

**Nombre de la asignatura**

---

Tema n.º 2

Diseño de Base de Datos

## Índice

	Pág.
2.1. Introducción	3
2.2. Diseño de Base de Datos	3
2.2.1 Diseño - Etapa 1. Diseño Conceptual: Modelo Entidad/Relación (E/R)	3
2.2.2 Diseño - Etapa 2. Diseño Lógico: Modelo Relacional	4
2.2.3 Diseño - Etapa 2. Diseño Lógico: Modelo Relacional	4
2.2.4 Diseño - Etapa 3. Diseño Físico: Modelo Físicos	5
2.3. MODELO ENTIDAD/RELACIÓN (E/R)	5
2.3.1. Entidades	6
2.3.2. Atributos	6
2.3.3. Relaciones	8
2.4. MODELO ENTIDAD/RELACIÓN EXTENDIDO	14
2.5. MODELO RELACIONAL	16
2.5.1. Introducción	16
2.5.2. Estructura de datos relacional	16
2.5.3. Elementos y propiedades del modelo relacional	17
2.5.4. Esquema relacional a partir del Modelo Entidad/relación	18
Recursos complementarios	22
Bibliografía	22
Autoevaluación	23

## 2.1. Introducción

Después de haber aprendido los conceptos básicos sobre las bases de datos, es necesario aprender a diseñarlas e implementarlas de manera correcta. Existen varios tipos de diagramas que nos ayudaran a diseñar una base de datos eficiente. Estos diagramas están pensados por niveles unos más cercanos al entendimiento del usuario, otros más cercanos al nivel físico computacional y diagramas intermedios entre ellos. Los modelos que aprenderemos en esta unidad son:

- Modelo Entidad-Relación
- Modelo Relacional

## 2.2. Diseño de BD

La fase de diseño de una base de datos es primordial para el correcto funcionamiento final de la base de datos y sus aplicaciones clientas. Diseñar una base de datos se puede comparar con el diseño de planos para un edificio, si estos están bien la construcción del edificio será exitosa, de igual manera sucede en el diseño de la base de datos, este diseño es la base para el resultado final de la base de datos.

Diseñar una base de datos es analizar el problema que se quiere solucionar a profundidad y dentro de este análisis discernir qué datos son necesarios y necesitan ser guardados en la base de datos y cuáles no. Una vez determinados los datos esenciales procedemos a generar modelos adecuados con herramientas de apoyo.

Estos modelos nos ayudarán a seguir puliendo el diseño final de nuestra base de datos y trasladar el diseño desde el punto de vista humano hasta el nivel físico computacional.

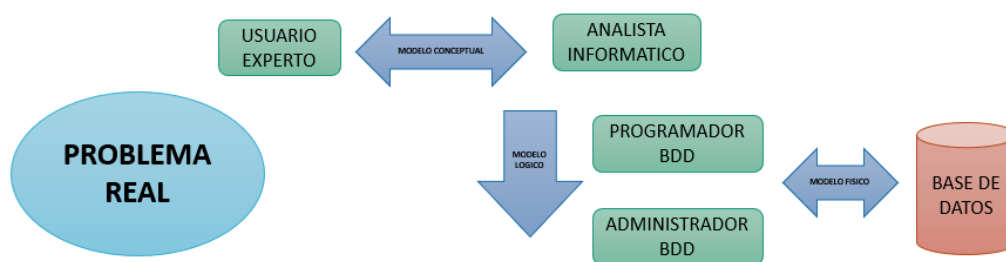
### 2.2.1. Especificación de requerimientos de Software

Antes de generar un modelo se debe tener muy claro y definido las necesidades del usuario final de nuestra aplicación que utilizará la base de datos a ser creada.

En esta fase es donde con mucha pericia el diseñador de la base de datos a través de una entrevista con los usuarios finales descubre las necesidades del cliente y las traslada a un documento formal de Especificación de requerimientos de software. Es fundamental que a través de las preguntas al usuario se pueda definir

todos los objetivos que se espera cumpla la base de datos y definir el alcance y magnitud de esta. El documento resultante de esta fase es el Input para poder generar correctamente los modelos de las fases siguientes, y por eso la importancia de saber trasladar el lenguaje y necesidades del usuario a un documento técnico correcto (E.R.S).

Figura 1 Diseño de una Base de Datos



Nota. La figura muestra el proceso de diseñado de una BD. Creación Propia

### 2.2.2. Diseño – Etapa 1. Diseño Conceptual: Modelo Entidad/Relación (E/R)

**Prerrequisitos:** Documentos de requerimientos de software

**Resultado:** Modelo de datos de alto nivel. Ejemplo: Modelo Entidad Relación

Esta etapa tiene como propósito generar un esquema conceptual de la información dada por el usuario final y del documento de especificación de requerimientos y plasmarlos en estructuras conceptuales con una notación específica. En otras palabras, es diagramar bajo ciertos modelos técnicos la funcionalidad y estructura de nuestra BD, pero este modelamiento debe ser de alto nivel y debe estar desligado del tipo de base de datos o SGBD que se vaya a utilizar. Más adelante profundizaremos sobre esta etapa.

### 2.2.3. Diseño – Etapa 2. Diseño Lógico: Modelo Entidad/Relación

**Prerrequisitos:** Modelo Entidad Relación

**Resultado:** Modelo Lógico Ej. Modelo Relacional

Esta etapa es enfocada en un diagrama más técnico, aquí definimos el tipo de base de datos que se va a utilizar, es decir si va a ser de red, jerárquica, relacional, porque esta fijará el esquema lógico resultante. En este curso nos centraremos en el modelo relacional la cual nos dará como resultado el Modelo Relacional que tendrá como componentes: relaciones, atributos, llaves foráneas y llaves primarias.

Para hacer más fácil el desarrollo de esta fase se la puede dividir en 3 subfases:  
Correcciones al modelo conceptual.

Migración del Modelo ER al modelo lógico (Modelo relacional)  
Normalizar el modelo relacional

#### 2.2.4. Diseño – Etapa 4. Diseño Físico: Modelo Físico

**Prerrequisitos:** Modelo Relacional

**Resultado:** Script creación de la base de datos

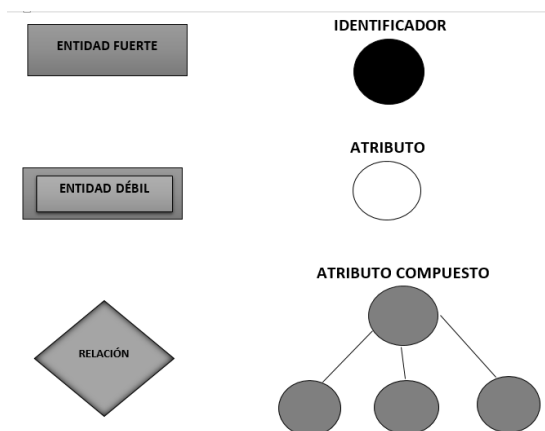
Para desarrollar esta etapa debemos escoger el Sistema de Gestión de Base de Datos que vamos a utilizar, porque el propósito general de esta fase es que el modelo relacional sea adaptado al SGBD, con su sintaxis, sus tipos de datos y restricciones y/o ventajas del SGDB escogido.

Además de darnos el script de la creación de la base de datos, en esta adaptación o transformación del modelo lógico al físico se puede incluir mejoras a nuestro diseño como dividir una relación en varias, aumentar campos calculados a alguna tabla, creación de índices, etc.

#### 2.3. Modelo Entidad/Relación (E/R)

Creado en 1976, este modelo es uno de los más populares. Está basado en dos pilares, por un lado, las *entidades* que son la representación de cada objeto conceptual, y las relaciones que son las diferentes asociaciones entre las entidades. Sus diseños gráficos para construirlos son los siguientes:

Figura 2 Notación Diagrama ER



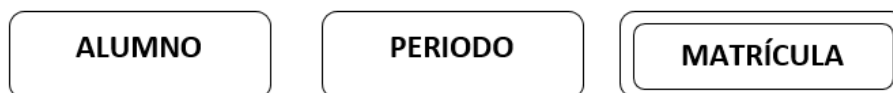
Nota. La figura muestra los elementos para diagramar en modelo ER. Creación Propia

### 2.3.1. Entidades

Definimos entidad como “un objeto o cosa que actúa como una unidad poseyendo una serie de atributos propios y que en conjunto la hacen diferente de otros objetos”. Esta entidad puede ser una representación del mundo abstracto o del mundo real. En el diagrama conceptual no se pueden repetir nombres de entidades, y son representadas con una figura rectangular con el nombre de la entidad dentro.

Hay dos clases de entidades en nuestro diagrama, la entidades fuertes y débiles. Una entidad fuerte es la que existe sin depender de otra entidad y tiene independencia. Si una entidad existe dependiendo que exista otra entidad, esta es llamada entidad débil y se la representa con doble rectángulo. Por ejemplo, en un entorno académico la entidad “Alumno” es una entidad fuerte ya que no depende de otra para existir, pero la entidad “Matrícula” es una entidad débil porque no depende de sí misma para existir, depende que haya un Alumno que se va a matricular y otros factores más.

Figura 3 Entidades modelo E/R



Nota. La figura muestra ejemplos de entidades. Creación Propia

### 2.3.2. Atributos

Cada entidad tiene sus propiedades propias y distintivas de otras entidades, a estas características se las denomina Atributos. Hay atributos que tiene un grupo determinado de valores que pueden tomar, esto es llamado el dominio del atributo. Cuando los atributos tienen sus valores se puede ver los diferentes registros en la misma entidad.

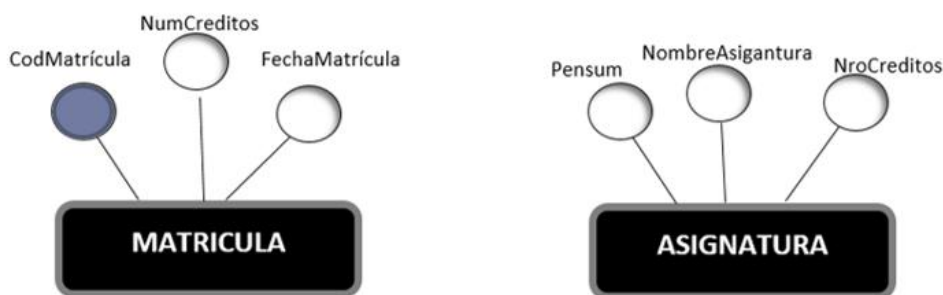
De manera global existen dos tipos de atributos:

**Atributos Identificativos:** son llamados claves o llave primaria, y son los encargados de identificar de manera única a cada registro de la entidad. Cada entidad debe tener al menos un atributo de este tipo.

**Atributos descriptivos:** cuando un atributo describe o muestra características de la entidad se llama descriptivo, pueden repetirse entre diferentes registros.

Los atributos se representan gráficamente por un círculo unido por una línea a la entidad. En caso de ser atributo identificativo el círculo es relleno de negro, mientras que para los otros atributos es un círculo vacío.

Figura 4 Entidades con Atributos



Nota. La figura muestra entidades con sus atributos. Creación Propia

### Clases de atributos

**Atributos identificativos:** Son atributos que tienen un valor único y que su valor no se puede encontrar dos o más veces en la misma entidad. Es el campo que sirve para diferenciar cada registro dentro de la entidad. Por ejemplo, en una entidad persona puede ser la cedula que es única, o en una tabla de usuario es el nombre de usuario o login. Un atributo identificativo también puede ser compuesto por ejemplo en una Entidad Carrera puede ser compuesto del campo código Universidad, código Facultad y código Carrera.

**Atributos descriptivos:** estos atributos son los más encontramos con frecuencia. Son los que describen o muestran una característica de la entidad. En el ejemplo de la entidad Persona podemos encontrar los atributos descriptivos: nombre, género, fecha nacimiento, estatura que son cualidades o características que describen a una persona.

**Atributos compuestos:** Cuando varios atributos de distintos dominios se unen y se componen en otro atributo es llamado atributo compuesto. Un ejemplo de esto puede ser el atributo “dirección domicilio” que puede ser el resultado del atributo “calle primaria” más “numero de casa” más el atributo “calle secundaria”.

**Atributos calculados:** Son atributos que son el resultado de un cálculo entre otros atributos de la entidad, estos atributos no se pueden actualizar ni asignar directamente un valor, porque son derivados o dependientes de otros atributos. Ejemplos: el atributo calculado “IVA” que es el resultado del atributo “subtotal” multiplicado por el atributo “% IVA” o el atributo “Edad” que es el resultado de la diferencia de años entre la fecha actual y el atributo “fecha de nacimiento”.

**Atributos multivalor:** Cuando se necesita varios valores de un dominio dentro de un atributo se convierten en atributos multivalor. Ejemplo: correo electrónico.

Figura 5 Tipos de Atributos

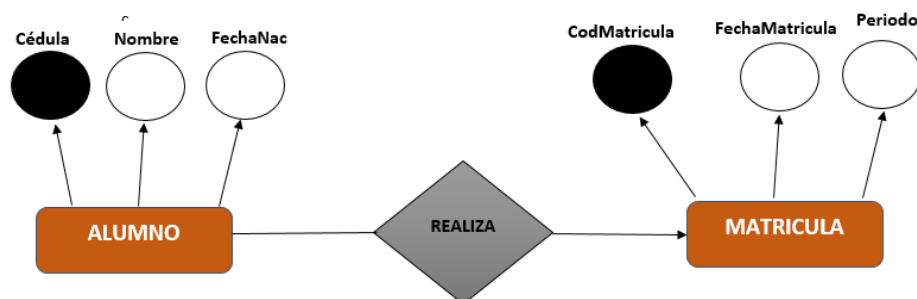


Nota. La figura muestra las clases de atributos en un modelo E/R. Creación Propia

### 2.3.3. Relaciones

Quando hay vínculo, nexo o asociación entre dos o más entidades nacen las “relaciones”. EL nombre de la relación puede ser la letra inicial de las entidades asociadas o la función descriptiva de la relación. Existe otra característica de las relaciones llamada grado de la relación que se define por el número de entidades participantes. Una relación entre dos entidades será una relación de grado 2 o binaria, en caso de que participen 3 entidades será de grado 3 o ternaria.

Figura 6 Relación entre Entidades

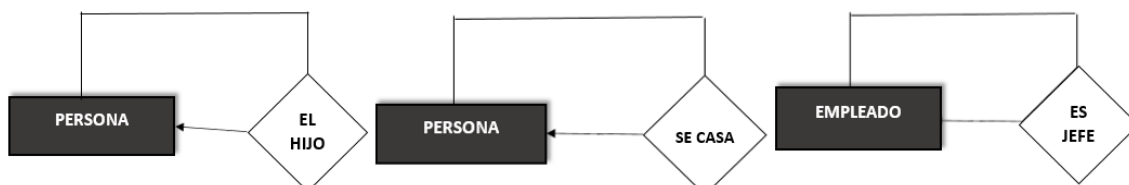


Nota. La figura muestra la relación entre dos entidades. Creación Propia



Hay entidades que están asociadas a sí mismo, estas son llamadas recursivas o reflexivas. Por ejemplo, la entidad “persona” con la asociación “es hijo”: una entidad “persona” es hijo/a de una “persona”. La relación vuelve a la entidad inicial

Figura 7 Entidades Recursivas

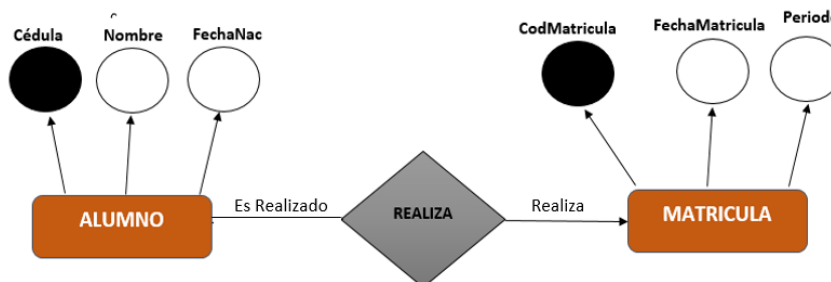


Nota. La figura muestra la relación de entidades consigo mismo. Creación Propia

### Papel de una entidad en una asociación

Sirve para aclarar mejor la función de la entidad en la asociación, es decir el rol de la entidad en la relación. Se escribe sobre el conector de la relación y la entidad. A continuación, el ejemplo anterior incluido con roles de las entidades:

Figura 8 Roles en las relaciones



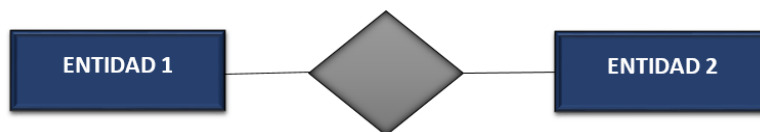
Nota. La figura muestra una relación del modelo ER con sus roles. Creación Propia

### Cardinalidad

“Cardinalidad es el número de entidades con la cual otra entidad se puede asociar mediante una relación binaria”. En una relación binaria que son la mayoría en nuestros modelos existen 3 clases de cardinalidades:

**Cardinalidad 1:1 (uno a uno):** a cada registro de la Entidad 1 le corresponde uno y solo un registro de la Entidad 2 y a cada registro de la Entidad 2 le corresponde solo un registro de la Entidad 1.

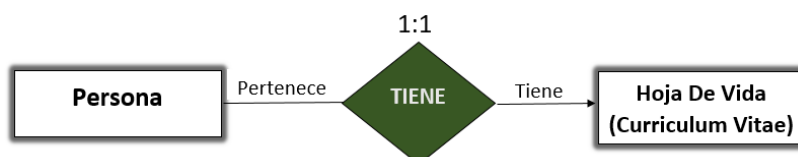
Figura 9 Relación 1 a 1



Nota. La figura muestra la cardinalidad 1 a 1. Creación Propia

Ej.: Una persona tiene una y solo una hoja de vida y cada hoja de vida pertenece solamente a una persona:

Figura 20 Ejemplo relación 1 a 1



Nota. La figura muestra un ejemplo de cardinalidad 1 a 1. Creación Propia

**Cardinalidad 1: n (uno a varios):** a cada registro de la Entidad 1 se relaciona con cero, uno o muchos registros de la Entidad 2 y a cada registro de la Entidad 2 se asocia con un solo registro de la Entidad 1.

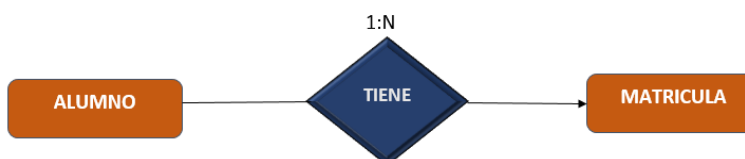
Figura 31 Relación 1 a N



Nota. La figura muestra la cardinalidad 1 a N. Creación Propia

Ej.: Una entidad “alumno” tiene 1 o varias “matriculas”, y una entidad “matricula” corresponde a uno y solo un alumno.

Figura 42 Ejemplo Relación 1 a N



Nota. La figura muestra un ejemplo de la cardinalidad 1 a varios. Creación Propia

**Cardinalidad n:m (muchos a muchos):** Un registro de la Entidad 1 se puede asociar con 1 o muchos registros de la entidad 2 y un registro de la entidad 2 se puede relacionar con muchos registros de la Entidad 1.

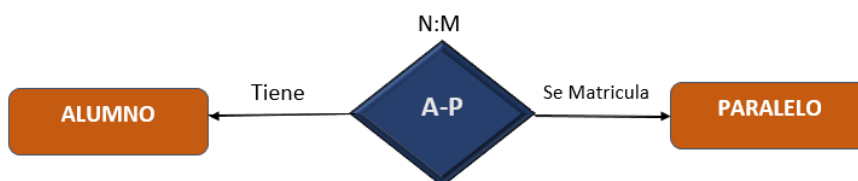
Figura 53 Relación N a M



Nota. La figura muestra la cardinalidad N:M. Creación Propia

Ej.: Un Alumno se matricula en 1 o varios paralelos en un semestre y un Paralelo de una materia puede tener 1 o varios alumnos.

Figura 64 Ejemplo Relación N a M



Nota. La figura muestra ejemplo de la cardinalidad N:M. Creación Propia

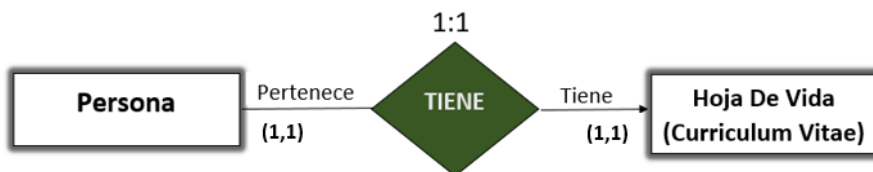
### Restricciones de participación de una entidad en una relación.

Se puede ver que en diferentes asociaciones una entidad puede tener diferentes cardinalidades, a esto lo llamamos participación. La forma de determinar esta participación es fijar un registro o ocurrencia de una relación y deducir cuantos máximo y mínimos registros le corresponden de la otra entidad. Esto se debe realizar en ambos sentidos.

En el diagrama se las representa junto a la entidad que se fija la relación entre paréntesis con el mínimo y el máximo separado por una coma, y la cardinalidad de la relación la obtenemos de los valores máximos de cada participación, y lo colocamos sobre el rombo de la asociación.

Ejemplo 1:

Figura 75 Participación 1 a 1

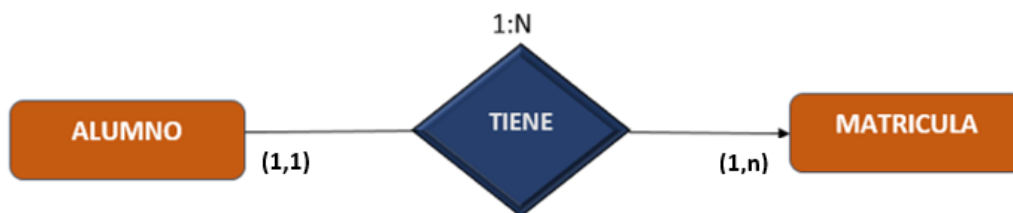


Nota. La figura muestra ejemplo de participación de relación de 1 a 1. Creación Propia

Una persona tiene mínimo un currículum vitae y máximo un currículum vitae, lo que resulta una participación de  $(1,1)$ . En el sentido contrario un currículum pertenece mínimo a una persona y máximo a una sola persona dando una participación de  $(1,1)$ . La cardinalidad la obtendremos sacando los máximos de ambas participaciones será 1:1

Ejemplo 2:

Figura 86 Participación 1:N

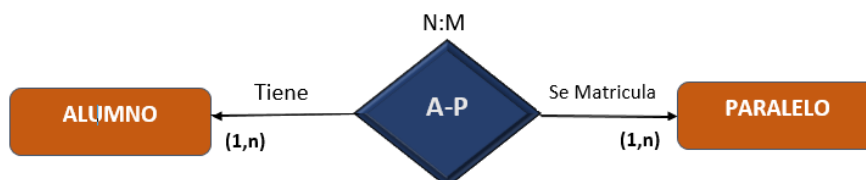


Nota. La figura muestra un diagrama de participación 1: N. Creación Propia

Un alumno puede tener mínimo 1 matrícula y máximo muchas o varias matrículas dando una participación de  $(1, n)$  y una matrícula puede pertenecer mínimo a un alumno y máximo a un solo estudiante dando una participación de  $(1,1)$ . La cardinalidad tomamos los máximos de estas 2 participaciones siendo n el máximo en la primera y 1 en la segunda, cardinalidad igual a 1: N

Ejemplo 3:

Figura 97 Participación N:M



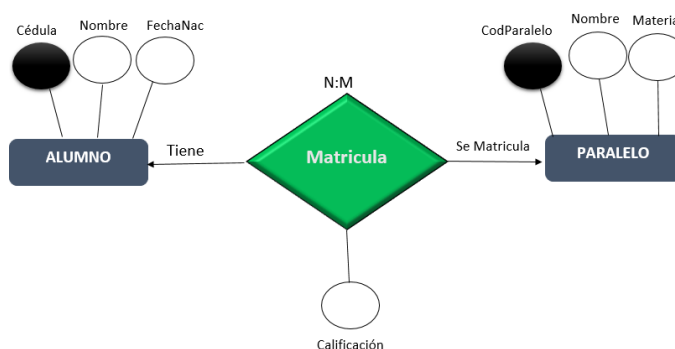
Nota. La figura muestra diagrama de participación N: M. Creación Propia

Un alumno puede matricularse en mínimo 1 paralelo y máximo en muchos paralelos dando participación (1, n); y un paralelo puede tener mínimo un alumno y máximo muchos alumnos dando participación (1, n). La cardinalidad sacando los máximos sería N:N, reemplazada por N:M para evitar confusiones.

## Relaciones con atributos propios

No solo las entidades pueden poseer atributos, también las relaciones pueden tener atributos, porque estos atributos dependen de ambas entidades relacionados y se hacen reales o existen en la mezcla de ambas entidades es decir en la relación.

Figura 108 Relación con Atributos



Nota. La figura muestra ejemplo de un atributo propio de una relación. Creación Propia

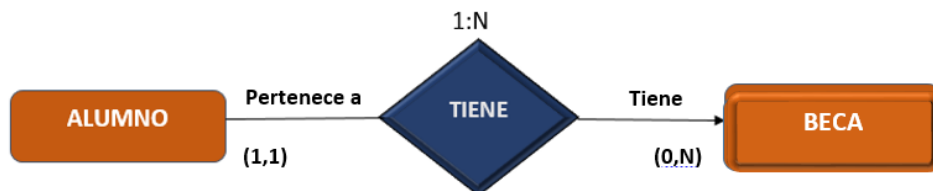
Ej.: Como podemos ver la entidad Alumno tienes sus atributos y la entidad Paralelo también tiene sus atributos, pero el atributo calificación no pertenece directamente a ninguna de las dos, este atributo es posible o real por la existencia y la relación del Alumno y del Paralelo, es decir la calificación existe cuando una entidad “Alumno” se matricula en una entidad “Paralelo”, solo allí existe la calificación.

## Entidades Débiles y Entidades Fuertes, relación de dependencia

Cuando hablamos de las entidades vimos que hay entidades débiles que dependen de entidades fuertes. Dentro de esta dependencia hay dos tipos:

**Dependencia en existencia:** Se da cuando la entidad débil existe solamente porque existe la entidad fuerte, si no estuviera la entidad fuerte la entidad débil no existiría o no tendría sentido. Se representa con una letra E en la asociación.

Figura 119 Dependencia en existencia

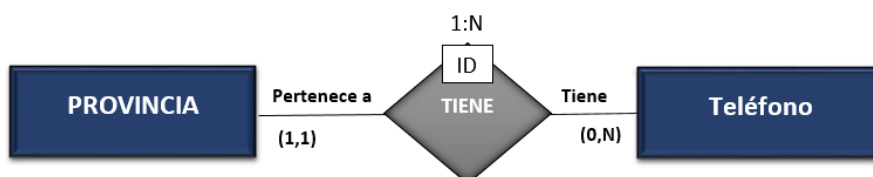


Nota. La figura muestra un ejemplo de dependencia de existencia de entidades. Creación Propia

La entidad débil “Beca” solo tiene sentido si existe una entidad “Alumno”, si el alumno desaparece no puede existir la beca ni tiene sentido una beca sin alumno.

**Dependencia en identificación:** Se da cuando al atributo identificativo de la entidad débil incluye a la entidad fuerte, es decir la entidad fuerte ayuda a identificar a la entidad débil.

Figura 20 Dependencia en Identificación



Nota. La figura muestra un ejemplo de entidades con dependencia en identificación. Creación Propia

Los números de teléfono pueden ser iguales entre diferentes provincias, pero la entidad provincia con su código de provincia los hace únicos. El teléfono 2957434 puede ser el mismo en Guayas que en Pichincha, pero si se une al código de provincia que es la entidad fuerte se hace único, Pichincha (022957434) Guayas (042957434)

## 2.4. Modelo ER Extendido

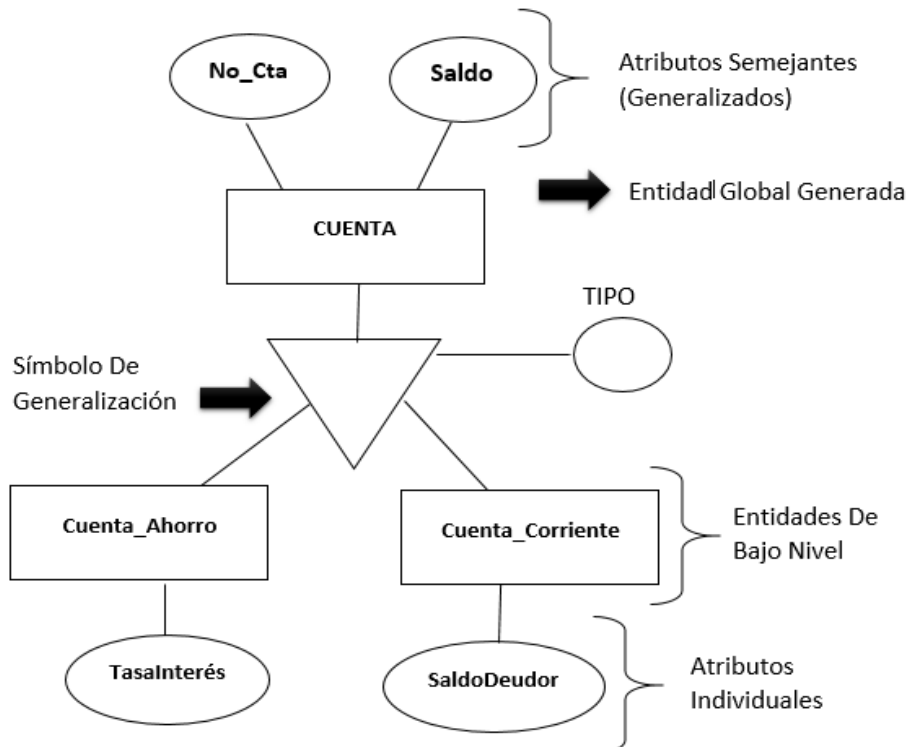
El modelo entidad relación extendido es cuando sobre el modelo ER se aplica la lógica de generalización y especificación similares al proceso de clases y superclases del paradigma Orientado a Objetos, con la finalidad de mejorar y no redundar atributos en varias clases que pueden agruparse todos en una súper clase. Es así como se van creando las relaciones de jerarquía. Para realizar esta

jerarquización debemos hacer el proceso de generalización y el proceso de especialización de las entidades/clases.

Se hace la abstracción de la relación de nivel más alto (súper clases) con todos los atributos que son compartidos por las clases inferiores de la jerarquía, es decir las subclases. Estas subclases heredarán todos los atributos de la clase generalizadora. A nivel de modelo ER la superclase se llamará superentidad y las hijas subentidades. En el diagrama para sustituir todos los rombos de la relación entre la superentidad y cada subentidad se pone un triángulo invertido que significa “es un”.

El ejemplo que haremos para este caso será de cuentas bancarias, todas las cuentas bancarias del tipo que sean tienen atributos comunes y de ahí que saldrá nuestra superentidad “Cuenta” que tendrá los atributos de todas las cuentas como son: número de cuenta y saldo. Las subclases serán las cuentas de ahorro por un lado y cuentas corrientes por otro lado, que heredarán los atributos de la superentidad e incluirán atributos propios de cada subclase:

Figura 20 Modelo E/R Extendido



Nota. La figura muestra un ejemplo del modelo ER extendido y con jerarquías.  
Creación Propia

## 2.5. Modelo Relacional

### 2.5.1. Introducción

La “relación” es la estructura básica y principal de nuestro modelo tanto ER como el modelo relacional de este subtema. Este es el modelo más usado y su popularidad se da porque está basado en una teoría ampliamente estudiada y llena de fundamentos como: el álgebra relacional.

El modelo relacional tiene 3 ejes sobre cual se basa: escritura y lectura de los datos, la estructura de los datos que representen la abstracción del entorno real, y la integridad de los datos.

Para lograr cumplir todos estos objetivos del modelo, entra en ayuda el SGBD, que pueden haber de muchos tipos según el tipo de base que se escogió (red, jerárquica, relaciona, OO). Para nuestro estudio hemos escogido cualquier SGBD relacional que nos ayudara a cumplir con la estructura y tipos de datos, las operaciones de lectura/escritura y la integridad de los datos.

La migración del modelo ER al modelo relacional se da básicamente en cada entidad del ER pasa a ser una tabla en el modelo relacional y los atributos del ER viene a ser campos/columnas de la tabla en el modelo relacional.

### 2.5.2. Estructura de datos relacional

Si en el modelo relacional tenemos la entidad Alumnos en el modelo relacional tendremos la tabla Alumno con sus campos que describen las cualidades de un alumno, y cada registro de esta tabla corresponderá a un alumno en concreto.

Figura 21 Entidad en Modelo Relacional

ALUMNO	
• Cédula	PK
○ Nombre	
○ Apellido	
○ Nacimiento	
○ Sexo	
○ Estado Civil	

Nota. La figura muestra la entidad alumno en el modelo relacional. Creación Propia

Una tabla/relación en el modelo relacional es la descripción de la tabla, que incluye los campos que la describen y el nombre de la tabla. Por ejemplo, para la entidad Universidad, la tabla en el modelo relacional quedaría de la siguiente manera.



Figura 22 Tabla en el modelo Relacional

UNIVERSIDAD	
CodUniversidad	PK
Nombre	
Ruc	
RazónSocial	
Dirección	
NombreRector	
CédulaRector	
Teléfono	

Nota. La figura muestra la entidad Universidad en notación del modelo relacional.  
Creación Propia

La llave primaria será el campo o campos (una clave primaria puede ser compuesta de varios campos) que nos ayuden a distinguir un registro de otro unívocamente. En el ejemplo de los alumnos la cédula es única en la tabla y diferencia a cada registro o tupla.

Figura 23 Registros de una Entidad

### ALUMNOS

Cédula	Nombre	Apellido	Nacimiento	Sexo	Estado Civil
1714365663	Tania	Moya	4/08/1978	M	Casada
1751583608	Daniel	Ortiz	14/01/2002	H	Soltero
1722707773	Sebastián	Vásquez	18/01/1999	H	Soltero

Nota. La figura muestra el ejemplo de filas o registros en la tabla alumnos. Creación Propia

### 2.5.3. Elementos y propiedades del modelo relacional

**Tabla (relación):** Cada entidad u objeto real o abstracto que se quiere almacenar en la base de datos:

**Tuplas:** es cada instancia u ocurrencia de la entidad en la tabla. También son llamados filas o registros.

**Campos:** Son las características de la tabla o Entidad. También son llamados atributos o columnas.

Las características de las tablas son:

En una misma base de datos solo hay una tabla con el mismo nombre, no puede haber dos tablas con el mismo nombre.

Dentro de una tabla cada columna tiene un nombre y no se puede repetir en la misma tabla.

En una tabla cada registro es único y mínimo se diferencia de cualquier otro por su valor de la llave primera.

**Clave candidata:** UN atributo o conjuntos de atributos que son únicas en cada registro de la tabla, puede haber varios. Por ejemplo, en la tabla alumnos podría ser la cedula o un campo autonumérico creado como código de alumno. Las dos serian claves candidatas.

**Llave Principal:** La llave primaria es cuando se escoge una de las claves candidatas como clave principal.

**Llave Alternativa:** Son las claves candidatas que no fueron tomadas como llaves primarias.

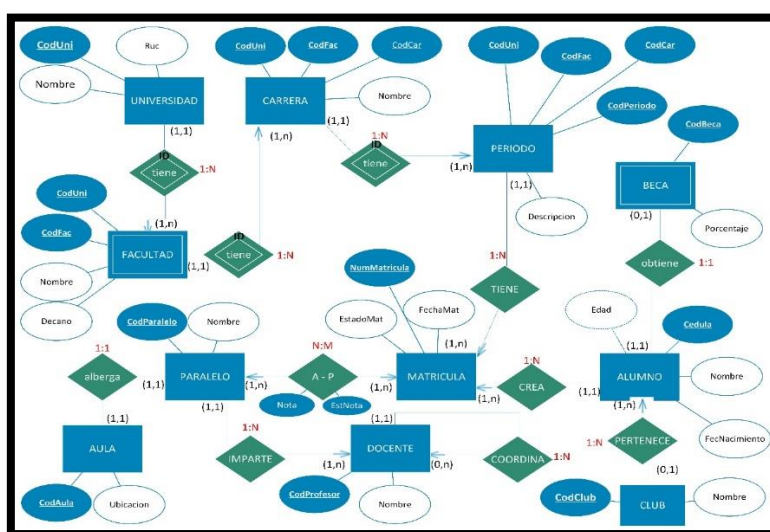
Por integridad de datos las claves primarias no pueden tener valor nulo.

**Clave Externa:** En una tabla B la clave externa o foránea son el/los atributo(s) que son clave primaria en otra tabla A, quedando así relacionadas ambas tablas. Para mantener la integridad de datos, en la tabla b no se puede introducir la clave foránea si previamente no está creada en la tabla principal A.

## 2.5.4. Esquema relacional a partir del modelo Entidad/Relación

Hay varios pasos para pasar del modelo entidad relación a un modelo relacional. Este E/R nos servirá para revisar este proceso:

Figura 24 Modelo Entidad Relación



Nota. La figura muestra un ejemplo de un modelo E/R semicomplejo para el ámbito educativo. Creación Propia

## Tablas (Relaciones)

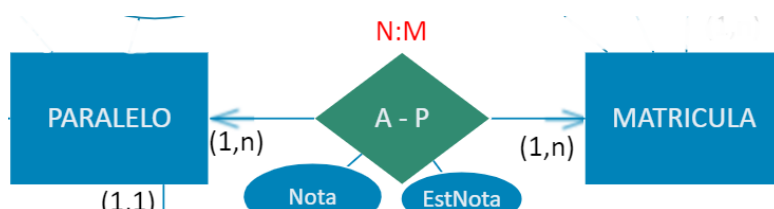
Cada entidad del modelo ER de ejemplo se convertirán en una tabla en el modelo relacional. Cada atributo identificativo pasará a ser la clave primaria de la tabla y debe ir en **negrita subrayada** y con la nomenclatura de Primary Key (PK). Ejemplo de entidad Universidad cambiado a modelo relacional:

### Asociaciones Binarias (2do grado)

**Relación muchos a muchos (N:M).** Esta relación en el modelo ER generará una tabla adicional a las dos entidades participantes. Esta tabla nueva tendrá como Foreign Key (claves foráneas) las claves primarias de ambas tablas iniciales. Y la clave primaria de esta nueva tabla será la unión de estas claves foráneas. Hay que recalcar que no son dos claves primarias que se generarán sino una clave primaria compuesta por las dos foreign key de las tablas iniciales. Las columnas de esta nueva tabla serán los atributos de la “relación” entre las dos entidades del diagrama ER.

*Ej.:*

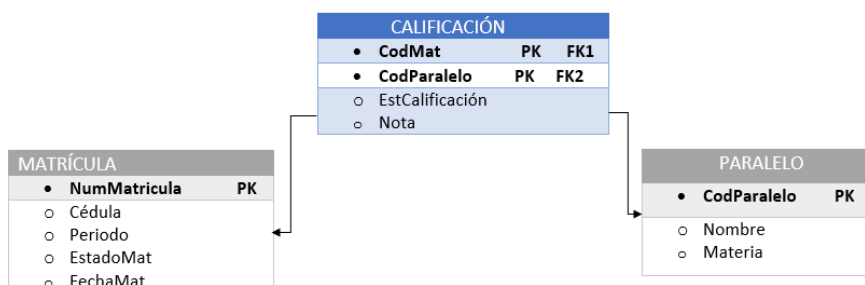
Figura 25 Relación muchos a muchos modelos ER



Nota. La figura muestra la relación de varios a varios entre Entidades.  
Creación Propia

En esta relación N:M se debe generar una nueva tabla en el lugar del rombo de la asociación. La nueva tabla se llamará “Calificación” y tendrá como llaves foráneas el CodParalelo y el NumMatricula que son PK en las entidades iniciales. La clave primaria de la nueva tabla será una llave compuesta ( codParalelo, NumMatricula) y los atributos de la nueva tabla serán “Nota” y “estNota” que son los atributos de la relación. El resultado será el siguiente:

Figura 26 Resolución relaciones N:M

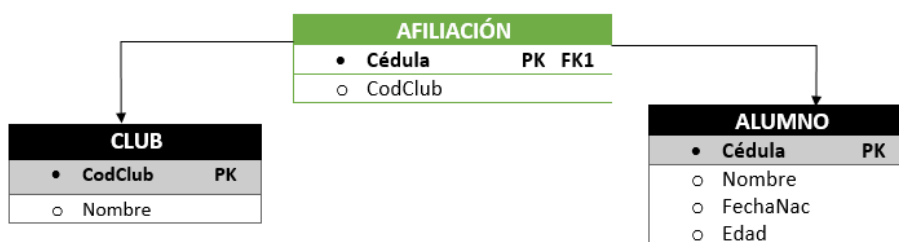


Nota. La figura muestra la tabla nueva “Calificación” que resuelve la relación N:M.  
Creación Propia

**Relación uno a muchos (1:n).** Este tipo de asociación tiene dos tipos:

**Tipo 1:** Cuando la participación es (0,1) en la entidad del lado de cardinalidad 1. En este caso se deberá crear una tabla nueva con las claves foráneas las claves primarias de cada entidad. Y la llave primaria de la nueva tabla será la clave foránea que viene del lado N de cardinalidad. En nuestro ejemplo de ER este caso se da en la relación entre Alumno y Club. Un club puede tener varios alumnos y un alumno puede pertenecer a ningún club o máximo 1 (participación 0,1). Ejemplo de cómo quedaría en el modelo relacional:

Figura 27 Relación 1 a Varios



Nota. La figura muestra Relación 1 a varios con cardinalidad (0,1). Creación Propia

**Tipo 2:** Para todos los demás casos de uno a varios, que son los más comunes. Para diagramarlos, simplemente la tabla de la cardinalidad N recibe como Foreign Key a la primary key de la entidad con cardinalidad 1. Como ejemplo en nuestra modelo ER tenemos la relación Periodo – Matricula. En un periodo académico puede haber 1 o muchas matrículas. Lo que haremos es pasar la primary key de la tabla Periodo como foreign de la tabla Matrícula.

Figura 28 Relación 1 a N



Nota. La figura muestra las relaciones 1:N de tipo 2. Creación Propia

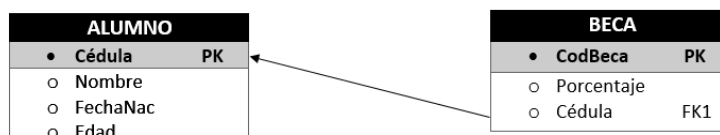
**Relaciones uno a uno (1:1).** Existen dos casos para este tipo y los dos se solucionan de igual manera.

**Tipo 1:** La participación de la entidad A es (0,1) y de la Entidad B es (1,1).

**Tipo 2:** La participación de la entidad A es (1,1) y de la Entidad B es (1,1)

En ambos casos a la Entidad que tenga participación mínima igual a 1 se lleva como llave foránea la llave primaria de la otra entidad. En el tipo 1 sería a la Entidad B y en al tipo 2 a cualquiera de las dos la entidad A o la entidad B. En nuestro diagrama ER principal tenemos este caso en la relación Alumno-Beca, ya que un alumno puede tener 0 o 1 beca Máximo y una beca solo puede pertenecer a un alumno. La resolución al modelo relacional sería la siguiente:

Figura 29 Relación uno a uno



Nota. La figura muestra las relaciones uno a uno. Creación Propia

**Relaciones de Dependencia identificativas.** Se dan en relaciones uno a varios (1:N) y no generarán tablas extras como es común. Generalmente esta clase se dan entre una entidad fuerte y una entidad débil. Se soluciona introduciendo como primer atributo de la entidad débil a la primary key de la entidad fuerte. La primary key de le entidad débil será la foreign key traída de la entidad fuerte mas las primary key de la entidad débil es decir una clave primaria compuesta.

Ej.: La entidad fuerte “Universidad” con la entidad débil “Facultad”. Como es una relación identificativa la primary key de la entidad débil va a estar formada por el codUni proveniente de la entidad fuerte y el codFac proveniente de la entidad débil:

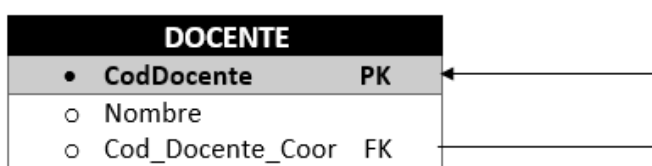
Figura 30 Relación Identificativa



Nota. La figura muestra tablas con relación de dependencia identificativa. Creación Propia

**Relaciones recursivas.** La primary key está presente dos veces en la tabla, una como llave primaria y otra como llave ajena. Como los nombres de las columnas no se pueden repetir la columna de la llave foránea debe ser renombrada de preferencia a un nombre descriptivo de su función. En la tabla profesor tenemos este ejemplo, pues un profesor puede ser coordinador de otro profesor. La clave foránea nueva se llamará algo relacionado a coordinador:

Figura 31 Relación Reflexiva



Nota. La figura muestra una tabla recursiva. Creación Propia

### Recursos complementarios

- En la dirección encontrará un video sobre Relaciones en Base de Datos  
Dirección URL: <https://www.youtube.com/watch?v=P7dAj64b6m4>
- En la dirección encontrará un video sobre ejemplos de MER en Base de Datos  
Dirección URL: <https://www.youtube.com/watch?v=BqS-1KNakoc>

### Referencias

- Kroenke, D. P. (2003). *Procesamiento de Base de Datos*.  
Piattini, M. (2007). *Tecnología y Diseño de Base de Datos*. Alfaomega.