

Tema 3:

MODELO RELACIONAL-DISEÑO LÓGICO

1. Introducción

El diseño lógico consiste en transformar el esquema conceptual (ENTIDAD-RELACIÓN), que se encuentra descrito usando un cierto modelo de datos, en estructuras y transacciones descritas en términos del modelo de datos en el cuál se base el sistema de gestión de bases de datos que se vaya a utilizar. Ya que el modelo de datos elegido es el modelo relacional, se deberá realizar este proceso para obtener finalmente un esquema relacional y un conjunto de transacciones sobre las relaciones de este esquema. Este proceso se dividirá en dos fases:

Transformación de los aspectos estáticos del esquema conceptual: para ello se verá cómo es posible transformar cada una de las estructuras de un diagrama Entidad-Relación, modelo utilizado en el diseño conceptual, en relaciones. Esta transformación dará lugar a un primer esquema relacional.

2) Transformación de los aspectos dinámicos del esquema conceptual: para ello se verá cómo se obtienen transacciones sobre las relaciones del esquema relacional a partir de las transacciones del esquema conceptual.

En algunos casos puede suceder que haya varios esquemas relacionales posibles para un mismo esquema conceptual; el criterio de elección que se aplicará cuando esto suceda es el siguiente: "Elija el esquema con menos restricciones de integridad añadidas. Ante igualdad de restricciones, elija el esquema con menos relaciones". Este criterio se justifica por el hecho de que las restricciones de integridad suponen, usualmente, controles costosos en tiempo; por otra parte, cuantas menos relaciones tenga el esquema más eficientes serán las operaciones de consulta. Como ya se verá más adelante sólo en algunos casos muy concretos puede no seguirse este criterio.

2. Estructura modelo Relacional

El elemento básico del Modelo Relacional es la entidad, que será una tabla o matriz bidimensional con unas características o restricciones que comentaremos más adelante.

EMPLEADO

Dni	Nombre	Dirección	Teléfono	Sueldo	Fecha_n
18.876.543	Llopis Bernat, Jaume	C/ Artana, 3	964-213243	200.000	7-12-55
18.900.111	Garrido Vidal, Rosa	C/ Herrero,54	964-253545	200.000	25-1-58
18.922.222	Nebot Aliaga, Carme	C/ Sant Vicent, 5	694-216191	150.000	8-6-59
18.932.165	Folch Mestre, Pilar	C/Palància, 22	964-234567	300.000	8-6-60

TEMA 3: MODELO RELACIONAL BDA-DAM

18.933.333	Peris Andreu, Joan	C/ Balmes, 3	964-223344	200.000	15-3-60
18.934.567	Sebastià Broch, Ferran	C/ Magallanes,38	964-281706	250.000	14-7-62
18.944.444	Garcia Tomàs, Alícia	C/ Amunt,15	964-205080	250.000	10-5-64

Las filas, donde tenemos la información de las ocurrencias, de los individuos, se denominan **TUPLAS** (en ocasiones por similitud con ficheros, también se llaman **REGISTROS**).

Las columnas, que serán características que nos interesen de los individuos y que en cada tupla toma un valor, los llamaremos **ATRIBUTOS** (o **CAMPOS**).

El conjunto de valores posibles que puede tomar un atributo determinado se llama **DOMINIO**. Posteriormente veremos que los dominios se intentaran definir lo mejor posible, para prevenir errores.

3. Restricciones

Igual que en otros modelos de datos, en el Modelo Relacional existen restricciones, es decir, estructuras o ocurrencias no permitidas.

Estas restricciones pueden ser de dos tipos fundamentales: restricciones inherentes, que son impuestas por el propio modelo, y restricciones de usuario (también denominadas restricciones semánticas) en las cuales es el usuario quien prohíbe, porque el Modelo se lo permite, determinadas circunstancias.

RESTRICCIONES INHERENTES.

Como hemos dicho son las que impone el propio modelo. Algunas son características que han de cumplir las relaciones. Podemos considerar las siguientes:

Valores atómicos: cada valor de la tabla, es decir, cualquier valor de cualquier atributo de cualquier tupla ha de ser simple, no divisible. Por tanto no valen atributos compuestos o repetitivos.

Así, si consideramos Nombre en la relación Empleado como nombre de pila más apellidos, no será divisible (no podré tomar posteriormente el nombre de pila por un lado y los apellidos por otra; si lo quisiera hacer, se habrían de definir los atributos simples Apellido1, Apellido2 y Nombre). Tampoco valen valores repetitivos, Por ejemplo un vector de 12 entradas. Ni los valores que se puedan derivar unos de otros mediante cálculos.

Tuplas distintas: no pueden haber dos tuplas iguales. Esto es una diferencia con respecto a las tablas matemáticas donde sí que se pueden duplicar filas.

Clave Candidata: Es aquel atributo o conjunto de atributos de una tabla que cumplen las condiciones de identificación única y minimalidad (no existe ningún subconjunto de atributos dentro de la clave candidata que cumpla la condición de identificación única)

Clave Principal: ha de existir una clave principal o primaria que identificará de forma unívoca las tuplas. Por tanto, no podrá tomar valores nulos y tampoco podrá repetirse. Puede estar formada por un atributo o por más de uno.

TEMA 3: MODELO RELACIONAL

BDA-DAM

Clave Alternativa: Son el resto de claves que son claves candidatas y no son clave principal.

El orden de las tuplas no es significativo.

El orden de los atributos no es significativo.

4.1 Restricciones de usuario

También llamadas Restricciones semánticas, serán condiciones que podremos poner para que el esquema de la B.D. refleje lo mejor posible la realidad.

Restricción de dominio: El valor de un atributo ha de ser un valor atómico del dominio. Definiendo claramente el dominio nos aseguramos (dentro de lo posible) que el atributo no pueda tomar valores incorrectos.

Por una parte, el dominio será de un tipo determinado, escogido de una gama suficientemente extensa: entero corto, entero, entero largo, real, doble precisión carácter, cadena de caracteres (texto), fecha, hora, ... Así, por ejemplo, definiendo el Dni como entero largo impediremos que por error pueda tomar el valor 18.934.R57, o que la fecha de nacimiento, con dominio de tipo fecha, sea 15-14-1958 o 31-2-1958.

También se podrán definir dominios que estén en un determinado intervalo (nota de un examen: 0-10) o de un tipo enumerado (nota de evaluación: MD, IN, SUF, BE, NOT, EXC).

Por ejemplo, Empleado podría quedar:

Empleado (Dni: entero(8); Nombre: carácter(30); Dirección: carácter(30); Teléfono: entero(9); Sueldo: entero(6); Fecha_n: fecha)

Restricción de clave: Permite declarar un atributo o un conjunto de atributos como CLAVE PRINCIPAL o PRIMARIA (Primary Key).

Vimos que la obligación de declarar una clave principal era una restricción inherente. Lo que es una restricción de usuario es la elección de la clave principal, y la consecuencia de que no podrá tomar valores nulos ni repetidos.

Estas últimas características también las podrán tener otros atributos, separándolas:

Unicidad (UNIQUE): no se puede repetir, clave alternativa.

Obligatoriedad (NOT NULL): no puede tomar un valor nulo.

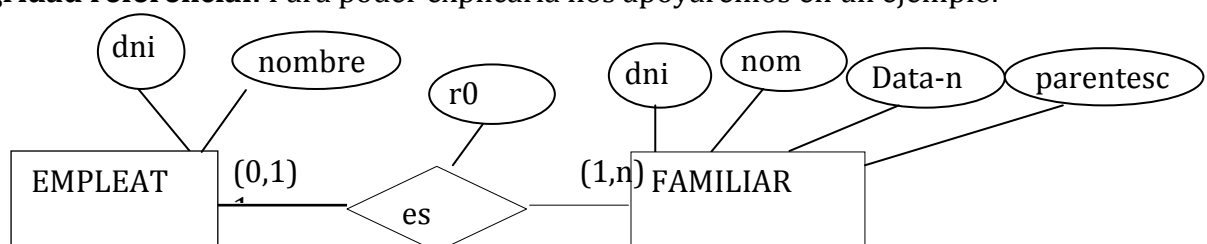
Podremos hacer que un atributo cumpla una o las dos. Por ejemplo el nombre podríamos hacerlo NOT NULL (y UNIQUE si no hay dos empleados con el propio Nombre).

En Access:

NOT NULL :Requerido

UNIQUE : Indexado sin duplicados

Integridad referencial: Para poder explicarla nos apoyaremos en un ejemplo.



TEMA 3: MODELO RELACIONAL BDA-DAM

que se podría traducir en las siguientes tablas:

EMPLEAT	
Dni	Nom
18.876.543	Llopis Bernat, Jaume
18.900.111	Garrido Vidal, Rosa
18.922.222	Nebot Aliaga, Carme
18.932.165	Folch Mestre, Pilar
18.933.333	Peris Andreu, Joan
18.934.567	Sebastià Broch, Ferran
18.944.444	Garcia Tomàs, Aïcia

FAMILIAR			
Dni_emp	Nom	Data_n	Parentesc
18.876.543	Jaume Llopis Domènech	01/06/85	Fill
18.900.111	Felip Gomis Pitarch	31/03/57	Conjuge
18.932.165	Josep Serra González	09/05/60	Conjuge
18.932.165	Xavier Serra Folch	05/04/90	Fill
18.933.333	Silvia Peris Prades	17/07/95	Filla
18.933.333	Silvia Prades Arnau	15/03/62	Conjuge
18.944.444	Andreu Garcia Torró	15/09/22	Pare
18.944.444	Laura Almela Garcia	05/12/92	Filla
18.944.444	Lluís Almela Garcia	10/05/94	Fill
18.944.444	Marc Almela Tomeu	25/03/64	Conjuge

Si en una tabla R2 (**Familiar**) tenemos un atributo (**Dni_emp**) que es clave (primaria o Alternativa) de otra tabla R1 (**Empleado** [**Dni**], todo valor de ese atributo ha de concordar con un valor de la clave de R1 (no he de poder poner en familiar un Dni que no lo tenga ningún empleado de la empresa). el atributo en R2 es, por tanto, una **CLAVE AJENA**. Ha de ser imposible poner en **Familiar** el Dni 18.754.321, porque no está en el otro, y por tanto no es un Dni de un empleado de la empresa.

Las tablas R1 y R2 no tienen por qué ser distintas, pueden ser la misma. Así, si consideramos el supervisor, este ha de ser de la empresa:

Supervisor es una clave ajena, pero de la misma tabla. No todos los SGBD permiten una

EMPLEADO						
Dni	Nombre	Dirección	Teléfono	Sueldo	Fecha_n	Supervisor
18.876.543	Llopis Bernat, Jaume	C/ Artana, 3	964.213.243	200.000	7-12-55	18.933.333
18.900.111	Garrido Vidal, Rosa	C/ Herrero, 54	964.253.545	200.000	25-1-58	18.932.165
18.922.222	Nebot Aliaga, Carme	C/ Sant Vicent, 5	694.216.191	150.000	8-6-59	18.944.444
18.932.165	Folch Mestre, Pilar	C/ Palància, 22	964.234.567	300.000	8-6-60	0
18.933.333	Peris Andreu, Joan	C/ Balmes, 3	964.223.344	200.000	15-3-60	18.944.444
18.934.567	Sebastià Broch, Ferran	C/ Magallanes, 38	964.281.706	250.000	14-7-62	0

clave ajena reflexiva (Access, por ejemplo, no lo puede hacer).

Una manera de representar las claves ajenas en el esquema es la siguiente

Empleado (Dni, Nombre, Dirección, Teléfono, Sueldo, Fecha_n)

Familiar(Dni_emp, Nombre, Parentesco)

Esto nos impedirá que introducimos valores no correctos, no existentes. ¿Pero que pasará si borramos un empleado, o si modificamos su Dni? ¿Qué hacemos con los familiares? Pues en principio tres podrían ser las acciones a realizar:

No dejar borrarlo o modificarlo (**NO ACTION**).

Borrar también los familiares o cambiar los datos **en cascada (CASCADE)**.

Cambiar el valor de la clave ajena al **valor nulo (SET NULL)** o un valor predeterminado. En el ejemplo de los familiares no tiene sentido ya que no nos interesan los familiares de los que no son de la empresa. Pero imaginemos, por ejemplo, un proveedor que nos ha proporcionado unos artículos. Por el hecho de no trabajar ya con el proveedor y quitarlo de

TEMA 3: MODELO RELACIONAL

BDA-DAM

la B.D. no habríamos de eliminar los artículos. Sería suficiente con dar un valor nulo al proveedor de este artículo.

Hay SGBD que hasta permiten acciones distintas para el caso de borrado y de actualización de la clave.

Otras restricciones: Los SGBDR más avanzados, más potentes, permiten otras restricciones consistentes en comprobar una determinada condición después (o antes) de una actualización. Tendremos dos tipos de actuación, según la acción que se realiza en caso que la condición sea cierta:

Verificación (CHECK): si la condición no se cumple después de la actualización, se rechaza esta. Sirve muy bien para definir muy claramente un dominio, entre otras cosas. Por ejemplo:

CHECK Sueldo > 0

CHECK (Año(Fecha_n) < Año(hoy)) and ((Año(hoy)- Año(Fecha_n)) < 65)

Disparador (TRIGGER): si se cumple la condición se ejecuta un procedimiento definido por el usuario. este concepto es muy potente, ya que da una respuesta procedimental donde se puede hacer cualquier cosa.

En resumen, podemos considerar que una B.D. Relacional es un conjunto de tablas sobre las que se han definido una serie de restricciones que garanticen la coherencia de la información almacenada. Pero uno de los objetivos de las B.D. era eliminar la redundancia. este aspecto, que aún no lo hemos tratado, lo solucionaremos mediante el proceso de **NORMALIZACIÓN**.

Cod_Art	Descripción	Cantid.	Codi_prov	Nom_prov	Dirección	Teléfono
01	Disquet	5	32	Distr. Garcia	C/ Ample, 32	964-224466
02	Cartutx	2	32	Distr. Garcia	C/ Ample, 32	964-224466
03	CD-ROM	10	48	Inf. Ascencior	C/ Corrents, 348	964-203040
04	Paper	2	32	Distr. Garcia	C/ Ample, 32	964-224466

Ejemplo:

En este ejemplo hay mucha redundancia, con los inconvenientes de que ocupa mucho espacio, y si se ha de actualizar alguna información (por ejemplo la de un proveedor), se habrá de actualizar en muchos sitios.

La manera de conseguir una B.D relacional normalizada será:

Realizar primero el esquema mediante el M E/R (modelo entidad relación).

A partir de este construir el esquema relacional.

Finalmente lo normalizaremos.

4. Transformación del Modelo E/R al Relacional

A continuación veremos las reglas de transformación de el esquema en el Modelo E/R al Modelo Relacional

4.1 Entidades

Cada tipo de entidad se convierte en una tabla. Cada tabla que corresponda a un tipo de entidad llevará su mismo nombre.

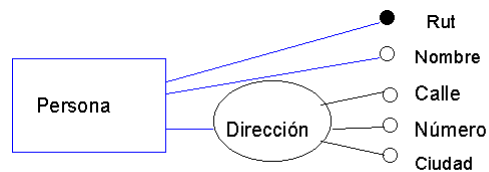
4.2 Atributos

Atributos compuestos y multivaluados.

El modelo relacional admite sólo atributos simples, atómicos.

Cada atributo compuesto se puede transformar según las siguientes dos alternativas:

Eliminar el atributo compuesto considerando todos sus componentes como atributos individuales, o eliminar los componentes individuales y considerar el atributo compuesto entero como un sólo atributo.



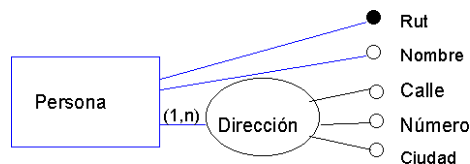
Esquemas relacionales:

Persona (Rut, Nombre, Calle, Número, Ciudad)

ó

Persona (Rut, Dirección)

Los atributos multivaluados requieren la introducción de relaciones nuevas; cada atributo polivalente distinto requiere una relación en la cual pueda estar representado como atributo atómico. La nueva relación contiene el atributo multivaluado más el identificador de la entidad original; el identificador de la nueva relación es el conjunto de todos sus atributos.



Esquema relacional:

Persona(Rut, Nombre)

Dirección(Rut, Calle, Número, Ciudad) con Rut clave ajena que referencia a Persona.

Atributos identificadores.

El o los atributos identificadores principales de cada tipo de entidad pasan a ser la **clave primaria** de la tabla. Se usa la cláusula **PRIMARY KEY**.

Respecto a los identificadores alternativos, son **clave alternativa** se utiliza la cláusula **UNIQUE**.

Otros atributos.

Los atributos no identificadores pasan a ser columnas de la tabla, las cuales tienen permitido tomar valores nulos, a no ser que se indique lo contrario.

4.3 Relaciones

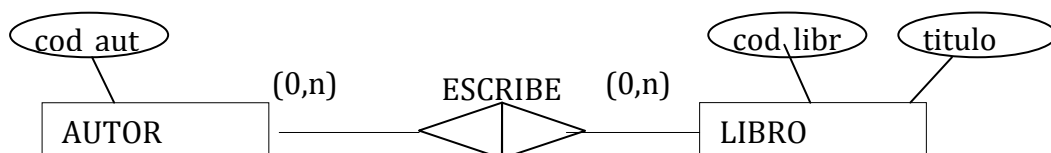
Dependiendo del tipo de correspondencia de la relación, variará la manera de realizar la transformación del esquema.

Relaciones Binarias N:M (Muchos a Muchos)

Un tipo de relación muchos a muchos se transforma en una tabla que tendrá como clave primaria la concatenación de los identificadores de los tipos de entidad que asocia.

Cada uno de los atributos que forman la clave primaria de una tabla derivada de un tipo de relación N:M son clave ajena respecto de cada una de las tablas derivadas de los tipos de entidad que relaciona. Esto se especifica a través de la cláusula FOREIGN KEY dentro de la sentencia de creación de la tabla.

El esquema relacional resultante:



Autor(cod autor)

CP{cod-autor}

Libro(cod-libro, título)

CP{cod-libro }

Escribe(cod-libro, cod-autor)

CP{cod- libro, cod-autor }

CAj{ cod-libro} referencia a Libro (cod_libro)

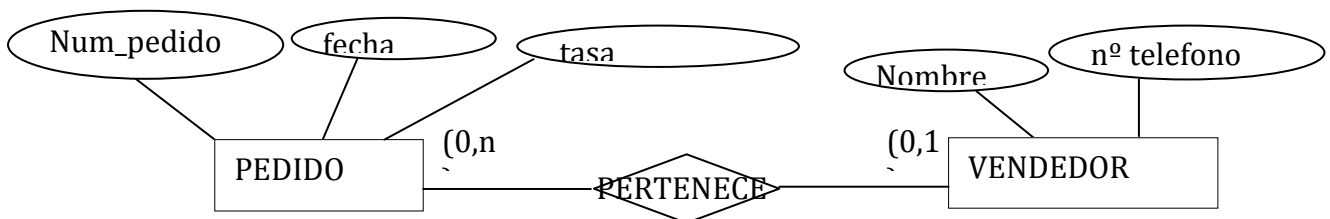
CAj {cod-autor} referencia a Autor (cod_autor)

TEMA 3: MODELO RELACIONAL BDA-DAM

Relaciones Binarias 1:N (Uno a Muchos)

La entidad del lado de "muchos" tiene una participación parcial. (No existe restricción de existencia).

Los pedidos pueden hacerse por medio de vendedores, en cuyo caso se aplica una tasa de descuento, y también directamente sin vendedores (sin aplicar una tasa de descuento). De este modo, existe la posibilidad de valores nulos de nombre-vendedor y tasa-descuento en

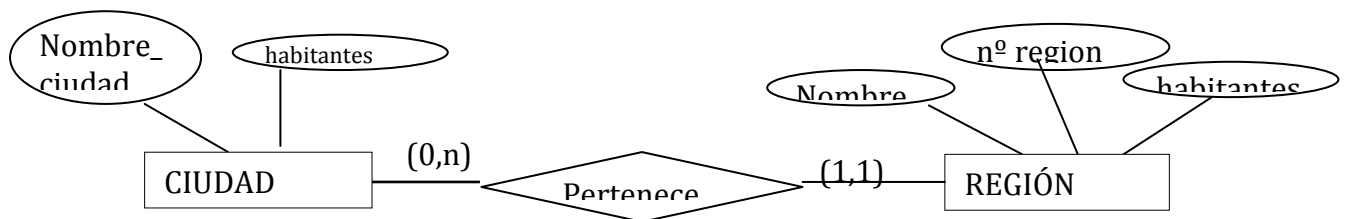


relación a Pedido si se usa el siguiente esquema:

Pedido(Num-Pedido, fecha, nombre-vendedor, tasa descuento)
CP{Num_pedido}
CAj{nombre-vendedor} Referencia a Vendedor(num_pedido)
Vendedor(Nombre, telefono)
CP{Nombre}

Relaciones Binarias 1:N (Uno a Muchos) Con restricción de existencia

La relación binaria con una restricción de existencia sobre la entidad de cardinalidad máxima Muchos.

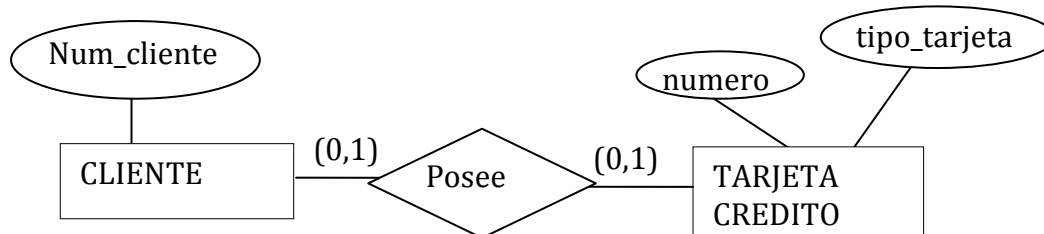


El esquema relacional resultante:

Ciudad(Nombre-Ciudad, nº Región, habitantes)
CP{ Nombre-Ciudad}
VNN{nº Región}
CAj {nº Región} referencia a Región(nº región)
Región(nº-Región, nombre, habitantes)
CP {nº Region}
(*)Mínima 1 no se puede representar

Relaciones Binarias 1:1 (Uno a Uno)

El siguiente diagrama muestra una estructura de este tipo

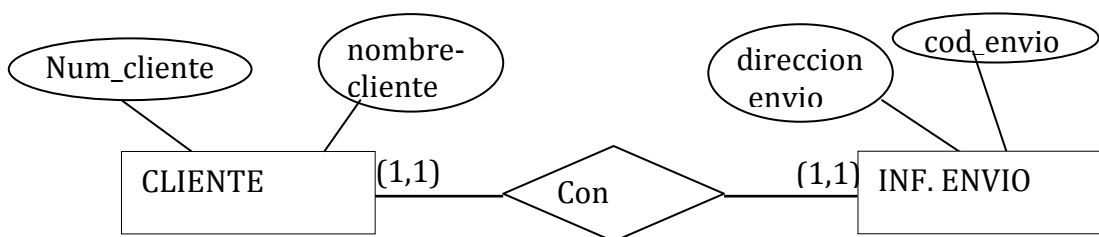


El esquema relacional resultante:

Cliente (num_cliente)
CP{num_cliente}
Tarjeta_credito(numero, tipo_tarjeta, num_cliente)
CP{numero}
CAj{num_cliente} referencia a Cliente(num_cliente)
CAlt{num_cliente}

Relaciones Binarias 1:1 (Uno a Uno): Con doble restricción de existencia

Dos restricciones de existencia una a cada parte de la relación uno a uno

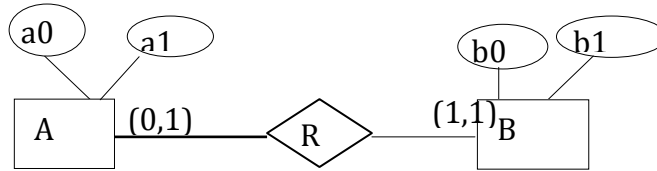


El esquema relacional resultante:

Envío-Cliente(num_cliente, nombre_cliente, direccion-envío, cod_envio)
CP {num_cliente}
CAlt{cod_envio}
VNN{cod_envio}

Relaciones Binarias 1:1 (Uno a Uno): Con 1 restricción de existencia

Una restricción de existencia a una parte de la relación uno a uno



El esquema relacional resultante:

$A(a_0, a_1, b_0)$

$CP\{a_0\}$

$CAj\{b_0\}$ referencia a $B(b_0)$

$CAlt\{b_0\}$

$VNN\{b_0\}$

$B(b_0, b_1)$

$CP\{b_0\}$

Relaciones Reflexivas

Relación reflexiva 1:1 sin restricción de existencia

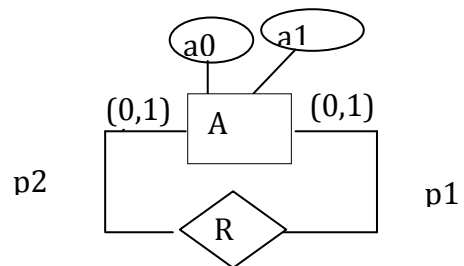
El esquema relacional resultante:

$A(a_0, a_1, a_{0_p2})$

$CP\{a_0\}$

$CAlt\{a_{0_p2}\}$

$CAj\{a_{0_p2}\}$ referencia a $A(a_0)$



Relación reflexiva 1:1 con restricción de existencia

El esquema relacional resultante:

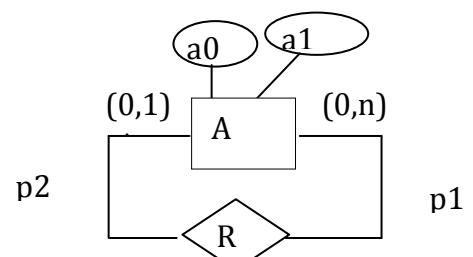
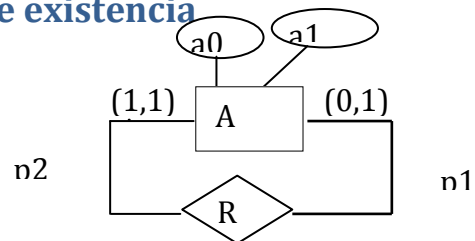
$A(a_0, a_1, a_{0_p2})$

$CP\{a_0\}$

$CAlt\{a_{0_p2}\}$

$CAj\{a_{0_p2}\}$ referencia a $A(a_0)$

$VNN\{a_{0_p2}\}$



TEMA 3: MODELO RELACIONAL
BDA-DAM

Relación reflexiva 1:M Sin restricción de existencia

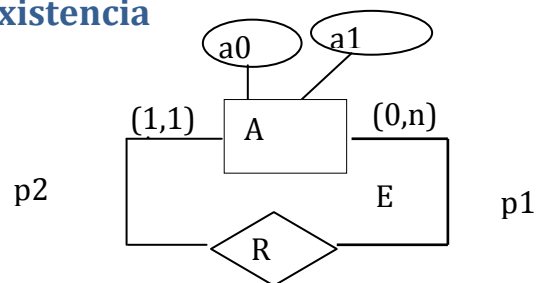
El esquema relacional resultante:

A (a0,a1,a0_p2)
CP{a0}
CAj{a0_p2} referencia a A(a0)

Relación reflexiva 1:M con restricción de existencia

El esquema relacional resultante:

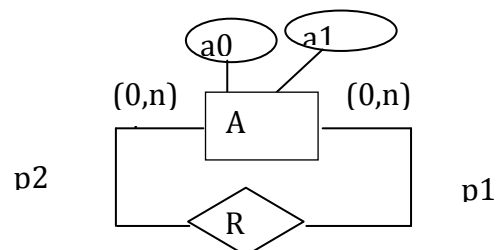
A (a0,a1,a0_p2)
CP{a0}
CAj{a0_p2} referencia a A
VNN{a0_p2}



Relación Reflexiva M:M sin restricción de existencia

El esquema relacional resultante:

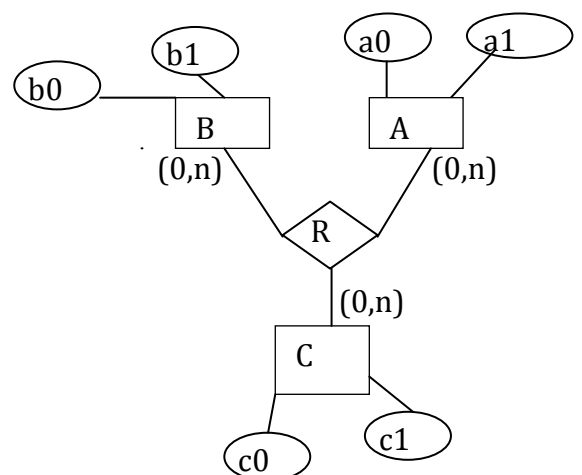
A (a0,a1)
CP{a0}
R(a0_p1,a0_p2)
CP{a0_p1,a0_p2}
CAj{a0_p1} referencia a A(a0)
CAj{a0_p2} referencia a A(a0)



Relaciones Ternarias

Relación ternaria M:M:M

A(a0,a1)
CP{a0}
B(b0,b1)
CP{b0}
C(c0,c1)
CP{c0}
R(a0,b0,c0)
CP{a0,b0,c0}
CAj {a0} Referencia a A(a0)



TEMA 3: MODELO RELACIONAL

BDA-DAM

CAj {b0} Referencia a B(b0)

CAj {c0} Referencia a C(c0)

TEMA 3: MODELO RELACIONAL
BDA-DAM

Relación ternaria 1:M:M

A(a0,a1)

CP{a0}

B(b0,b1)

CP{b0}

C(c0,c1)

CP{c0}

R(a0,b0,c0)

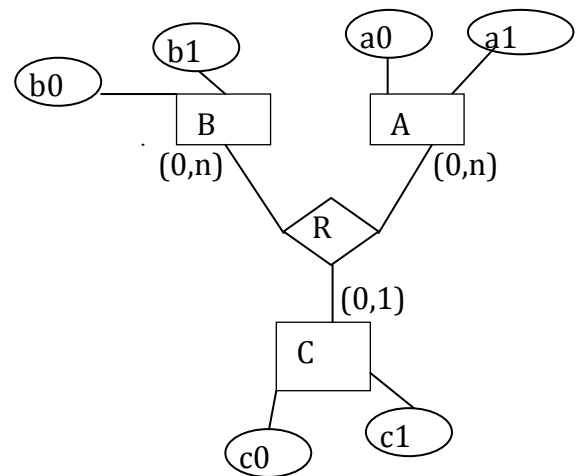
CP{a0,b0}

VNN{c0}

CAj {a0} Referencia a A (a0)

CAj {b0} Referencia a B(b0)

CAj {c0} Referencia a C (c0)



Relación ternaria 1:1:M

El esquema relacional resultante:

A(a0,a1)

CP{a0}

B(b0,b1)

CP{b0}

C(c0,c1)

CP{c0}

R(a0,b0,c0)

CP{a0,b0}

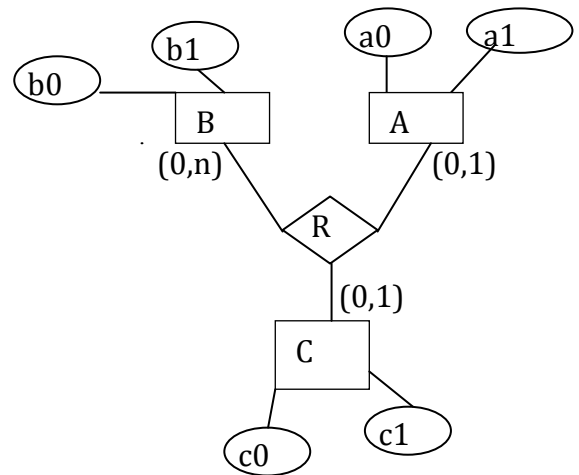
CAIt{b0,c0}

VNN{c0}

CAj {a0} Referencia a A(a0)

CAj {b0} Referencia a B(b0)

CAj {c0} Referencia a C(c0)



Relación ternaria 1:1:1

El esquema relacional resultante:

A(a0,a1)

CP{a0}

B(b0,b1)

CP{b0}

C(c0,c1)

CP{c0}

R(a0,b0,c0)

CP{a0,b0}

CAIt{a0,c0}

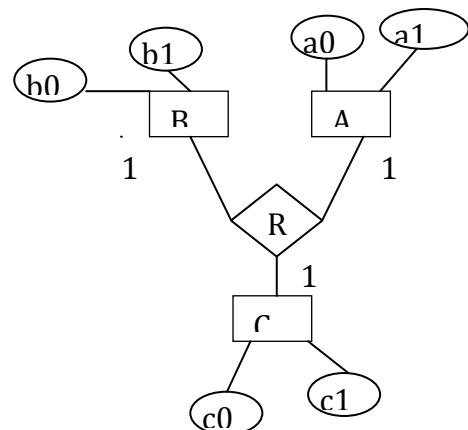
CALT {b0,c0}

VNN{c0}

CAj {a0} Referencia a A(a0)

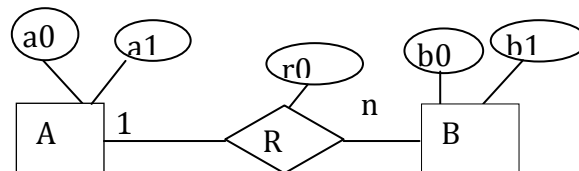
CAj {b0} Referencia a B(b0)

CAj {c0} Referencia a C(c0)



4.4 Atributos en las relaciones

Cuando una relación entre entidades, tiene atributos propios, estos deben incluirse



siempre en la relación donde se esté representando la relación.

A(a0,a1)

CP{a0}

B(b0,b1,a0,r0)

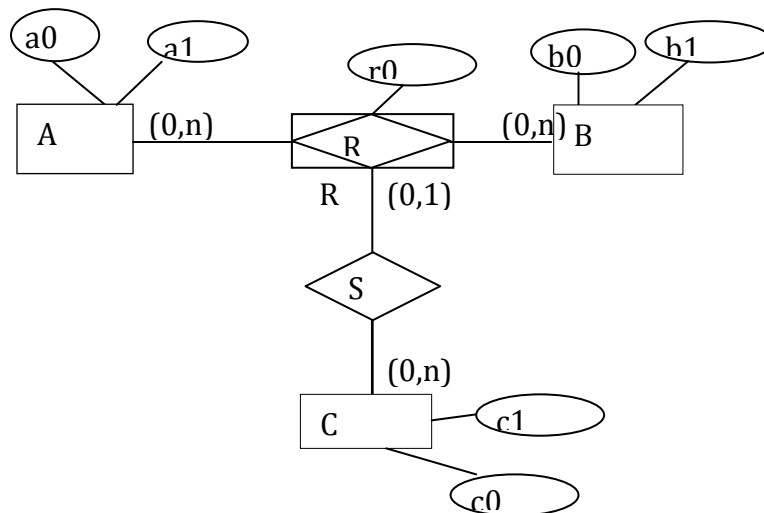
CP{b0}

CAj {a0} Referencia a A(a0)

4.5 Transformación de la agregación

Generalizar el diseño de los objetos agregados que pueden aparecer en un diagrama Entidad-Relación es complicado ya que éstos pueden estar definidos sobre relaciones de cualquier grado.

Ejemplo1



A(a0,a1)

CP{a0}

B(b0,b1)

CP{b0}

R(a0,b0)

CP{a0,b0}

CAj{a0} referencia a A (a0)

CAj{b0} referencia a B(b0)

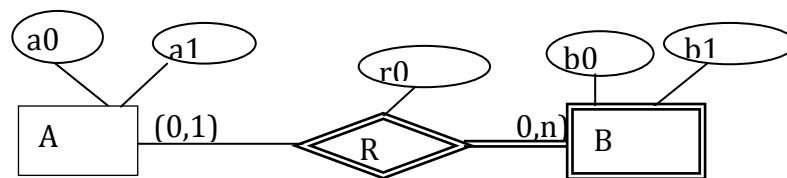
TEMA 3: MODELO RELACIONAL BDA-DAM

$C(c0, c1, a0, b0)$
CP{c0}
CAj{a0, b0} referencia a R(a0, b0)

4.6 Entidad débil

La transformación de una entidad débil es análoga a la de una entidad que no sea débil; la única diferencia es que es necesario incorporar, como atributos, las clave primarias de las relaciones que representan a las entidades gracias a las cuales se identifica.

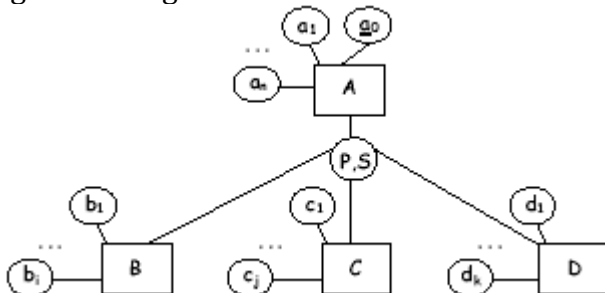
Supongase el siguiente diagrama



$A(a0, a1)$
CP{a0}
 $B(b0, b1, a0, r0)$
CP{a0, b0}
CAj{a0} Referencia a A(a0)

4.7 Entidad especializada

Sólo cuando la especialización es Parcial y Solapada existe una transformación en relaciones totalmente adecuada; en los demás casos es necesario incluir ciertas restricciones de integridad que permitan la definición exacta de estos objetos. La transformación consiste en definir una relación para cada entidad especializada que incluye los atributos propios y también la clave primaria de la relación que representa a la entidad general, que pasa a ser también la clave primaria de la relación. Sea, por ejemplo, el siguiente diagrama Entidad-Relación:



TEMA 3: MODELO RELACIONAL

BDA-DAM

El conjunto de relaciones que representan ese esquema es el siguiente:

$A(a_0: \text{dom_a0}, a_1: \text{dom_a1}, \dots, a_n: \text{dom_an})$

CP: $\{a_0\}$

$B(a_0: \text{dom_a0}, b_1: \text{dom_b1}, \dots, b_i: \text{dom_bi})$

CP: $\{a_0\}$

CAj: $\{a_0\}$ referencia a A (a_0)

$C(a_0: \text{dom_a0}, c_1: \text{dom_c1}, \dots, c_j: \text{dom_cj})$

CP: $\{a_0\}$

CAj: $\{a_0\}$ referencia a A (a_0)

$D(a_0: \text{dom_a0}, d_1: \text{dom_d1}, \dots, d_k: \text{dom_dk})$

CP: $\{a_0\}$

CAj: $\{a_0\}$ referencia a A (a_0)

Obsérvese que la clave primaria de las relaciones donde se representan las entidades especializadas se define también como clave ajena a la relación donde está representada la entidad general; de esta forma se expresa la restricción de que toda ocurrencia de cualquier entidad especializada corresponde a una ocurrencia de la entidad general. En caso de que la generalización sea de otro tipo, el esquema relacional que se debe definir es el mismo, pero es necesaria la inclusión de algunas restricciones de integridad como se muestra a continuación:

Total y solapada: En este caso el conjunto de relaciones anterior no representa exactamente el objeto general ya que queda por expresar el hecho de que toda ocurrencia de la entidad general A tiene que estar asociada con al menos una ocurrencia de alguna entidad especializada. Es necesaria la definición de una restricción de integridad como se muestra:

$$A[a_0] \subset B[a_0] \cup C[a_0] \cup D[a_0] \text{ (total)}$$

Parcial y disjunta: Para este tipo de especialización, el conjunto de relaciones anterior no expresa la restricción de que las entidades especializadas son disjuntas, esto es, el hecho de que cada ocurrencia de la entidad general sólo puede estar asociada con una ocurrencia de una entidad especializada. Las expresiones en cálculo relacional de tuplas y en SQL que representan esta restricción son las siguientes:

$$B[a_0] \cap C[a_0] = 0$$

$$B[a_0] \cap D[a_0] = 0 \quad \text{(Disjunta)}$$

$$C[a_0] \cap D[a_0] = 0$$

Total y disjunta: En este caso es necesaria la inclusión de las dos restricciones para obtener un esquema relacional que represente exactamente este tipo de objeto general.

$$B[a_0] \cap C[a_0] = 0$$

$$B[a_0] \cap D[a_0] = 0 \quad \text{(Disjunta)}$$

$$C[a_0] \cap D[a_0] = 0$$

TEMA 3: MODELO RELACIONAL

BDA-DAM

$A[a0] \subset B[a0] \cup C[a0] \cup D[a0]$ (total)

4.8 Restricciones de integridad.

Las restricciones de integridad nos permiten representar restricciones que no podemos definir en el paso a tablas del modelo relacional, tal como hemos visto en el caso de las especializaciones. Las restricciones de integridad se pueden representar bien en lenguaje natura, o mejor en lenguaje lógico.

Relación M:M con restricción de existencia

$A(a0,a1,b0)$

$CP\{a0\}$

$B(b0,b1)$

$CP\{b0\}$

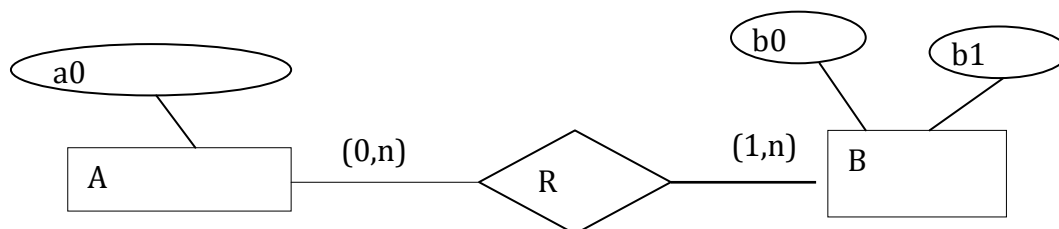
$R(a0,b0)$

$CP\{a0,b0\}$

$CAj\{a0\}$ referencia a A

$CAj\{b0\}$ referencia a B

RI: $B[b0] \subset R[b0]$



Relación M:M con 2 restricciones existencia

$A(a0,a1)$

$CP\{a0\}$

$B(b0,b1)$

$CP\{b0\}$

$R(a0,b0)$

$CP\{a0,b0\}$

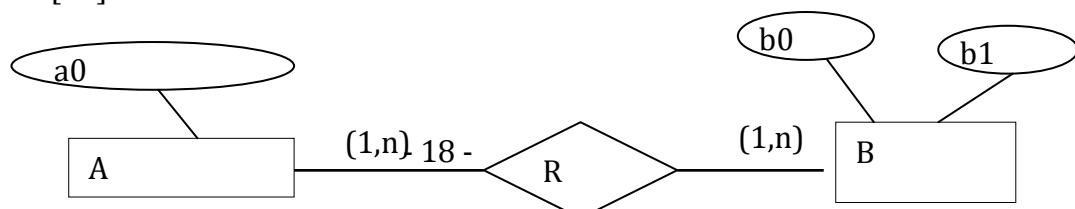
$CAj\{a0\}$ referencia a A(a0)

$CAj\{b0\}$ referencia a B(b0)

RI: $B[b0] \subset R[b0]$

$A[a0]$

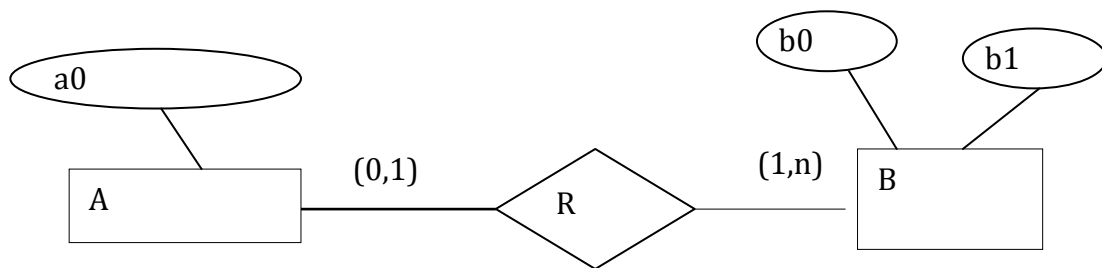
$\subset R[a0]$



Relación binaria 1:M CON RESTRICCIÓN DE EXISTENCIA

La relación binaria con una restricción de existencia sobre la entidad de cardinalidad máxima Uno.

A(a0,a1)
CP{a0}
B(b0,b1,a0)
CP{b0}
CAj {a0} referencia a A
RI: A[a0] \subset B[a0]



Relación binaria 1:1 con 2 restricciones de existencia

A(a0,a1,b0)
CP{a0}
CAj{b0} referencia a B(b0)
CAIt{b0}
VNN{b0}
B(b0,b1)
CP{b0}
RI: B[b0] \subset A[b0]

