

# Tema 6: Introducción a las Bases de Datos y al Modelo Relacional

Introducción a la Ingeniería del Software y los Sistemas de Información I  
Ingeniería Informática – Tecnologías Informáticas  
Departamento de Lenguajes y Sistemas Informáticos



1.Introducción

2.Conceptos básicos

3.Claves

4.Integridad

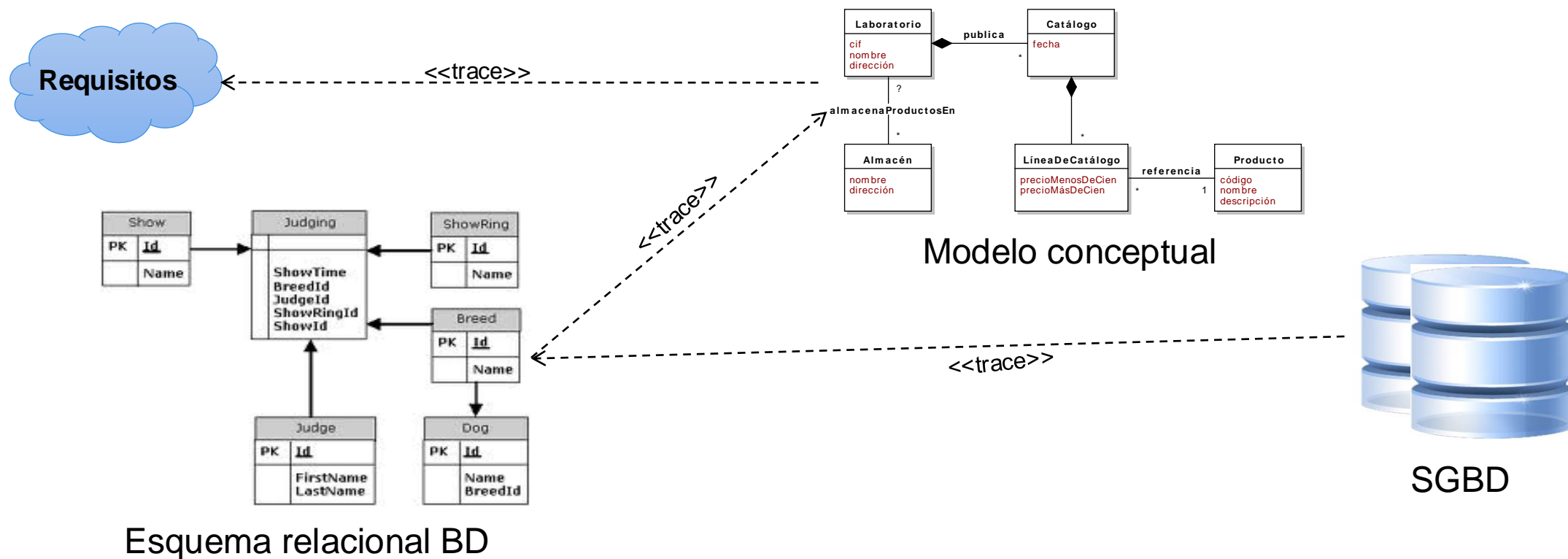
## ¿Qué es una base de datos?

Un sistema de gestión de base de datos (SGBD) es un sistema informático que:

- Implementa (principalmente) las funciones de **memoria** e **informativa** de un sistema de información.
- Almacena grandes volúmenes de datos.
- Gestiona el acceso concurrente a los datos.
- Mantiene la integridad semántica de los datos.
- Controla el acceso a los datos.

## Trazabilidad hacia otros productos

- El producto previo al diseño de una BD es el modelo conceptual.



## Principales hitos:

- **Sistemas pre-relacionales (antes de 1970)**
  - Basados en archivos (secuenciales e indexados); bases de datos jerárquicas; bases de datos en red (CODASYL)
- **Sistemas relacionales (desde 1970)**
  - Experimentales: RDMS, Ingres, ...
  - Comerciales: Oracle, DB2, MS SQL Server
  - Open source: PostgreSQL, MySQL, MariaDB, SQLite...
- **Sistemas post-relacionales (desde 1990)**
  - Orientados a objetos; Objeto-relacionales; NoSQL.

## Orígenes del modelo relacional

- [Codd, E. F. \(1970\). \*A Relational Model of Data for Large Shared Data Banks\*. Communications of the ACM 13 \(6\): pp. 377–387.](#)
- [Modelo sencillo y flexible basado en fundamentos matemáticos de \*\*teoría de conjuntos\*\* y \*\*lógica de predicados\*\*.](#)
- [Primeras implementaciones:](#)
  - [RDMS \(MIT\), primeros años de los 70](#)
  - [Ingres \(Universidad de Berkeley\), 1974](#)
  - [Oracle V2, 1979](#)
  - [IBM System R/DB2, 1979](#)

## ¿Qué es una **relación**?

- Una relación **está compuesta por una *intensión* y por una *extensión*.**
- **La *intensión* define un conjunto de *atributos*, cada uno de los cuales toma valores sobre un *dominio*.**
- Por **ejemplo**, para una relación denominada *Empleado*, su *intensión* podría ser:
  - **nif**: Naturales de 8 dígitos seguidos de una letra mayúscula
  - **nombre**: Cadenas menores de 50 caracteres
  - **salario**: Reales positivos

## ¿Qué es una relación?

- La **extensión** es un **conjunto de tuplas**, cada una formada por **conjuntos de pares (atributo, valor)**, de forma que a cada atributo de la intensión se le asocia un **valor** del dominio sobre el que está definido.
- Por ejemplo, para la relación *Empleados*, su extensión podría ser:

relación		Empleados		
atributos/intensión		nif	nombre	salario
tuplas/extensión		12.345.678-Z	Abel Abad	12.000
		23.456.789-D	Braulio Brío	23.000
		34.567.890-V	Carlos Cepa	34.000
		45.678.901-G	David Díaz	45.000
		56.789.012-B	Enrique Estepa	56.000

A dashed blue arrow labeled **valor** points to the value 34.000 in the salary column, which is also circled in blue.



## ¿Qué es una relación?

- El número de atributos definidos en la intensión se denomina **grado** de la relación.
- El número de tuplas de la extensión se denomina **cardinalidad** de la relación.

**grado = 3**

Empleados		
nif	nombre	salario
12.345.678-Z	Abel Abad	12.000
23.456.789-D	Braulio Brío	23.000
34.567.890-V	Carlos Cepa	34.000
45.678.901-G	David Díaz	45.000
56.789.012-B	Enrique Estepa	56.000

**cardinalidad = 5**

## Implementación del concepto de relación

- El concepto de relación se implementa mediante una **tabla**, cuyas **columnas** representan los atributos y las **filas** las tuplas.
- Las diferencias con una relación son:
  - Las filas están ordenadas, las tuplas no.
  - Las filas pueden estar repetidas, las tuplas no.
  - Las columnas (además de un nombre) tienen un orden, los atributos no.

## ¿Qué es un dominio?

- Es el **conjunto de valores admisibles** que puede tomar un atributo de una relación.
- Pueden ser:
  - Un tipo básico sin restricciones: enteros, cadenas, ...
  - Un tipo básico con restricciones: reales  $< 100$ , cadenas de longitud  $\leq 50$ , naturales  $> 20$ , ...
  - Un tipo enumerado: lunes, martes, miércoles, ...
  - Un tipo compuesto: fechas (dd/mm/aaaa), horas (hh:mm:ss), ...
- Todos los dominios deben tener definido el operador de **igualdad (=)**.
- Algunos dominios tienen su propia **álgebra** con operadores como +, -, \*, /, etc.

## ¿Qué es el **valor nulo**?

- Si un atributo tiene **valor nulo** (null), **significa que no se conoce su valor, que es desconocido.**
- El valor nulo **puede asignarse a atributos definidos sobre cualquier dominio, pero no pueden compararse valores nulos de atributos definidos sobre dominios diferentes.**
- **La introducción del valor nulo implica la necesidad de una **lógica trivaluada:****

A	B	A OR B	A AND B	NOT A
true	null	true	null	false
false	null	null	false	true
null	null	null	null	null

## Definiciones:

- Una **Superclave** es un subconjunto de atributos (**descriptor**) de una relación que **identifica** de manera **única** a las tuplas de dicha relación.
- **Criterio de unicidad**
  - Un **descriptor** es **único** si no tiene sentido\* que existan dos tuplas distintas en una relación con los mismos valores de dicho **descriptor**.
  - Toda **Superclave** cumple este criterio

\* Si no tiene sentido en el dominio del problema, asumiendo que una relación representa una parte de un modelo conceptual, como se verá en los próximos temas.

## Definiciones:

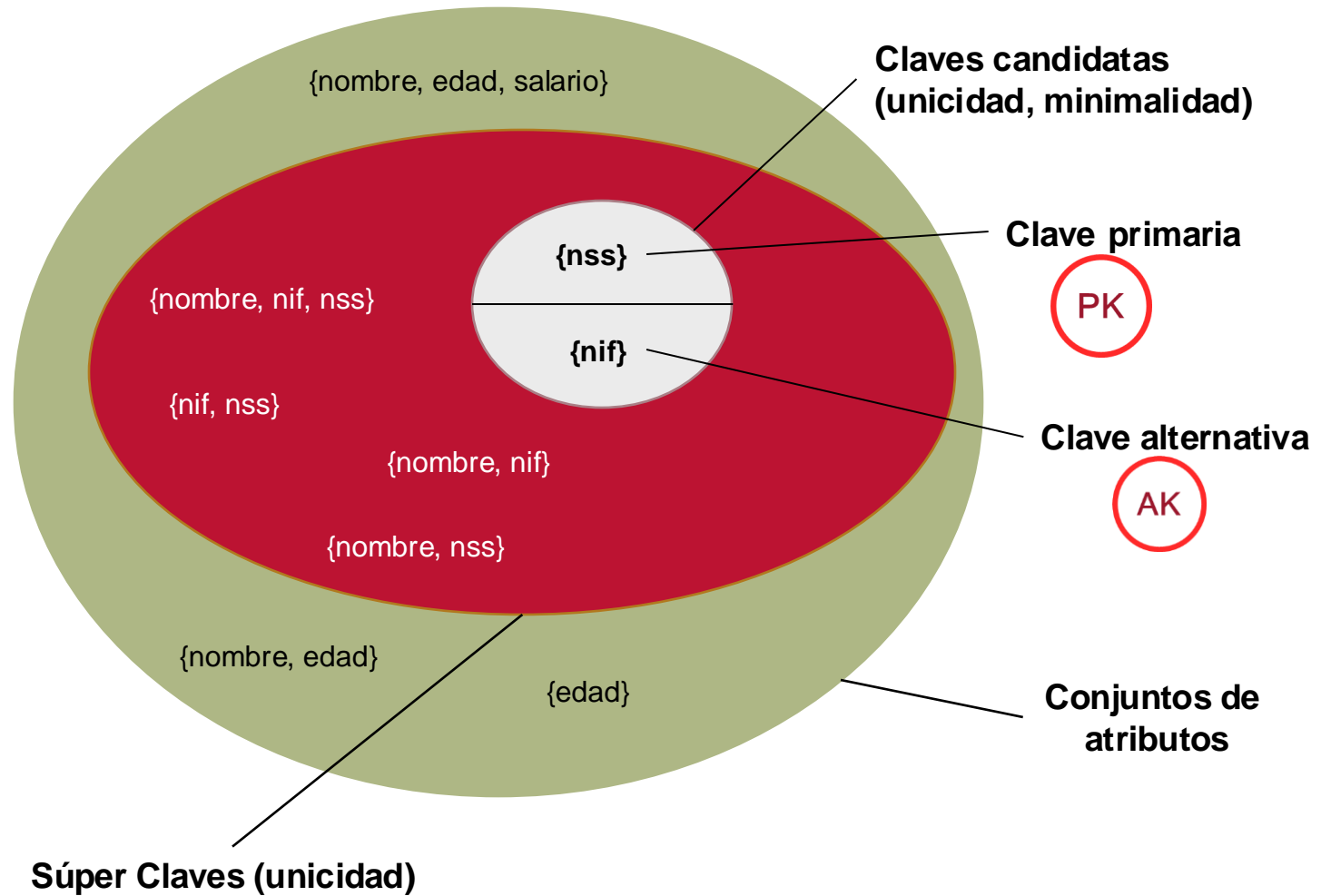
- En cualquier relación, siempre existe una **superclave** formada por todos sus atributos, ya que no puede haber tuplas repetidas.
- Una relación puede tener varias **superclaves**.

## Definiciones:

- Una **Clave Candidata** es una superclave que cumple el criterio de **minimalidad**.
- Criterio de **Minimalidad**
  - Una **Superclave** es mínima si al eliminar cualquier atributo de su descriptor deja de cumplir el **criterio de unicidad**.
- Hay que evitar las Superclaves que no son Claves Candidatas.

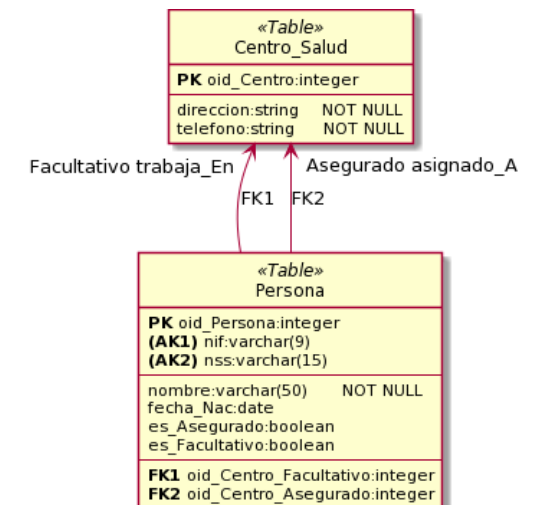
- Claves candidatas
  - Una relación siempre tiene al menos una clave candidata, aunque puede tener varias.
- Clave primaria (primary key, **PK**)
  - Clave candidata seleccionada arbitrariamente como mecanismo de identificación de la relación.
- Clave alternativa (alternative key, **AK**)
  - Cualquier clave candidata no seleccionada como primaria.





## Claves ajenas (foreign keys, **FK**)

- Las claves ajenas son conjuntos de atributos de una relación cuyos valores deben **coincidir** con los de la clave primaria de otra relación.
- Es la forma de representar las **asociaciones** del modelo conceptual en el modelo relacional.
- Se debe evitar ponerles nombres distintos a la clave primaria.



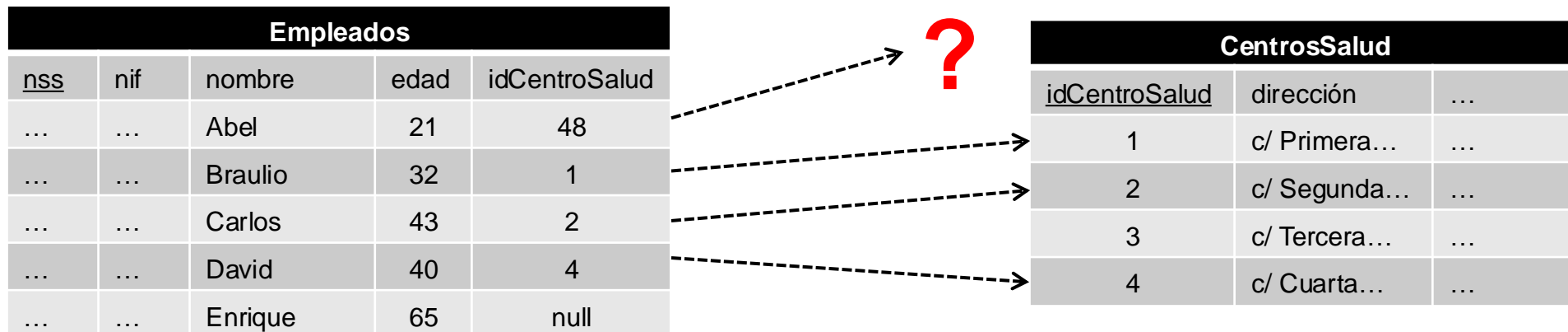
## Integridad de la entidad

- Ningún atributo que forme parte de la clave primaria de una relación puede tomar el valor nulo.
- Esta regla de integridad garantiza la identificación de tuplas mediante valores de la clave primaria.

Empleados					
<u>nif (PK)</u>	nss	nombre	edad	salario	estadoCivil
12.345.678-Z	123.456.789	Abel Abad	21	12.000	soltero
null 	234.567.890	Braulio Brío	32	23.000	casado
34.567.890-V	345.678.901	Carlos Cepa	43	34.000	separado
45.678.901-G	456.789.012	David Díaz	54	45.000	divorciado
56.789.012-B	567.890.123	Enrique Estepa	65	56.000	casado

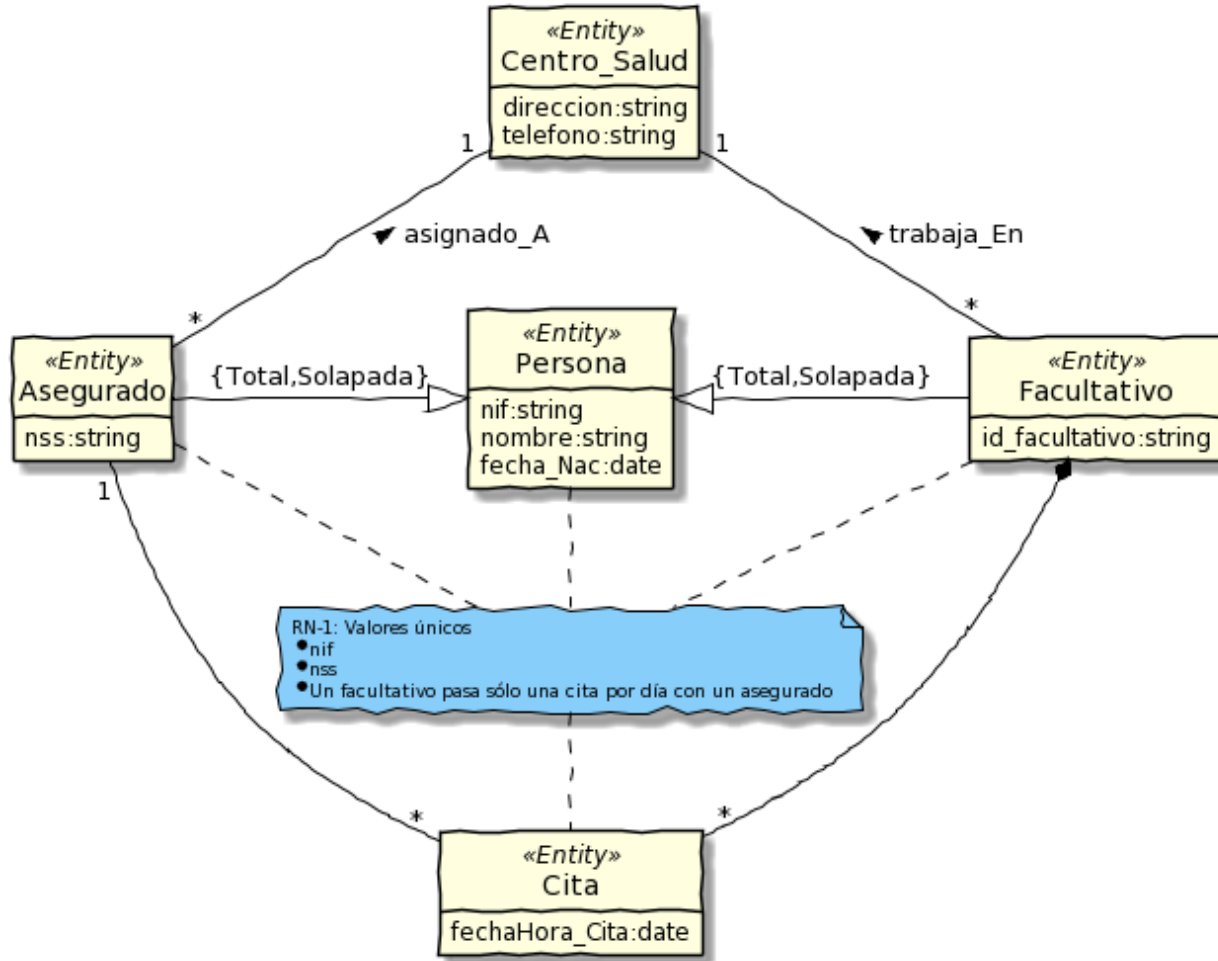
## Integridad referencial

- Todos los atributos de una clave ajena deben tomar valores que coincidan con valores de la clave primaria correspondiente o bien tomar valores nulos.
- Esta regla de integridad garantiza que todas las tuplas con claves ajenas se relacionan con otras tuplas existentes o bien con ninguna.

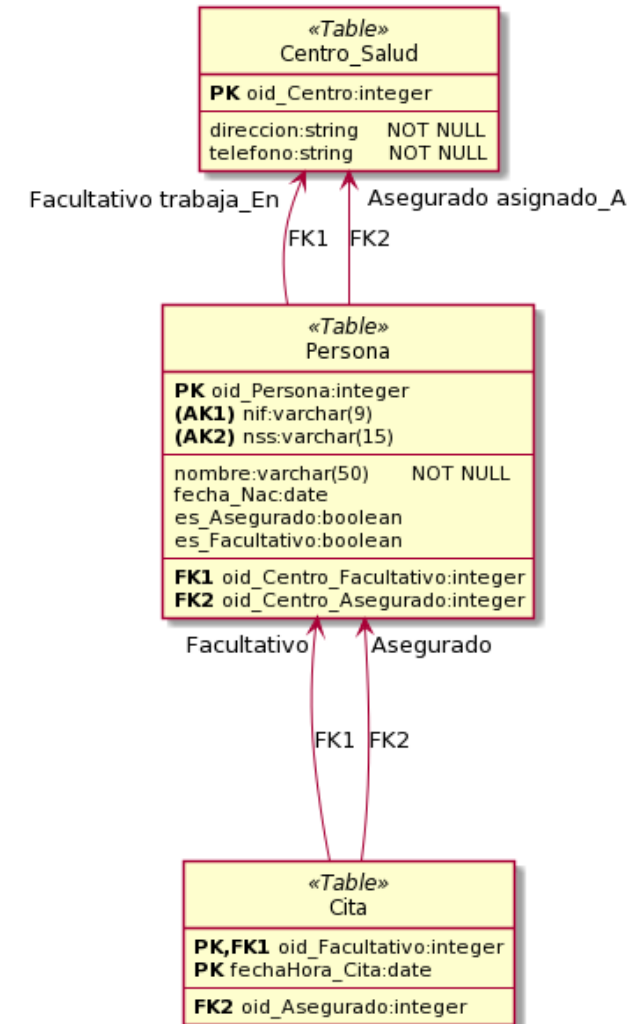


# Esquema conceptual vs Esquema relacional

Esquema Conceptual UML



Esquema Relacional -Diagrama relacional-



# Tema 6: Introducción a las Bases de Datos y al Modelo Relacional

Introducción a la Ingeniería del Software y los Sistemas de Información I  
Ingeniería Informática - Ingeniería del Software  
Departamento de Lenguajes y Sistemas Informáticos

