
- \* Agregación
- \* Herencia de atributos<sup>2</sup>

---

<sup>2</sup> Si quieres revisar aspectos de estas características puedes leer Silberschatz, et. al. (2006: 190-196).



## Unidad IV. Diseño



### ACTIVIDAD 1

Realiza dos Modelos E/R de la siguiente forma:

1. Relación entre las Entidades Clientes y Ventas, y
2. Relación entre las Entidades Clientes y Almacén

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.

### ACTIVIDAD 2

Desarrolla el Modelo E/R de las entidades:

1. Libro, Autor y Tema
2. Videocasete, Director y Película

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.





### ACTIVIDAD 3

Responde el siguiente cuestionario.

1. Anote el concepto de:

- a) Semántica
- b) Modelo Semántico
- c) Realidad
- d) Entidad
- e) Atributos clave

2. Escribe algunos ejemplos de Información contenida en las entidades.

3. Defina el Modelo E/R

4. El Modelo E/R fue creado por:

5. El Modelo Lógico esta basado en:

6. El Modelo conceptual se deriva de:

Para enviar tu respuesta, pulsa el botón **Editar mi envío**; se mostrará un editor de texto en el cual puedes redactar tu información; una vez que hayas concluido, salva tu actividad pulsando el botón **Guardar cambios**.



## Unidad IV. Diseño



### Autoevaluación

Indica si las oraciones son verdaderas (V) o falsas (F). Al finalizar obtendrás tu calificación de manera automática.

	Verdadera	Falsa
1. La Semántica define una relación.	( )	( )
2. El modelo relacional es un modelo incompleto.	( )	( )
3. El modelo semántico elimina desventajas del modelo relacional.	( )	( )
4. El modelo E/R resuelve el problema de las relaciones.	( )	( )
5. El modelo E/R es un desarrollo del modelo semántico.	( )	( )
6. J. C. Date creo el Modelo E/R.	( )	( )
7. Una entidad es un Atributo de una Tabla.	( )	( )
8. La Cardinalidad es una característica de las interrelaciones.	( )	( )
9. La Agregación es una característica del modelado extendido.	( )	( )
10. Un número determinado de veces expresado en un 10, a esta relación se llama Papel.	( )	( )



## Unidad IV. Diseño



### Bibliografía básica

Autor	Capítulo	Páginas

### Sitios electrónicos

Sitio	Descripción



## Unidad IV. Diseño



### Tema 4. Modelo físico

#### Objetivo del tema

Identificar las características del modelo físico e ilustrarlo en el modelo E-R.

#### Desarrollo

El modelo Entidad-Relación nos permite representar la realidad sobre la cual vamos a almacenar información. Mediante el Diagrama Entidad-Relación (DER), captamos el significado de todo aquello que queremos almacenar en la base de datos. Gracias a esto, podemos pasar de un modelo conceptual a un modelo lógico y finalmente a uno físico.

Para realizar lo anterior, es necesario determinar el modelo de datos con el que construiremos nuestra base. En nuestro caso, utilizaremos el modelo relacional, que como recordarás, está basado en relaciones o tablas. En el modelo físico se determinan aspectos físicos de almacenamiento como: *registros, punteros, direccionamiento y asignación de espacio para memorias intermedias.*

La realidad es que hoy en día son los manejadores de bases de datos los que se encargan de esto y el diseñador de bases de datos no juega un papel decisivo en estos aspectos físicos del almacenamiento. La parte donde sí se involucra el experto en bases de datos es en la modificación de parámetros de rendimiento del manejador.

Dado que los RDBMS (Sistema Administrador de Bases de Datos Relacionales) son los más utilizados hoy en día, es común que el proceso de diseño de una base de datos se realice inicialmente con el modelo E/R a través de un DER y después se realice un mapeo o transformación a relaciones o tablas de un modelo relacional. A continuación veremos cómo se hace esto, por lo que te solicito



## Unidad IV. Diseño



descargues el archivo de **Implementación de un modelo E/R al modelo relacional (ANEXO 2)**.

### ACTIVIDAD 1

Del Diagrama Entidad-Relación de las Tablas Cliente, Ventas y Almacén, desarrolla el Modelo Conceptual y Lógico de éstas mismas.

Haz lo mismo con las tablas Videocasete, Director y Película.

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.

### ACTIVIDAD 2

Contesta las siguientes preguntas:

1. ¿Qué significan las siglas DER?
2. ¿Para qué se empleaba inicialmente el Modelo? E/R?
3. Anota las tres reglas generales para convertir el Modelo E/R en Modelo Relacional.
4. ¿Qué aspectos físicos se determinan en el modelo físico?
5. ¿Para qué sirve el Check?

Para enviar tu respuesta, pulsa el botón **Editar mi envío**; se mostrará un editor de texto en el cual puedes redactar tu información; una vez que hayas concluido, salva tu actividad pulsando el botón **Guardar cambios**.



## Unidad IV. Diseño



### Autoevaluación

Contesta falso o verdadero a las siguientes afirmaciones. Al concluir obtendrás tu calificación de manera automática.

	Verdadera	Falsa
1. El modelo lógico es el término final para llegar a la representación de la realidad.	( )	( )
2. El manejador de bases de datos se encarga de los aspectos físicos de almacenamiento.	( )	( )
3. El experto en bases de datos modifica los parámetros de rendimiento del software.	( )	( )
4. La cardinalidad implica restricción.	( )	( )
5. El direccionamiento es un aspecto del modelo E/R.	( )	( )

### Bibliografía básica

Autor	Capítulo	Páginas

### Sitios electrónicos

Sitio	Descripción



## Unidad IV. Diseño



### Tema 5. Modelo de clases (UML)

#### Objetivo del tema

Identificar las características del modelo de clases UML.

#### Desarrollo

El modelo de clases es una herramienta para modelado de sistemas orientado a objetos. Muestra la estructura estática del sistema mediante clases. Estas clases representan objetos involucrados en el sistema; pueden mantener relaciones entre ellas, ser especializaciones de otras clases o tener dependencias.

Algunas de estas clases pueden terminar almacenadas en la base de datos, pero no necesariamente. El diagrama de clases es producto del modelado de sistemas orientado a objetos, de tal modo que si la base de datos es orientada a objetos, sería de esperarse que las clases del sistema sean persistentes de forma transparente.

El problema radica en si la implementación se realiza en una base de datos relacional. En este caso parece necesario, además de este diagrama, elaborar un Diagrama Entidad-Relación y luego pasarlo a tablas. El diagrama de clases ayuda a modelar las entidades que intervienen en el sistema junto con sus atributos, pero recordemos que no está orientado al modelado de datos.

Existen estándares de representación de este tipo de modelos de clases. Uno muy conocido y utilizado es el Lenguaje de Modelado Unificado (UML). Se trata de una norma de modelado mediante aspectos gráficos auspiciada por el Grupo de



## Unidad IV. Diseño



Administración de Objetos (*Object Management Group*, OMG <sup>3</sup>dedicado al desarrollo de especificaciones y estándares para crear componentes de software.

Hemos visto un acercamiento al diseño de base de datos llamado modelado semántico. Éste se realiza mediante un modelo Entidad-Relación que se representa en un Diagrama Entidad Relación. Una vez hecho, hemos obtenido un modelo conceptual de la realidad que queremos almacenar en la base de datos.

Después, siguiendo las reglas que te propusimos en la sección correspondiente, transformamos el modelo E/R en un modelo específico de base de datos. En nuestro caso, se transformó en un modelo relacional, el cual representa el modelo lógico de la base de datos que se implementa en un manejador de base de datos específico mediante instrucciones de SQL. Es precisamente éste último paso, el objetivo de la siguiente unidad.

---

<http://www.uml.org/>





## Unidad IV. Diseño



### ACTIVIDAD 1

Realiza el modelado de clases del siguiente caso de diseño, y represéntalo mediante un DER.

*Se necesita un registro de los becarios y los proyectos en los que participan. Los becarios pueden participar en varios proyectos y en cada proyecto siempre trabajan varios becarios. Los becarios tienen los siguientes atributos: número de cuenta, nombre y tareas (grupo de tareas que simultáneamente realiza el becario en el proyecto). De los proyectos necesitamos conocer la fecha de inicio, nombre, número de proyecto y si cuenta con patrocinio de PAPIIT O CONACYT. Existen tres tipos de becarios de acuerdo a sus estudios: licenciatura, maestría y doctorado. Los de licenciatura, además de sus datos generales, cuentan con créditos; los de maestría, nombre de tesis; y los de doctorado, comité doctoral, compuesto por tres o cuatro profesores*

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.

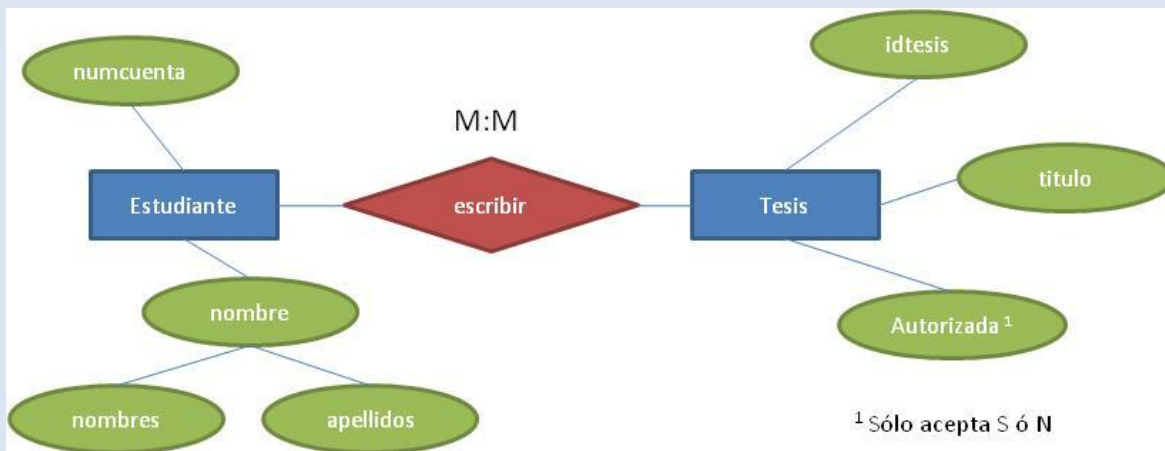


## Unidad IV. Diseño



### ACTIVIDAD 2

Observa el siguiente DER y, empleando los pasos proporcionados en la lectura, obtén un modelo relacional de tablas.



Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.



## Unidad IV. Diseño



### Cuestionario de la unidad

Responde el siguiente cuestionario.

1. ¿En qué consiste el modelado semántico de base de datos?
2. ¿Cuáles son las carencias del modelo relacional que dieron paso al surgimiento del modelo semántico?
3. ¿Cómo se llama el modelo semántico más utilizado para diseñar bases de datos?
4. ¿Qué es cardinalidad?
5. ¿Qué es una interrelación?
6. Explica cada uno de los tipos de atributos que existen en el modelo E/R.
7. ¿Cuáles son los tipos de entidades?
8. Describe cada una de las características de una interrelación.
9. ¿Cuáles son los pasos generales a seguir para la implementación de un modelo E/R en un modelo relacional?
10. ¿Cuál es el proceso que se sigue para implementar una interrelación de M:M al modelo relacional?

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.



## Unidad IV. Diseño



### Autoevaluación

Indica si las oraciones son verdaderas (v) o falsas (f). Al finalizar obtendrás tu calificación de manera automática.

	Verdadera	Falsa
1. El modelo semántico es una representación a nivel de modelo lógico de la base de datos.	( )	( )
2. Edgar Codd propuso el modelo entidad-relación.	( )	( )
3. Un atributo multivaluado es aquel que se compone de múltiples subatributos.	( )	( )
4. Un atributo derivado es aquel que se obtiene de una operación entre otros atributos.	( )	( )
5. La cardinalidad de un atributo consistiría en el número mínimo y máximo de valores que puede tomar ese atributo en cada ejemplar del tipo de entidad al cual pertenece.	( )	( )
6. Una interrelación tiene tres grados: 1 a 1, 1 a muchos y muchos a muchos.	( )	( )
7. Una interrelación M:M se implementa al modelo relacional como una nueva relación con los atributos clave de las relaciones involucradas.	( )	( )
8. Para la implementación del DER en el modelo relacional todos los atributos se convierten en columnas.	( )	( )
9. La relación 1 a muchos se implementa en el modelo relacional mediante la propagación de clave del lado de 1 al lado de muchos.	( )	( )
10. En la implementación del modelo E/R al modelo relacional todas las entidades se convierten en relaciones.	( )	( )



## Unidad IV. Diseño



### Bibliografía básica

Autor	Capítulo	Páginas

### Sitios electrónicos

Sitio	Descripción



## Unidad IV. Diseño



### Lo que aprendí

De las siguientes tablas que integran la base de datos “Video”, construye el modelo E/R para luego desarrollar su modelo semántico y finalmente su modelo físico. Su medio de almacenamiento será disco magnético. Para las condiciones de cardinalidad se tiene que la tabla TITULO tiene 45 títulos, la Tabla DISTRIBUIDOR tiene tres distribuidores, y la Tabla TIENDA tiene 10 tiendas.

TITULO			
ID_VIDEO	TIT_VIDEO	PRODUCT_VIDEO	DIEC_VIDEO

DISTRIBUIDOR			
ID_DIST	NOMB_DIST	PAIS_DIST	ID_VIDEO

TIENDA			
ID_TD	NOMB_TD	GERENTE_TD	DIREC_TD

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma



## Unidad IV. Diseño



### GLOSARIO

#### Atributo

Es una característica o propiedad de un tipo de entidad o interrelación, que puede tomar distintos valores.

#### Atributos clave

Todo ejemplar de una entidad debe ser identificado de manera única por un atributo llamado *clave*. Para representarlo gráficamente es necesario subrayar el nombre del atributo.

#### Atributos multivaluados

Otro tipo de atributo es el multivaluado, caracterizado porque cada ejemplar de entidad puede tomar varios valores del dominio, por ejemplo, un curso puede impartirse en dos idiomas. Su representación es con elipse doble.

#### Atributos derivados

Otro tipo de atributo es el derivado de una operación entre otros atributos. Éste lo podemos ubicar mediante una elipse punteada.

Es importante comentar que un atributo puede presentar cardinalidad, la que consistiría en el número mínimo y máximo de valores que puede tomar ese atributo en cada ejemplar del tipo de entidad al cual pertenece.

#### Cardinalidad

“Se define como el número máximo y mínimo de ejemplares de un tipo de entidad que pueden estar interrelacionados con un ejemplar del otro tipo” (De Miguel 2001: 63). Los valores que podrían tomar son (0,1), (1,1), (0,N) o (1,N), donde N significa muchos ejemplares.



## Unidad IV. Diseño



### **Diseño de bases de datos**

Consiste en traducir un conjunto de datos inmersos en una realidad a un modelo manejable en una base de datos. Esta traducción debe decirnos la estructura lógica de las estructuras para almacenar los datos y las restricciones sobre estos.

### **Dominio**

Es el conjunto de valores.

### **Entidad**

Es cualquier “objeto (real o abstracto) que existe en la realidad y acerca del cual queremos almacenar información en la base de datos”.

### **Grado**

“Es el número de tipos de entidad que participan en un tipo de interrelación” (De Miguel 2001: 61). Una interrelación de grado dos se refiere a la vinculación de dos tipos de entidades. Un tipo especial de interrelación de grado dos es la reflexiva, que asocia un tipo de entidad consigo misma.

### **Interrelación**

Es “una asociación, vinculación o correspondencia entre entidades.

### **Modelo de clases**

Es una herramienta para modelado de sistemas orientado a objetos. Muestra la estructura estática del sistema mediante clases. Estas clases representan objetos involucrados en el sistema; pueden mantener relaciones entre ellas, ser especializaciones de otras clases o tener dependencias.

### **Modelo Entidad-Relación (E/R)**

Ayuda a realizar un diseño de bases de datos sin atender a un modelo en especial (jerárquico, relacional, orientado a objetos).





## Unidad IV. Diseño



### **Papel (rol)**

“Es la función que cada uno de los tipos de entidad realiza en el tipo de interrelación” (De Miguel 2001: 63). Cuando la función de una entidad en la interrelación es ambigua o no se puede inducir de manera clara, se recomienda colocar el papel (rol) en la línea que conecta a la entidad con la interrelación.

### **Tipo**

“Es el número máximo de ejemplares de un tipo de entidad que pueden estar asociados” (De Miguel 2001: 62). Los tipos son uno a uno (1:1), uno a muchos (1:M) y muchos a muchos (M:M).



## Unidad IV. Diseño



### **MESOGRAFÍA**

#### **Bibliografía básica**

#### **Bibliografía complementaria**

#### **Sitios electrónicos**



## Unidad IV. Diseño ANEXO 1



### **Modelo E-R**

#### **Interrelación**

Una interrelación es “una asociación, vinculación o correspondencia entre entidades” (De Miguel 2001: 51). Igual que las entidades, las interrelaciones se agrupan en diferentes tipos. Por ejemplo, el tipo de interrelación IMPARTE es la vinculación entre las entidades PROFESOR y CURSO. Un ejemplar de esta interrelación sería la vinculación entre el profesor Sr. Ruiz y el curso Bases de datos. Las interrelaciones se representan gráficamente con un rombo que incluye el nombre del tipo de interrelación en su interior.

En algunos casos de diseño, será posible ver una interrelación como un tipo especial de entidad débil. Para decidir si lo modelaremos como entidad o como interrelación dependerá de la conveniencia en el modelo y de la realidad que analicemos.

Las interrelaciones cuentan con varias características:

#### **a) Grado**

“Es el número de tipos de entidad que participan en un tipo de interrelación” (De Miguel 2001: 61). Una interrelación de grado dos se refiere a la vinculación de dos tipos de entidades. Un tipo especial de interrelación de grado dos es la reflexiva, que asocia un tipo de entidad consigo misma.

#### **b) Tipo**

“Es el número máximo de ejemplares de un tipo de entidad que pueden estar asociados” (De Miguel 2001: 62). Los tipos son uno a uno (1:1), uno a muchos (1:M) y muchos a muchos (M:M).



## Unidad IV. Diseño

### ANEXO 1



#### **c) Papel (rol)**

“Es la función que cada uno de los tipos de entidad realiza en el tipo de interrelación” (De Miguel 2001: 63). Cuando la función de una entidad en la interrelación es ambigua o no se puede inducir de manera clara, se recomienda colocar el papel (rol) en la línea que conecta a la entidad con la interrelación.

#### **d) Cardinalidad**

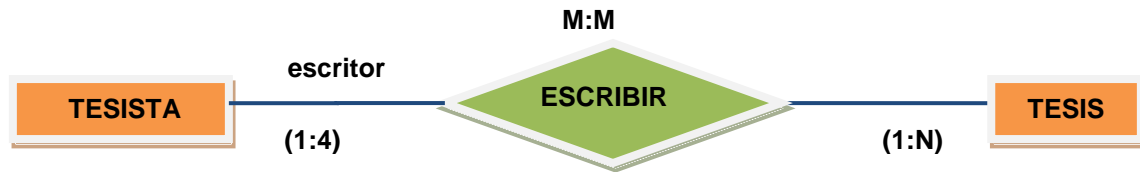
“Se define como el número máximo y mínimo de ejemplares de un tipo de entidad que pueden estar interrelacionados con un ejemplar del otro tipo” (De Miguel 2001: 63). Los valores que podrían tomar son (0,1), (1,1), (0,N) o (1,N), donde N significa muchos ejemplares.

Para comprender mejor las interrelaciones, veamos el siguiente ejemplo. En éste podemos observar una interrelación de tipo, muchos a muchos (M:M), entre TESISTA y TESIS. Esto significa que muchos tesisistas escriben una tesis, pero también que muchas tesis son escritas por un tesisista. Pero, si miramos bien, tenemos una restricción establecida por la cardinalidad en TESISTA, ésta nos indica que pueden participar en la relación al menos 1 tesisista y como máximo 4; en otras palabras, una tesis es escrita por mínimo 1 y máximo 4 tesisistas.

En el caso de la cardinalidad de TESIS, ésta nos indica que un tesisista escribe 1 o muchas tesis, sin restricción de cuántas. También se indica el papel o rol del tesisista en la interrelación, consistiendo su rol en “escritor”.

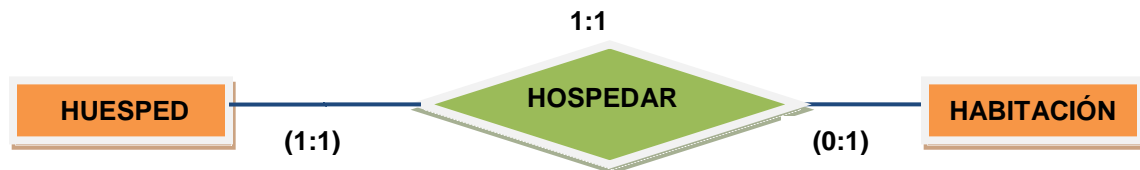


## Unidad IV. Diseño ANEXO 1



### Ejemplo 1. Interrelaciones

En otro ejemplo, las situaciones cambian. De acuerdo con este tipo de relación, una habitación sólo puede estar ocupada por un huésped y un huésped sólo puede ocupar una habitación (1:1). La cardinalidad nos indica que un huésped sólo puede tomar una habitación como máximo (1:1), pero una habitación puede estar desocupada o puede estar ocupada por máximo un huésped, por lo que la cardinalidad de su interrelación es 0:1.



### Ejemplo 2. Interrelaciones



## ***Implementación de un modelo E/R al modelo relacional***

Una vez realizado el modelado para obtener el Diagrama Entidad-Relación, es necesario seguir un conjunto de pasos para convertirlo en un modelo relacional. Las tres reglas generales para realizar esto son:

- 1) Los tipos de entidades se convierten en relaciones.
- 2) Las interrelaciones N:M se transforman en relaciones.
- 3) Para las interrelaciones 1:N se realiza una propagación de clave a partir del lado de 1 hacia el lado de N.

A continuación, explicaremos a detalle los pasos necesarios para esta derivación. Es necesario recordar que, de forma ideal, se debe mantener la semántica del modelo conceptual en el modelo lógico. Desafortunadamente, no siempre es posible hacerlo. Las reglas específicas para derivar un modelo Entidad-Relación son:

### **1. Derivación de dominios**

Debe crearse los dominios de todos los atributos. Para esto utilizamos la sentencia CREATE DOMAIN, por ejemplo:

```
CREATE DOMAIN edo_civil AS CHAR(1)  
CHECK (VALUE IN ('S', 'C'));
```

### **2. Derivación de entidades**

Cada tipo de entidad se transforma en una relación o tabla. Para ello, utilizamos la sentencia CREATE TABLE.



## Unidad IV. Diseño

### ANEXO 2



### 3. Derivación de atributos

Todo atributo de una entidad se transforma en una columna de su relación correspondiente.

- a. Atributos identificadores (claves o principales). Se transforman en claves primarias. Para ello se usa la restricción de integridad PRIMARY KEY. Por ejemplo:

```
cod_alumno INTEGER CONSTRAINT pk_cod_alumno PRIMARY  
KEY
```

- b. Atributos identificadores alternativos (claves candidatas). Se les aplica la restricción de integridad UNIQUE. Por ejemplo:

```
rfc CHAR(13) CONSTRAINT un_rfc UNIQUE
```

- c. Atributos no identificadores (no clave, no principales). Se les aplica la restricción de integridad de NOT NULL, sólo si es necesario. Por ejemplo:

```
nombre VARCHAR(30) NOT NULL
```

### 4. Derivación de interrelaciones

- a. **Interrelaciones N:M.** La interrelación se transforma en una relación que tendrá como superclave los atributos identificadores (claves o principales) de las entidades que relaciona. Los atributos de la relación resultante (relación o tabla de interrelación) se convierten en llaves foráneas (claves ajenas) mediante la restricción FOREIGN KEY. Para



## Unidad IV. Diseño

### ANEXO 2



esta restricción es necesario indicar la tabla padre y la llave primaria en esa tabla. Además, podemos indicar las restricciones de borrado y actualización en cascada. Por ejemplo:

```
CREATE TABLE imparte
(
cod_curso INTEGER,
cod_profesor INTEGER,
PRIMARY KEY (cod_curso, cod_profesor), /*Primaria
compuesta*/
CONSTRAINT fk_cod_curso FOREIGN KEY (cod_curso)
REFERENCES curso(cod_curso)
ON DELETE CASCADE
ON UPDATE CASCADE, /* Foránea con borrado y
actualización en cascada*/
CONSTRAINT fk_cod_profesor FOREIGN KEY
(cod_profesor)
REFERENCES profesor(cod_profesor)
ON DELETE CASCADE
ON UPDATE CASCADE /* Foránea con borrado y
actualización en cascada*/
)
```

Otro aspecto importante a derivar, es la cardinalidad de las entidades participantes. Como sabemos, existen casos en los que hay restricciones de negocio que impiden que un determinado número de ejemplares de una entidad interactúe con ejemplares de otra entidad. En este caso es necesario agregar una restricción de aserción. Desafortunadamente no todos los manejadores de bases de datos incluyen esta posibilidad. En ese





## Unidad IV. Diseño

### ANEXO 2



caso, debemos utilizar disparadores (*triggers*) de integridad. Por ejemplo, supongamos que un curso sólo puede ser impartido por cuatro profesores, es decir, cardinalidad (0, 4). En ese caso tendríamos que crear la siguiente aserción:

```
CREATE ASSERTION profesor_curso
CHECK NOT EXIST (SELECT COUNT(*)
                  FROM imparte
                  GROUP BY cod_curso
                  HAVING COUNT(*)>=4);
```

- b. Interrelaciones 1:N.** La manera general de transformar esta interrelación consiste en propagar (pasar) el atributo identificador (clave o principal) de la entidad con cardinalidad 1 hacia la entidad con cardinalidad N. Como resultado la interrelación se pierde, es decir, no se convierte en entidad como en el caso de N:M.

El atributo propagado se convierte en llave foránea, para el cual debemos indicar si existe borrado en cascada. La cardinalidad de la interrelación también debe ser implementada como ya establecimos, esto es, utilizando restricciones de aserción o triggers. Adicionalmente, si la cardinalidad indica que el número de ejemplares puede ser 0, la llave foránea aceptará valores nulos. Si, por el contrario, indica que al menos un ejemplar debe estar relacionado con uno o más ejemplares (cardinalidad 1), la llave foránea debe tener restricción de NOT NULL.

- c. Interrelaciones 1:1.** No hay regla fija para la transformación de esta interrelación. Lo más común es aplicar la regla 4.2, aunque puede ser posible la aplicación de la regla 4.1. Es importante notar que al aplicar



## Unidad IV. Diseño

### ANEXO 2



la regla 4.2 necesitamos decidir en qué dirección se propaga la llave primaria. Para decidirlo podemos tomar en cuenta al menos dos aspectos:

- i. Pasar la llave primaria de una entidad fuerte a una débil es generalmente más conveniente.
- ii. Pasar la llave primaria de la entidad con cardinalidad (1, 1) a la entidad con cardinalidad (0, 1) permite aplicar NOT NULL a la llave foránea.

#### 5. Atributos de interrelaciones

Si la interrelación se transforma en relación, los atributos de la interrelación se transforman en columnas de la relación resultante. En caso de que se aplicase una propagación de identificador (clave), los atributos de la interrelación pasan en la misma dirección que dicha clave.

#### 6. Otras restricciones

Podemos encontrar restricciones de valores posibles para un atributo. Para transformarlas al modelo relacional utilizamos la restricción CHECK. Además, si el manejador de bases de datos no acepta creación de dominios, con el CHECK es posible restringir los valores de un dominio.

#### 7. Transformación de dependencias de existencia e identificación

En este caso se realiza una propagación de identificador (clave) asegurándonos de poner en la llave foránea, si es necesaria, la restricción de borrado y actualización en cascada. Cuando se trate de una



## Unidad IV. Diseño

### ANEXO 2



dependencia de identificación, la llave primaria propagada se combina con algún atributo de la entidad débil para formar una superclave.

#### 8. Derivación de tipos y subtipos

No hay una manera de conservar la semántica de los tipos y subtipos en el modelo relacional. Lo más recomendable es crear relaciones para cada supertipo y para cada subtipo. Después, propagar la clave principal del supertipo en cada relación de los subtipos. Esta clave propagada será, además de llave foránea, la llave primaria de cada subtipo. Finalmente, si queremos asegurarnos de que un tipo no pueda aparecer en varios subtipos, tendremos que usar disparadores (triggers) que revisen que la clave del tipo no exista ya en algún subtipo.