



# Metodologías de Modelamiento

**2024**

Profesores del curso



# ÍNDICE

1. El Modelo Relacional
  - Estructura y operaciones relacionales
  - Reglas de integridad
2. Técnicas de modelamiento de datos
  - Notación Barker
    - Convenciones
  - IDEFIX
3. Conclusiones
4. Referencias



- El modelo conceptual: análisis formal para representar la realidad con un modelo semántico
- Herramienta por utilizar: Diagrama Entidad Relación (DER)
- Notación en el DER: entidad, atributos y relación



## Saberes previos

### Elaboración del Modelo Entidad Relación

- Identificar entidades
- Definir atributos
- Establecer relaciones
- Precisar cardinalidades



# Modelo Relacional



## Modelo Relacional

Los datos se agrupan como relaciones de datos y se **representan en forma de tablas**, donde cada fila corresponde a una "tupla" de la relación y cada columna representa a un dominio. No existe otra forma de agrupar datos.

Cada dato es atómico, representa un solo valor del dominio. Además, tenemos que:

- Grado de la relación: es la cantidad de columnas de la tabla
- Cardinalidad: es la cantidad de filas
- Cada columna de la relación representa diferente información. Aunque dos columnas de una misma relación pueden tener exactamente el mismo dominio, representarán información diferente. En efecto, cada columna es un uso particular de un dominio.
  - (1ra regla de Codd - 1985 "Regla de la información": Toda información debe expresarse sólo en términos de valores en relaciones).



## Modelo Relacional

El Modelo Relacional es un modelo de datos y, como tal, tiene en cuenta los tres aspectos siguientes de los datos:

1. La **estructura**, que debe permitir representar la información que nos interesa del mundo real.
2. La **manipulación**, a la que da apoyo mediante las operaciones de actualización y consulta de los datos.
3. La **integridad**, que es facilitada mediante el establecimiento de reglas de integridad, es decir, condiciones que los datos deben cumplir.



## Modelo Relacional

Además:

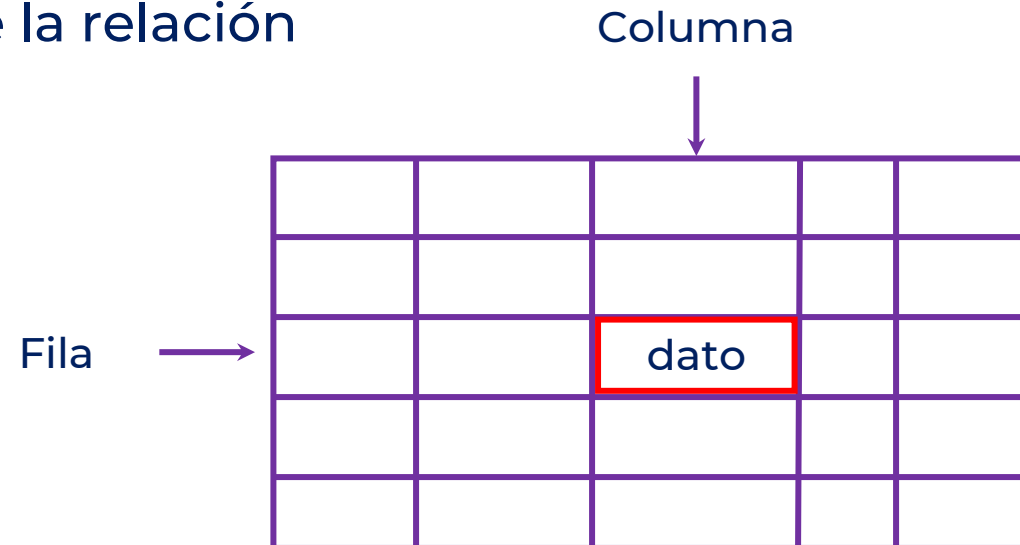
- Representación de los datos como una colección de tablas. Cada tabla tiene un nombre único. Contiene un solo tipo de registro.
- Las filas (**tuplas**) representan relaciones entre un conjunto de valores. Tiene un número fijo de atributos. No existen filas duplicadas.
- Las columnas son los dominios de los atributos.





## Modelo Relacional: Estructura

- **Dato:** unidad básica (atómica), representa un solo valor del dominio
- **Relación** (de datos): estructura básica (compuesta)
- **Tabla:** representación de una relación entre dominios de datos
- **Columna:** representa a un dominio
- **Fila:** representa a una tupla de la relación



# Modelo Relacional: Estructura

## Tabla

No tiene conceptos posicionales: no interesa el orden de las filas, ni el de las columnas.

## Fila

Contenido único respecto a la Relación: no se permiten filas duplicadas en una tabla.

## Clave Primaria (PK)

- Propiedad de unicidad.- El valor de la clave primaria en cada fila identifica a esa fila de manera única (distingue a la fila de cualquier otra de la tabla).
- Propiedad de mínima cantidad de columnas.- Si la clave primaria es compuesta y una de sus columnas es eliminada, la propiedad de unicidad ya no se cumple.



# Modelo Relacional: Estructura

## Clave Primaria (PK)

- Propiedad de unicidad interpretada como propiedad de identificación.- El valor de la clave primaria en cada fila identifica al objeto particular representado por esa fila dentro de la clase de objetos que representa esa relación.  
Si en el modelo existe la necesidad de referirse al mismo objeto, este mismo valor es usado.

## Clave Foránea (FK)

- Columna de una tabla que contiene los valores de un dominio que sirven al mismo de la clave primaria en otra(s) tabla(s) para identificar al mismo objeto.



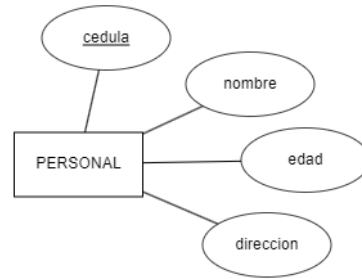
## Paso de MER a Modelo Relacional (Tablas)

### Pocas “reglas”:

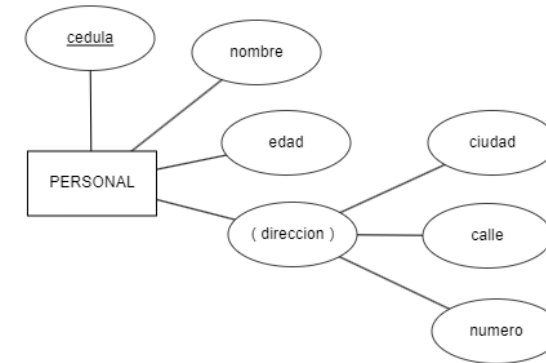
- Por entidad del esquema conceptual se crea una tabla  
Se selecciona uno de los atributos determinantes de la entidad como clave primaria de la tabla.
- Las relaciones N:M se modelan como una tabla nueva, donde se colocan las claves primarias de las tablas que representan a cada una de las entidades participantes  
Si existen atributos en la relación se tratan como si fueran los de una entidad.
- Por cada entidad débil (relaciones 1:N) se crea una tabla. Se incluye como atributos los de la clave primaria de la tabla que representa a la entidad fuerte  
La clave primaria la conforman la clave primaria de la tabla que representa a la entidad fuerte + atributo(s) que representan al identificador parcial.



## Paso de MER a Modelo Relacional (Tablas)



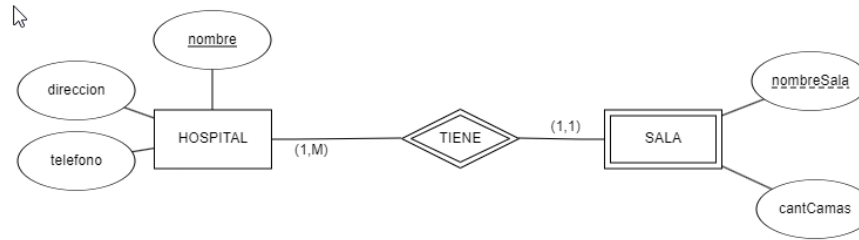
- PERSONAL(cedula, nombre, edad, direccion)



- PERSONAL(cedula, nombre, edad, ciudad, calle, numero)

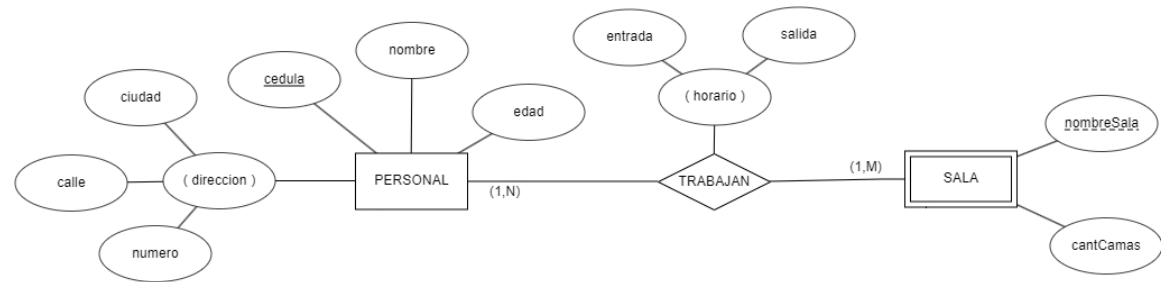


## Paso de MER a Modelo Relacional (Tablas)



HOSPITAL(nombre, dirección, teléfono)

SALA(nombreHospital, nombreSala, cantCamas)



HOSPITAL(nombre, dirección, teléfono)

SALA(nombreHospital, nombreSala, cantCamas)

TRABAJAN(cedula, nombreHospital, nombreSala, *entrada*, *salida*)



## Modelo Relacional: Lenguajes Formales

- Álgebra Relacional
  - Tipo *Procedural*, el usuario indica al sistema que lleve a cabo una serie de operaciones en la Base de Datos para calcular el resultado.
- Cálculo Relacional (de Tuplas / de Dominios)
  - Tipo *Non Procedural*, el usuario describe la información deseada sin dar un procedimiento concreto para obtener el resultado.

# Álgebra Relacional





# Álgebra Relacional

- Es un lenguaje de consulta procedimental basado en álgebra de conjuntos.
  - Para especificar una consulta en álgebra relacional, es preciso definir uno o más pasos que sirven para ir construyendo, mediante operaciones de álgebra relacional, una nueva relación que contenga los datos que responden a la consulta a partir de las relaciones almacenadas.
  - El álgebra relacional es importante porque:
    - Proporciona un fundamento formal para las operaciones del Modelo Relacional
    - Se utiliza como base para la implementación y optimización de consultas en los Sistemas de Gestión de Base de Datos (SGBD) relacional

# Álgebra Relacional

- Operaciones básicas
  - SELECT ( $\sigma$ )
  - PROJECT ( $\pi$ )
  - CARTESIAN PRODUCT ( $\times$ )
  - UNION ( $\cup$ )
  - SET DIFFERENCE ( $-$ )
- Operaciones adicionales
  - SET INTERSECTION ( $\cap$ )
  - JOIN (producto cartesiano con condición)
  - DIVISION ( $\div$ )




## Selección $\sigma(p)(R)$

- Donde R es la relación sobre la que se actúa y P es el predicado que debe cumplirse

**Empleado**

Número de Empleado	Nombre del Empleado	Salario	Código Depto.
1	Pedro	12,000	VE
2	Luís		NO
3	Francisco	36,000	
4	Lorena	24,000	
5	Gabriela	24,000	NO

- La SELECCIÓN de todos los empleados del departamento de Nómina, es:

 Cod.Depto='NO' (**Empleado**)

**RESULTADO**

Número de Empleado	Nombre del Empleado	Salario	Código Depto.
2	Luís		NO
5	Gabriela	24,000	NO



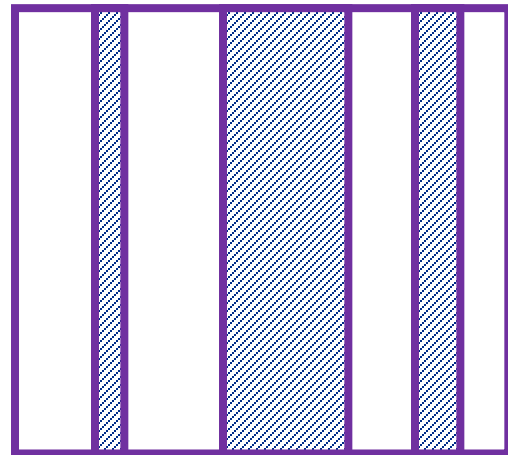
## Proyección $\pi (A) (R)$

Operación unitaria que produce una relación con las columnas de la relación operando (R) que se den como argumento (A) y eliminando de este resultado las tuplas repetidas.

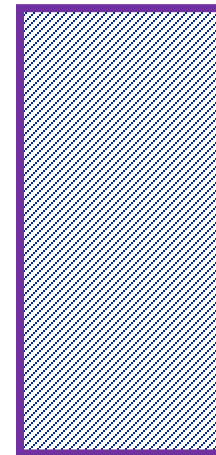
Devuelve los datos de ciertas columnas de la tabla y descarta otras.

El argumento :

Lista de columnas  $A = (a_1, a_2, \dots, a_n)$   
R



$\pi (A) (R)$



## Proyección $\pi (A) (R)$

Sintaxis:  $\pi A_1, \dots, A_n(R)$

Donde  $A_1, \dots, A_n$  es la lista de atributos y "R" la relación sobre la que se actúa.

Empleado			
Número de Empleado	Nombre del Empleado	Salario	Código Depto.
1	Pedro	12,000	VE
2	Luís		NO
3	Francisco	36,000	
4	Lorena	24,000	
5	Gabriela	24,000	NO

La proyección de los atributos Nombre y Número de empleado es:

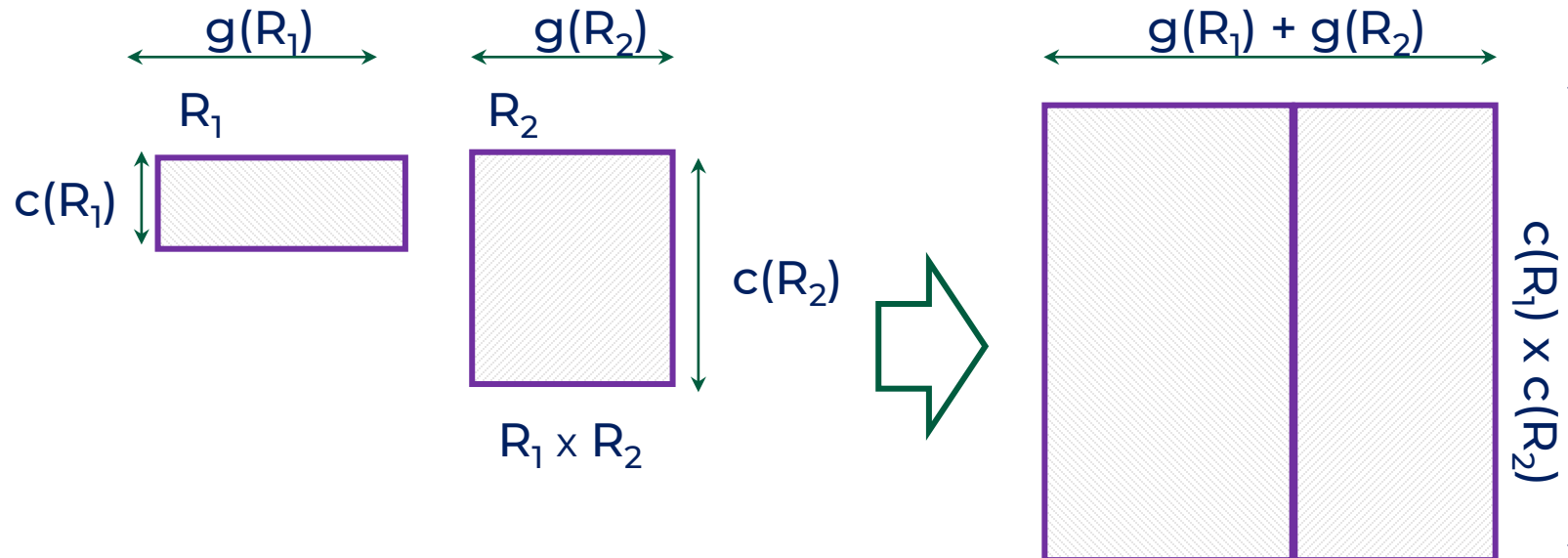
$\pi$  Nombre del Empleado, Número de empleado (Empleado)

RESULTADO	
Nombre del Empleado	Número de Empleado
Pedro	1
Luís	2
Francisco	3
Lorena	4
Gabriela	5



## Producto Cartesiano $R_1 \times R_2$

Operación binaria que produce una relación que contiene como tuplas a todas las combinaciones de las tuplas de la primera relación operando ( $R_1$ ) con las de la segunda ( $R_2$ ). La tabla que se obtiene como resultado tiene como columnas la concatenación de las columnas de ambas tablas operandos.



## Producto Cartesiano $R_1 \times R_2$

Sintaxis:  $(R_1 \times R_2)$

Donde  $R_1$  y  $R_2$  son las relaciones sobre las que se actúa

Departamento

Cod. Depto.	Nombre del Depto.
VE	Ventas
NO	Nómina
IN	Investigación
ME	Mercadotecnia
RE	Resultados

Empleado

Número de Empleado	Nombre del Empleado	Salario	Cod. Depto.
1	Pedro	12,000	VE
2	Luís		NO
3	Francisco	36,000	
4	Lorena	24,000	
5	Gabriela	24,000	NO

El Producto Cartesiano de las relaciones anteriores es:

(Departamento X Empleado)

RESULTADO

Cod. Depto.	Nombre del Depto.	Número de Empleado	Nombre de Empleado	Salario	Cod. Depto.
VE	Ventas	1	Pedro	12,000	VE
VE	Ventas	2	Luís		NO
VE	Ventas	3	Francisco	36,000	
VE	Ventas	4	Lorena	24,000	
VE	Ventas	5	Gabriela	24,000	NO
NO	Nómina	1	Pedro	12,000	VE
NO	Nómina	2	Luís		NO
NO	Nómina	3	Francisco	36,000	
...	...				
RE	Resultados	5	Gabriela	24,000	NO





## Unión $R_1 \cup R_2$

Sintaxis:  $R_1 \cup R_2$

Donde  $R_1$  y  $R_2$  son las relaciones sobre las que se actúa.

**Empleados Antiguos**

Número de Empleado	Nombre del Empleado	Salario
1	Pedro	12,000
2	Luís	
3	Francisco	36,000

**Empleados Nuevos**

Número de Empleado	Nombre del Empleado	Salario
3	Francisco	36,000
4	Lorena	24,000
5	Gabriela	24,000

La UNIÓN de las relaciones anteriores es:

**Empleados Antiguos  $\cup$  Empleados Nuevos**

**RESULTADO**

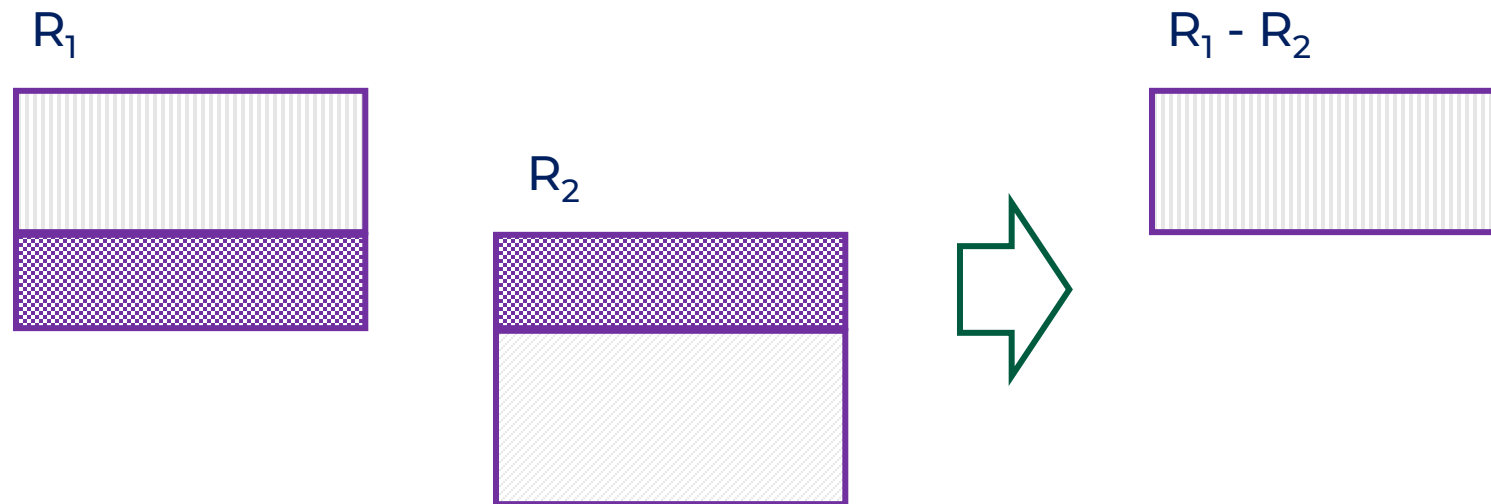
Número de Empleado	Nombre del Empleado	Salario
1	Pedro	12,000
2	Luís	
3	Francisco	36,000
4	Lorena	24,000
5	Gabriela	24,000



## Diferencia $R_1 - R_2$

Operación binaria que produce una relación que contiene todas las tuplas de la primera relación operando ( $R_1$ ) que no se encuentren en la segunda ( $R_2$ ).

Ambas relaciones operandos deben ser “compatibles para unirse”.



## Diferencia $R_1 - R_2$

- Ejemplo:

La diferencia de dos relaciones R y S con el mismo esquema.

NIF	Nombre	Localidad
1254	Juan	Lima
1578	Sergio	Trujillo
1865	Patricia	Arequipa
1904	Fernanda	Cuzco

**R**

NIF	Nombre	Localidad
1248	Nancy	Tumbes
1578	Sergio	Trujillo
1865	Patricia	Arequipa
1908	Andrés	Iquitos

**S**

**R - S**

NIF	Nombre	Localidad
1254	Juan	Lima
1904	Fernanda	Cuzco

# Álgebra Relacional

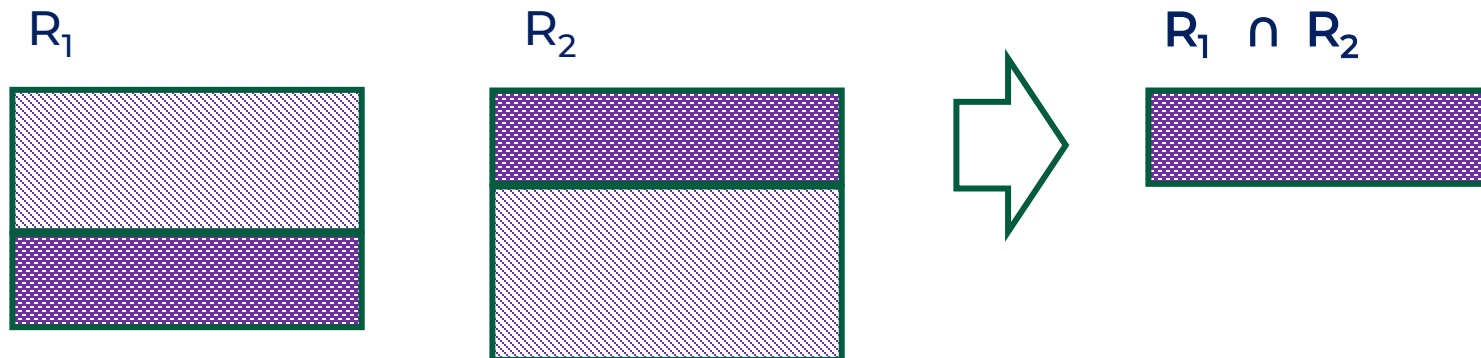
Operaciones adicionales



## Intersección $R_1 \cap R_2$

Operación binaria que produce una relación que contiene todas las tuplas de la primera relación operando ( $R_1$ ) que también están en la segunda relación operando ( $R_2$ ). Ambas relaciones operandos deben ser “compatibles para unirse”.

Es una operación derivada pues puede reemplazarse por un par de diferencias:  $R_1 \cap R_2 = R_1 - (R_1 - R_2)$



## Intersección $R_1 \cap R_2$

- Sintaxis:  $(R_1 \cap R_2)$  o  $R_1 - (R_1 - R_2)$

Donde  $R_1$  y  $R_2$  son las relaciones sobre las que se actúa.

Empleados Antiguos		
Número de Empleado	Nombre del Empleado	Salario
1	Pedro	12,000
2	Luís	
3	Francisco	36,000

Empleados Nuevos		
Número de Empleado	Nombre del Empleado	Salario
3	Francisco	36,000
4	Lorena	24,000
5	Gabriela	24,000

La INTERSECCIÓN de las relaciones anteriores, es:

**Empleados Antiguos  $\cap$  Empleados Nuevos**

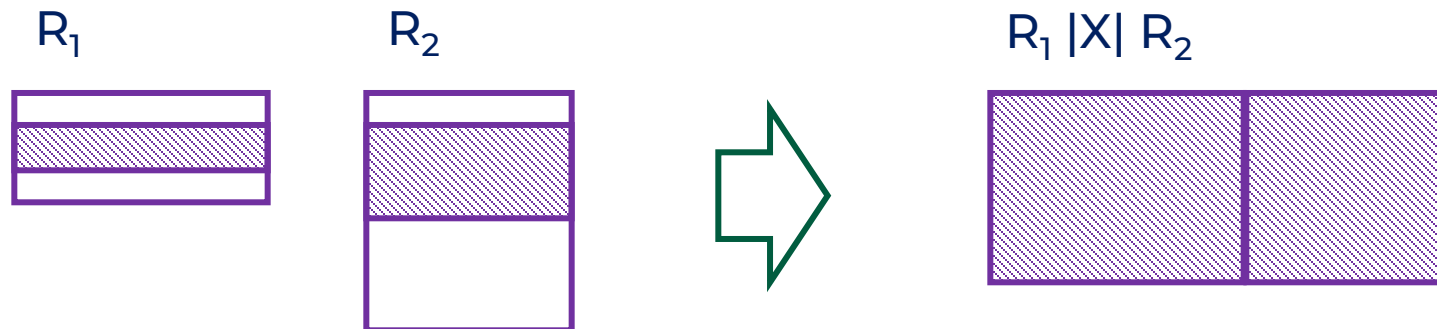
RESULTADO		
Número de Empleado	Nombre del Empleado	Salario
3	Francisco	36,000



## Conjunción Natural (Equi-Join) $R_1 \bowtie R_2$

Caso particular de conjunción donde el predicado (condición de conjunción) es de comparación por igualdad de cada pareja de columnas de las tablas operando ( $R_1$  y  $R_2$ ) que tengan igual dominio y significado (no necesariamente igual nombre de columna).

Supone el uso de condiciones de conexión solo con comparaciones de igualdad. En el resultado de un EQUI-JOIN siempre tenemos uno o más pares de atributos que cuentan con valores idénticos den cada tupla.



## Conjunción Natural (Equi-Join) $R_1 \bowtie R_2$

- Sintaxis:  $(R_1 \theta_p R_2)$  o  $\sigma_p(R_1 \times R_2)$

Donde  $R_1$  y  $R_2$  son las relaciones sobre las que se actúa y  $P$  representa el predicado de la selección.

**Departamento**

Cod. Depto.	Nombre del Depto.
VE	Ventas
NO	Nómina
IN	Investigación
ME	Mercadotecnia
RE	Resultados

**Empleado**

Número de Empleado	Nombre del Empleado	Salario	Cod. Depto.
1	Pedro	12,000	VE
2	Luís		NO
3	Francisco	36,000	
4	Lorena	24,000	
5	Gabriela	24,000	NO

El JOIN con la columna en común del código del departamento en las relaciones anteriores, es:

(Departamento  $\theta_{\text{Departamento.codDepto} = \text{Empleado.codDepto}}$  Empleado)

**RESULTADO**

Cod. Depto.	Nombre del Depto.	Número de Empleado	Nombre de Empleado	Salario
VE	Ventas	1	Pedro	12,000
NO	Nómina	2	Luís	
NO	Nómina	5	Gabriela	24,000



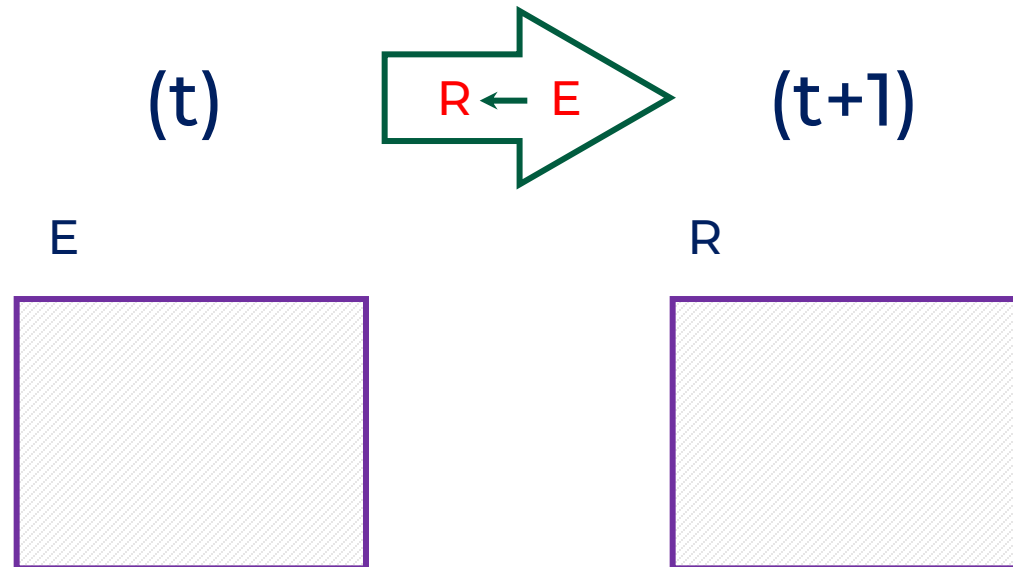
# Álgebra Relacional

## Operaciones de Modificación



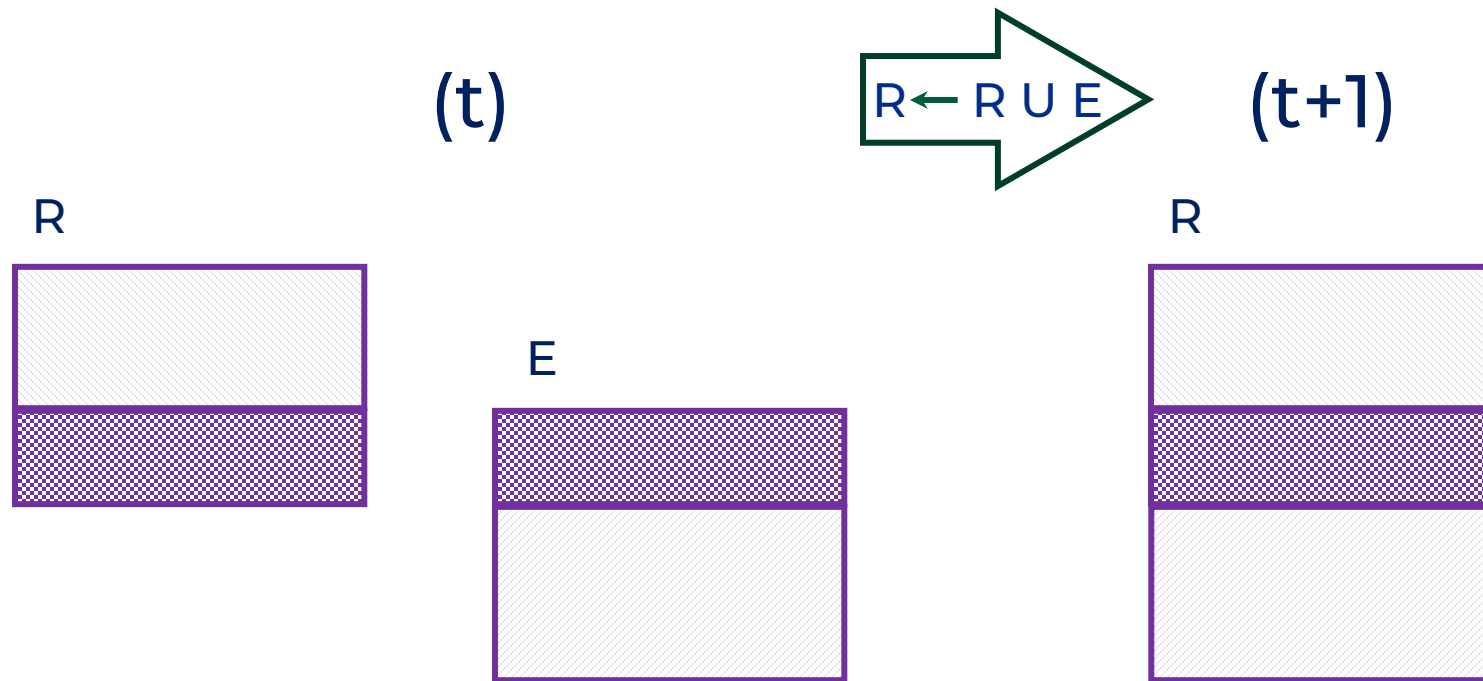
## Asignación $R \leftarrow E$

Operación que asigna a una relación que el usuario pone nombre (R) el resultado de una expresión relacional (E).



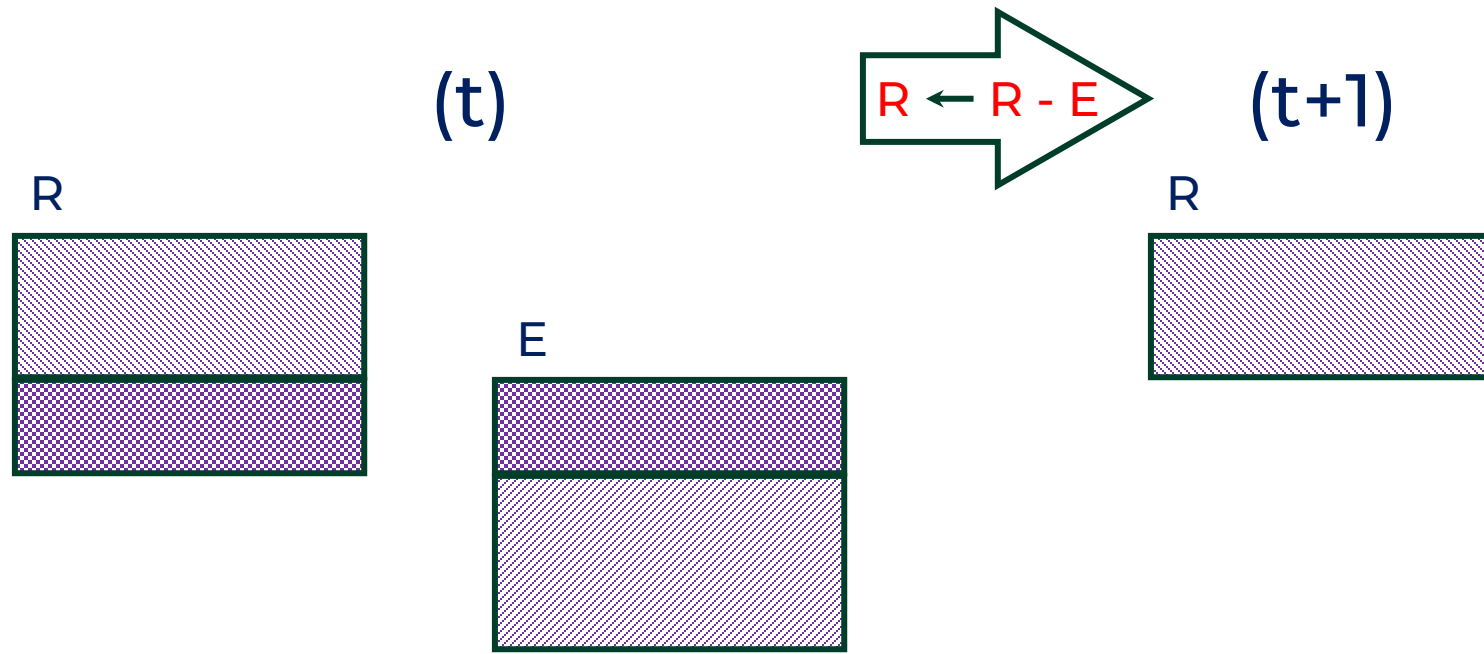
## Inserción $R \leftarrow R \cup E$

Operación que permite insertar una relación de tuplas, que puede ser resultado de una expresión relacional (E), a una relación (R), excluyendo las que ya se encuentren en R (evitando repetición de filas).



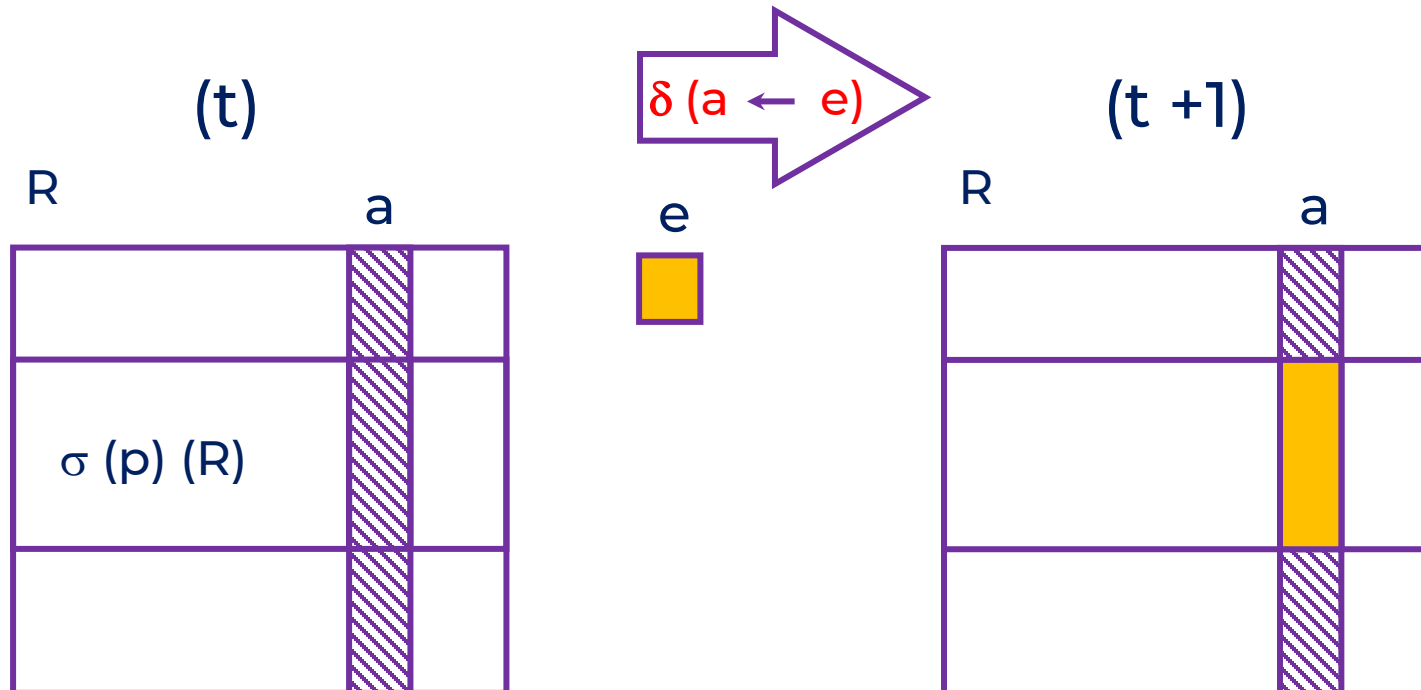
## Eliminación $R \leftarrow R - E$

Operación que permite eliminar una relación de tuplas, que puede ser resultado de una expresión relacional (E), de una relación (R).



## Actualización $\delta(a \leftarrow e)(\sigma(p)(R))$

Operación que permite **cambiar el valor** de un atributo (a) a las tuplas seleccionadas de la relación operando (R) que satisfagan un predicado (p), por otro valor que a su vez puede ser resultado de una expresión (e).



# Álgebra Relacional

Caso práctico



# Modelo Relacional

## Restricciones de Integridad



## Restricciones de Integridad

Aseguran la consistencia de los datos. Tenemos las siguientes:

- Integridad de dominio
- Integridad de columna
- Integridad de entidad
- Integridad referencial
- Integridad definida por el usuario



## Integridad de dominio

El valor que se puede asignar a un dato debe estar en el dominio.

- La excepción de la “información faltante” :

El modelo relacional permite a un dato estar marcado para no contener ningún valor del dominio. Algunos dicen que contiene el “valor **NULO**”.

Un dato puede estar marcado como NULO si :

- Se desconoce la información (nulo-aplicable). Permite NULO y valores de dominio.
- No tiene sentido asignar un valor del dominio (nulo-inaplicable).



## Integridad de dominio

Una definición adecuada de integridad de dominio no sólo permitirá controlar valores insertados/actualizados sino controlar la coherencia de consultas (si las comparaciones en los predicados tienen sentido).

- Ejemplo :

Entidad alumnos, columna número de libros a prestarse, debe tomar valores de los números naturales.

Atributo	Dominio	Descripción Dominio
Nombre	NOMBRES	cadenas de hasta 30 caracteres alfabéticos
Telefono	TELEFONOS	cadenas de hasta 9 caracteres numéricos
Altura	MEDIDAS	números reales entre 0 y 2.5 (metros)
...	...	...



## Integridad de columna

Dentro de la integridad de dominio puede adicionarse restricciones a nivel columna.

- Ejemplo:

La entidad Alumno

columna cantidad de libros a prestarse menor a 5,

columna número de créditos a matricularse menor a 18 créditos.

En los SGBD:

- Los dominios se implementan con tipos de datos
- Los subconjuntos de dominios se implementan con la integridad de columnas



## Integridad de entidad

- Ningún componente de una clave primaria puede tener valores nulos.
- Ningún componente de una clave foránea debe permitir un valor nulo por inaplicable.

En una base de datos relacional, nunca registraremos información acerca de algo que no podamos identificar.

- Ejemplo:



## Integridad definida por el usuario

Se permiten restricciones definidas por el usuario que pueden tener:

- aspectos de tiempo
- acciones de usuarios y de programas de aplicación
- condiciones de verificación
- acciones para tomarse en caso de una violación de la confidencialidad, disponibilidad o integridad de los datos

### Ejemplo

Una “regla de negocio”: el sueldo mínimo de un nuevo empleado debe ser mayor o igual a S/ 1,025 soles



# BARKER

## Técnica de modelamiento y convenciones



## Notación Barker

La siguiente notación fue desarrollada originalmente por la consultora británica CACI. Posteriormente, fue promovido por Richard Barker y adoptado por Oracle Corporation para su "método CASE\*" (posteriormente rebautizado como "método de desarrollo personalizado").

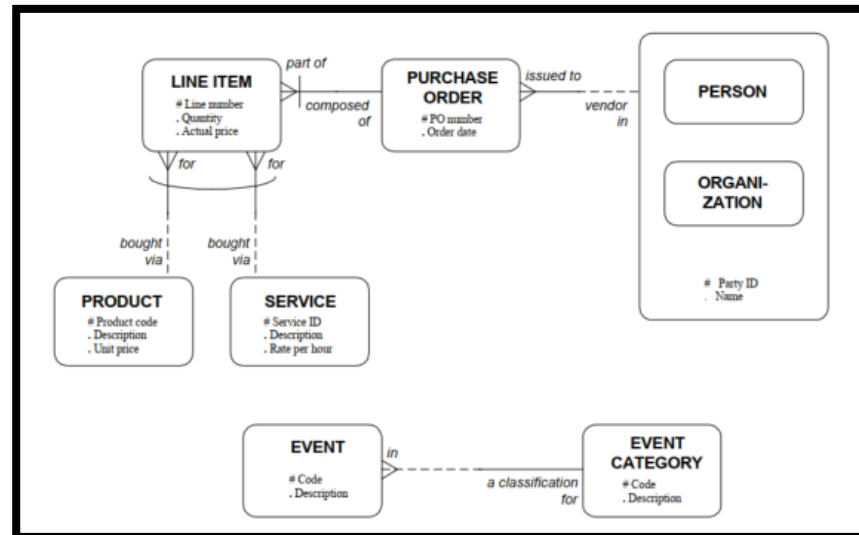


Figura. Un modelo de datos del método CASE\*





## Notación Barker

### Convenciones de Diagramas: Entidad

- Los nombres de entidades van en los cuadros editables con las esquinas redondeadas.
- Los nombres de entidades están siempre en singular y escritos con todas las letras mayúsculas.

**EMPLEADO**

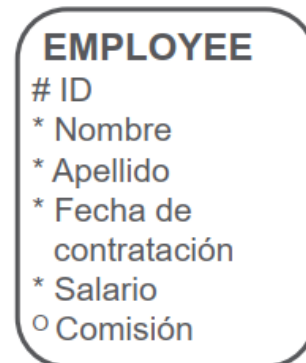
**JOB**



## Notación Barker

### Convenciones de Diagramas: **Atributo**

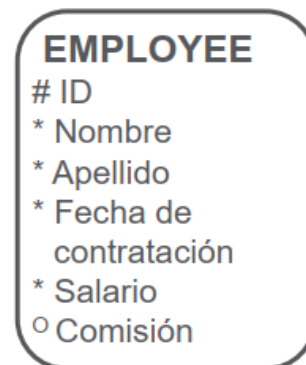
- Los atributos se muestran bajo los nombres de entidades.
- Los atributos obligatorios (valor no pueden ser nulos) están marcados con un asterisco: "\*"
- Los atributos opcionales (valor puede ser nulo) están marcados con un círculo: "o"
- Los identificadores únicos están marcados con un símbolo de almohadilla: "#"



## Notación Barker

### Convenciones de Diagramas: **Atributo**

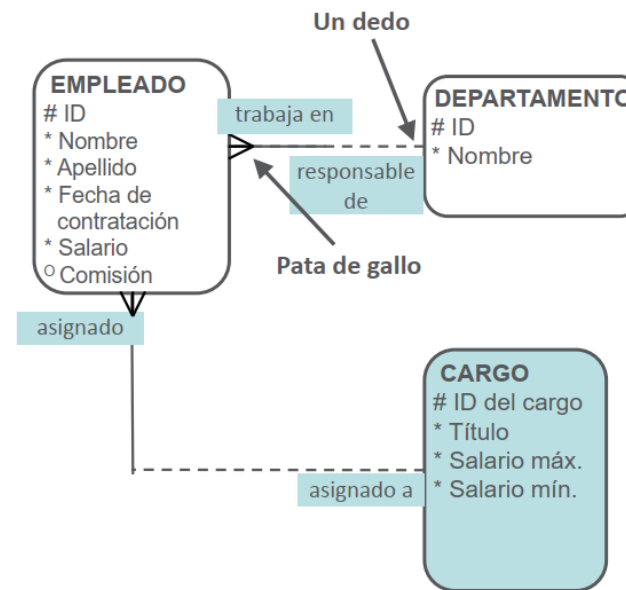
- Los atributos deben estar escritos de forma que todas las personas puedan comprenderlos y no solo los desarrolladores
- Los atributos deben estar escritos con la primera letra de cada palabra en mayúscula y el resto en minúscula.
- Al lado de cada atributo se debe colocar un símbolo que represente el tipo de atributo.



## Notación Barker

### Convenciones de Diagramas: Relación

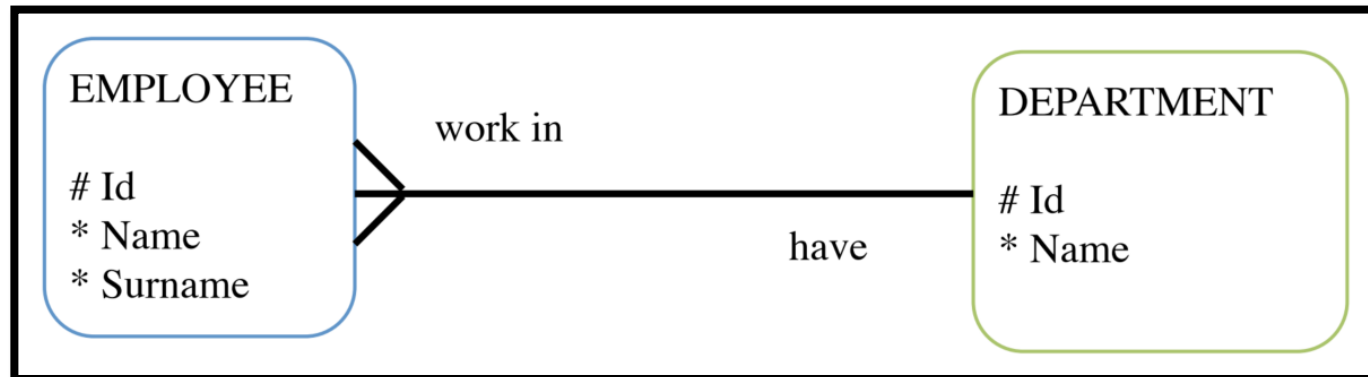
- Las relaciones son líneas que conectan entidades
- Estas líneas son continuas o discontinuas
- Estas líneas terminan en un "dedo" o "pata de gallo" al final de cada entidad



## Notación Barker

### Convenciones de Diagramas: Relación

Una relación se implementa como una clave externa y, por lo tanto, el atributo FK no aparece como atributo en la entidad de destino.



## Notación Barker

Convenciones de Diagramas: **Relación**

Al establecer una relación utilizando la notación de Barker, se deben respetar las siguientes reglas:

- Una relación puede existir entre un máximo de dos entidades
- Una relación puede existir en la misma entidad
- Una relación tiene dos perspectivas
- Ambas perspectivas de una relación deben estar etiquetadas





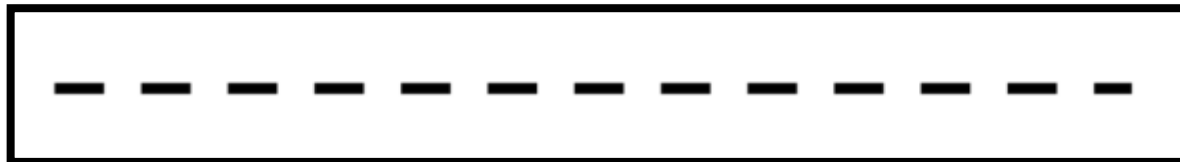
## Notación Barker

### Convenciones de Diagramas: Opcionalidad de una Relación

- Relación obligatoria: una relación obligatoria especifica que cada instancia de una entidad **debe** estar relacionada con otra instancia. Esto se representa mediante una línea recta



- Relación opcional: una relación opcional especifica que cada instancia de una entidad **puede** estar relacionada con otra instancia. Esto se representa mediante una línea discontinua.





## Notación Barker

### Relaciones

Una relación siempre se compone de dos perspectivas utilizando la siguiente notación:

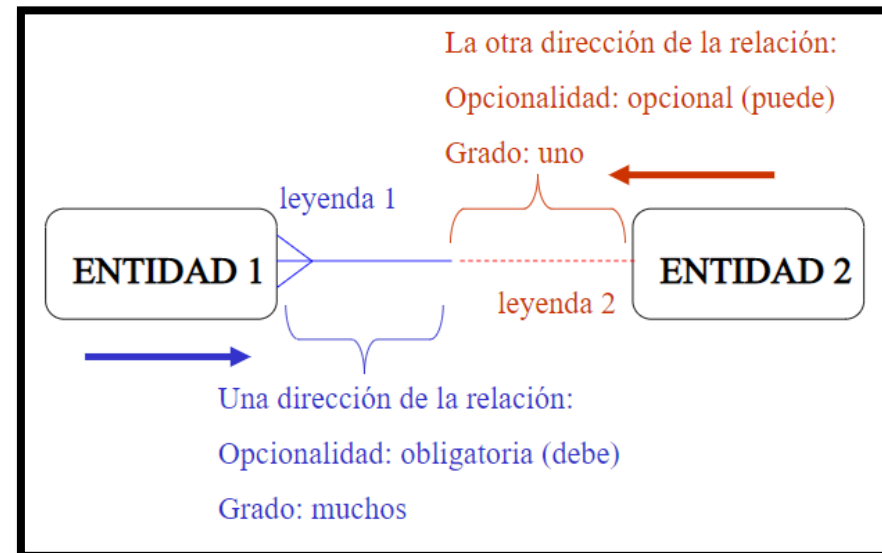


Figura. Notación Richard Barker



## Notación Barker

### Relaciones

- Lectura según perspectiva:
  - De CURSO a PROFESOR (**primera perspectiva**)
    - Un CURSO debe ser dictado por un PROFESOR
  - De PROFESOR a CURSO (**segunda perspectiva**)
    - Un PROFESOR puede estar encargado de muchos CURSOS

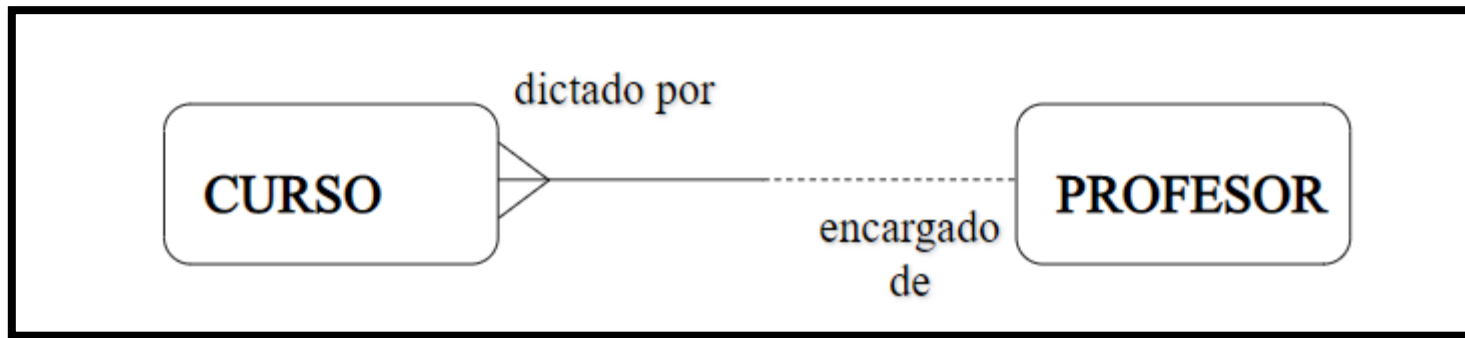


Figura. Notación Richard Barker



# Relaciones

-

## Notación Barker

### Relaciones

- Puesto que cada relación tiene dos partes, leemos la primera perspectiva de izquierda a derecha (o de arriba a abajo, según el diseño del DER)



Figura. ERD. Fuente: ORACLE Academy



# Notación Barker

## Relaciones

- Ahora leeremos la relación de derecha a izquierda (segunda perspectiva)

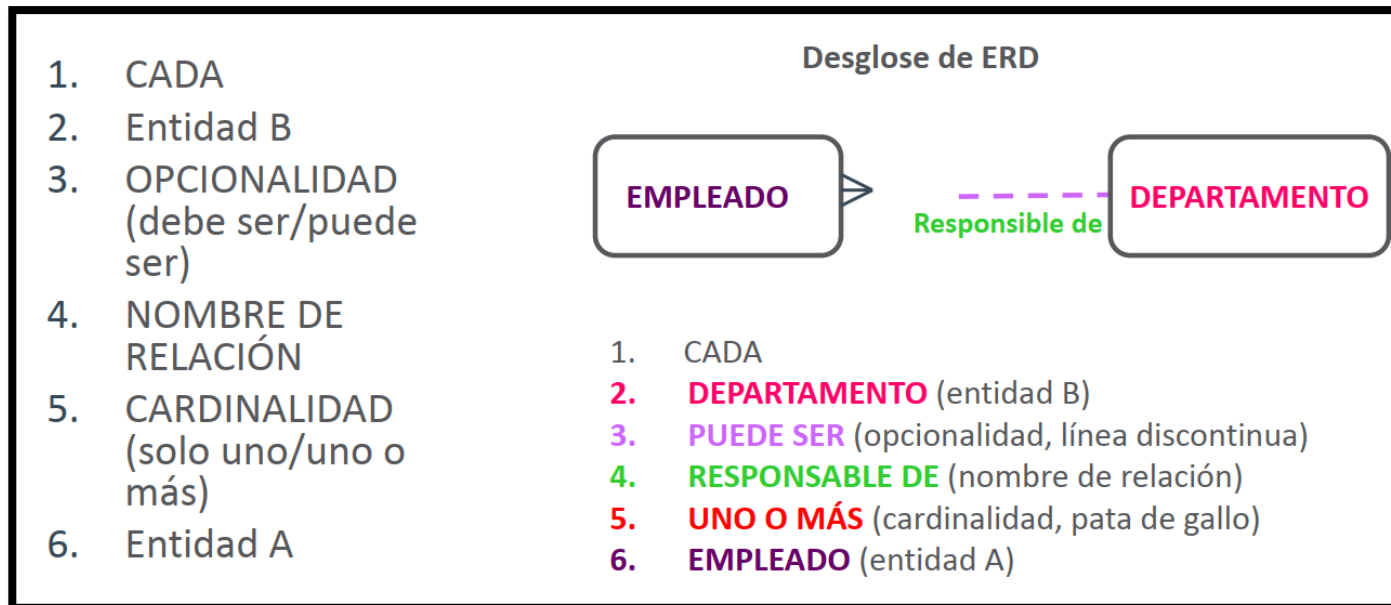


Figura. ERD. Fuente: ORACLE Academy



# Notación Barker

## Relaciones

- Ahora reúnalo todo

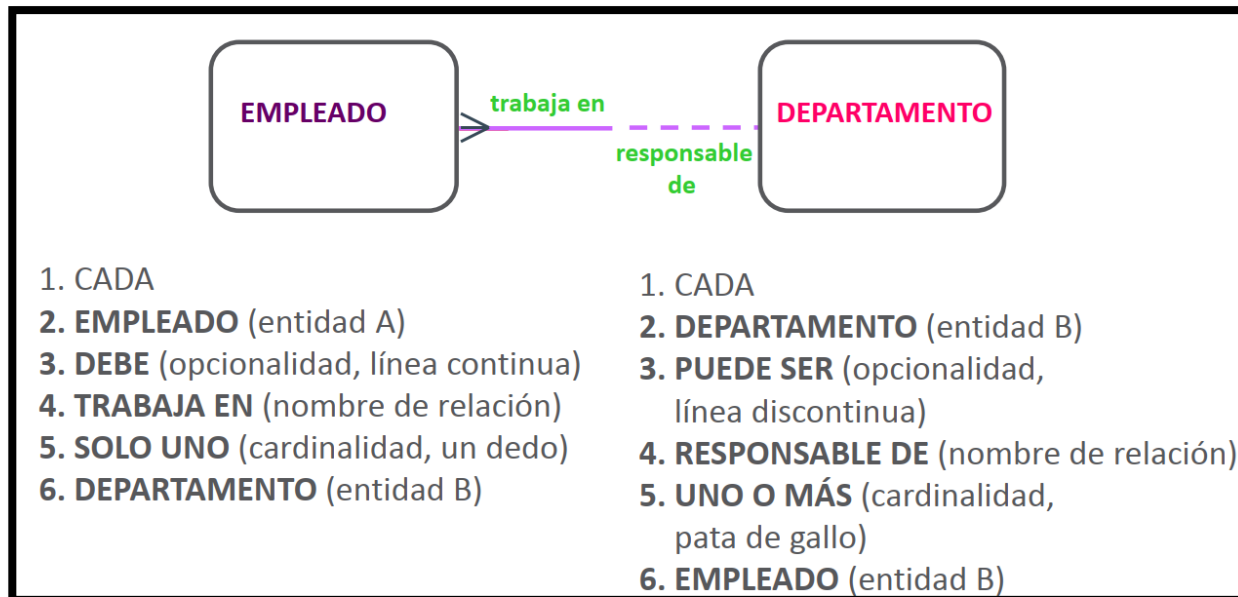


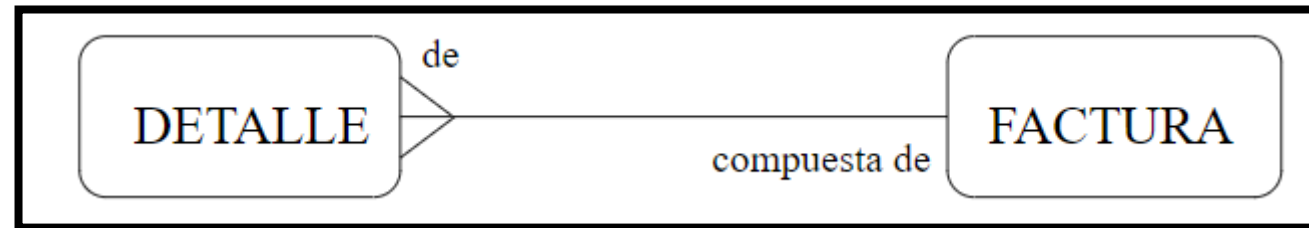
Figura. ERD. Fuente: ORACLE Academy



## Notación Barker

### Tipos de Relaciones

- Uno a muchos



- Uno a uno

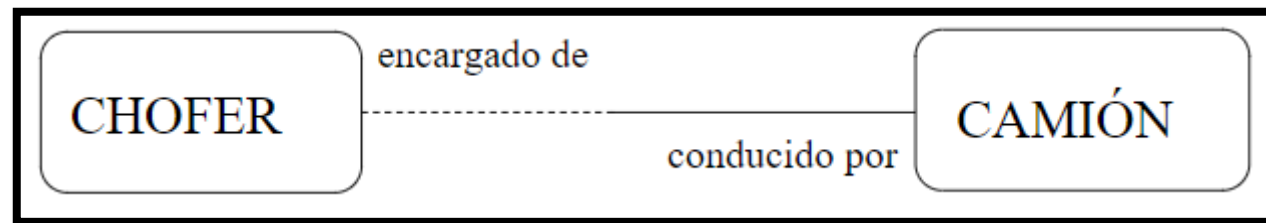


Figura. Notación Richard Barker



## Notación Barker

### Tipos de Relaciones

- Muchos a muchos

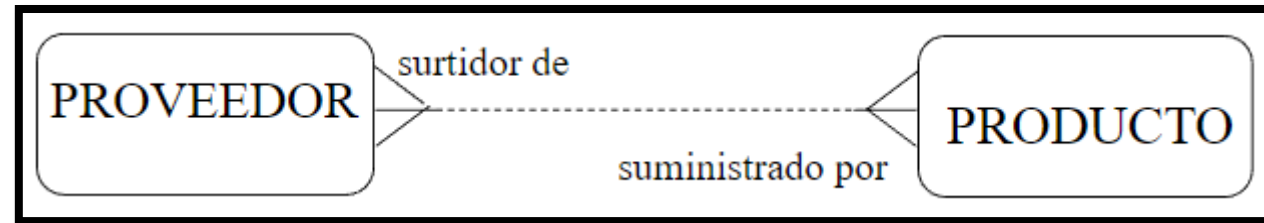


Figura. Notación Richard Barker





# Técnica de modelamiento y convenciones

Notación IDEF1X





## Conclusiones

Se han revisado los principales conceptos del Modelo Relacional como son:

- Estructura
- Operaciones relacionales (Álgebra relacional)
- Restricciones de integridad

Además, técnicas de modelamiento de datos como:

- Barker
- IDEF1X





## Referencias

- R. Elmasri y S.B. Navathe. (2007). Fundamentos de Sistema de Base de Datos, 5ta edición
- Korth, H. and A. Silberschatz (1996). Fundamentos de Bases de Datos. España, McGraw Hill.
- F. Inguanez (2012). A short guide to designing Entity Relationship Models using Barker's notation



**¡Gracias!**

