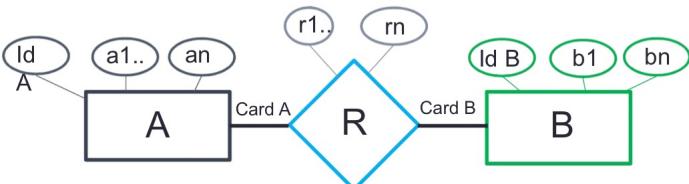


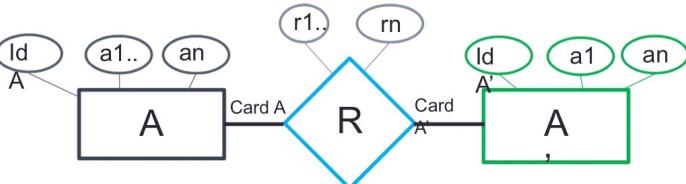
# MAPEO DE MODELOS

## Resumen Binarias



Card A	Card B	Relaciones	CC	PK	FK
(1,N)	(1,N)	A ( <u>idA</u> , a1, ... an)	{}{{idA}}	{}{{idA}}	-
(1,N)	(0,N)	B ( <u>idB</u> , b1, ... bn)	{}{{idB}}	{}{{idB}}	-
(0,N)	(1,N)	R ( <u>idA</u> , <u>idB</u> , r1..rn)	{}{{idA,idB}}	{}{{idA},{}{idB}}	{}{{idA}};{}{{idB}}
(0,N)	(0,N)				
(1,N)	(1,1)	A ( <u>idA</u> , a1, ... an, idB, r1..rn)	{}{{idA}}	{}{{idA}}	{}{{idB}}
(0,N)	(1,1)	B ( <u>idB</u> , b1, ... bn)	{}{{idB}}	{}{{idB}}	-
(1,N)	(0,1)	A ( <u>idA</u> , a1, ... an)	{}{{idA}}	{}{{idA}}	-
(0,N)	(0,1)	B ( <u>idB</u> , b1, ... bn)	{}{{idB}}	{}{{idB}}	-
(1,1)	(1,1)	R ( <u>idA</u> , idB, r1..rn)	{}{{idA}};{}{{idB}}	{}{{idA}}	{}{{idB}}
(0,1)	(1,1)	A ( <u>idA</u> , a1, ... an, idB, r1..rn, b1, ... bn)	{}{{idA}};{}{{idB}}	{}{{idA}}	{}{{idB}}
(0,1)	(1,1)	B ( <u>idB</u> , b1, ... bn)	{}{{idB}}	{}{{idB}}	-
(0,1)	(0,1)	A ( <u>idA</u> , a1, ... an)	{}{{idA}}	{}{{idA}}	-
		B ( <u>idB</u> , b1, ... bn)	{}{{idB}}	{}{{idB}}	-
		R ( <u>idA</u> , idB, r1..rn)	{}{{idA}};{}{{idB}}	{}{{idA}}	{}{{idB}}

## Resumen Unarias



Card A	Card A'	Relaciones	CC	PK	FK
(1,N)	(1,N)	A ( <u>idA</u> , a1, ... an)	{}{{idA}}	{}{{idA}}	-
(1,N)	(0,N)	R ( <u>idA</u> , <u>idA'</u> , r1..rn)	{}{{idA}};{}{{idA'}}	{}{{idA}},{}{{idA'}}	{}{{idA}};{}{{idA'}}
(0,N)	(1,N)				
(1,N)	(1,1)	A ( <u>idA</u> , a1, ... an, idA', r1..rn)	{}{{idA}}	{}{{idA}}	{}{{idA'}}
(0,N)	(1,1)				
(1,N)	(0,1)	A ( <u>idA</u> , a1, ... an)	{}{{idA}}	{}{{idA}}	-
(0,N)	(0,1)	R ( <u>idA</u> , idA', r1..rn)	{}{{idA}}	{}{{idA}}	{}{{idA}};{}{{idA'}}
(0,1)	(1,1)	A ( <u>idA</u> , a1, ... an, idA', r1..rn)	{}{{idA}};{}{{idA'}}	{}{{idA}}	{}{{idA'}}
(0,1)	(0,1)	A ( <u>idA</u> , a1, ... an)	{}{{idA}}	{}{{idA}}	-
		R ( <u>idA</u> , idA', r1..rn)	{}{{idA}};{}{{idA'}}	{}{{idA}}	{}{{idA'}}

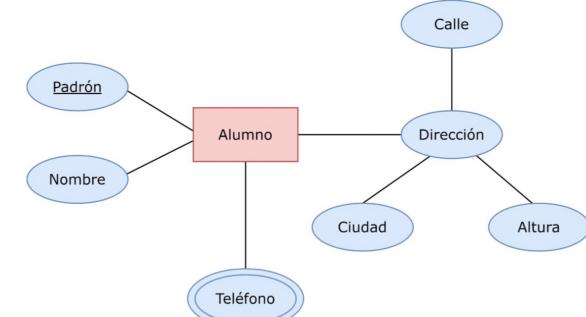
## Resumen Ternarias

Relaciones		CC	PK	FK
A ( <u>idA</u> , a1, ... an)		{}{{idA}}	{}{{idA}}	-
B ( <u>idB</u> , b1, ... bn)		{}{{idB}}	{}{{idB}}	-
C ( <u>idC</u> , c1, ... cb)		{}{{idC}}	{}{{idC}}	-

Para relación R

Card A	Card B	Card C	Relación	CC	PK	FK
N	N	N	R ( <u>idA</u> , <u>idB</u> , <u>idC</u> , r1..rn)	{}{{idA}, idB, idC}}	{}{{idA}, idB, idC}}	{}{{idA}};{}{{idB}};{}{{idC}}
N	N	1	R ( <u>idA</u> , <u>idB</u> , idC, r1..rn)	{}{{idA}, idB}}	{}{{idA}, idB}}	{}{{idA}};{}{{idB}};{}{{idC}}
N	1	1	R ( <u>idA</u> , <u>idB</u> , idC, r1..rn)	{}{{idA}, idB};{}{{idA}, idC}}	{}{{idA}, idB}}	{}{{idA}};{}{{idB}};{}{{idC}}
1	1	1	R ( <u>idA</u> , <u>idB</u> , idC, r1..rn)	{}{{idA}, idB};{}{{idA}, idC};{}{{idB}, idC}}	{}{{idA}, idB}}	{}{{idA}};{}{{idB}};{}{{idC}}

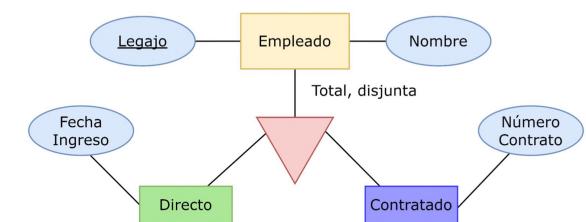
## T.E. FUERTES



Alumnos (Padrón, Nombre, Calle, Altura, Ciudad)

Teléfonos (Padrón, teléfono)

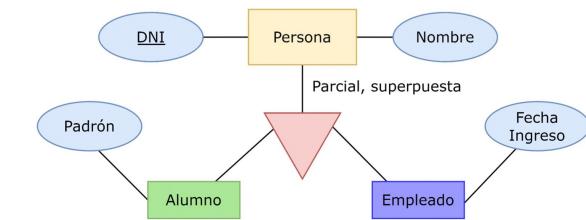
## JERARQUÍAS



Empleados(Legajo, Nombre)

Empleados\_Directos (Legajo, Fecha\_Ingreso)

Empleados\_Contratados (Legajo, Número\_Contrato)

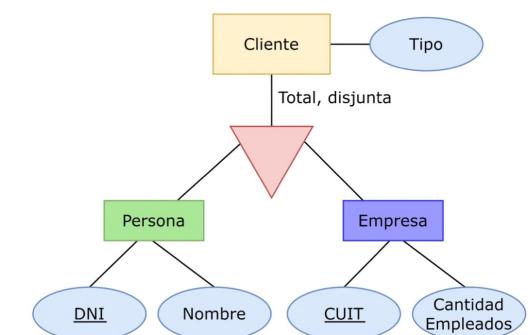


Personas (DNI, Nombre)

Alumnos (DNI, Padrón)

Empleados (DNI, Fecha\_Ingreso)

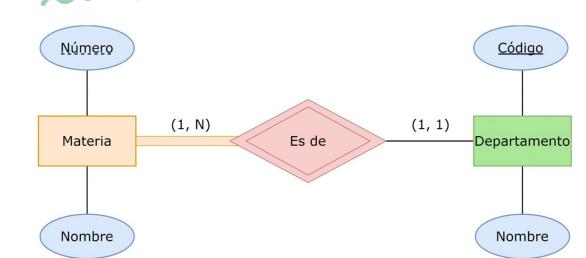
## UNIÓN



Personas (DNI, Nombre, Tipo\_Cliente)

Empresas(CUIT, Cantidad\_Employeados, Tipo\_Cliente)

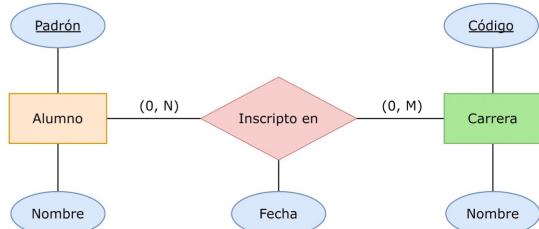
## T.E. DÉBIL



Departamentos (Código, Nombre)

Materias (Código, Número, Nombre)

## T.I. N:M

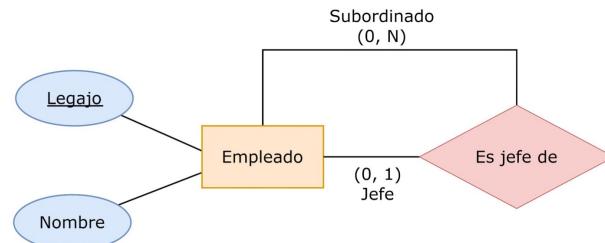


Alumnos (Padrón, Nombre)

Carreras (Código, Nombre)

Inscripciones (Padrón, Código, Fecha)

## VARIOS ROLES

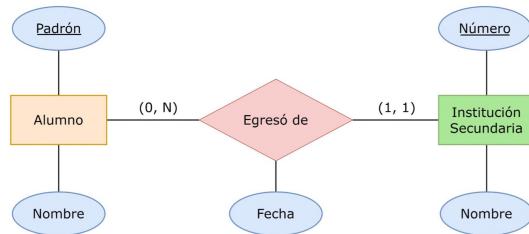


Empleados (Legajo, Nombre)

Jefes (Legajo<sub>Subordinado</sub>, Legajo<sub>Jefe</sub>)

## T.I. N:1

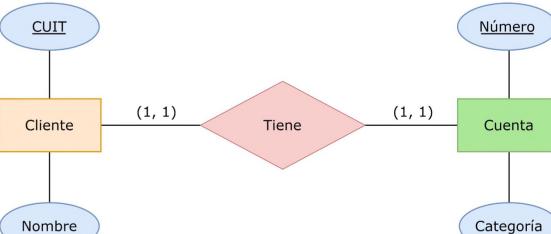
### Participación Total



Alumnos (Padrón, Nombre, Número<sub>Институция</sub>, Fecha<sub>Егриз</sub>)

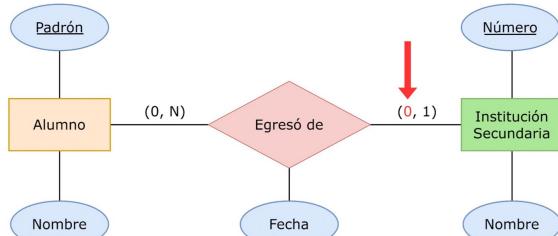
Instituciones (Número, Nombre)

### Participación Total



Clientes (CUIT, Nombre, Número, Categoría)

### Participación Parcial

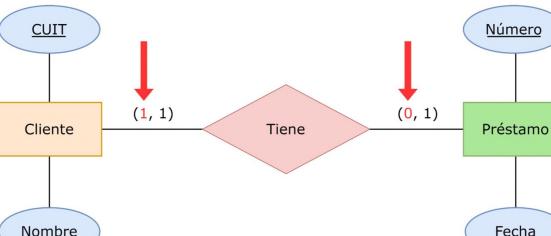


Alumnos (Padrón, Nombre)

Instituciones (Número, Nombre)

Egresos (Padrón, Número, Fecha)

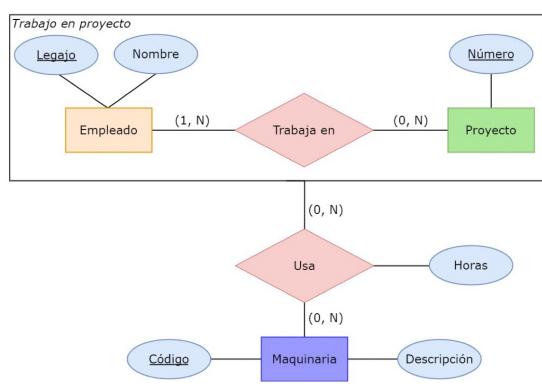
### sin Participación Total



Clientes (CUIT, Nombre)

Préstamos (Número, Fecha, CUIT)

## AGREGACIÓN



Usos (Legajo, Número, Código, Horas)

## CLAVES SUSTITUTAS

Alumnos( <u>Padrón</u> , Nombre)		Departamentos( <u>Código</u> , Nombre)	
<u>Padrón</u>	<u>Nombre</u>	<u>Código</u>	<u>Nombre</u>
103784	Julian Alvarez	66	Electrónica

Materias( <u>Código</u> , <u>Número</u> , Nombre)			Notas( <u>Padrón</u> , <u>Código</u> , <u>Número</u> , Nota)		
<u>Código</u>	<u>Número</u>	<u>Nombre</u>	<u>Padrón</u>	<u>Código</u>	<u>Nota</u>
66	06	Análisis de Circuitos	103784	66	9
66	71	Sistemas Gráficos			

Alumnos( <u>Id</u> , <u>Padrón</u> , Nombre)			Departamentos( <u>Id</u> , <u>Código</u> , Nombre)		
<u>Id</u>	<u>Padrón</u>	<u>Nombre</u>	<u>Id</u>	<u>Código</u>	<u>Nombre</u>
1	103784	Julian Alvarez	1	66	Electrónica

Materias( <u>Id</u> , <u>Id_Departamento</u> , <u>Número</u> , Nombre)				Notas( <u>Id_Alumno</u> , <u>Id_Materia</u> , Nombre)		
<u>Id</u>	<u>Id_Departamento</u>	<u>Número</u>	<u>Nombre</u>	<u>Id_Alumno</u>	<u>Id_Materia</u>	<u>Nota</u>
1	66	06	Análisis de Circuitos	1	2	9
2	66	71	Sistemas Gráficos			

# ALGEBRA RELACIONAL

1. Muestre las películas dirigidas por 'Scorsese' de la década de 1970 (del año 1970 al año 1979).
2. Muestre la/las películas que fueron dirigidas por Hitchcock (Alfred (I)) en las cuales Hitchcock además es actor.
3. Nombre todas las películas en que participó por lo menos un actor de la película 'Life of Brian'. Muestre el título de la película y el actor, omita en la respuesta 'Life of Brian'.
4. Mostrar los nombres y años de filmación, de la/las películas de género dramático (Drama) más vieja/s de la base.
5. Muestre la/las películas que han sido dirigidas con más de 2 directores.

-- EJ 1

```
movies_id = π movie_id, director_id (movies_directors) ∩ π director_id (ρ director_id←id (σ last_name = 'Scorsese' (directors)))
σ year ≥ 1970 ∧ year ≤ 1979 (π name, year, movie_id (ρ movie_id←id (movies)) × π movie_id (movies_id))
```

-- EJ 2

```
movies_hitchcock_director = π movie_id, director_id (movies_directors) × π director_id (ρ director_id←id (π id (σ last_name = 'Hitchcock' (directors))))
```

```
movies_hitchcock_actor = π actor_id (ρ actor_id←id (π id (σ last_name = 'Hitchcock' ∧ first_name = 'Alfred (I)' (actors))) × π actor_id, movie_id (roles))
```

```
movies_id = π movie_id (movies_hitchcock_director) ∩ π movie_id (movies_hitchcock_actor)
```

```
π name, movie_id (ρ movie_id←id (movies)) × π movie_id (movies_id)
```

-- EJ 3

```
actores = π actor_id, movie_id (roles) × π movie_id (σ name = 'Life of Brian' (ρ movie_id←id (movies)))
```

```
movies_id = π movie_id, actor_id (roles) × π actor_id (actores)
```

```
peliculas = π movie_id, title (ρ movie_id←id, title←name (σ name ≠ 'Life of Brian' (movies))) × π actor_id, movie_id (movies_id)
```

```
π title, actor_id (peliculas) × π actor_id, first_name, last_name (ρ actor_id←id (actors))
```

-- EJ 4

```
drama_movies = π movie_id (σ genre = 'Drama' (movies_genres)) × π movie_id, name, year (ρ movie_id←id (movies))
```

```
years = π movies.year drama_movies
```

```
movies_years = π drama_movies1.year (σ (drama_movies1.year > drama_movies2.year) (ρ drama_movies1 drama_movies × ρ drama_movies2 drama_movies))
```

```
year = years - movies_years
```

```
π year, name (movies) × year
```

-- EJ 5

```
R1 = ρ r1 π director_id, movie_id movies_directors
```

```
R2 = ρ r2 π director_id, movie_id movies_directors
```

```
R3 = ρ r3 π director_id, movie_id movies_directors
```

```
dos_directores = π r1.movie_id, r1.director_id, r2.director_id (R1 × r1.director_id ≠ r2.director_id ∧ r1.movie_id = r2.movie_id (R2))
```

```
tres_directores = π r1.movie_id (dos_directores × r1.movie_id = r3.movie_id ∧ r1.director_id ≠ r3.director_id ∧ r2.director_id ≠ r3.director_id (R3))
```

```
π name, movie_id (ρ movie_id←id (movies)) × π movie_id (tres_directores)
```

SINTAXIS ⊕ : =, ≠, >, ≥, <, ≤

Π<sup>v</sup><sub>(col1 ⊕ val1) ∧ (col2 ⊕ val2)</sub> (Tabla) → Selección      Tabla 1 - Tabla 2 → diferencia

Π<sub>col1, col2</sub> (Tabla) → proyección

Tabla 1 × Tabla 2 → producto cartesiano

Ρ<sub>Tabla' (col1')</sub> (Tabla) → redenominación

Tabla 1 ×<sub>col1 ⊕ col2</sub> Tabla 2 → junta

Tabla 1 ∪ Tabla 2 → unión

Tabla 1 \* Tabla 2 → junta matrrix

Tabla 1 ∩ Tabla 2 → intersección

Tabla 1 - Tabla 2 → división

# DISEÑO RELACIONAL

## Cubrimiento Minimal ( $f_{min}$ )

- 1º Dejar todos los lados derechos con 1 único atributo.
- 2º Eliminar los atributos redundantes del lado izquierdo.  
Si  $FD = \{ \dots, AC \rightarrow E \}$  me fijo si  $A^+ \subseteq E$  ó  $C^+ \subseteq E$  y saco algunos
- 3º Eliminar los  $\alpha_f$ 's redundantes:  
Analizo para cada  $\alpha_f : X \rightarrow Y$  si  $Y \subseteq X^+ f - \{X \rightarrow Y\}$  y veo si es redundante

## Cálculo de Claves

- 1º Calcular el cubrimiento minimal.
- 2º Detectar atributos independientes (en ninguno  $\alpha_f$ ) y quitarlos para después.
- 3º Eliminar atributos equivalentes ( $A \rightarrow B$ ,  $B \rightarrow A$ ), dejar sólo uno y reservar para después.
- 4º Construir  $K$  con todos los elementos sólo implicantes (sólo en la parte izquierda). Si  $K^+ = R \Rightarrow K$  es clave.
- 5º Si  $K$  no resultó clave, buscar el conjunto de elementos que estén entre los implicantes pero que pueden ser implicados (están en la parte izquierda y derecha). Agregar alternativamente a  $K$  todos los posibles subconjuntos (todos los de 1 elemento, 2 elementos, etc.) y verificar que  $K$  sea clave calculando la clausura. Observar los subconjuntos que contienen una clave ya calculada.
- 6º Agregar a la clave los elementos independientes.
- 7º Calcular las otras claves con los elementos equivalentes.

## Atributos

### PRIMOS

pertenece a  
alguna clave  
candidate

### NO PRIMOS

no pertenece a  
ninguna clave  
candidate

## Formas Normales

- [2] Ningún atributo no primo depende parcialmente de alguna clave.
- [3] O el lado izquierdo es siempre una superclave o el lado derecho es un atributo primo.
- [BC] El lado izquierdo es siempre una superclave.

## Descomposición en 3FN

$$R(A, F) \rightarrow P = R_1, \dots, R_n, R_i \in 3\text{FN}$$

- ✓ Preservación de dependencias e información

## Construcción

- 1º Calcular el cubrimiento mínimo y las claves de  $R$ .
- 2º Para cada  $\delta f \times \rightarrow A_i$  crear un ER  $R_{\delta} = \{\times \cup A_i\}$
- 3º Si ninguno de los ER contiene una clave de  $R$ , crear un ER adicional que contenga todos los atributos de alguna clave candidate de  $R$ .

## Reagrupamiento

- 4º Unir los ER que tengan una misma clave candidate.
- 5º Si un ER contiene los atributos de otro, eliminar el redundante y unir el mayor con los  $\delta f$  de ambos

## Descomposición en FNBC

$$R(A, F) \rightarrow P = R_1, \dots, R_n, R_i \in \text{FNBC}$$

- ✓ Preservación de la información

Mientras existe un ER  $P \notin \text{FNBC}$ :

Elegir  $P \notin \text{FNBC}$  y encontrar lo  $\delta f \times \rightarrow Y$  que lo viola

Reemplazar  $P$  por  $R_1 = \{\times\}^+$  y  $R_2 = \{P - \{\times\}^+ \cup \times\}$

Procesar recursivamente en  $R_1$  y  $R_2$