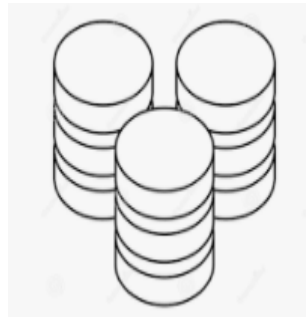


Capítulo 2

Modelos de entidad – relación y su pasaje a tablas parte 1



Diseño de Esquemas de BD

- En este capítulo vamos a estudiar el siguiente enfoque para diseño de esquemas de BD:
 - **Diseño de entidad-relación (ER)**: modelado de entidades y relaciones con **toma de buenas decisiones de diseño**.
 - **Traducción** de modelos de entidad – relación a esquemas relacionales.

Diseño de Esquemas de BD

- **Encaramos este capítulo de la siguiente manera:**
 - Primero explicamos los conceptos básicos de modelado de entidad-relación.
 - Luego traducimos esos conceptos a esquemas relacionales.
 - Esto es necesario saber cómo se hace esto para poder justificar las decisiones de diseño que se toman.
 - A continuación, estudiamos como tomar buenas decisiones de diseño involucrando los conceptos básicos.
 - Finalmente, aplicamos los tres pasos anteriores a conceptos más avanzados como entidades débiles y generalización.

Conjuntos de Entidades

- Una **entidad** es un objeto que existe y es distinguible de los otros objetos.
 - **Ejemplos:** persona específica, empresa, recital, planta.
- Las entidades tienen **atributos**.
 - **Ejemplo:** una persona tiene nombres y direcciones.
- Un **conjunto de entidades (CE)** es un conjunto de entidades del mismo tipo (i.e. Con los mismos atributos) que comparte las mismas propiedades.
 - **Ejemplo:** conjunto de todas las personas con los atributos del ejemplo anterior.

Conjuntos de Entidades

- **Ejemplo:**

- Se quiere modelar una base de datos de *estudiantes* e *instructores*, identificados por *número de matrícula* y *legajo*, respectivamente. Un *instructor* puede instruir a varios *estudiantes*, pero un *estudiante es instruido* por un solo instructor.
- ¿Qué conjuntos de entidades tenemos aquí?

Conjuntos de Entidades

instructor_ID instructor_name

76766	Crick
45565	Katz
10101	Srinivasan
98345	Kim
76543	Singh
22222	Einstein

instructor

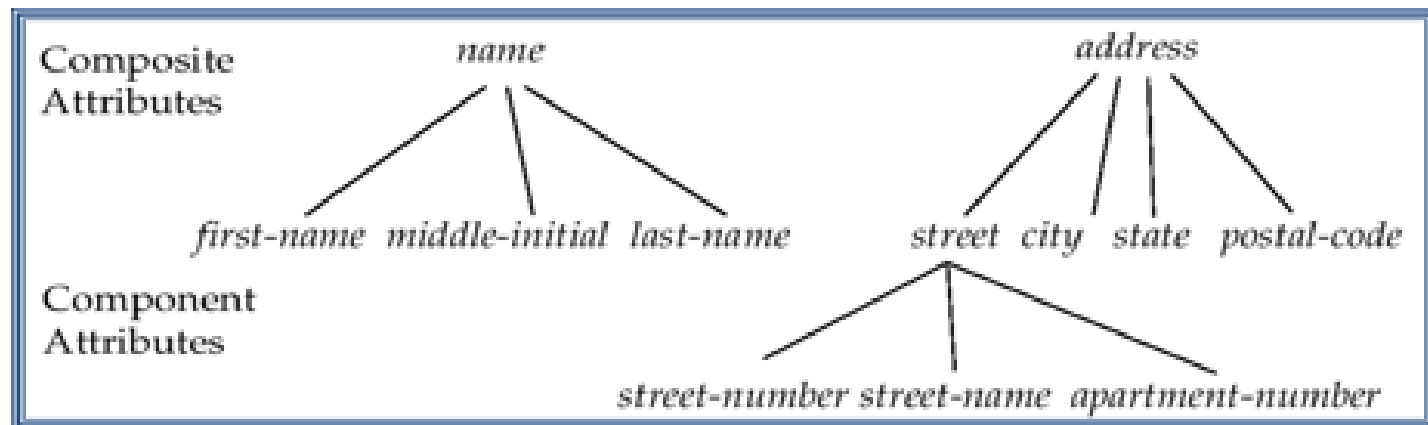
student-ID student_name

98988	Tanaka
12345	Shankar
00128	Zhang
76543	Brown
76653	Aoi
23121	Chavez
44553	Peltier

student

Atributos

- **Situación:** A veces se tienen **partes de un atributo** que nos interesan porque queremos acceder a ellas en consultas.
- **Solución:** Usar **atributos compuestos**.
 - Un atributo compuesto se descompone en otros más simples.
 - Se puede definir así una jerarquía de atributos.



Atributos

- **Situación:** A veces tenemos **varios valores** para un atributo.
 - Usar un atributo simple no alcanza.
 - Usar varios atributos simples tampoco sirve.
- **Solución:** Usar **atributos multivalorados**
 - Significan: hay **0 o más** valores para el atributo.
 - **Ejemplo:** *números-telefónicos* de una persona es atributo multivalorado,

Atributos

- Al conjunto de valores permitidos para cada atributo se le llama **dominio**.
- **Clasificaciones de los atributos:**
 - Atributos *Simples* y *compuestos*.
 - Los atributos hoja son simples (no se siguen descomponiendo)
 - Atributos *uni-valorados* (toman un valor) y *multi-valorados* (pueden tomar varios valores)
 - **Ejemplo:** *números-telefónicos* de una persona es atributo multivalorado, *nombre* de una persona es atributo univalorado.
- Atributos **derivados**
 - Pueden computarse de otros atributos.
 - **Ejemplo:** *edad* dada la *fecha de nacimiento*.

Claves

- Ahora se enseñan los conceptos de **superclave**, **clave candidata** y **clave primaria** para CE.
 - Es muy similar a como lo hicimos para modelo relacional.
- **Ejemplo:** el CE *instructor* tiene atributos:
ID, name, dept_name, salary
 - Con el *ID* y el *name* sabemos de qué instructor estamos hablando.
 - Basta considerar el *ID* para saber a cuál instructor nos referimos.
 - Con el *salary* no sabemos de qué instructor se habla.
- Una **superclave** de un CE es un conjunto de uno o más atributos cuyos valores unívocamente determinan cada entidad.

Claves

- Una **clave candidata (CC)** de un CE es una superclave minimal (i.e. si se quita atributo dejamos de tener superclave).
 - *ID* es clave candidata de *instructor* (y no tiene otra CC)
 - *course_id* es clave candidata de *course*
- Una clave candidata no necesariamente tiene cardinalidad mínima.
- **Ejemplo:** un CE *biblioteca* de las bibliotecas de una ciudad tiene los atributos: *nombreBib*, *calle*, *número*
 - *nombreBib* clave candidata
 - *calle*, *número* clave candidata

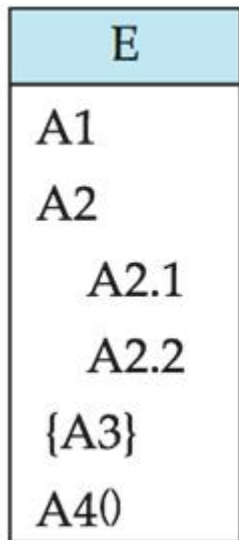
Claves

- Aunque varias claves candidatas pueden existir, una de las claves candidatas es seleccionada para ser la **clave primaria**.
- **Ejemplo:** en el CE biblioteca se puede elegir:
 - *nombreBib* clave primaria,
 - o
 - *calle, número* clave primaria
- No necesariamente es la de menor cantidad de atributos

Modelado ER

- **Propósito:** diagramar conjuntos de entidades con sus atributos.
- **Notaciones** de diagramas que usaremos:
 1. La de las ediciones 4 y 5 del libro del Silberschatz
 2. La de la edición 6 del libro de Silberschatz
 - Principalmente difieren en cómo se representan los atributos de los conjuntos de entidades.
 - En las demás cosas son bastante similares.

Conjuntos de Entidades y Atributos



Notación de clases: sexta edición del Silberschatz

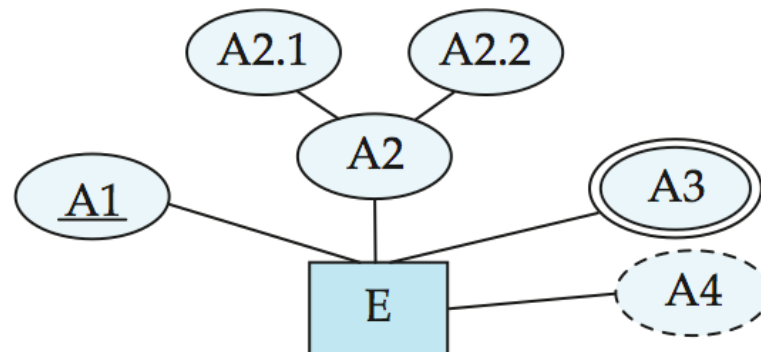
attributes:
simple (A1),
composite (A2) and
multivalued (A3)
derived (A4)

¿Qué ventajas tiene la segunda notación?

¿Qué ventajas tiene la primera notación?

Otra forma de decirlo: cuarta y quinta edición del Silberschatz

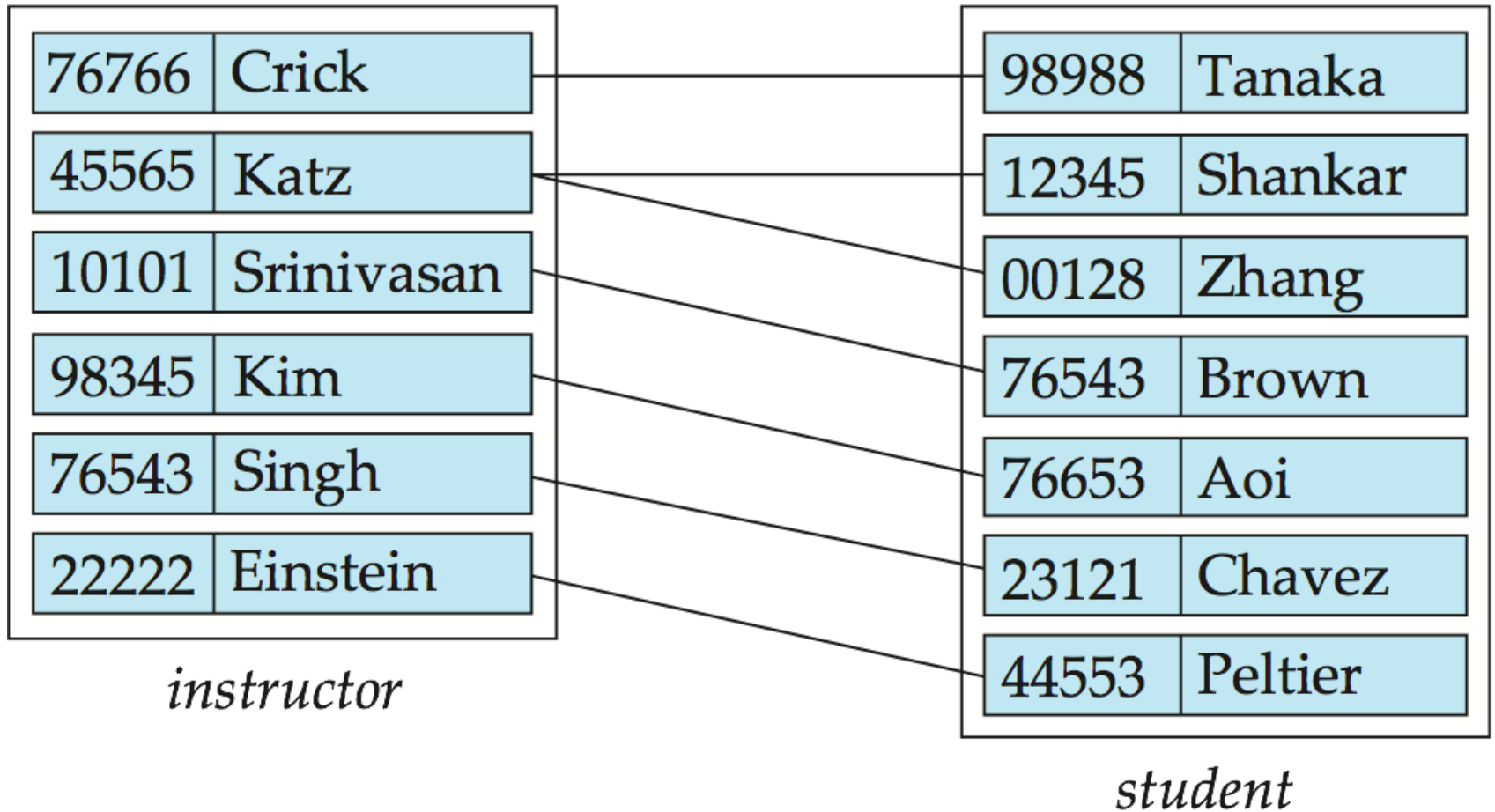
entity set E with
simple attribute A1,
composite attribute A2,
multivalued attribute A3,
derived attribute A4,
and primary key A1



Conjuntos de Entidades y Atributos

- **Ejercicio:** representar la siguiente situación: hay docentes. Un docente tiene un identificador que es único entre los docentes. Además, un docente tiene un nombre que consiste de: primer nombre, inicial del medio y apellido. Además, un docente tiene un domicilio que se compone de posición de calle, ciudad, estado y código postal; la posición de calle consiste de número, nombre de calle y número de departamento (el cuál es opcional). Además, un docente tiene cero o más números telefónicos. Finalmente, un docente tiene una fecha de nacimiento y una edad.

Conjuntos de Relaciones



Conjunto de relaciones *supervisa*

Conjuntos de Relaciones

- Una **relación** es una asociación entre varias entidades.

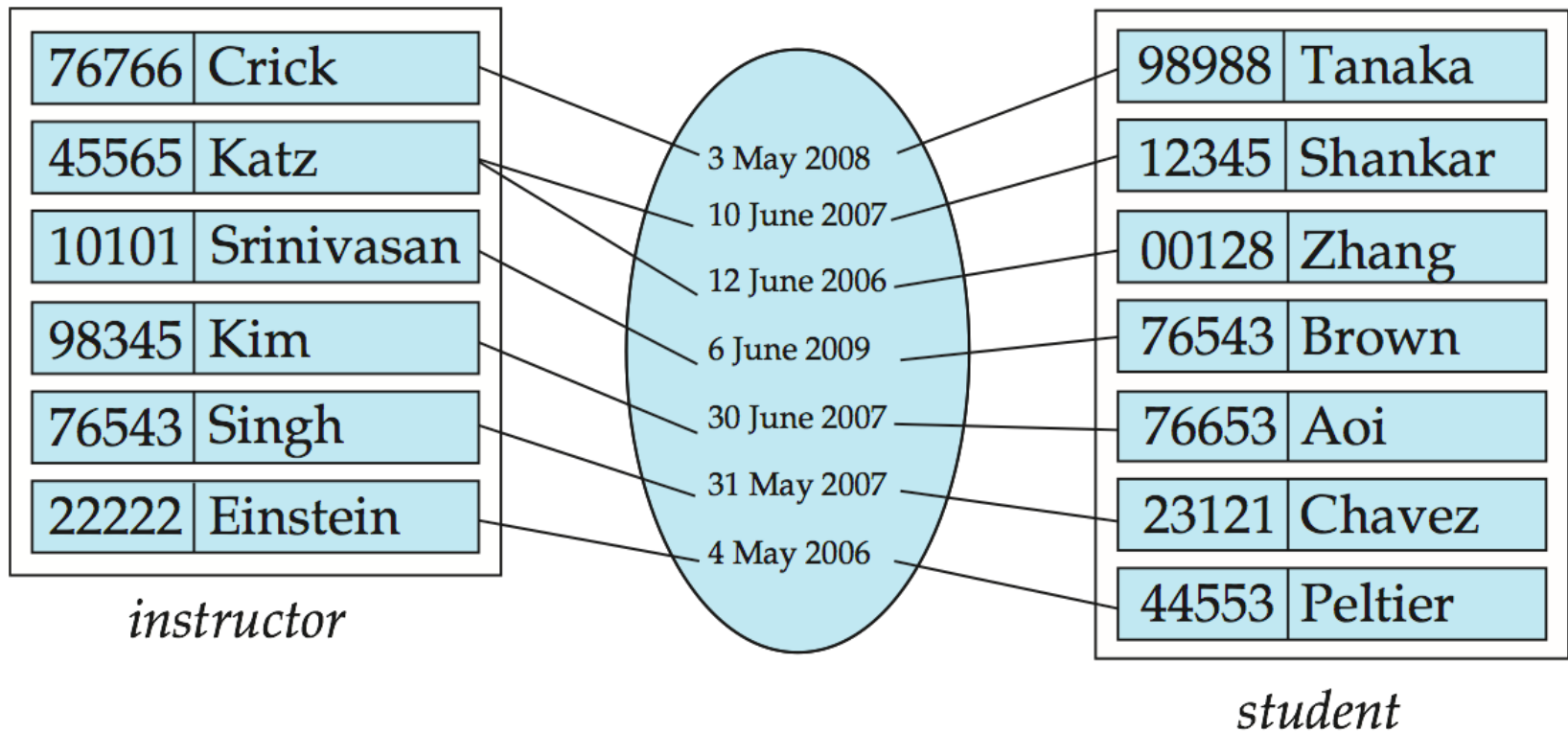
Ejemplo:

22222 (Einstein)
entidad instructor

44553 (Peltier)
entidad estudiante

- Tenemos la relación: ((22222, Einstein), (44553, Peltier))
- Un **conjunto de relaciones** (**CR**) es una relación matemática entre $n \geq 2$ conjuntos de entidades,
 $\{(e_1, e_2, \dots, e_n) \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n\}$
donde (e_1, e_2, \dots, e_n) es una relación.
- **Por ejemplo:** el CR *supervisa* entre instructor y estudiante.

Conjuntos de Relaciones



- Un atributo puede ser también una propiedad de un conjunto de relaciones.
- Conjunto de relaciones *supervisor* entre *instructor* y *estudiante* con atributo *fecha* que indica cuándo a un alumno se le asignó un profesor supervisor.

Correspondencias de cardinalidades

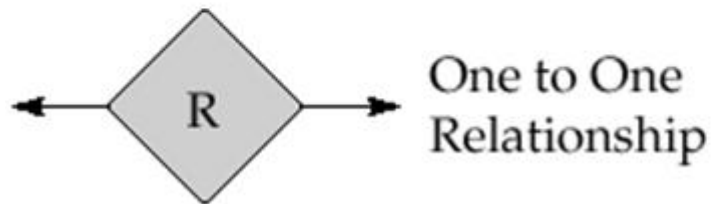
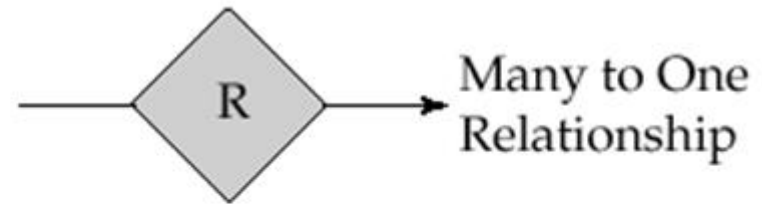
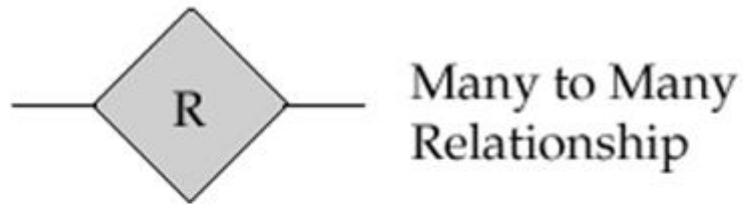
- Sean $E1$ y $E2$ conjuntos de entidades y R conjunto de relaciones entre $E1$ y $E2$.
- Si tenemos $E1 \text{ } R \text{ } -- \text{ } E2$ (línea de R a $E2$) esto significa que a cada entidad de $E1$ le corresponde por R varias entidades de $E2$ (precisamente: 0 o más entidades).
- De manera similar se define $E1 \text{ } -- \text{ } R \text{ } E2$.
- Si tenemos $E1 \text{ } R \text{ } \rightarrow \text{ } E2$ (flecha de R a $E2$) esto significa que a cada entidad de $E1$ le corresponde por R a lo más una entidad de $E2$ (precisamente: 0 o 1 entidad)
- De manera similar se define $E1 \text{ } \leftarrow \text{ } R \text{ } E2$.

Correspondencias de cardinalidades

- Entonces tenemos 4 combinaciones posibles de líneas y flechas (llamadas **correspondencias de cardinalidades**):
 - $E1 - R - E2$: varios – varios
 - $E1 \leftarrow R \rightarrow E2$: uno – uno
 - $E1 - R \rightarrow E2$: varios – uno
 - $E1 \leftarrow R - E2$: uno – varios

Correspondencias de Cardinalidades

Propósito: Diagramar conjuntos de relaciones binarios.
Usamos la siguiente notación para correspondencia de cardinalidades.



Correspondencias de Cardinalidades

- **Ejercicio 1:** representar el CR *supervisa* (trabajo especial) entre *instructor* y *estudiante*.
 - Un docente puede supervisar varios estudiantes y un estudiante puede tener a lo sumo un supervisor.
- **Ejercicio 2:** representar el CR *pertenece* entre *cliente* y *carrito de compras* de un sitio de comercio electrónico.
 - ¿Cómo conviene que sea la correspondencia de cardinalidades?
- **Ejercicio 3:** representar el CR *contiene* entre *artículo* y *carrito de compras*.
 - Reflejar además que para cada artículo en un carrito se tiene además una cantidad del artículo.

Correspondencias de Cardinalidades

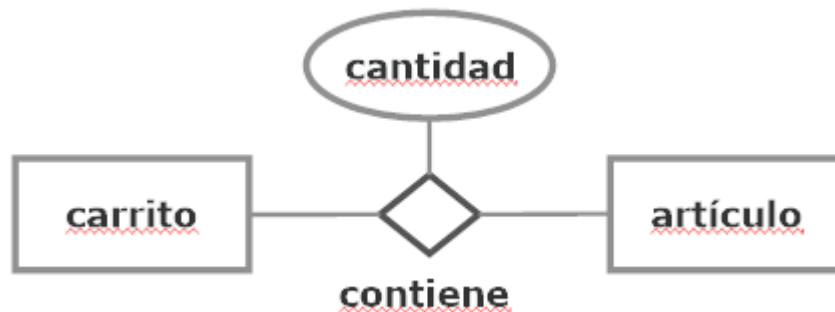
- Solución ejercicio 1:



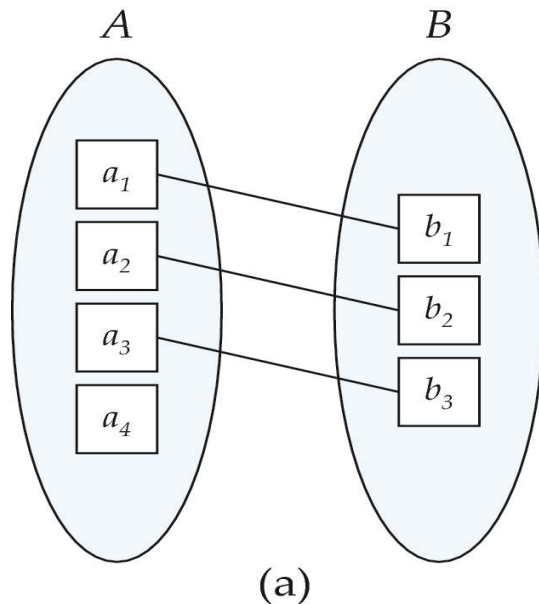
- Solución ejercicio 2:



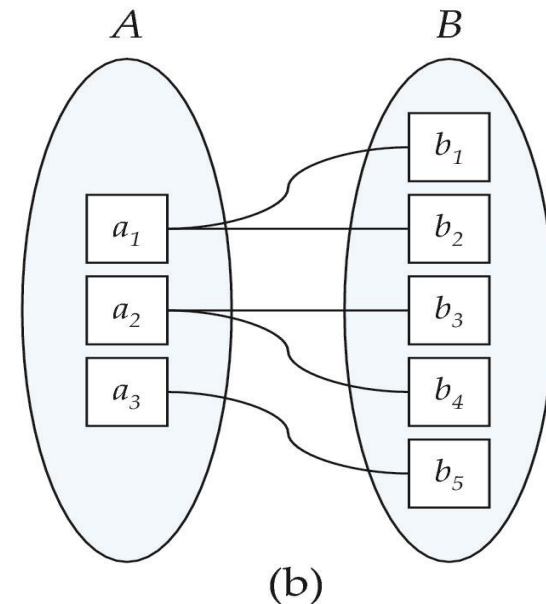
- Solución ejercicio 3:



Formas de participación de CE en CR



A participa parcialmente

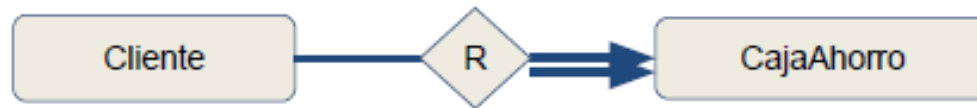


A participa totalmente

- **Requisito:** quiero poder distinguir entre estos dos casos y expresarlos como restricciones de integridad en un diagrama ER.
- **Ejemplos:**
 - CE *estudio*, CR *posee* y CE *película*.
 - Toda *película* es poseída por un *estudio*.
 - CE *docente*, CR *supervisa* y CE *alumno*.
 - Un *docente* puede no ser supervisor y un *alumno* puede no ser supervisado.

Formas de participación de CE en CR

- **Participación total:** (indicada por línea doble): toda entidad en el conjunto de entidades participa en al menos una relación en el conjunto de relaciones.
 - ❑ **Ejemplo:** participación de *caja de ahorro* en *cliente* es total.
 - Toda *caja de ahorro* debe tener *clientes* asociados.



Participación parcial: algunas entidades no participan en alguna relación en el conjunto de relaciones.

- ❑ Si la participación es parcial no se usa doble línea.
- ❑ **Ejemplo:** participación de *instructor* en *supervisor* es parcial.

Conjuntos de Relaciones

- **Ejercicio:** Indicar conjuntos de relaciones y restricciones de integridad (i.e. correspondencia de cardinalidades y forma de participación) para el siguiente enunciado.
 - *“Un socio puede tener prestado varios libros y todo libro ha sido prestado a a lo sumo un socio. Una biblioteca puede tener varios libros, y todo libro debe pertenecer a a lo sumo una biblioteca. Un socio puede estar inscripto en varias bibliotecas y una biblioteca puede tener varios socios. Un bibliotecario trabaja en a lo más una biblioteca y en una biblioteca puede haber varios bibliotecarios”.*

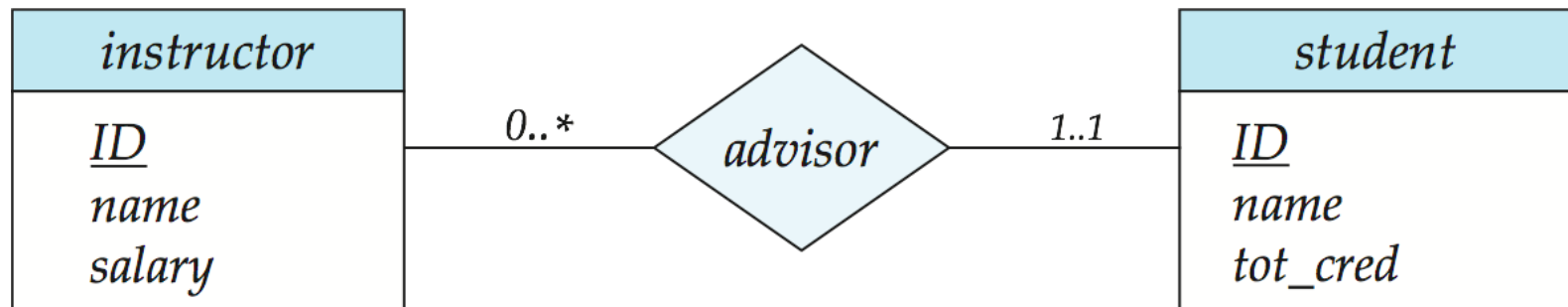
Notación Alternativa preferida para Correspondencia de Cardinalidades

□ Notación de intervalos o de cardinalidades

Usar $[a..b]$ o $[a..*]$.

□ Utilidad de la notación de intervalos:

- Mayor expresividad.
- Se puede expresar participación total y parcial.



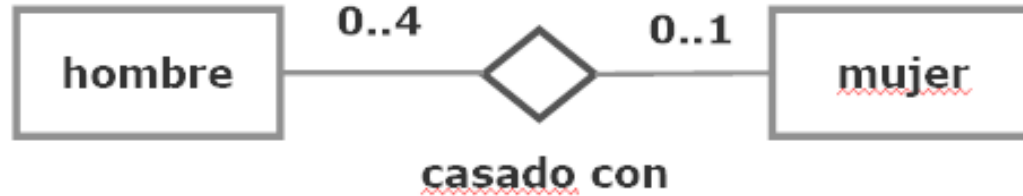
Importante: Notar que el lugar donde se pone la información es al revés (o sea, del otro lado) que en correspondencia de cardinalidades.

Notación Alternativa preferida para Correspondencia de Cardinalidades

- Como deberán poder leer diagramas de entidad-relación deberán saber todas las notaciones enseñadas para conjuntos de relaciones.
- **Ejercicio:** reflejar la siguiente situación usando notación de intervalos: en varios países árabes un hombre puede casarse con hasta 4 mujeres.

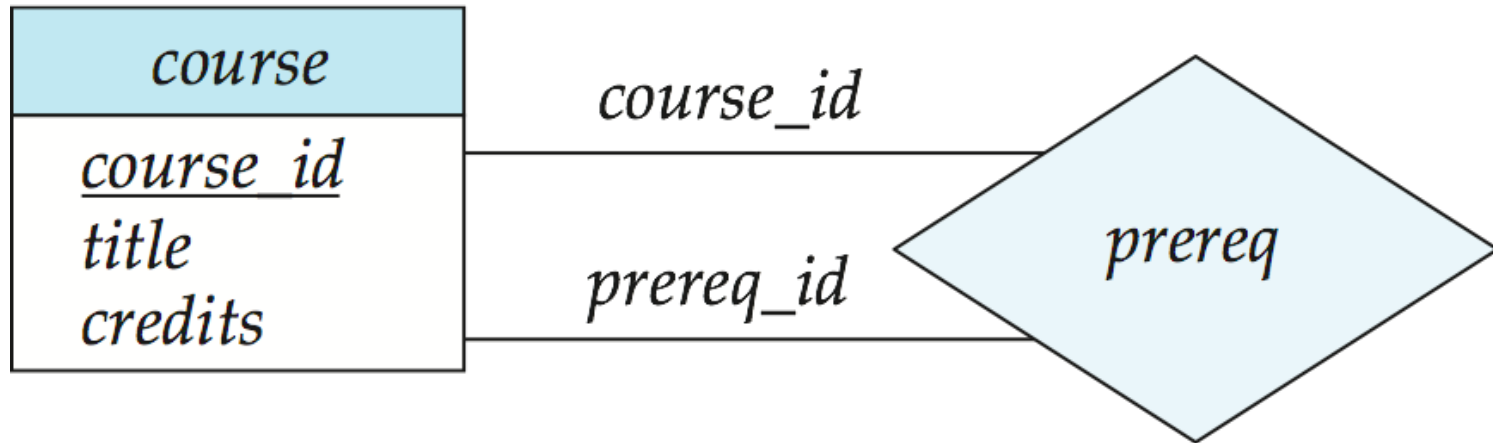
Notación Alternativa para Correspondencia de Cardinalidades

- **Solución:**



Esto no se puede expresar en la otra notación para correspondencia de cardinalidades.




Roles



- Cuando los CE en un CR, son iguales.
 - Cada occurrencia de un CE juega un “rol” en el CR.
- Las etiquetas “*course_id*” y “*prereq_id*” en *prereq* (correlatividades) son llamadas **roles**.

Razones para tener diseño ER y relacional

Comparativa: Diseño Entidad-Relación vs. Diseño Relacional

Dimensión	Diseño Entidad-Relación (E-R)	Diseño Relacional
 Nivel de abstracción	Conceptual: representa entidades, relaciones y atributos del mundo real.	Lógico: define tablas, columnas, claves y restricciones para implementación técnica.
 Separación de preocupaciones	Facilita el diálogo interdisciplinario; accesible para usuarios no técnicos.	Optimiza rendimiento, integridad y consistencia; requiere conocimiento técnico.
 Evolución y mantenimiento	Puede mantenerse como referencia conceptual estable a lo largo del tiempo.	Puede adaptarse por razones técnicas sin alterar el modelo conceptual subyacente.

Reducción a Esquemas Relacionales

- **Problema:** Puede ser que se tiene un buen diseño de esquema de BD usando modelado-ER, pero en la práctica se usan esquemas de BD relacionales.
 - O sea que necesito un buen diseño de esquema de BD relacional.
- **Solución:** **Traducción** de diagrama ER a un conjunto esquemas relacionales con claves primarias y restricciones de clave foránea.
 - Hay **reglas de traducción** útiles.
 - Pero esas reglas **no contemplan todos los casos** y
 - a veces hay que decidir entre alternativas,
 - o mejorar una traducción que no es óptima.

Reducción a Esquemas Relacionales

- **Estudiaremos:**

- Reglas de traducción automáticas.

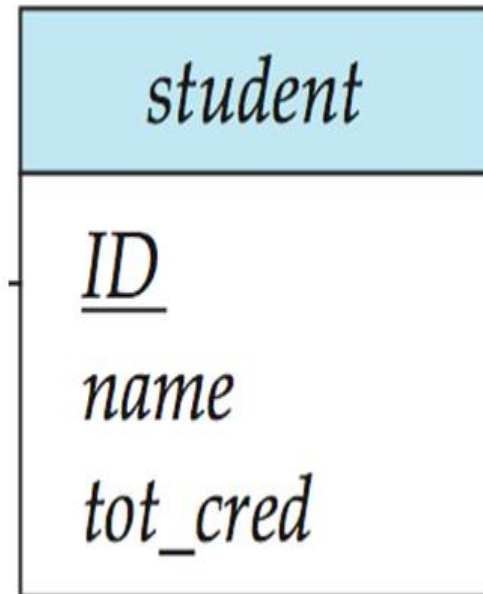
- **Situación más complicada:**

- Cuando *no hay* regla de traducción automática que se pueda usar, pero:
 - Hay alguna idea para hacer la traducción
 - ¿cómo evaluarla?
 - ¿Cómo mejorarla?
 - O hay más de una alternativa para hacer la traducción que se puede aplicar,
 - ¿cómo decidir cuál de ellas conviene usar?
- En ambos casos usar **criterios de decisión** que nos ayudarán:
 - Redundancia de información
 - Facilidad de comprensión
 - Facilidad para hacer consultas

Reducción a Esquemas Relacionales

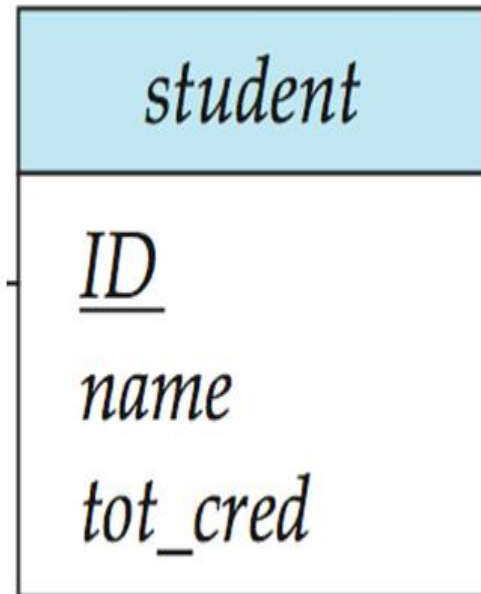
- *Además de los esquemas de relación obtenidos por la traducción automática, se necesita obtener:*
 - **Claves primarias** de esquemas de relación.
 - Para esto se usa la información de claves primarias de CE.
 - **Restricciones de clave foránea.**
 - Para esto se usa el tipo de elemento de modelo ER a mapear (p.ej. CR, CE débiles, atributo multivalorado, etc.).

Reducción a Esquemas Relacionales



¿A qué se mapea?

Reducción a Esquemas Relacionales



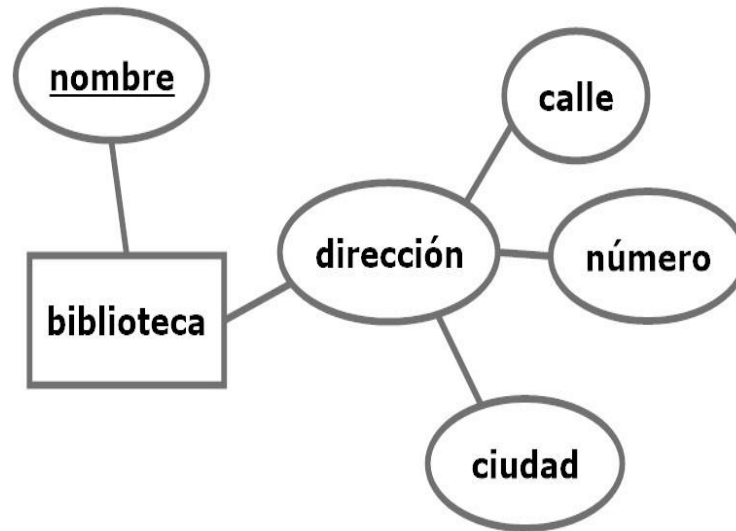
Se mapea a:

Student = (ID, name, tot_cred)

Reducción a Esquemas Relacionales

- **Regla:** Un CE fuerte que no involucra atributos compuestos ni atributos multi-valorados se mapea a un esquema relacional con los mismos atributos.
 - La clave primaria del CE se convierte en la clave primaria del esquema relacional.

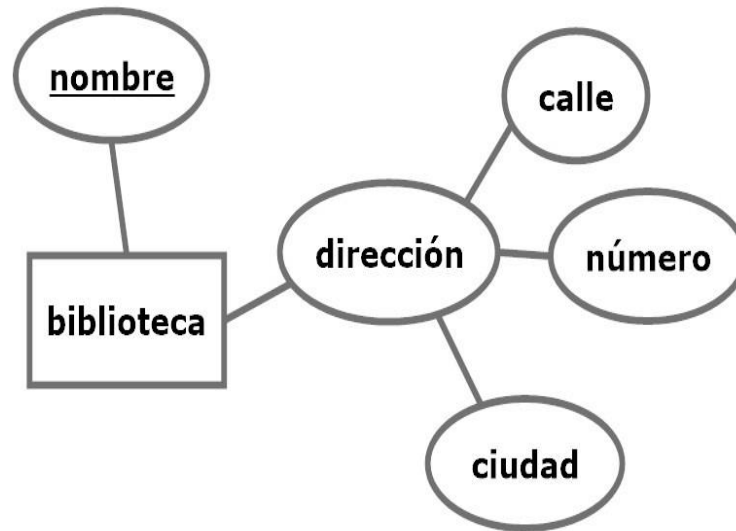
Reducción a Esquemas Relacionales



¿A qué se mapea?

Asumir que queremos hacer consultas que se meten en atributos particulares de *dirección*, como: ¿cuáles son las bibliotecas de la ciudad de Córdoba?

Reducción a Esquemas Relacionales



Se mapea a:

Biblioteca = (nombre, calle, número, ciudad)

Reducción a Esquemas Relacionales



¿A qué se mapea?

Queremos poder hacer consultas sobre un autor específico, por ejemplo: ¿Cuáles son los libros de un autor en particular?

Reducción a Esquemas Relacionales



Se mapea a:

Libro = (título, ISBN, editorial, edición)

Libro-autor = (ISBN, autor)

For Libro-autor foreign key ISBN references Libro

Reducción a Esquemas Relacionales

- **Aclaración:** Cada valor del atributo multivalorado mapea a una tupla separada en la tabla del esquema *Libro-autor*.
 - Por ejemplo, un libro con ISBN 970-17-0256-5 y autores con DNIs: 11122256 y 15345678 se mapea a dos tuplas: (970-17-0256-5, 11122256) y (970-17-0256-5, 15345678)

Reducción a Esquemas Relacionales

- **Decisión:** Los *atributos derivados* no son explícitamente representados en el modelo de datos relacional.
 - Se verá que si se los necesita una forma de computarlos es por medio de consultas.

Reducción a Esquemas Relacionales

- **Ejercicio:** reducir a tablas el siguiente conjunto de entidades

<i>instructor</i>
<u>ID</u>
<i>name</i>
<i>first_name</i>
<i>middle_initial</i>
<i>last_name</i>
<i>address</i>
<i>street</i>
<i>street_number</i>
<i>street_name</i>
<i>apt_number</i>
<i>city</i>
<i>state</i>
<i>zip</i>
{ <i>phone_number</i> }
<i>date_of_birth</i>
<i>age</i> ()

Reducción a Esquemas Relacionales

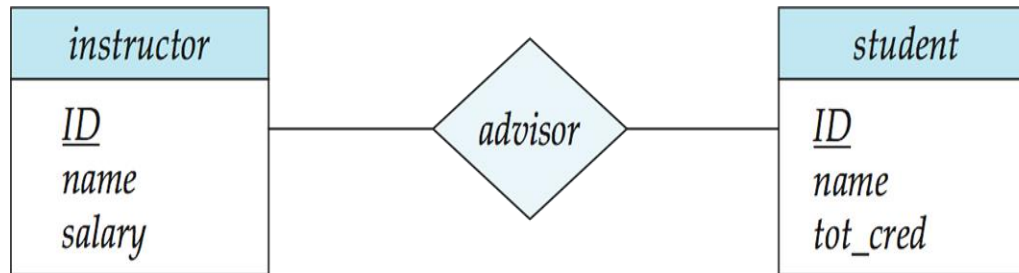
Solución:

Instructor = (ID, name, Street, city, state, zip,
phone_numbers, año_nac, mes_nac, día_nac)

Aclaraciones:

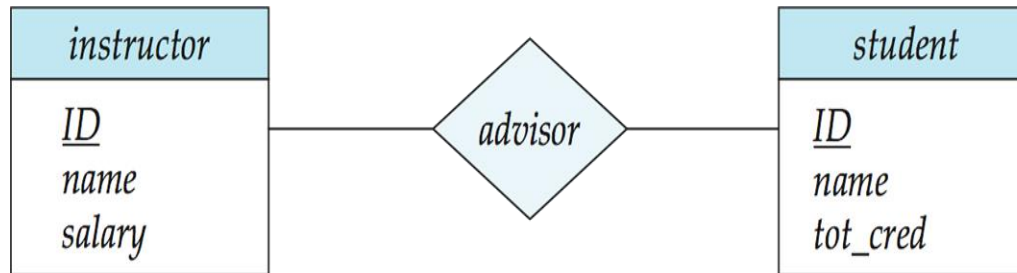
- El *nombre* es manejado como un todo (es poco probable tener consultas que se metan con partes del nombre.) Lo mismo se puede decir del atributo *street*.
- Del Instructor solo interesa saber cómo contactarlo; luego los números telefónicos son vistos como un todo.
- Para el cálculo de la edad es necesario mirar dentro de las partes de fecha nacimiento; por eso agregamos tres atributos para ella.
- Finalmente, edad no se pone por ser derivado.

Traducción a Esquemas Relacionales



¿A qué se mapea advisor?

Traducción a Esquemas Relacionales



Se mapea a:

Advisor = (*i-id*, *s-id*)

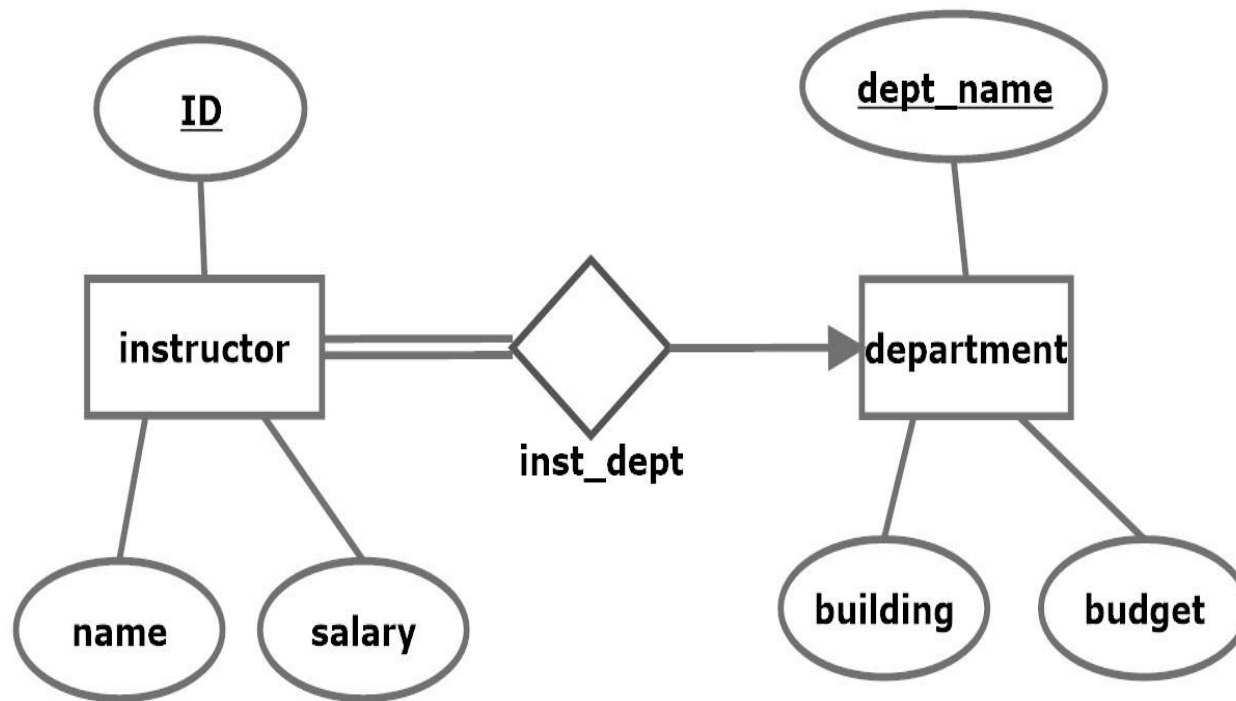
For *Advisor* foreign key *i-id* references *instructor*

For *Advisor* foreign key *s-id* references *student*

Traducción a Esquemas Relacionales

- **Regla CR1:** un CR varios-varios es representado con un esquema con atributos para las claves primarias de los dos CE participantes y todos los atributos descriptivos del CR (que no son multivalorados).
 - la clave primaria del esquema del CR es la unión de las claves primarias de los CE que participan en el CR.
 - Para cada CE que participa en el CR se crea restricción de clave foránea que referencia clave primaria de CE.

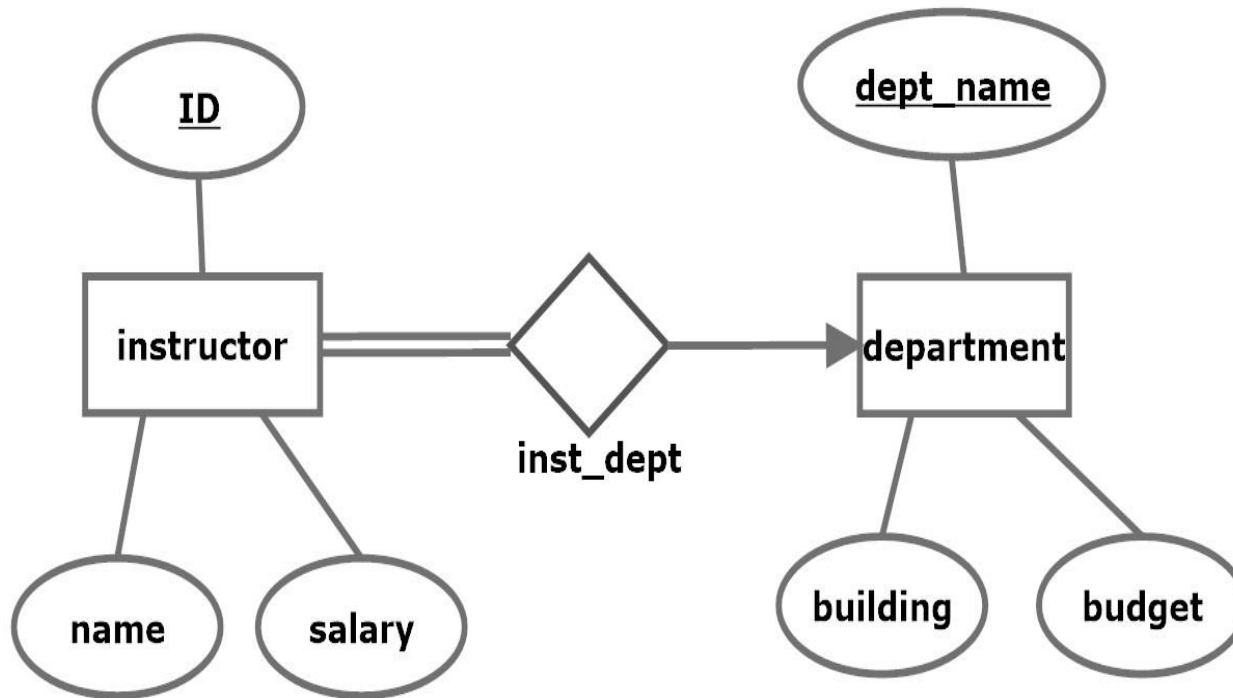
Traducción a Esquemas Relacionales



*¿A qué se mapea CR **inst_dept**?*

Usar el mapeo CR1 es ineficiente porque requiere de almacenamiento extra (si se hace tabla separada para **inst_dept**) y las consultas que vinculan *instructor* con *departamento* son ineficientes.

Traducción a Esquemas Relacionales



CR inst_dept se mapea a:

Instructor = (ID, name, salary, dept_name)

For Instructor foreign key dept_name references Department

traducción a Esquemas Relacionales

- **Regla CR2:** un CR varios a uno o uno a varios que es **total** en el lado varios, puede ser representado agregando atributos extra en el CE del lado varios, conteniendo la clave primaria del lado uno.
 - La clave primaria del CR es la clave primaria del CE del lado varios.
 - Se crea restricción de clave foránea de CR que referencia a clave primaria de CE de lado varios.

Traducción a Esquemas Relacionales

- **Ejercicio 1:** ¿Cómo traducir a un esquema relacional un CR varios uno donde del lado varios no es participación total?
- *Por ejemplo*, un estudiante puede no tener supervisor de trabajo de grado.
- Si la participación es parcial en el lado varios, aplicar la regla CR2 puede resultar en **valores nulos**.
- **Los valores nulos significan:**
 - el valor no existe; o el valor existe, pero no lo sabemos; o no sabemos si el valor existe.
 - Aquí no usamos valores nulos en un sentido tan amplio, solo en el sentido que el valor no existe.
- *Otro ejemplo*: un instructor puede no tener departamento.
 - Estudiaremos este ejemplo en más detalle.

Traducción a Esquemas Relacionales

- **Problema:** Si usamos la traducción de CR2, la clave foránea va a ser inválida (para los valores nulos de *dept_name* en la traducción de instructor).
 - Como *dept_name* es una clave primaria del departamento,
 - esto significa que en *departamento* este atributo no puede tener valores nulos.
- **Conclusión:** Luego no podemos poner la clave foránea y hace falta un requisito que dice:
 - cuando *dept_name* tiene un valor no nulo,
 - debe haber una tupla con ese valor en *departamento*.
- **Por lo tanto, hace falta otra solución.**

Traducción a Esquemas Relacionales

- **Solución:**

- Inst-dept = (ID, deptName)
- Instructor =(ID,name, salary)
- Department =(dept_name, building, budget)
- **For *Inst_dept* foreign key *ID* references *Instructor***
- **For *Inst_dept* foreign key *dept_name* references *Department***

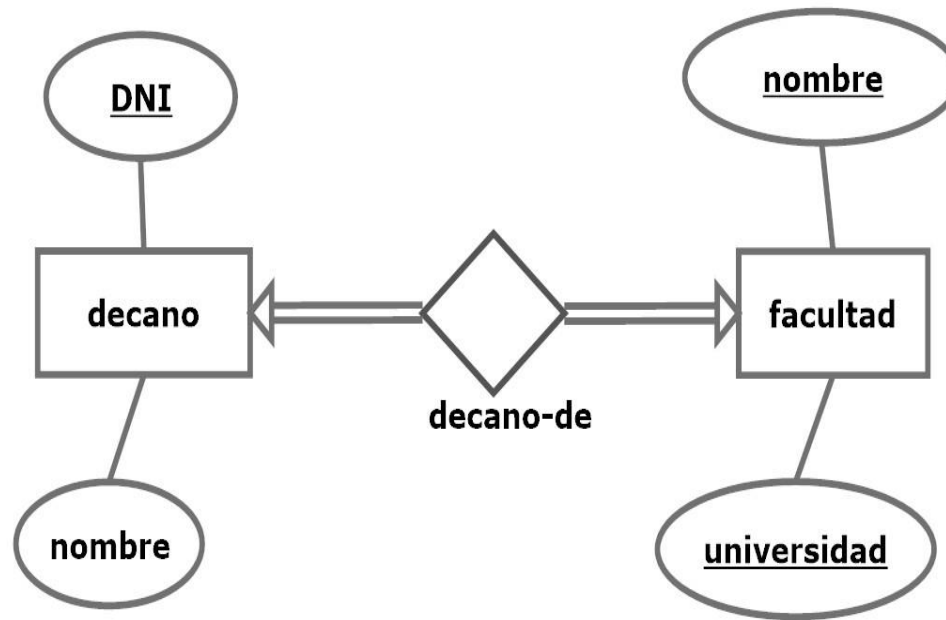
- **Aclaraciones**

- Como *ID* es clave primaria, el valor de *ID* determina unívocamente el valor de *dept_name*. Luego se respeta la condición: varios – uno.
- Por otro lado, nunca se usan valores nulos para *dept_name*; luego no tenemos los problemas de la filmina anterior. Además, se respeta la participación parcial de *instructor* en *departamento*.

- **Conclusión:** Al final terminamos privilegiando las restricciones de integridad del DER frente a las ineficiencias de tener una tabla de *inst_dept*.

Traducción a Esquemas Relacionales

Ejercicio 2: Sea el diagrama siguiente



Hacer la traducción del diagrama a esquemas relacionales.

Traducción a Esquemas Relacionales

- *Usando la idea de CR2 se nos ocurren las siguientes alternativas.*
- Decano = (DNI, nombre, nombreFacultad, universidad)
- Facultad = (nombre, universidad)
- **For** Decano **foreign key** nombreFacultad, universidad **references** Facultad (nombre, universidad)

y

- Facultad = (nombre, universidad, DNI)
- Decano = (DNI, nombre)
- **For** Facultad **foreign key** DNI **references** Decano

- **Aclaración:**

- Las dos alternativas cumplen las restricciones de integridad del DER.
- Una facultad puede tener dos decanos (diseño de arriba),
- una persona puede ser decano de dos facultades (diseño de abajo).
- Ambos diseños son ineficientes en términos de almacenamiento.
- Buscamos resolver esas ineficiencias.

Traducción a Esquemas Relacionales

- *Resolviendo las ineficiencias anteriores obtenemos las siguientes alternativas (que se derivan de las previas):*
- Decano = (DNI, nombre, nombreFacultad, universidad)
- **Aclaración:** Facultad no hace falta por tener sus atributos en decano. Al sacar Facultad no hace falta la clave foránea.

y

- Facultad = (nombre, universidad, DNI, nombreDecano)
- **Aclaración:** una tabla para *decano* no hace falta por tener sus atributos en *facultad*. Al sacar *decano*, no hace falta la clave foránea.
- El problema de que facultad puede tener dos decanos no se puede resolver,
 - porque hay que decir que nombreFacultad y universidad son clave candidata.
 - De ultima poner en español eso.
- Resta entonces elegir la mejor entre ambas alternativas.

Traducción a Esquemas Relacionales

- Para una base de datos universitaria es más importante la facultad que el decano (usar una tabla facultad tiene más sentido desde el punto de vista lógico).
 - Por lo tanto, conviene tener una tabla que se llame *facultad* y que adentro tenga los atributos de quien es el decano.
 - Entonces, elegimos la segunda alternativa.
- Los ejercicios 1 y 2 son ejemplos de ejercicio para parcial o final, donde tienen que:
 - llegar a una traducción que cumpla restricciones de integridad del DER,
 - que sea eficiente (para consultas y almacenamiento) y
 - que sea adecuada desde el punto de vista lógico.