



**UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS**  
(Universidad del Perú, DECANA DE AMÉRICA)

FACULTAD DE INGENIERIA DE SISTEMAS E INFORMATICA  
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

“Adaptado en el marco de la emergencia sanitaria por el COVID-19”

# Base de Datos I

## Semana 3

*Docente: Dr. Luis Soto Soto*

# Agenda

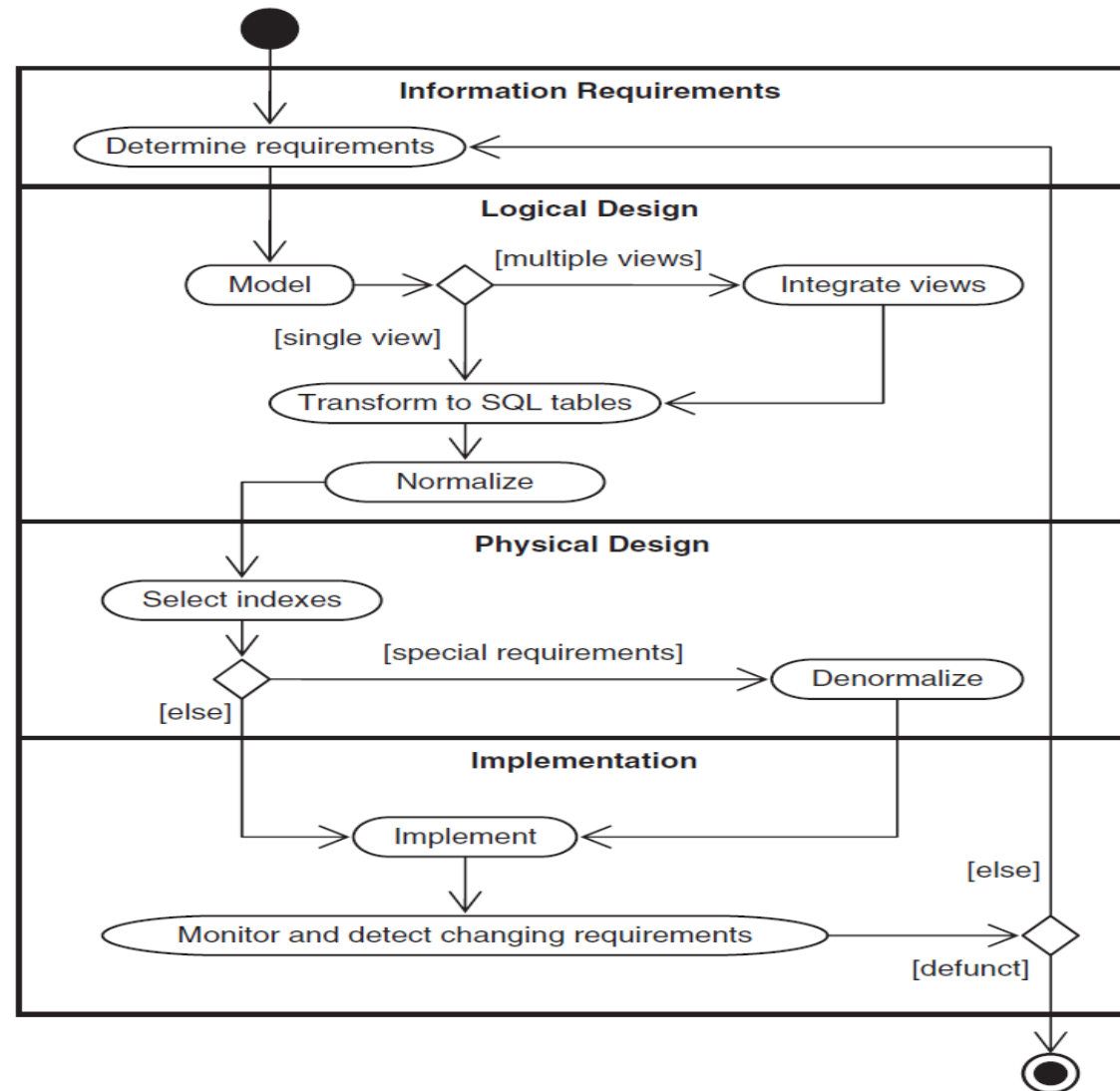
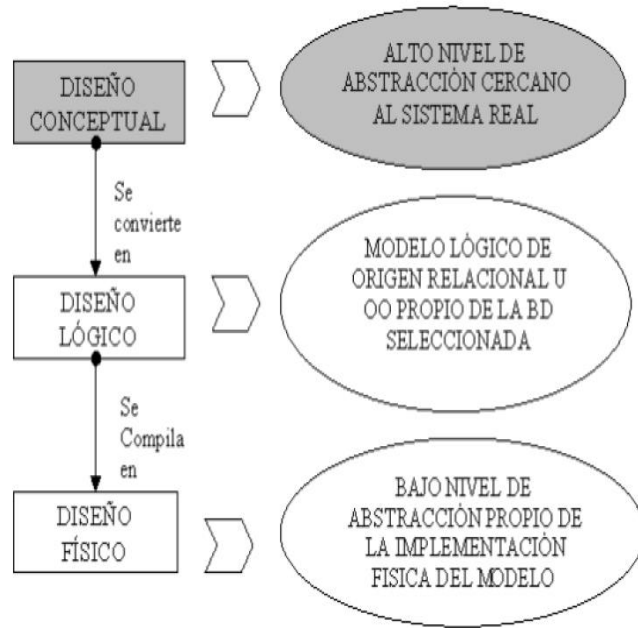
- Diseño Lógico de Base de Datos.
- El Modelo Relacional Restricciones implícitas del Modelo Relacional

# 1. Diseño Lógico de Base de Datos

# Diseño Lógico de Base de Datos

- En esta etapa se transforma el diseño conceptual al modelo lógico de base de datos.
- Un **modelo lógico de datos** es un **modelo** que no es específico de una **base de datos** que describe aspectos relacionados con las necesidades de una organización para recopilar **datos** y las relaciones entre estos aspectos.
- Un esquema **lógico** es una descripción de la estructura de la **base de datos** en términos de las estructuras de **datos** que puede procesar un tipo de SGBD.
- El **diseño lógico** depende del tipo de SGBD que se vaya a utilizar, no depende del producto concreto.
- Consiste en un diagrama entidad relación a nivel lógico, es decir un modelo ER independiente del sistema gestor de base de datos (SGBD) un SGBD es por ejemplo Sql Server u Oracle, Otros.
- El modelo lógico debe contar con entidades, Pk, Fk, otras restricciones de datos.
- El modelo lógico se concluye con un modelo de datos normalizado es decir eliminar las redundancia de datos.

# El ciclo de vida de la BD

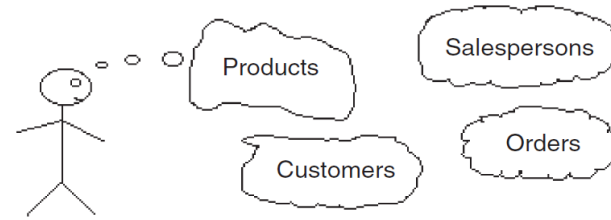


**Figure 1.1** The database life cycle.

# El ciclo de vida de la BD

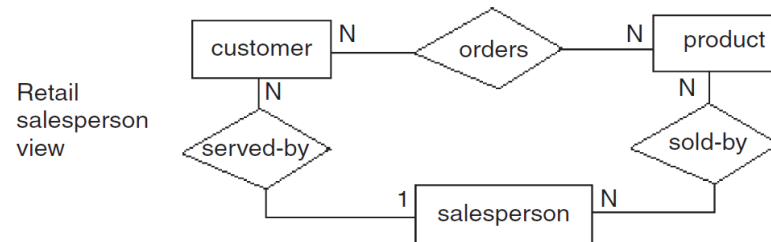
## Database Life Cycle

### Step I Information Requirements (reality)

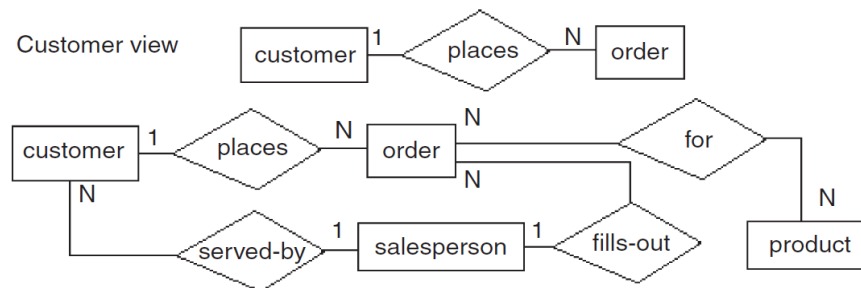


### Step II Logical design

#### Step II.a Conceptual data modeling



#### Step II.b View integration



Integration of retail salesperson's and customer's views

**Figure 1.2** Life cycle results, step by step (continued on following page).

## 2. El Modelo Relacional Restricciones implícitas del Modelo Relacional

# Restricciones implícitas del Modelo Relacional

## Restricciones inherentes

---

- ▶ Las restricciones inherentes vienen impuestas por el propio Modelo de Datos.
- ▶ En el caso del MR, una **relación** tiene unas propiedades intrínsecas que no tiene una tabla, y que se derivan de la misma definición matemática de relación, ya que, al ser un **conjunto**:
  - ▶ No puede haber dos tuplas iguales => obligatoriedad de la PK
  - ▶ El orden de las tuplas no es significativo.
  - ▶ El orden de los atributos no es significativo.
  - ▶ Cada atributo sólo puede tomar un único valor del dominio subyacente
    - ▶ Se dice que la relación está normalizada (en 1FN).
- ▶ Otra restricción es **la regla de integridad de entidad**:

“Ningún atributo que forme parte de la clave primaria de una relación puede tomar un valor nulo”



# Diseño Lógico de Base de Datos

## Restricciones semánticas

- Son definidas por el usuario
- Son facilidades que el modelo ofrece a los diseñadores para que puedan reflejar en el esquema, la semántica del mundo real.
- Los tipos de restricciones semánticas permitidos en el modelo relacional (incorporados a SQL 92) son:
  - Clave Primaria (PK)
  - Clave foránea (Fk)
  - Obligatoriedad (Not null)
  - Unicidad (Unique)
  - Verificación (Check)
  - Disparador (Trigger)
  - Aserción (Assertion – Check)

# Diseño Lógico de Base de Datos

**Entidad:** La entidad es cualquier cosa de interés para la empresa, acerca del cual almacenamos datos y es identificable de manera única a través de su clave primaria. Ejemplo de entidades son producto, Empleado, Cliente, Etc.

Las Entidades son el vehículo para construir base datos.

**Atributo:** Los atributos son información que se necesita conocer o tener acerca de una entidad. Los atributos describen una entidad para calificar, identificar, clasificar, cuantificar o expresar el estado de una entidad.

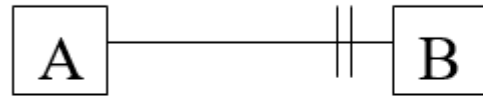
**Clave primaria:** La clave primaria (PK), es un atributo o grupo de atributos que identifican de manera única a cada instancia (fila de la tabla o registro). La clave primaria nunca es nula, por ejemplo el código del producto para la entidad Producto, código del empleado para la Entidad Empleado y código del cliente para la Entidad Cliente.

**Relación:** Una relación es bidireccional y representa la asociación entre dos entidades o entre una entidad consigo mismo. Cada dirección de una relación se da por una regla de negocio. No se debe perder de vista su cardinalidad en cada dirección de la relación.

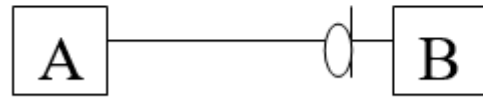
# Diseño Lógico de Base de Datos

## 32. SIMBOLOS UTILIZADOS DER

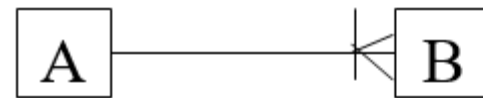
Los símbolos utilizados en para elaborar el diagrama entidad relación son:



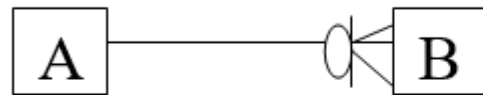
CADA A DEBE SER UNA Y SOLA UNA B



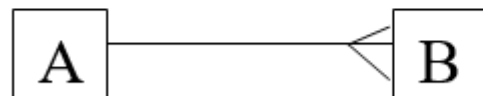
CADA A PUEDE SER UNA Y SOLA UNA B



CADA A DEBE SER UNA Ó MUCHAS B



CADA A PUEDE SER UNA Ó MUCHAS B

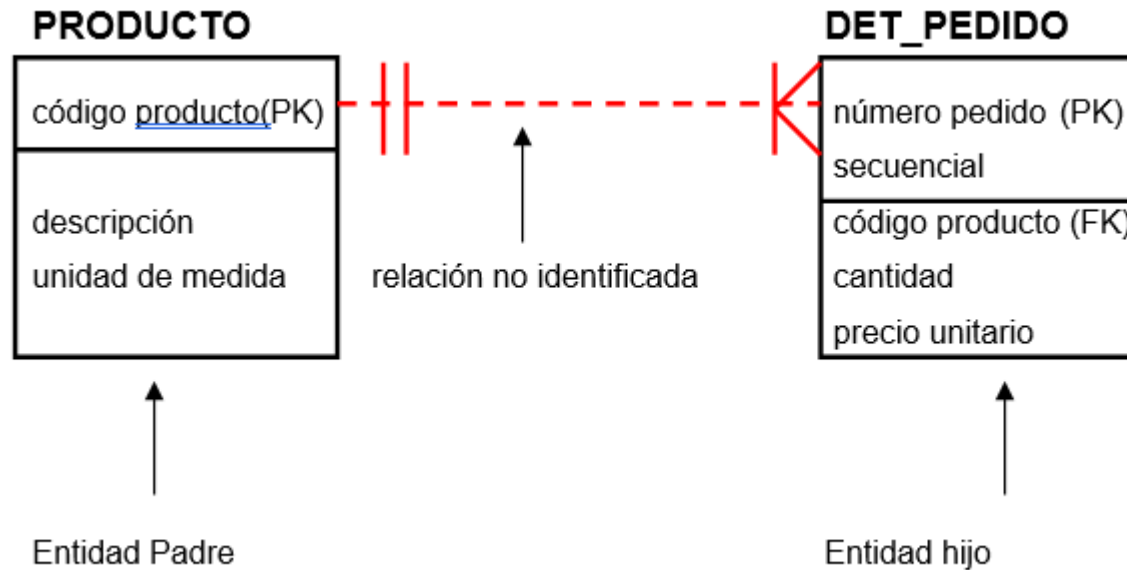


CADA A DEBE SER MUCHAS B

# Diseño Lógico de Base de Datos

## Clave Foránea (FK)

Una llave foránea es un atributo o conjunto de atributos en una entidad (denominada entidad hijo) y que es clave primaria en otra entidad (denominada entidad padre). Los atributos foráneos pueden formar parte de la clave primaria (sí la relación es identificada). Ejemplo:



- ¿DÓNDE UBICAR LA CLAVE FORANEA en una relación 1:1 o 1:M o M:M?
- RELACIONES RECURSIVAS

# Diseño Lógico de Base de Datos

**Obligatoriedad (Not null):** Especifica que la columna no puede contener un valor nulo.

**Unicidad (Unique):** Especifica una o varias columnas cuyos reglones deben ser únicos a través de todos los reglones. ORACLE crea un índice único para asegurar este constraint.

**Verificación (Check):** Especifica una condición que cada reglón en la tabla debe satisfacer.