

Diseño lógico: Modelo relacional

Diseño de Bases de Datos Multimedia

Grado en Ingeniería Multimedia



Universitat d'Alacant
Universidad de Alicante



Departamento de
Lenguajes y Sistemas Informáticos

Fases de diseño

ANALISIS DE REQUERIMIENTOS

Una empresa distribuidora de recambios de autom6vil nos ha encargado el dise1o de una Base de Datos (s6lo modelo EER) que de respuesta a sus necesidades de cara a la gesti6n de los recambios que distribuye.

La empresa obtiene las piezas que distribuye de distintas f6bricas de las que se tiene el identificador 6nico interno de la misma, el n6mero de empleados (no siempre), el pa6s donde est6 ubicada la sede central (siempre) y adem6s se tiene su nombre comercial que es 6nico en el mundo.

Las piezas se identifican por un n6mero de serie y por la f6brica que las fabrica, ya que un n6mero de serie se puede repetir de una f6brica a otra (una misma f6brica nunca repite n6meros).

De las piezas adem6s se conoce el precio de cada una de ellas y de algunas piezas se dispone de las instrucciones necesarias para su montaje. Se catalogan en piezas de torniller6a, piezas de motores y piezas de chapas, tambi6n se catalogan en calidad alta, media y baja. El IVA a aplicar en las piezas depende del nivel de calidad, a igual nivel de calidad igual IVA a aplicar.

Si las piezas de motores necesitan para su montaje de piezas de torniller6a que la empresa distribuye, se tiene la lista de estas piezas y su cantidad necesarias para su montaje, es obvio, que una pieza de motor puede tener m6s de un tornillo y viceversa.

Por otro lado la empresa dispone de empleados con diferentes roles, t6cnicos seniors, t6cnicos juniors y agentes comerciales. De todos ellos se conoce su NIF (que lo identifica) su nombre y su localidad de nacimiento.

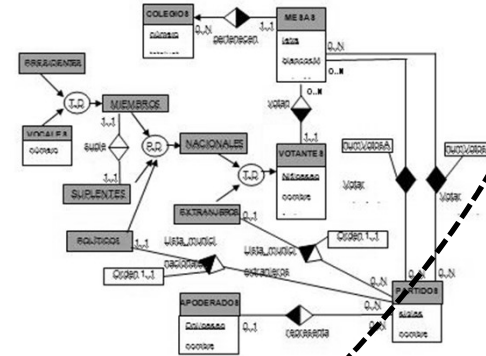
Para controlar la calidad de las piezas de chapa que se distribuyen, la empresa forma parejas 6nicas entre un t6c. Senior y un t6c. Junior (solo se puede pertenecer a una pareja, o a ninguna). Estas parejas eval6an distintas piezas de chapas y le ponen unos puntos. Se desea conservar la puntuaci6n que cada pareja hace de cada pieza de chapa. Una misma pieza puede ser evaluada por m6s de una pareja.

Las f6bricas realizan inversiones en diferentes pa6s y para ello son subvencionadas por organismos internacionales (de los que se conocen sus siglas que los identifican y su nombre) de forma que una f6brica en un pa6s solo puede ser subvencionada por un organismo, y si un organismo subvenciona una f6brica X en un pa6s ya no puede subvencionar a otra en este mismo pa6s.

En cuanto a los agentes comerciales la empresa los tiene asignados a los pa6s en los que trabaja (un agente puede estar asignado a m6s de un pa6s y viceversa) teniendo almacenado el tel6fono de cada agente en cada pa6s, adem6s, en algunos casos la empresa asigna a estos agentes a las relaciones con los organismos subvencionadores del pa6s en el que est6n asignados, de forma que un agente en un pa6s solo puede relacionarse como m6ximo con un organismo, pero un organismo puede estar asignado a m6s de un agente en un mismo pa6s.

DISE1O CONCEPTUAL

Modelo relacional → Esquema EER



DISE1O LOGICO

Modelo relacional -> Esquema Relacional

VISOR (nombre, empresa)
 C. Primaria: nombre
 FORMATO (nombre, descrip, a1o)
 C. Primaria: nombre
 VNN: descrip
 SE_VISUALIZA_CON(nombre_visor, nombre_formato, c6dec)
 C. Primaria: (nombre_visor, nombre_formato)
 C. Ajena: nombre_visor → VISOR
 C. Ajena: nombre_formato → FORMATO
 VNN: C6dec
 RECURSO (c6digo, descrip, falta, Tama1o, Taprx_desc, nombre_formato)
 C. Primaria: c6digo
 C. Ajena: nombre_formato → FORMATO
 V.N.N.: nombre_formato
 COMPATIBLE_CON (nom_formato1, nom_formato2)
 C. Primaria : (nom_formato1, nom_formato2)
 C. ajena : nom_formato1 -> FORMATO

DISE1O FISICO

Scripts de BD (dependiente SGBDR)

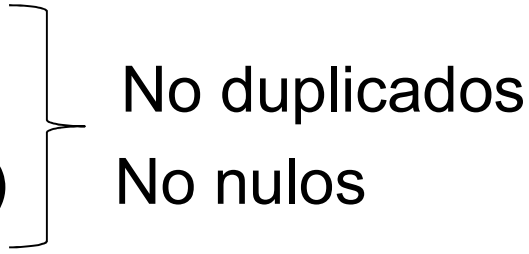
```
CREATE TABLE VISOR
(
  nombre VARCHAR2(100) CONSTRAINT PK_VISOR PRIMARY KEY,
  empresa VARCHAR2 (100)
);

CREATE TABLE FORMATO
(
  nombre VARCHAR2(10) CONSTRAINT PK_FORMATO PRIMARY KEY,
  descripcion VARCHAR2(100) NOT NULL,
  anyo DATE
);

CREATE TABLE SE_VISUALIZA_CON
(
  nombre_visor VARCHAR2(100),
  nombre_formato VARCHAR2(10),
  c6dec VARCHAR2(100),
  CONSTRAINT PK_SE_VISUALIZA_CON PRIMARY KEY (nombre_visor, nombre_formato),
  CONSTRAINT FK_SE_VISUALIZA_CON_VISOR FOREIGN KEY (nombre_visor) REFERENCES VISOR
);

CREATE TABLE RECURSO
(
  c6digo VARCHAR2(50),
  descripcion VARCHAR2(100),
  falta DATE,
  nombre_formato VARCHAR2(10),
  CONSTRAINT PK_RECURSO PRIMARY KEY (c6digo),
  CONSTRAINT FK_RECURSO_FORMATO FOREIGN KEY (nombre_formato) REFERENCES FORMATO
);
```

modelo relacional

- Clave candidata
 - Clave primaria (CP)
 - Clave alternativa (cAlt)
 - SÓLO PUEDE EXISTIR UNA CP PERO VARIAS CAIt
- Clave ajena
- Valor no nulo

una entidad

A	
<u>a0</u>	
a1	
a2	1 ..1
a3	

A(a0, a1, a2, a3)

C.P.: a0

V.N.N.: a2

una entidad

clave primaria compuesta

A	
<u>a0</u>	
<u>a1</u>	
a2	1 ..1
a3	

A(a0, a1, a2, a3)

C.P.: (a0,a1)

V.N.N.: a2

una entidad

con atributos multivalor

A	
<u>a0</u>	
a1	
a2	1 ..1
A3	0 .. N

$A(a_0, a_1, a_2)$

C.P.: a_0

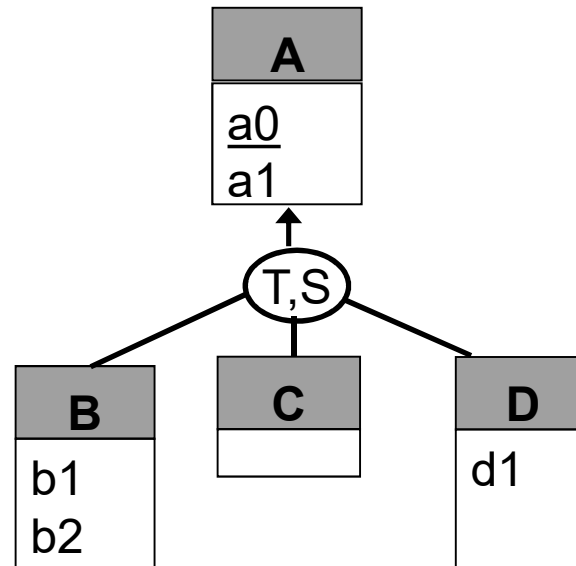
V.N.N.: a_2

$M(a_0, a_3)$

C.P.: (a_0, a_3)

C.aj.: $a_0 \rightarrow A$

generalización



A(a0, a1)

C.P.: a0

No se puede captar
que es TOTAL

B(rA, b1, b2)

C.P.: rA

C. Ajena: $rA \rightarrow A$

C(rA)

C.P.: rA

C. Ajena: $rA \rightarrow A$

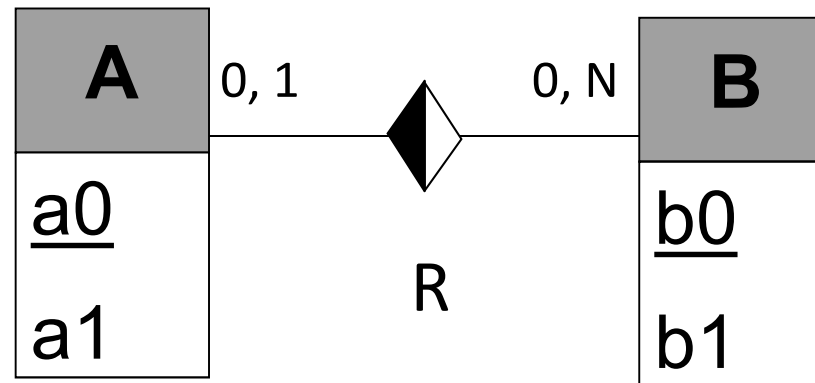
D(rA, d1)

C.P.: rA

C. Ajena: $rA \rightarrow A$

**en las tablas sólo se representan
bien las generalizaciones P,S**

binaria 1:m



$A(a0, a1, rB)$

C.P.: $a0$

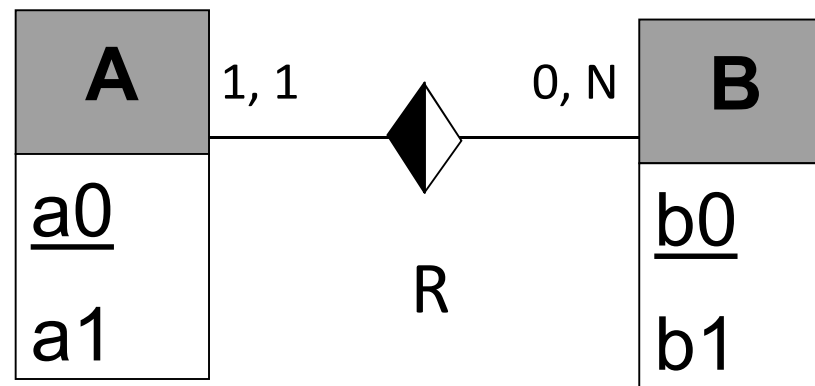
C.aj.: $rB \rightarrow B$

$B(b0, b1)$

C.P.: $b0$

binaria 1:m

con una restricción de existencia

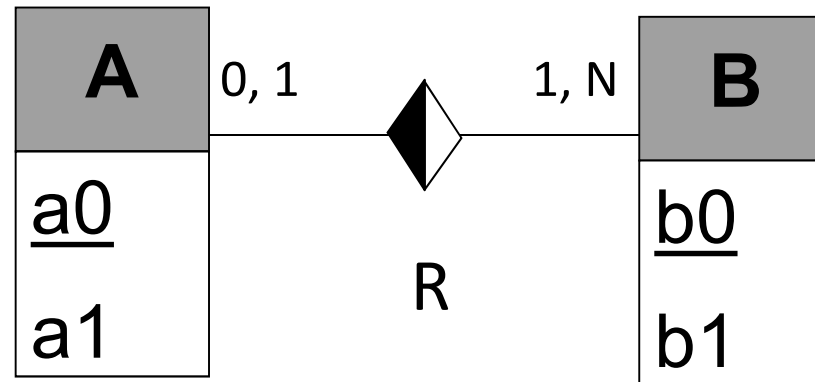


A(a0, a1, rB)
C.P.: a0
C.aj.: rB → B
V.N.N.: rB

B(b0, b1)
C.P.: b0

binaria 1:m

con una restricción de existencia



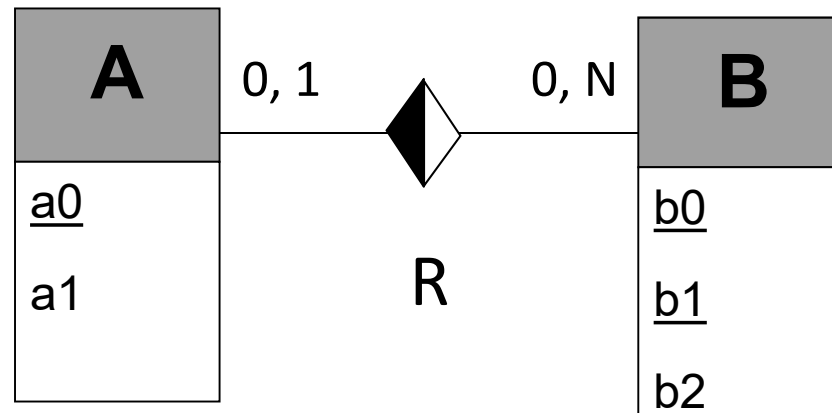
A(a0, a1, rB)
C.P.: a0
C.aj.: rB → B

B(b0, b1)
C.P.: b0

se pierde la R.E. de B hacia R

binaria 1:m

cuando hay clave primaria compuesta



A(a0, a1, rB0, rB1)

C.P.: a0

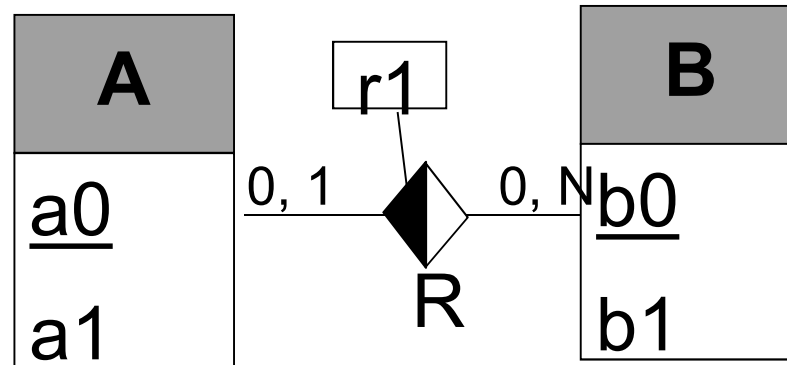
C.aj.: (rB0,rB1) → B

B(b0, b1,b2)

C.P.: (b0,b1)

binaria 1:m

con atributo



■ A(a0, a1, rB, r1*)

C.P.: a0

C.aj.: rB → B

■ B(b0, b1)

C.P.: b0

■ A(a0, a1)

C.P.: a0

■ B(b0, b1)

C.P.: b0

■ R(rA, rB, r1)

C.P.: rA

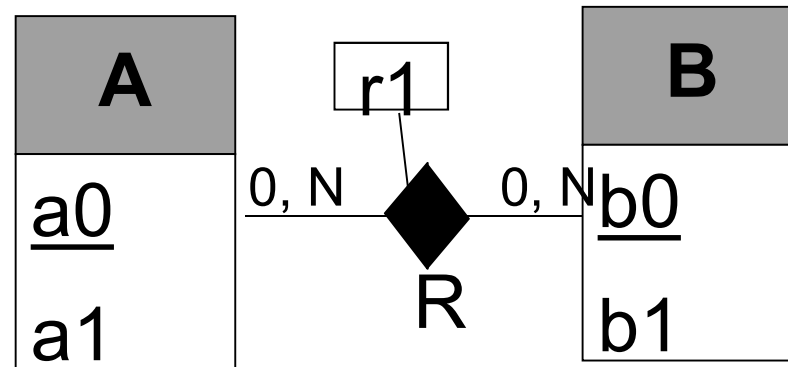
C.aj.: rA → A

C.aj.: rB → B

V.N.N.:rB

*** Existirán valores de r1 cuando dispongamos de valores para rB**

binaria m:m



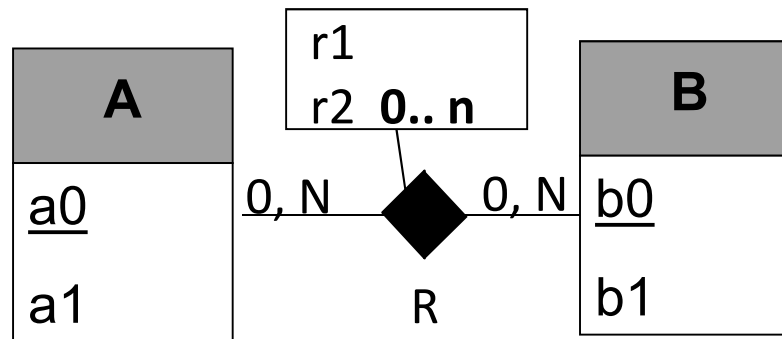
A(a0, a1)
C.P.: a0

B(b0, b1)
C.P.: b0

R(rA, rB, r1)
C.P.: (rA, rB)
C. Ajena: $rA \rightarrow A$
C. Ajena: $rB \rightarrow B$

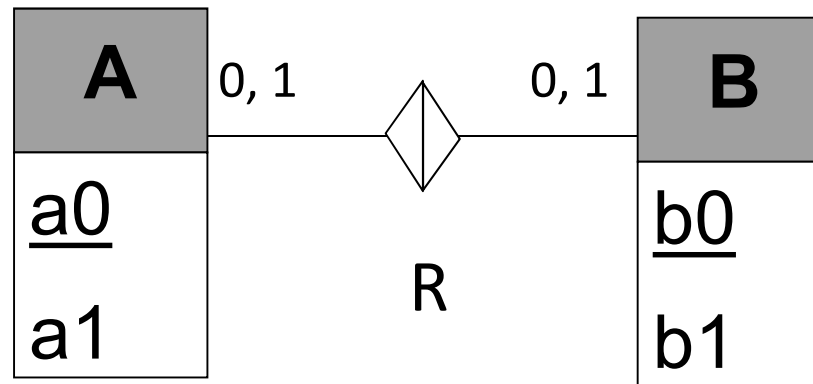
binaria m:m

con atributo multivalor



<p>A(a0, a1)</p> <p>C.P.: a0</p>	<p>R(rA, rB, r1)</p> <p>C.P.: (rA, rB)</p> <p>C. Ajena: rA → A</p> <p>C. Ajena: rB → B</p>	<p>R2(rA, rB, r2)</p> <p>C.P.: (rA, rB, r2)</p> <p>C. Ajena: (rA, rB) → R</p>
<p>B(b0, b1)</p> <p>C.P.: b0</p>		

binaria 1:1



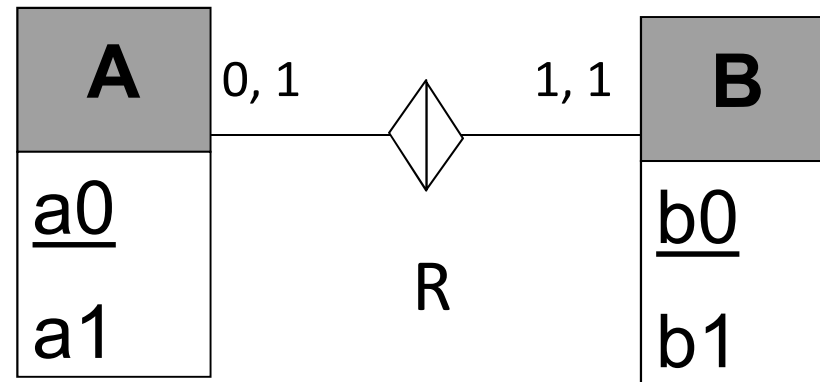
A(a0, a1)
C.P.: a0

B(b0, b1)
C.P.: b0

R(rA, rB)
C.P.: rA
C. Alt: rB
C. Ajena: rA → A
C. Ajena: rB → B

binaria 1:1

con una restricción de existencia

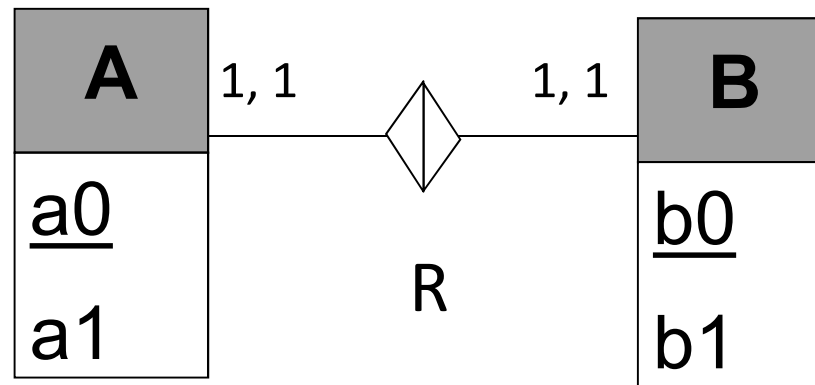


$A(a0, a1)$
C.P.: $a0$

$B(b0, b1, rA)$
C.P.: $b0$
C. Alt: rA
C. Ajena: $rA \rightarrow A$

binaria 1:1

con dos restricción de existencia

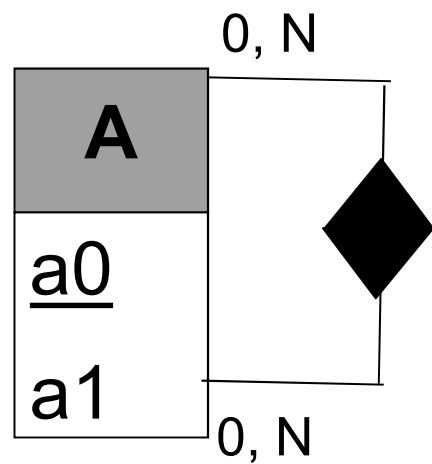


$R(a_0, a_1, b_0, b_1)$

C.P.: a_0

C.Alt: b_0

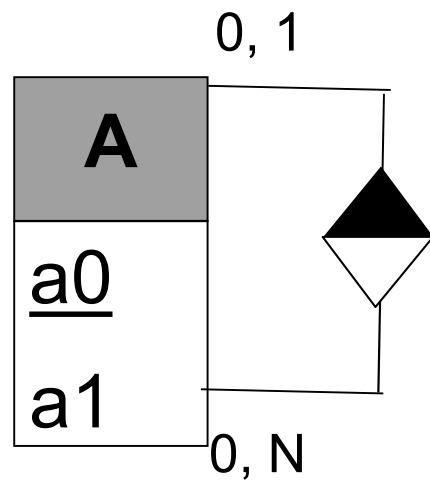
Reflexiva M:M



$A(a_0, a_1)$
C.P.: a_0

$R(r_{A1}, r_{A2})$
C.P.: (r_{A1}, r_{A2})
C. Ajena: $r_{A1} \rightarrow A$
C. Ajena: $r_{A2} \rightarrow A$

Reflexiva 1:M

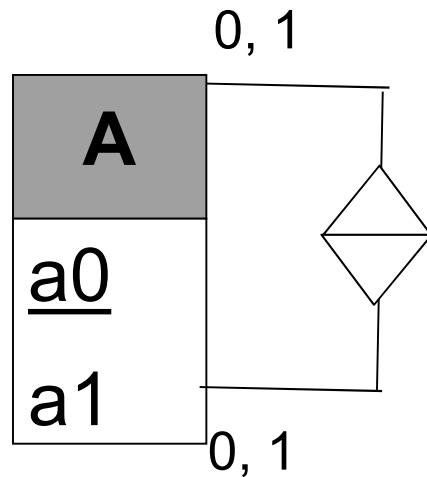


$A(a_0, a_1, r_A)$

C.P.: a_0

C. Ajena: $r_A1 \rightarrow A$

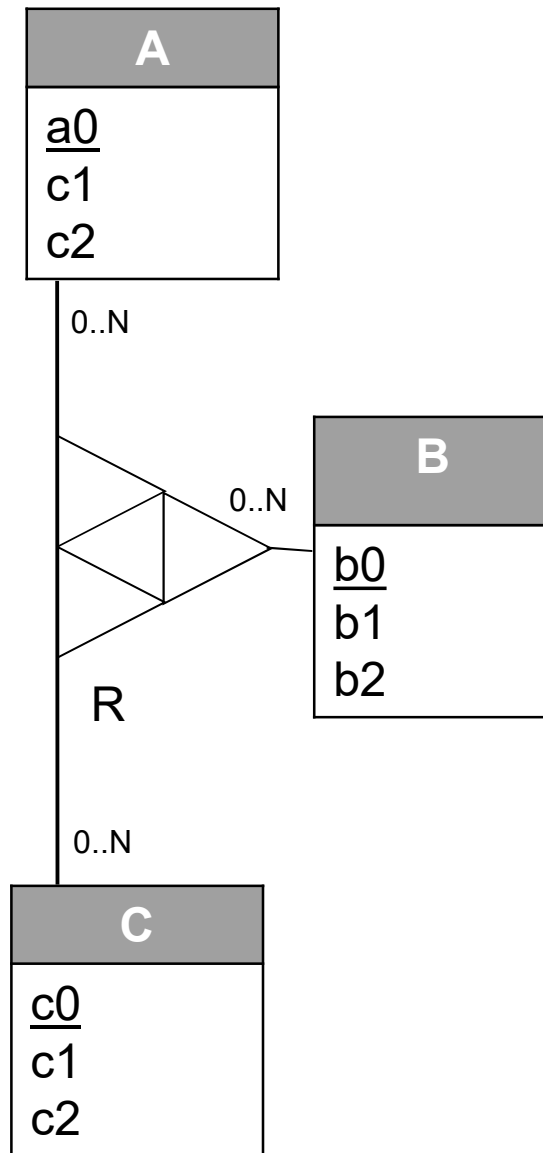
Reflexiva 1:1



$A(a_0, a_1)$
C.P.: a_0

$R(r_{A1}, r_{A2})$
C.P.: r_{A1}
C.Alternativa: r_{A2}
C. Ajena: $r_{A1} \rightarrow A$
C. Ajena: $r_{A2} \rightarrow A$

Ternaria 1:1:1



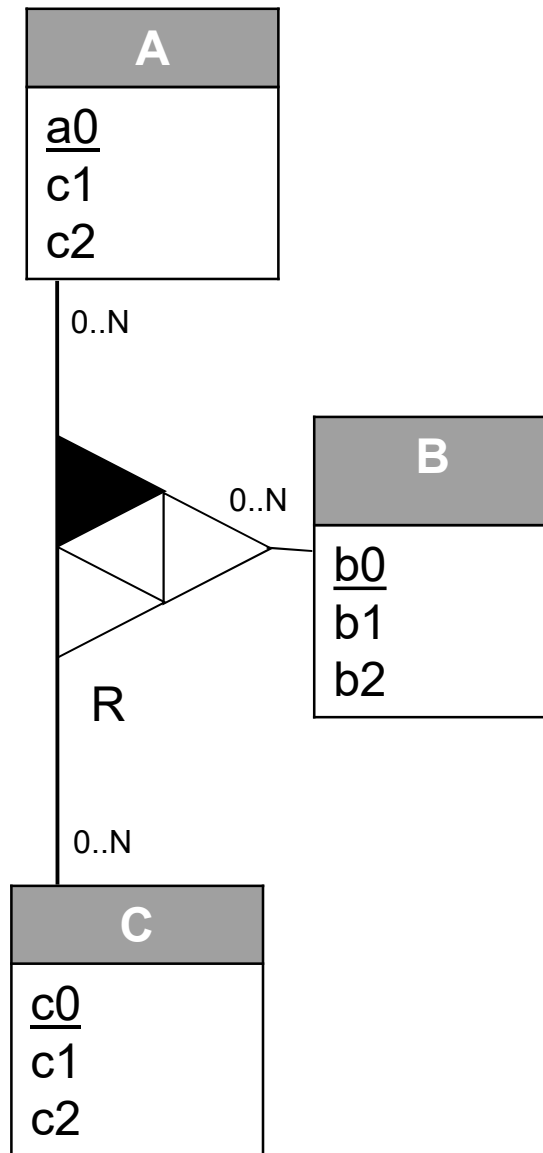
A(a0, a1)
C.P.: a0

B(b0,b1)
C.P.: b0

C(c0,c1)
C.P.: c0

R(rA, rB, rC)
C.P.: (rA, rB)
C.Alternat.: (rA, rC)
C.Alternat.: (rB, rC)
C. Ajena: $rA \rightarrow A$
C. Ajena: $rB \rightarrow B$
C. Ajena: $rC \rightarrow C$

Ternaria 1:1:N



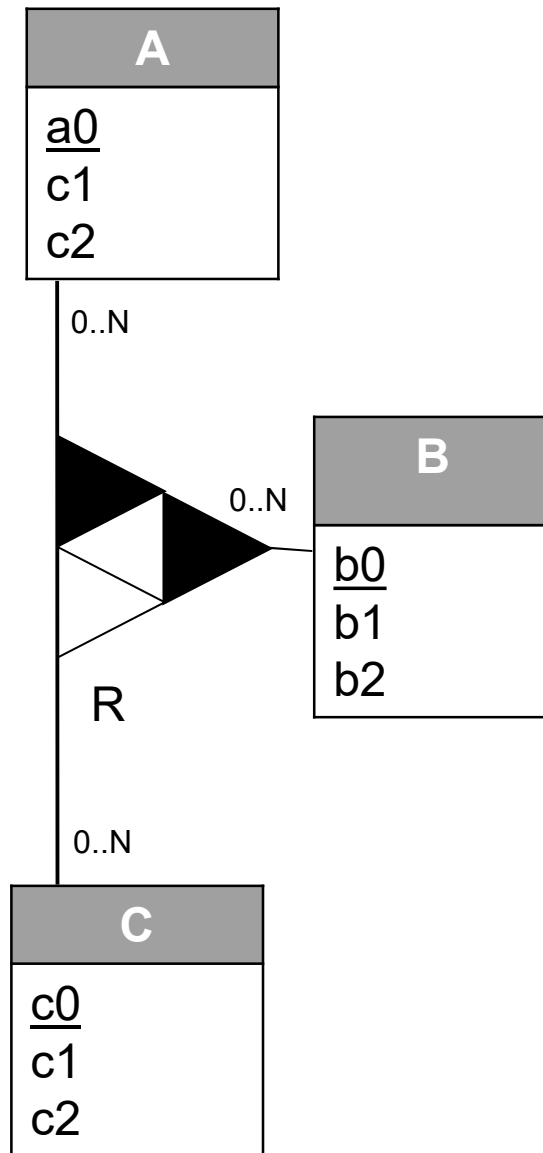
A(a0, a1)
C.P.: a0

B(b0,b1)
C.P.: b0

C(c0,c1)
C.P.: c0

R(rA, rB, rC)
C.P.: (rA, rB)
C.Alternat.: (rA, rC)
C. Ajena: $rA \rightarrow A$
C. Ajena: $rB \rightarrow B$
C. Ajena: $rC \rightarrow C$

Ternaria 1:N:N



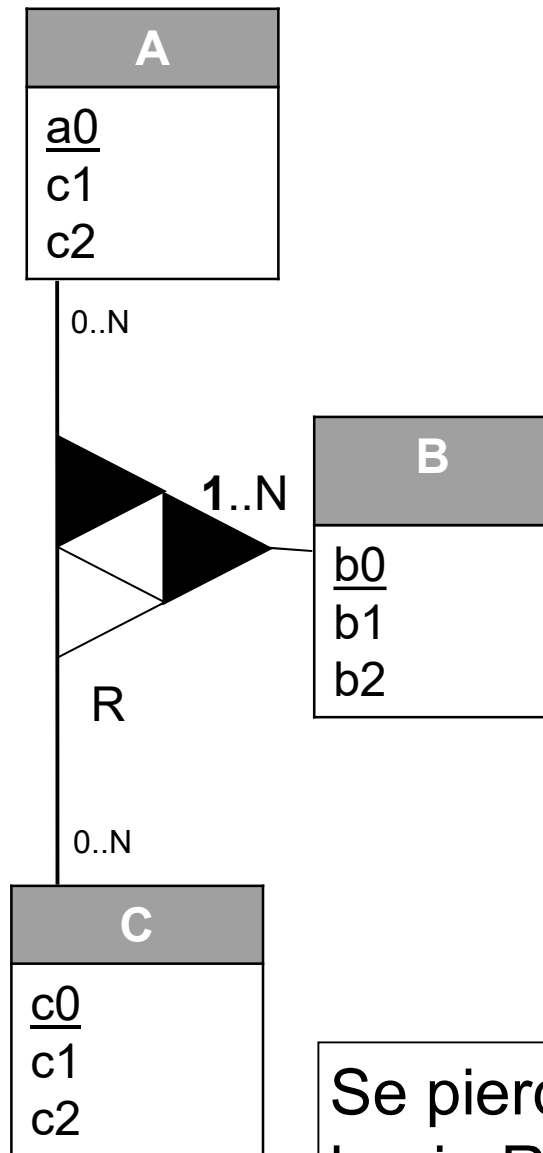
$A(a_0, a_1)$
C.P.: a_0

$B(b_0, b_1)$
C.P.: b_0

$C(c_0, c_1)$
C.P.: c_0

$R(r_A, r_B, r_C)$
C.P.: (r_A, r_B)
C. Ajena: $r_A \rightarrow A$
C. Ajena: $r_B \rightarrow B$
C. Ajena: $r_C \rightarrow C$
VNN: r_C

Ternaria 1:N.N



A(a0, a1)
C.P.: a0

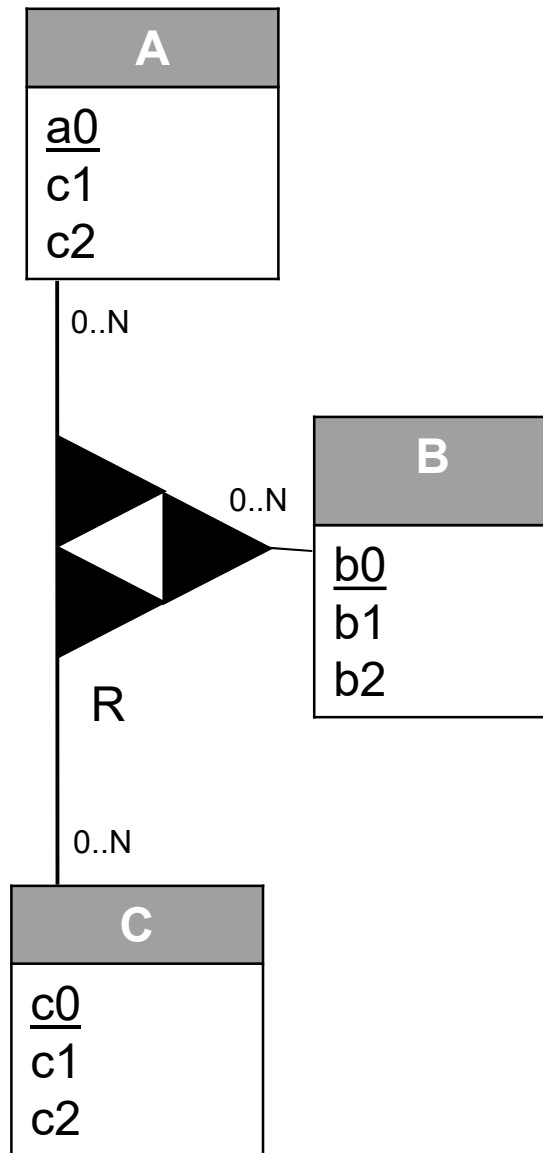
B(b0,b1)
C.P.: b0

C(c0,c1)
C.P.: c0

R(rA, rB, rC)
C.P.: (rA, rB)
C. Ajena: $rA \rightarrow A$
C. Ajena: $rB \rightarrow B$
C. Ajena: $rC \rightarrow C$
VNN: rC

Se pierde restricción de existencia de B
hacia R

Ternaria M:N:N



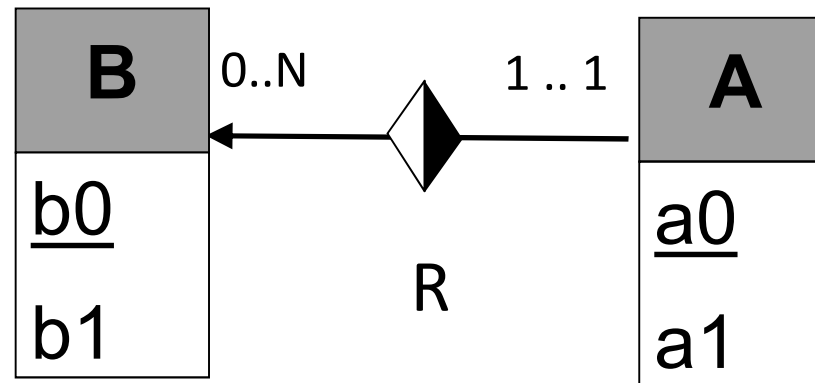
A(a0, a1)
C.P.: a0

B(b0,b1)
C.P.: b0

C(c0,c1)
C.P.: c0

R(rA, rB, rC)
C.P.: (rA, rB, rC)
C. Ajena: rA → A
C. Ajena: rB → B
C. Ajena: rC → C

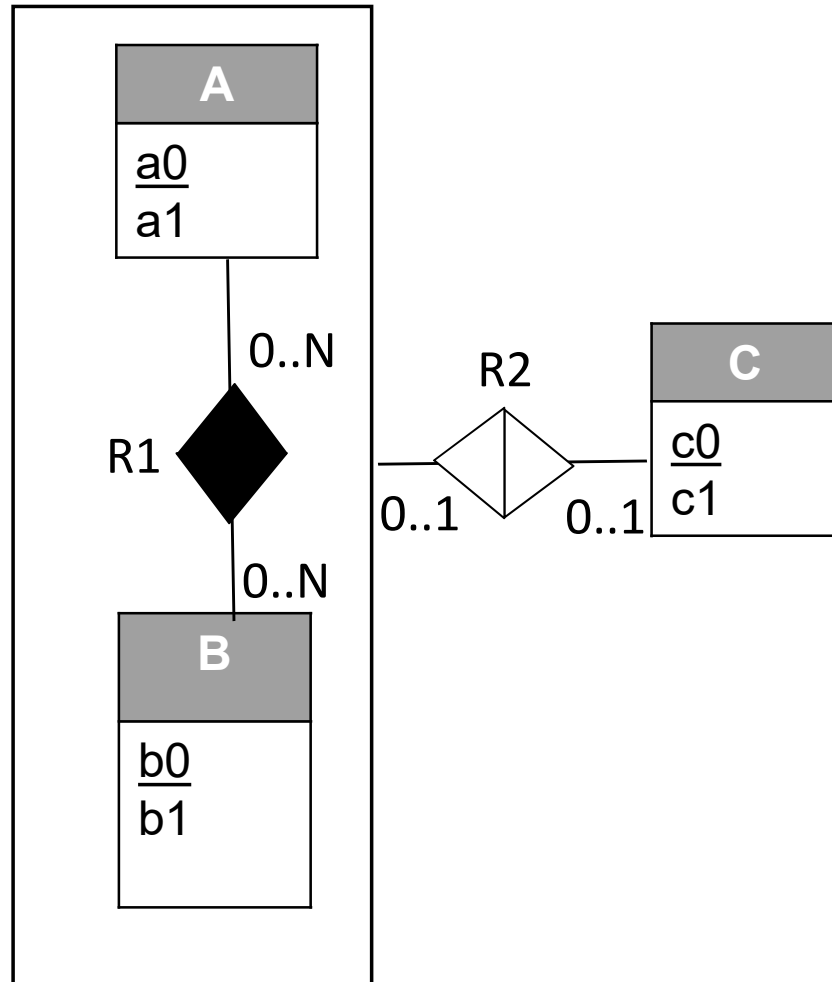
Restricción de identificador



$B(b0, b1)$
C.P.: $b0$

$A(a0, a1, rB)$
C.P.: $(a0, rB)$
C.aj.: $rB \rightarrow B$

Agregación



diseño de bases de datos

A(a0, a1)
C.P.: a0

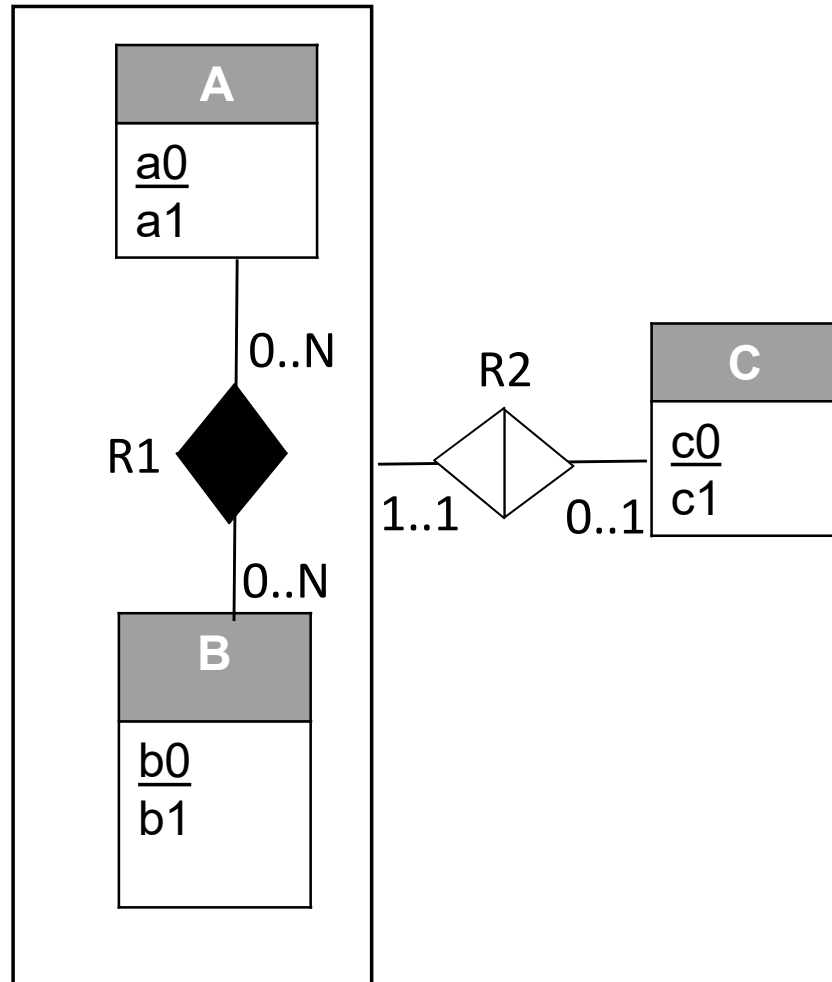
B(b0,b1)
C.P.: b0

C(c0,c1)
C.P.: c0

R1(rA, rB)
C.P.: (rA, rB)
C. Ajena: $rA \rightarrow A$
C. Ajena: $rB \rightarrow B$

R2(rA, rB, rC)
C.P.: (rA, rB)
C. Alt.: rC
C. Ajena: $(rA, rB) \rightarrow R1$
C. Ajena: $rC \rightarrow C$

Agregación



A(a0, a1)
C.P.: a0

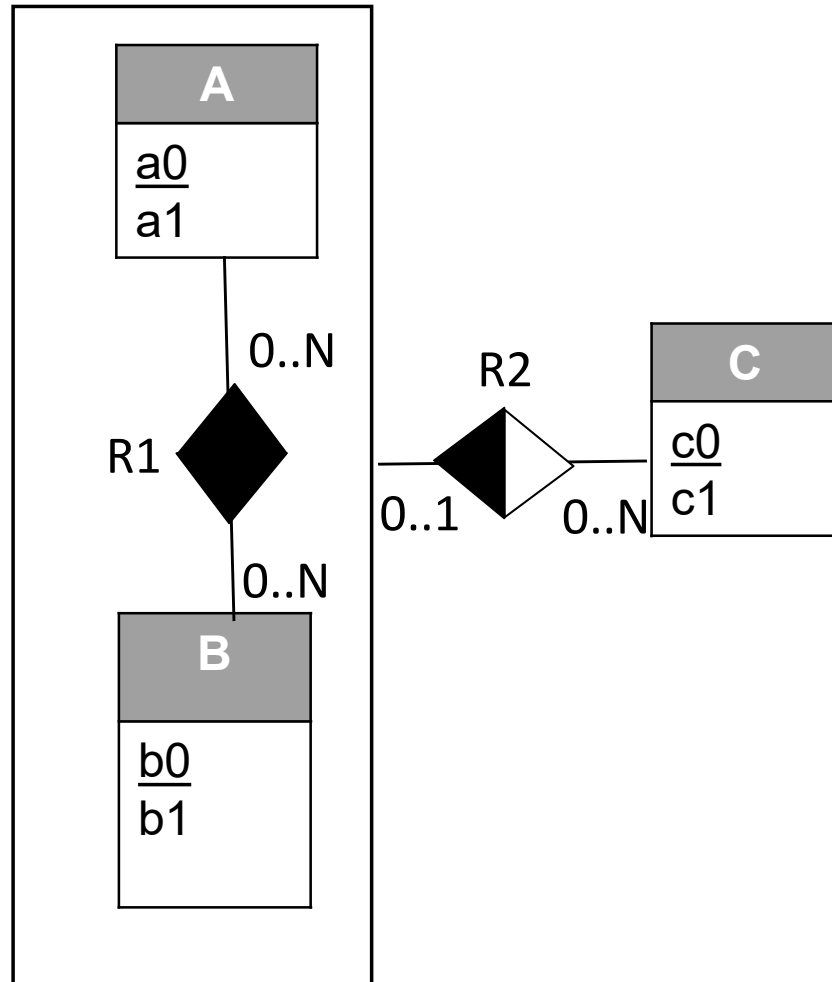
B(b0,b1)
C.P.: b0

C(c0,c1)
C.P.: c0

R1(rA, rB, rC)
C.P.: (rA, rB)
C. Alt.: rC
C. Ajena: $rA \rightarrow A$
C. Ajena: $rB \rightarrow B$
C. Ajena: $rC \rightarrow C$

diseño de bases de datos

Agregación



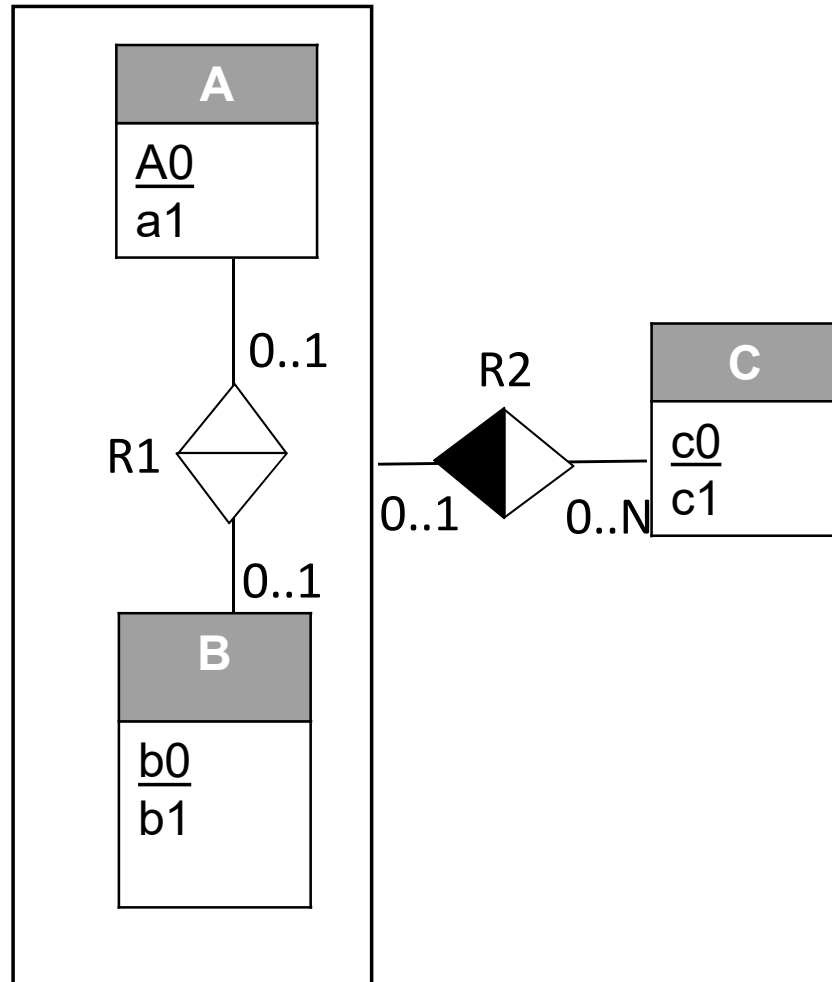
$A(a0, a1)$
C.P.: $a0$

$B(b0, b1)$
C.P.: $b0$

$C(c0, c1)$
C.P.: $c0$

$R1(rA, rB, rC)$
C.P.: (rA, rB)
C. Ajena: $rA \rightarrow A$
C. Ajena: $rB \rightarrow B$
C. Ajena: $rC \rightarrow C$

Agregación



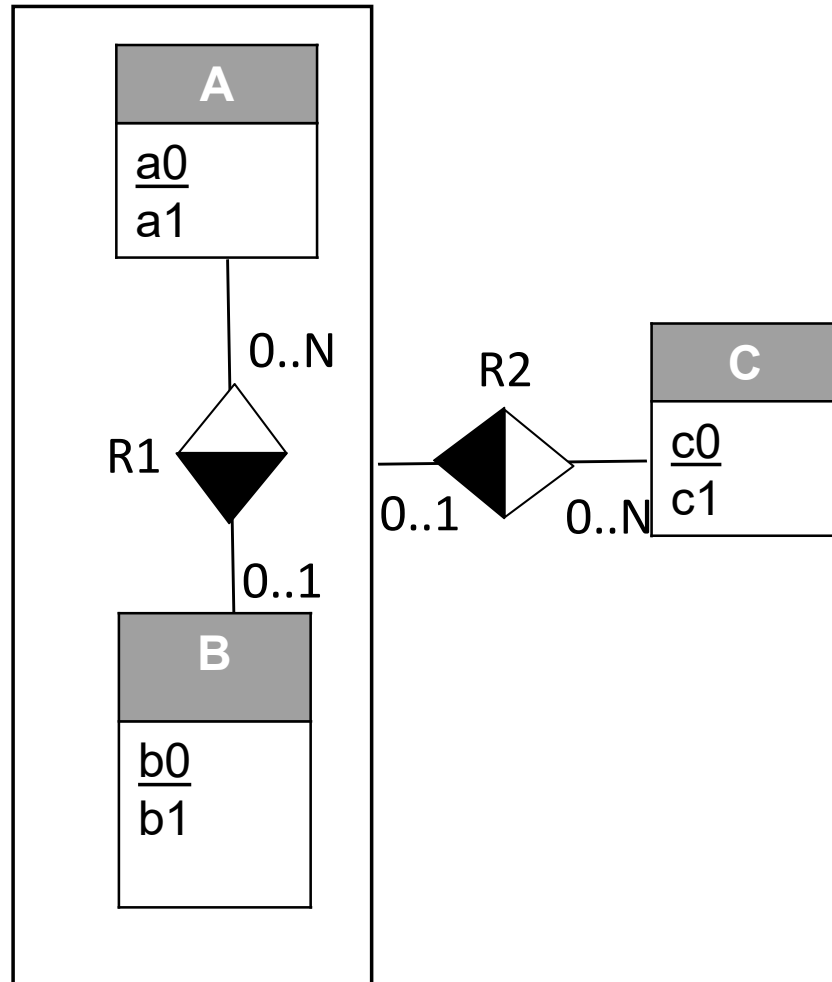
A(a0, a1)
C.P.: a0

B(b0,b1)
C.P.: b0

C(c0,c1)
C.P.: c0

R1(rA, rB, rC)
C.P.: rA
C.Alter.: rB
C. Ajena: $rA \rightarrow A$
C. Ajena: $rB \rightarrow B$
C. Ajena: $rC \rightarrow C$

Agregación



A(a0, a1)
C.P.: a0

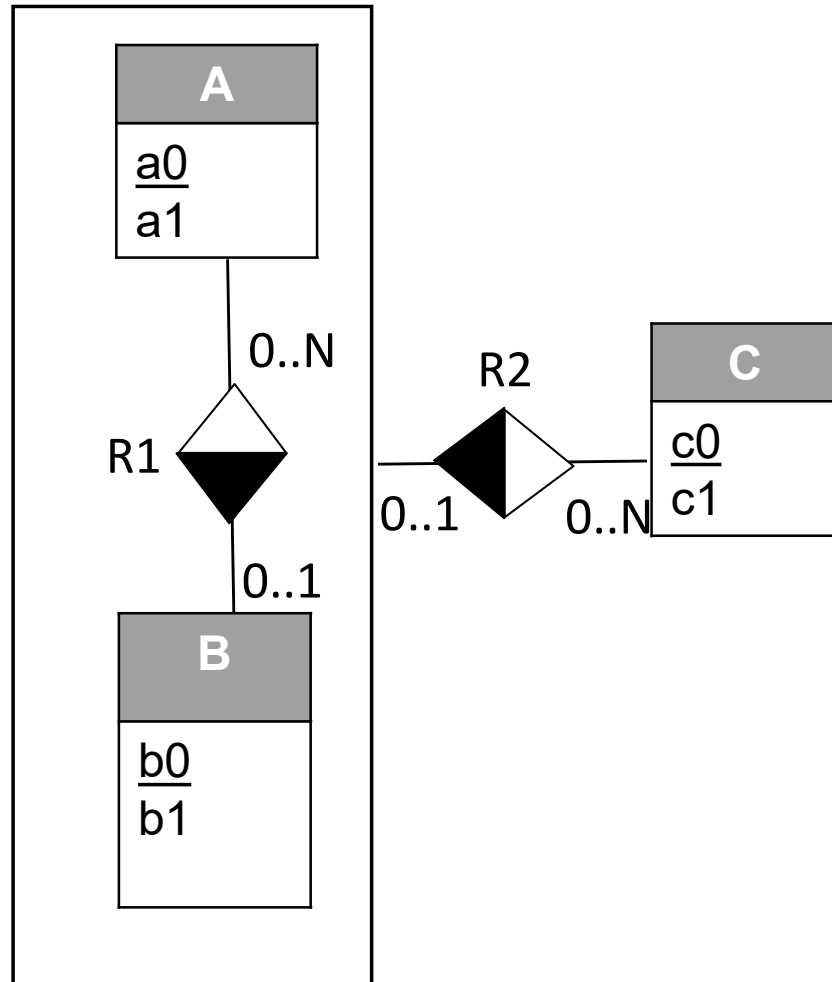
C(c0,c1)
C.P.: c0

B(b0, b1, rA, rC)
C.P.: b0
C. Ajena: $rA \rightarrow A$
C. Ajena: $rC \rightarrow C$
*La columna rC sólo
tendrá valor si lo
tiene rA*

Opción 1

diseño de bases de datos

Agregación



A(a0, a1)
C.P.: a0

B(b0, b1)
C.P.: b0

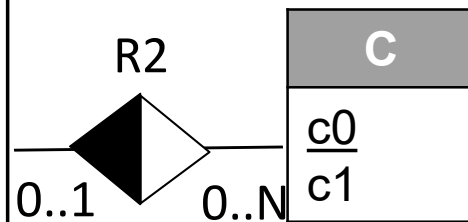
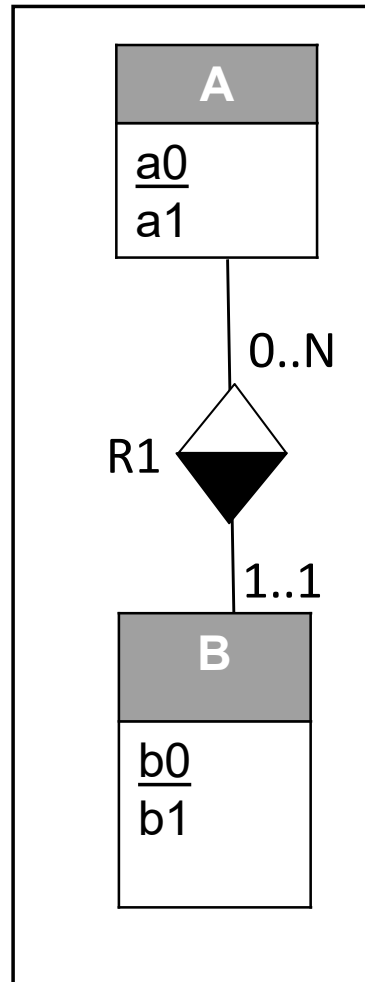
C(c0,c1)
C.P.: c0

R1(rB, rA, rC)
C.P.: rB
C. Ajena: $rB \rightarrow B$
C. Ajena: $rA \rightarrow A$
C. Ajena: $rC \rightarrow C$
V.N.N.: rA

Opción 2

diseño de bases de datos

Agregación



$A(a0, a1)$

C.P.: a0

$C(c0, c1)$

C.P.: c0

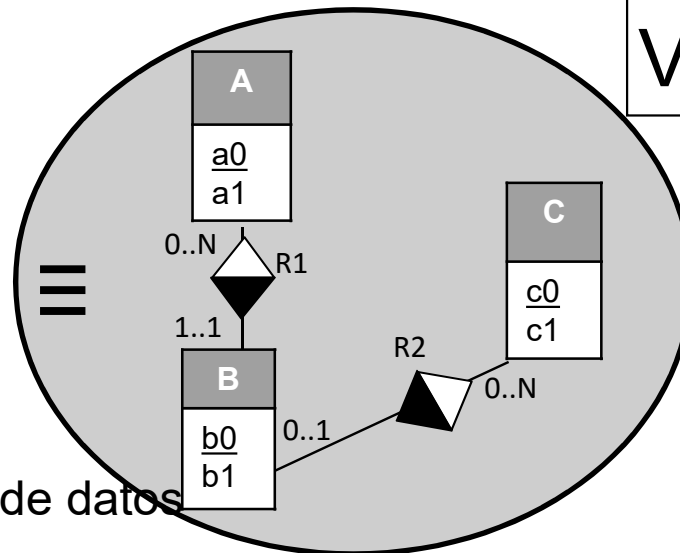
$B(b0, rA, rC)$

C.P.: b0

C. Ajena: $rA \rightarrow A$

C. Ajena: $rC \rightarrow C$

V.N.N.: rA



diseño de bases de datos