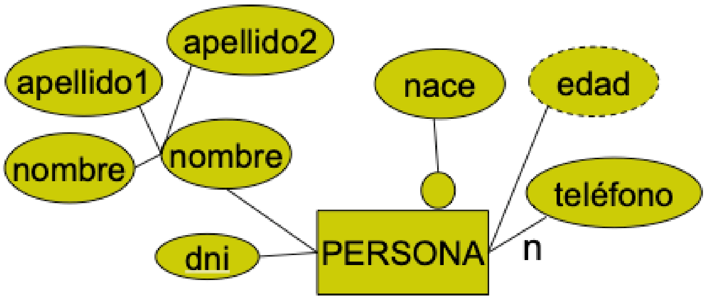
TEMA 09 – DER A MODELO RELACIONAL (MAPEO A TABLAS)

Paso a modelo relacional

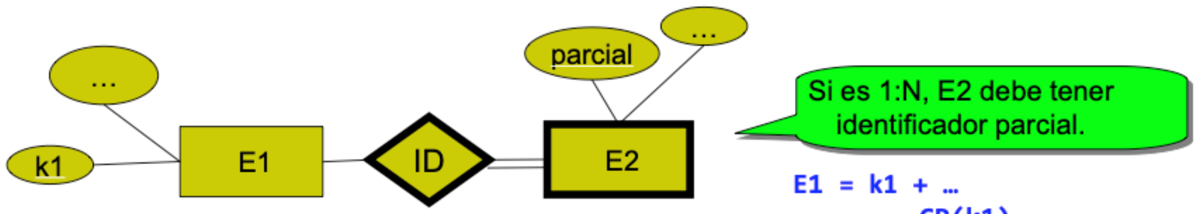
- Entidades y atributos
  - Toda entidad genera una tabla (casi siempre)
  - Cada atributo se corresponde con otro de la tabla
  - Excepciones:
    - Atributos compuestos
      - El modelo relacional solo admite atributos atómicos
      - La tabla tendrá los atributos que componen el atributo compuesto
    - Atributos multivaluados
      - Generan una nueva tabla
      - Clave primaria: la de la entidad origen junto al atributo multivaluado (mínimo uno de ellos)
      - Parte de la clave primera de la nueva tabla será una clave ajena de la tabla de la entidad




```
PERSONA = dni + nombre + apellido1 +
          apellido2 + nace
          C.P(dni)
          VNN(nace)
          Edad= hoy - nace

TELEFONOS = dni + telefono
          C.P(dni + telefono)
          C.Aj(dni) → PERSONA(dni)
```

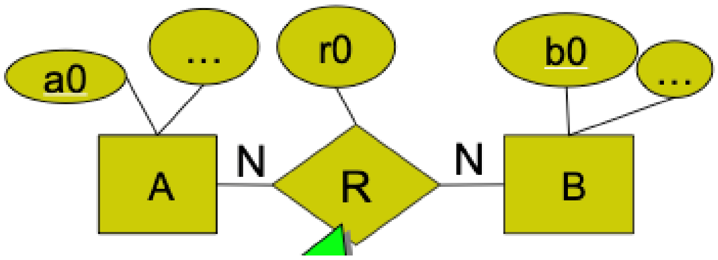
    - Atributos obligatorios: además de crear la columna, se indica VNN(atributo)
    - Atributos derivados: podría incluirse la fórmula con la que se calcula
- Entidades débiles
  - Generan tabla (la relación NUNCA genera tabla)
  - Primero vamos generando tablas desde las más fuertes hasta las más débiles (en caso de haber una cadena de relaciones débiles)
  - En las débiles, incluimos en su clave primaria la clave de la fuerte (o fuertes)
  - Se une junto a su clave parcial (si la tiene) para identificarse
  - Las claves primarias de las propietarias, que forman la clave primaria de la débil, son además claves ajenas de las tablas de las entidades fuertes




```
E1 = k1 + …
     CP(k1)
E2 = k1 + parcial + …
     CP(k1,parcial)
     C.Aj(k1) → E1(k1)
```

Relaciones

- Relaciones binarias
  - Cardinalidad N:N
    - En relaciones muchos a muchos siempre se genera tabla

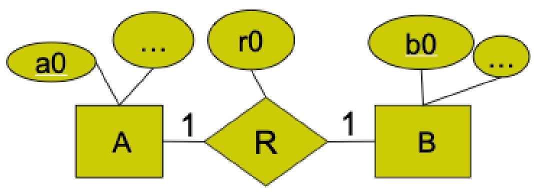



```
A = a0 + …
    CP(a0)
B = b0 + …

    CP(b0)
R = a0 + b0 + r0
    CP(a0,b0)
    C.Aj: a0 → A(a0)
    C.Aj: b0 → B(b0)
```

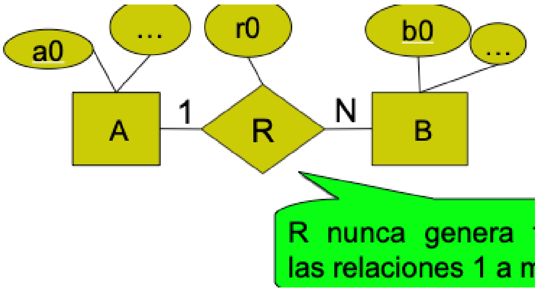
    - A y B siguen igual (cada una hace su tabla)
    - R genera una tabla con:
      - Sus atributos
      - Clave primaria: a0, b0
      - A su vez son claves ajenas de sus respectivas tablas
  - Cardinalidad 1:1
    - No genera tabla
    - La clave de una entidad pasa a la otra como clave ajena
    - Si en una fila no hay relación, la alternativa queda como *null*
    - Los atributos de la relación (r0) también se van a la tabla de la entidad del lado al que pasamos la ajena

```
A = a0 + …
    CP(a0)
B = b0 + a0 + …
    CP(b0)
    C.Alt(a0)
    C.Aj: a0 → A(a0)
```



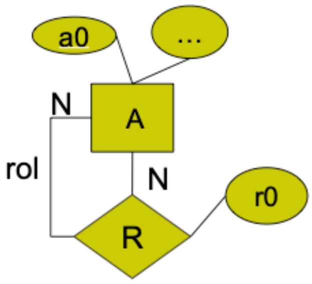
- Cardinalidad 1:N
  - No genera tabla
  - La clave primaria de la entidad con cardinalidad 1 va a la tabla de la entidad con cardinalidad N (clave ajena)
  - Los atributos de la relación (r0) también se van a la tabla de la entidad del lado N

```
A = a0 + …
       CP(a0)
B = b0 + … + a0 + r0
       CP(b0)
       C.Aj: a0 → A(a0)
```



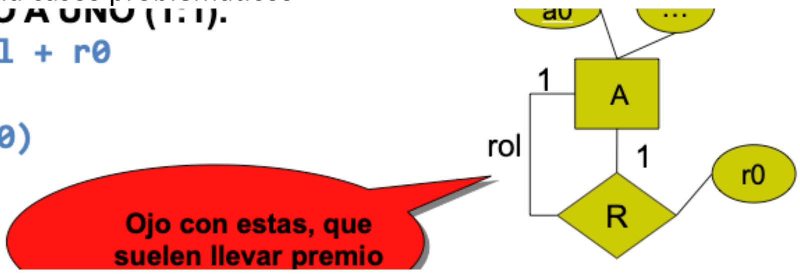
- Relaciones unarias
  - Cardinalidad N:N
    - Es una binaria: en vez de 2 entidades hay 1 y a los atributos (no pueden repetirse) se les cambia su nombre por el de rol
    - Tabla 1: la de A
    - Tabla 2: la de R (A + roles + r0)

```
A = a0 + …
    CP(a0)
R = a0 + rol + r0
    CP(a0,rol)
    C.Aj: a0 → A(a0)
    C.Aj: rol → A(a0)
```



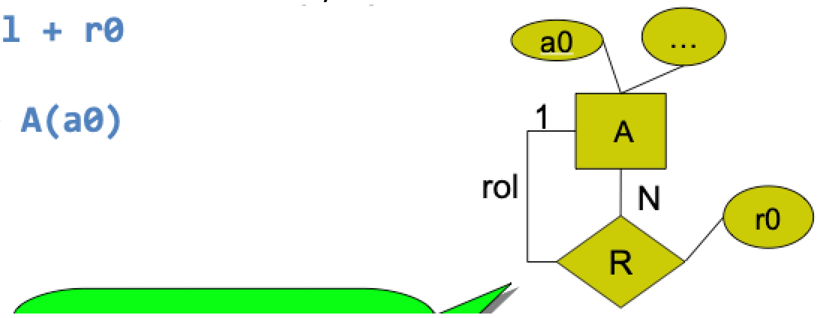
  - Cardinalidad 1:1
    - 1 tabla
    - Habría casos problemáticos

```
A = a0 + … + rol + r0
CP(a0)
C.Aj: rol → A(a0)
C.Alt(rol)
```



  - Cardinalidad 1:N
    - 1 tabla
    - Se almacenan en A los de rol y los r0

```
A = a0 + … + rol + r0
    CP(a0)
    C.Aj: rol → A(a0)
```

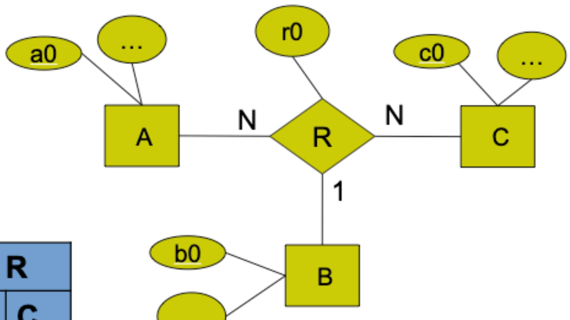


- Relaciones ternarias
    - Cardinalidad 1:N:N
        - Tabla A: normal con todos sus atributos
        - Tabla B: normal con todos sus atributos
        - Tabla C: normal con todos sus atributos
        - Tabla R:
            - Coge las primarias de las 3 tablas (claves ajenas)
            - Coge los atributos propios
            - Las claves primarias son las 2 de la cardinalidad N
            - La de la cardinalidad 1 (B) se pone VNN para evitar que sea una binaria
                - Si además queremos evitar que un mismo B tenga más de una pareja de (a,c), podemos añadir C.Alt(b0)

```
A = a0 + …
B = b0 + …
C = c0 + …
R = a0 + c0 + b0 + r0
    CP(a0,c0)
    C.Aj: a0 → A
    C.Aj: b0 → B
    C.Aj: c0 → C
    VNN(b0)
```

ar que
binaria

Tabla R		
A	B	C

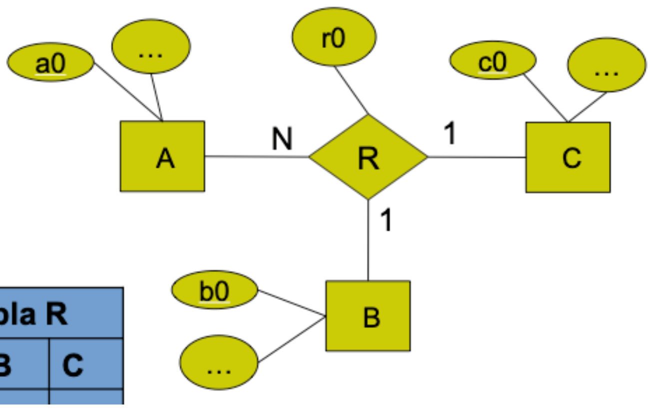


        - Cada pareja (a,c) puede, como máximo, relacionarse con un elemento de B
    - Cardinalidad 1:1:N
        - Tabla A: normal con todos sus atributos
        - Tabla B: normal con todos sus atributos
        - Tabla C: normal con todos sus atributos
        - Tabla R:
            - Coge las primarias de las 3 tablas (claves ajenas)
            - Coge los atributos propios
            - Las claves primarias son la de la cardinalidad N y una de la 1
            - Las de cardinalidad 1 son clave alternativa (a0, c0)
            - La clave ajena que no está como primaria es VNN

```
A = a0 + …
B = b0 + …
C = c0 + …
R = a0 + b0 + c0 + r0
    CP(a0,b0)
    C.Aj: a0 → A
    C.Aj: b0 → B
    C.Aj: c0 → C
    C.Alt(a0,c0)
    VNN(c0)
```

io (a,b) es PK un

Tabla R		
A	B	C



- Cardinalidad 1:1:1
    - Cada tabla normal
    - Tabla R:
        - Coge las primarias de las 3 tablas (claves ajenas)
        - Coge los atributos propios
        - 2 de las claves forman la clave primaria (a0, b0)
        - Las otras dos combinaciones son claves alternativas (a0, c0) y (b0, c0)

```
A = a0 + …
B = b0 + …
C = c0 + …
R = a0 + b0 + c0 + r0
   CP(a0,b0)
   C.Aj: a0 → A
   C.Aj: b0 → B
   C.Aj: c0 → C
   C.Alt(a0,c0)
   C.Alt(b0,c0)
   VNN(c0)
```

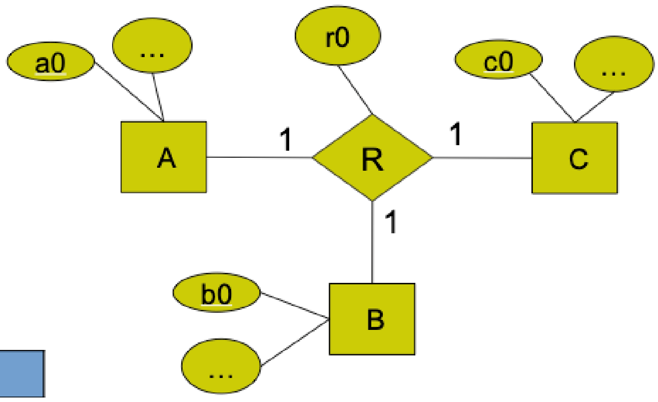


Tabla R

- Cardinalidad N:N:N
    - Cada tabla normal
    - Tabla R:
        - Coge las primarias de las 3 tablas (claves ajenas)
        - Coge los atributos propios
        - La clave primaria son las 3: CP(a0, b0, c0)
    - Cada pareja puede relacionarse con muchos elementos de la otra entidad

```
A = a0 + …
B = b0 + …
C = c0 + …
R = a0 + b0 + c0 + r0
  CP(a0,c0,b0)
   C.Aj: a0 → A
   C.Aj: b0 → B
   C.Aj: c0 → C
```

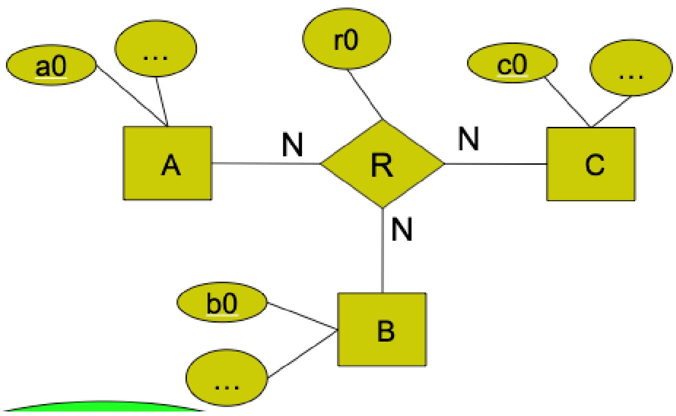
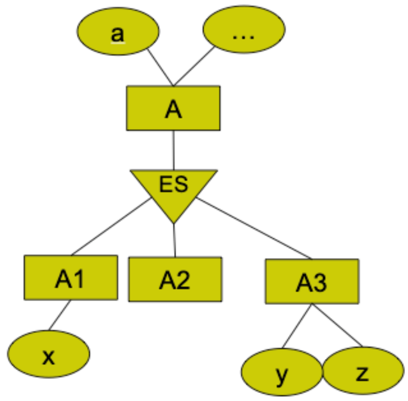


Tabla R		
A	B	C

- Especializaciones
    - Solapada (no disjunta) y parcial
        - Tabla A normal
        - Tabla A1: clave de A (CP) y sus atributos exclusivos
        - Tabla A2: clave de A (CP) y sus atributos exclusivos
        - Tabla A3: clave de A (CP) y sus atributos exclusivos

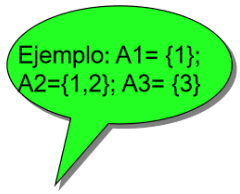
```
A  = a + …
     CP(a)
A1 = a + x
     CP(a)
     C.Aj: a → A
A2 = a
     CP(a)
     C.Aj: a → A
A3 = a + y + z
     CP(a)
     C.Aj: a → A
```

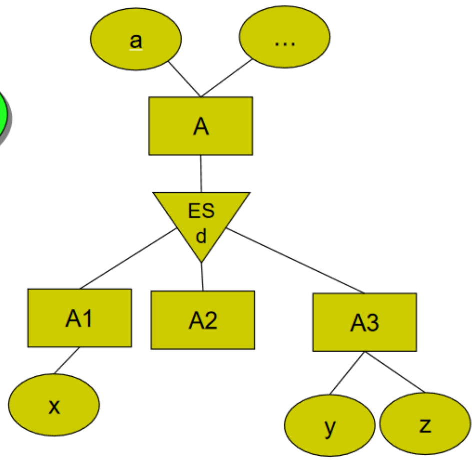


- Exclusiva (disjunta) y parcial
  - Tabla A normal
  - Tabla A1: clave de A (CP) y sus atributos exclusivos
  - Tabla A2: clave de A (CP) y sus atributos exclusivos
  - Tabla A3: clave de A (CP) y sus atributos exclusivos
  - Aserciones
    - ASERCIÓN 1: A1[a] ∩ A2[a] = ∅ (intersección, conjunto vacío)
      - Que no tienen valores en común
    - ASERCIÓN 2: A1[a] ∩ A3[a] = ∅
    - ASERCIÓN 3: A2[a] ∩ A3[a] = ∅
    - NO equivalen a: A1[a] ∩ A2[a] ∩ A3[a] = ∅

DISJUNTA Y PARCIAL

```
A  = a + …
     CP(a)
A1 = a + x
     CP(a)
     C.Aj: a → A
A2 = a
     CP(a)
     C.Aj: a → A
A3 = a + y + z
     CP(a)
     C.Aj: a → A
```

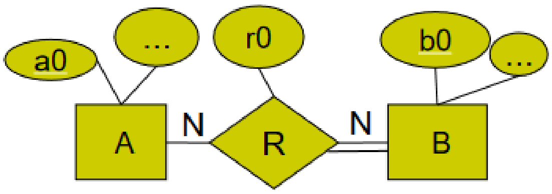




ASERCIÓN 1: A1[a] ∩ A2[a] = ∅
ASERCIÓN 2: A1[a] ∩ A3[a] = ∅
ASERCIÓN 3: A2[a] ∩ A3[a] = ∅

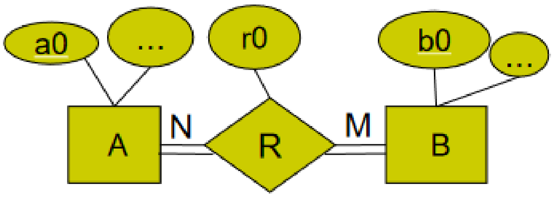
- Relaciones obligatorias
  - Relaciones binarias
    - Cardinalidad N:N y 1 lado participación total
      - Se obliga a la entidad B a que se relacione a través de B con A
      - Tabla A: igual
      - Tabla B: Igual
      - Tabla R:
        - Claves primarias de A y B más sus atributos propios
        - Claves ajenas de A y B
        - ASERCIÓN 1: B[b0] ⊆ R[b0] (está incluida sí o sí)

A = a0 + ….
B = b0 + ….
R = a0 + b0 + r0
C.Aj: a0→A
C.Aj; b0→B
ASERCIÓN 1: B[b0] ⊆ R[b0]



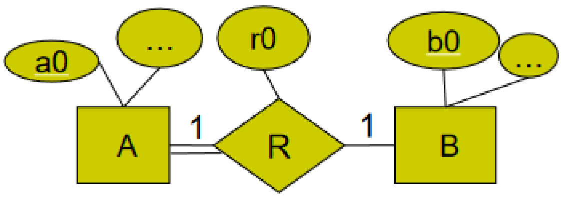
- Cardinalidad N:N y 2 lados participación total
    - Se obliga a la entidad B a que se relaciones a través de R con A y viceversa
    - Tabla A: igual
    - Tabla B: igual
    - Tabla R:
        - Claves primarias de A y B más sus atributos propios
        - Claves ajenas de A y B
        - ASERCIÓN 1: B[b0] ⊆ R[b0]
        - ASERCIÓN 2: A[a0] ⊆ R[a0]

A = a0 + ….
B = b0 + ….
R = a0 + b0 + r0
C.Aj: a0→A
C.Aj; b0→B
ASERCIÓN 1: B[b0] ⊆ R[b0]
ASERCIÓN 2: A[a0] ⊆ R[a0]



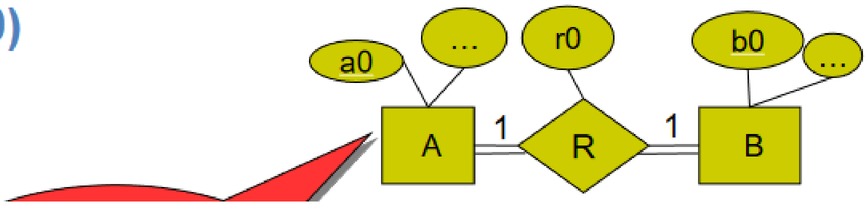
- Cardinalidad 1:1 y 1 participación total
    - Tabla A:
        - Clave de A y sus atributos
        - Clave ajena de B (clave alternativa)
        - VNN(b0)
        - Atributos de R
    - Tabla B: clave de B y sus atributos

```
A = a0 + … + b0 + r0
  C.Alt(b0)
  VNN(b0)
  C.Aj: b0 → B
B = b0 + …
```



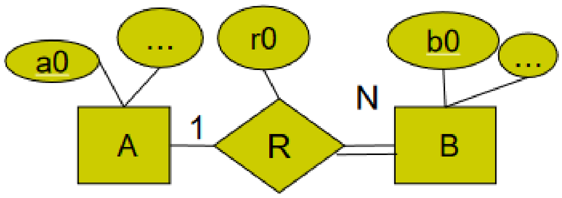
- Cardinalidad 1:1 y 2 participaciones totales
    - Fusiona todos los elementos en una única tabla
    - Es el único caso en el que las entidades no generan tablas
    - Tabla R:
        - CP de A y sus atributos
        - CP de B y sus atributos
        - CP(a0)
        - C.Alt(b0)
        - VNN(b0)

R = a0 + … + b0 + … + r0
C.Alt(b0)
VNN(b0)



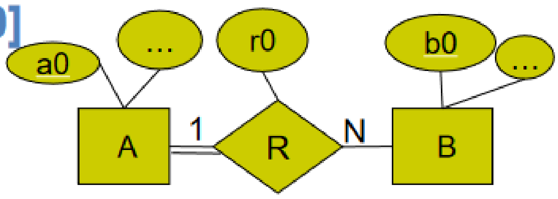
- Cardinalidad 1:N y participación total en el lado N
    - Tabla A: igual
    - Tabla B:
        - Su CP y atributos
        - C.Aj. de A
        - Atributos de R
        - VNN(a0)

A = a0 + …
B = b0 + … + a0 + r0
C.Aj: a0→A;
VNN(a0)



- Cardinalidad 1:N y participación total en el lado 1
    - Tabla A: igual
    - Tabla B:
        - Su CP y atributos
        - C.Aj. de A
        - ASERCIÓN 1: A[a0] ⊆ B[a0]

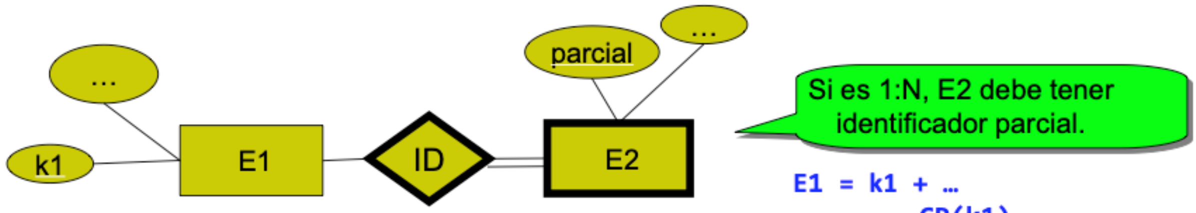
A = <u>a0</u> + …
B = <u>b0</u> + … + a0 + r0
C.Aj: a0→A;
ASERCIÓN 1: A[a0]⊆B[a0]



- Cardinalidad 1:N y participación total en ambos lados
    - Tabla A: igual
    - Tabla B:
        - Su CP y atributos
        - C.Aj. de A
        - Atributos de R
        - VNN(a)
        - ASERCIÓN 1: A[a0] ⊆ B[a0]
- Relaciones unarias

- Generalización

RESUMEN RELACIONES
Entidades débiles:

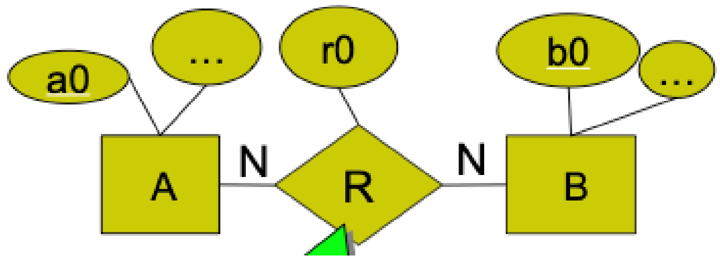



```
E1 = k1 + …
     CP(k1)
E2 = k1 + parcial + …
     CP(k1,parcial)
     C.Aj(k1) → E1(k1)
```

Binarias

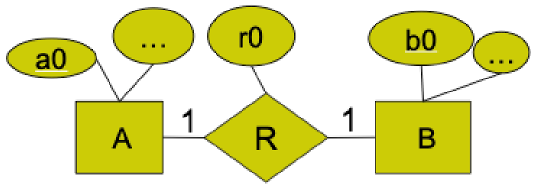
- N:N

```
A = a0 + …
    CP(a0)
B = b0 + …
    CP(b0)
R = a0 + b0 + r0
    CP(a0,b0)
    C.Aj: a0 → A(a0)
    C.Aj: b0 → B(b0)
```



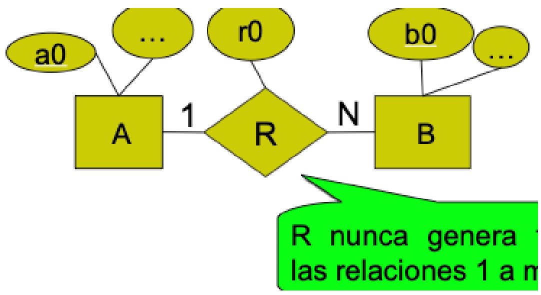
- 1:1

```
A = a0 + …
    CP(a0)
B = b0 + a0 + …
     CP(b0)
     C.Alt(a0)
     C.Aj: a0 → A(a0)
```



- 1:N

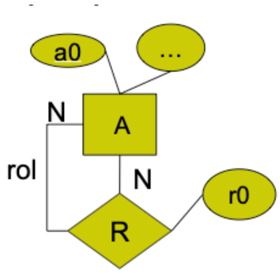
```
A = a0 + …
       CP(a0)
B = b0 + … + a0 + r0
       CP(b0)
       C.Aj: a0 → A(a0)
```



Unarias

- N:N

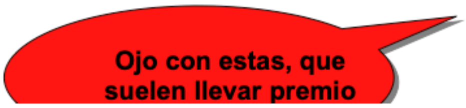
```
A = a0 + …
    CP(a0)
R = a0 + rol + r0
    CP(a0,rol)
    C.Aj: a0 → A(a0)
    C.Aj: rol → A(a0)
```

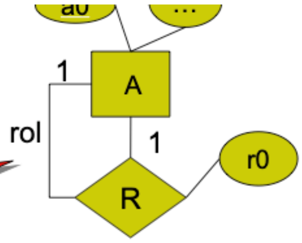


- 1:1

ARDINALIDAD UNO A UNO (1:1).

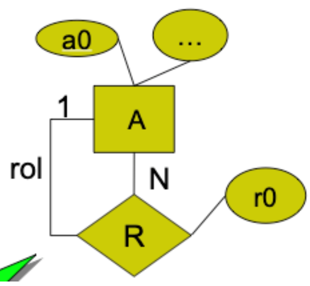
```
A = a0 + … + rol + r0
CP(a0)
C.Aj: rol → A(a0)
C.Alt(rol)
```





- 1:N

```
A = a0 + … + rol + r0
    CP(a0)
     C.Aj: rol → A(a0)
```



Relaciones ternarias (0 mínima)

- 1:N:N

```
A = a0 + …
B = b0 + …
C = c0 + …
R = a0 + c0 + b0 + r0
    CP(a0,c0)
    C.Aj: a0 → A
    C.Aj: b0 → B
    C.Aj: c0 → C
    VNN(b0)
```

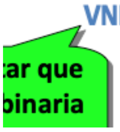
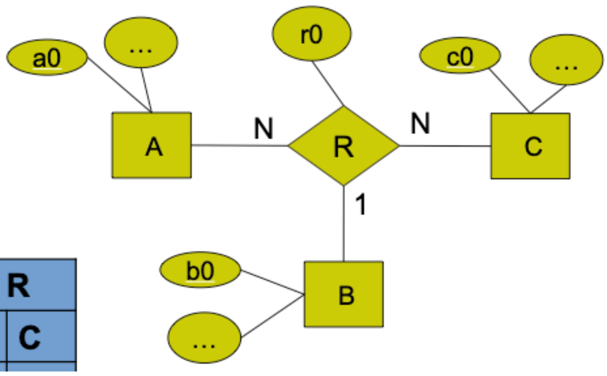


Tabla R		
A	B	C



- 1:1:N

A = a0 + …
B = b0 + …
C = c0 + …
R = a0 + b0 + c0 + r0
CP(a0,b0)
C.Aj: a0 → A
C.Aj: b0 → B
C.Aj: c0 → C
C.Alt(a0,c0)
VNN(c0)

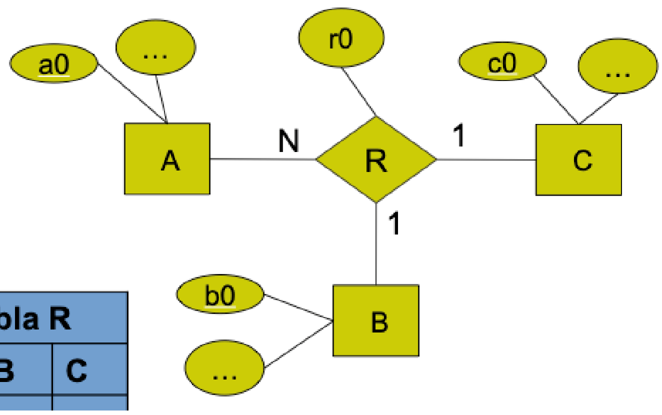


Tabla R		
A	B	C

o (a,b) es PK un

- 1:1:1

A = a0 + …
B = b0 + …
C = c0 + …
R = a0 + b0 + c0 + r0
CP(a0,b0)
C.Aj: a0 → A
C.Aj: b0 → B
C.Aj: c0 → C
C.Alt(a0,c0)
C.Alt(b0,c0)
VNN(c0)

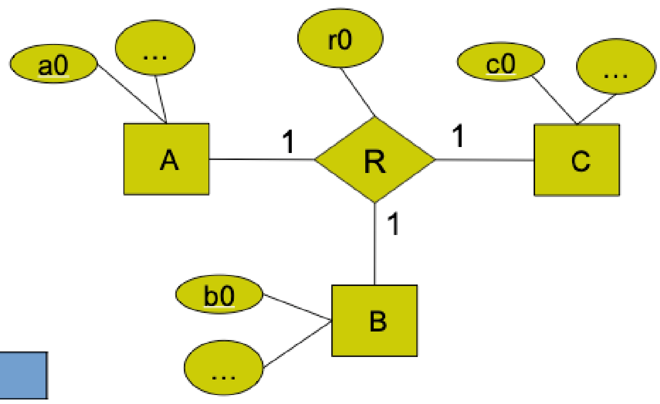
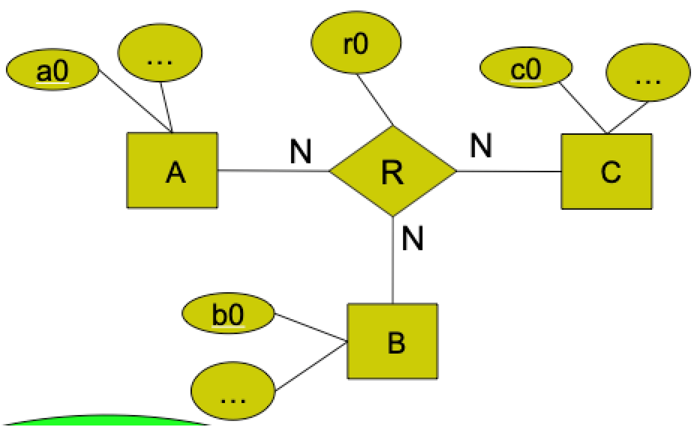


Tabla R

- N:N:N

A = a0 + …
B = b0 + …
C = c0 + …
R = a0 + b0 + c0 + r0
CP(a0,c0,b0)
C.Aj: a0 → A
C.Aj: b0 → B
C.Aj: c0 → C

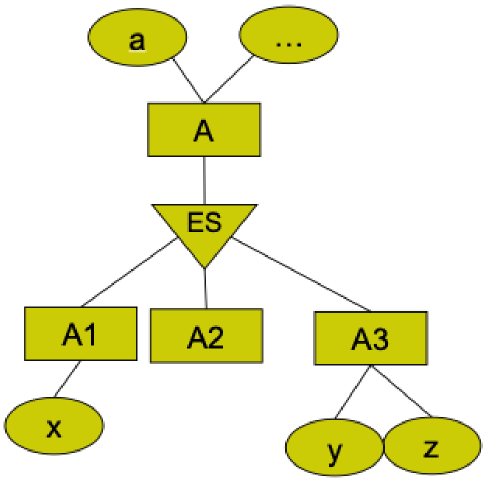
Tabla R		
A	B	C



Especializaciones

- No disjunta/solapada y parcial

```
A  = a + …
     CP(a)
A1 = a + x
     CP(a)
     C.Aj: a → A
A2 = a
     CP(a)
     C.Aj: a → A
A3 = a + y + z
     CP(a)
     C.Aj: a → A
```

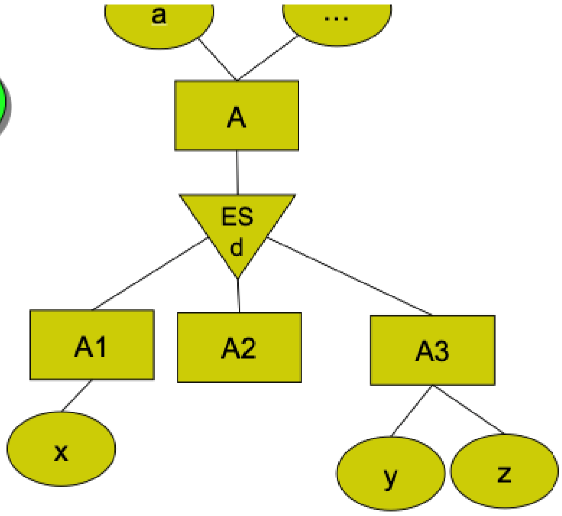


- Disjunta/exclusiva y parcial

```
A  = a + …
     CP(a)
A1 = a + x
     CP(a)
     C.Aj: a → A
A2 = a
     CP(a)
     C.Aj: a → A
A3 = a + y + z
     CP(a)
     C.Aj: a → A
```

ASERCIÓN 1: $A1[a] \cap A2[a] = \varnothing$
ASERCIÓN 2: $A1[a] \cap A3[a] = \varnothing$
ASERCIÓN 3: $A2[a] \cap A3[a] = \varnothing$

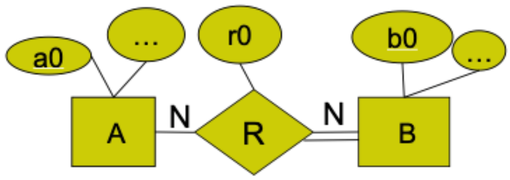
Ejemplo: A1= {1}; A2={1,2}; A3= {3}



Relaciones obligatorias binarias

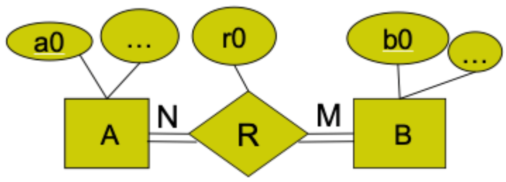
- N:N y 1 participación total

A = a0 + ….
B = b0 + ….
R = a0 + b0 + r0
C.Aj: a0→A
C.Aj; b0→B
ASERCIÓN 1: B[b0] ⊆ R[b0]



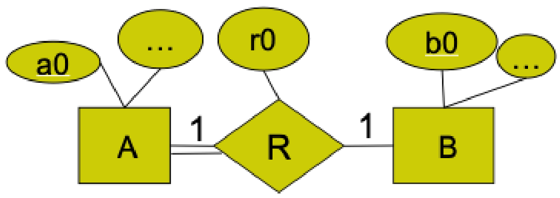
- N:N y 2 participaciones totales

A = a0 + ….
B = b0 + ….
R = a0 + b0 + r0
C.Aj: a0→A
C.Aj; b0→B
ASERCIÓN 1: B[b0] ⊆ R[b0]
ASERCIÓN 2: A[a0] ⊆ R[a0]



- 1:1 y 1 participación total

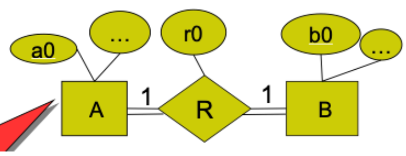
A = a0 + … + b0 + r0
C.Alt(b0)
VNN(b0)
C.Aj: b0 → B
B = b0 + …



- 1:1 y 2 participaciones totales

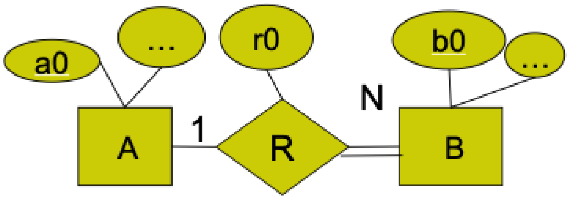
Es el único caso en que las entidades no generan tablas.

R = a0 + … + b0 + … + r0
C.Alt(b0)
VNN(b0)



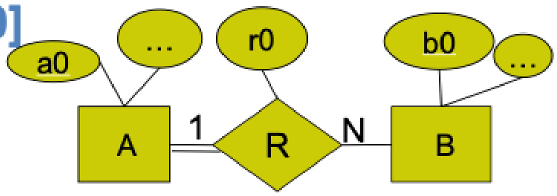
- 1:N y participación total en el lado N

A = a0 + …
B = b0 + … + a0 + r0
C.Aj: a0→A;
VNN(a0)



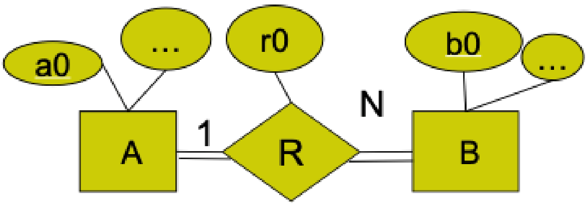
- 1:N y participación total en el lado 1

A = a0 + …
B = b0 + … + a0 + r0
C.Aj: a0→A;
ASERCIÓN 1: A[a0]⊆B[a0]



- 1:N y 2 participaciones totales

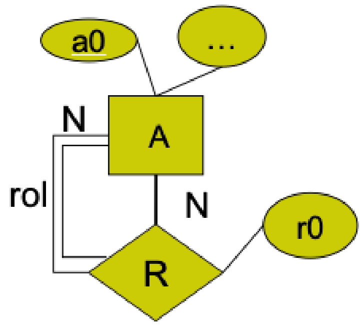
A = a0 + …
B = b0 + … + a0 + r0
C.Aj: a0→A;
VNN(a0)
ASERCIÓN 1: A[a0]⊆B[a0]



Relaciones obligatorias y unarias

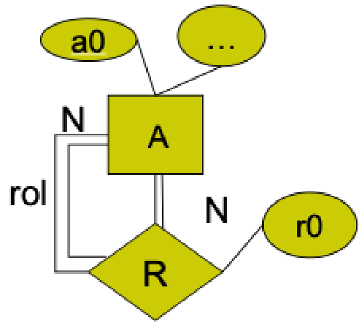
- N:N y 1 participación total

A = a0 + …
R = a0 + rol + r0
C.Aj: a0→A
C.Aj: rol→A
ASERCIÓN 1: A[a0]⊆R[rol]



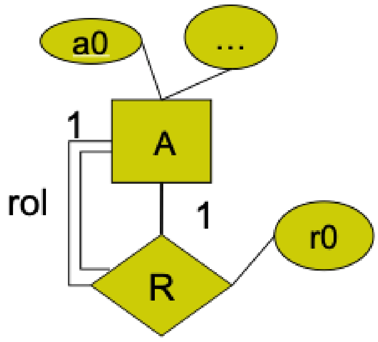
- N:N y 2 participaciones totales

A = a0 + …
R = a0 + rol + r0
C.Aj: a0→A
C.Aj: rol→A
ASERCIÓN 1: A[a0]⊆R[a0]
ASERCIÓN 2: A[a0]⊆R[rol]



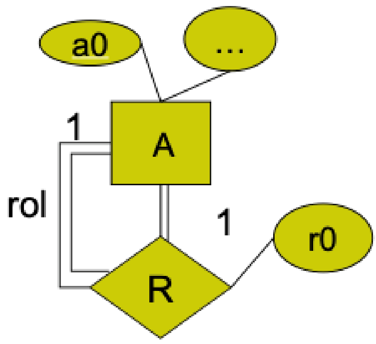
- 1:1 y 1 participación total

A = a0 + … + rol + r0
C.Alt(rol)
VNN(rol)
C.Aj: rol→A



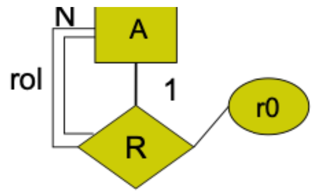
- 1:1 y 2 participaciones totales

A = a0 + … + rol + r0
C.Alt(rol)
VNN(rol)
C.Aj: rol → A
ASERCIÓN 1: A[a0] ⊆ A[rol]



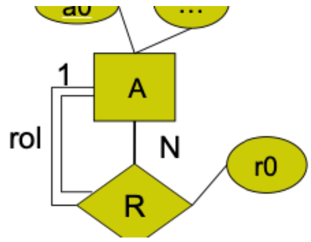
- 1:N y participación total en N

A = a0 + … + rol + r0
C.Aj: rol→A;
VNN(rol)



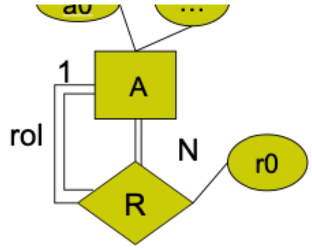
- 1:N y participación total en 1

A = a0 + … + rol + r0
C.Aj: rol→A;
ASERCIÓN 1: A[a0]⊆A[rol]



- 1:N y 2 participaciones totales

A = a0 + … + rol + r0
C.Aj: rol→A;
VNN(rol)
ASERCIÓN 1: A[a0]⊆A[rol]



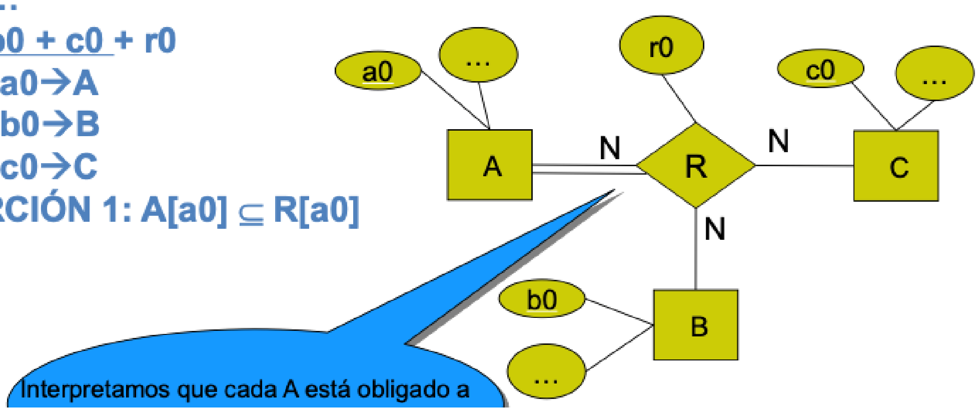
Relaciones obligatorias y ternarias

- 1 sola participación total

Sea cual sea la cardinalidad, hay que usar aserciones.

EJEMPLO N:N:N.

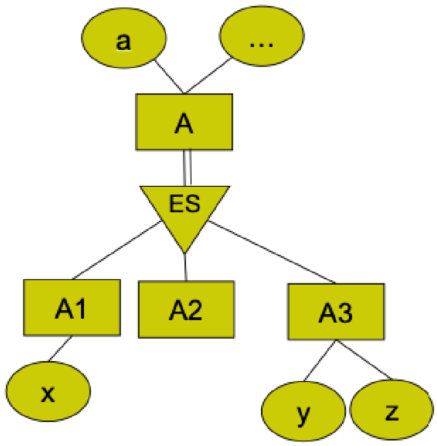
A = a0 + …
B = b0 + …
C = c0 + …
R = a0 + b0 + c0 + r0
C.Aj: a0→A
C.Aj: b0→B
C.Aj: c0→C
ASERCIÓN 1: A[a0] ⊆ R[a0]



Generalizaciones

- No disjunta y total

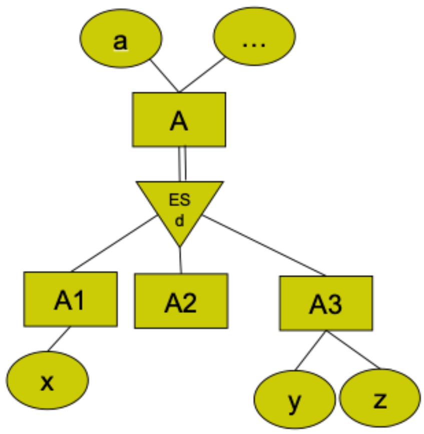
```
A  = a + …
     CP(a)
A1 = a + x
     CP(a)
     C.Aj: a → A
A2 = a
     CP(a)
     C.Aj: a → A
A3 = a + y + z
     CP(a)
     C.Aj: a → A
```



ASERCIÓN1: $A[a] \subseteq A1[a] \cup A2[a] \cup A3[a]$

- Disjunta y total

```
A  = a + …
     CP(a)
A1 = a + x
     CP(a)
     C.Aj: a → A
A2 = a
     CP(a)
     C.Aj: a → A
A3 = a + y + z
     CP(a)
     C.Aj: a → A
```



ASERCIÓN 1: $A1[a] \cap A2[a] = \varnothing$
ASERCIÓN 2: $A1[a] \cap A3[a] = \varnothing$
ASERCIÓN 3: $A2[a] \cap A3[a] = \varnothing$
ASERCIÓN 4: $A[a] \subseteq A1[a] \cup A2[a] \cup A3[a]$