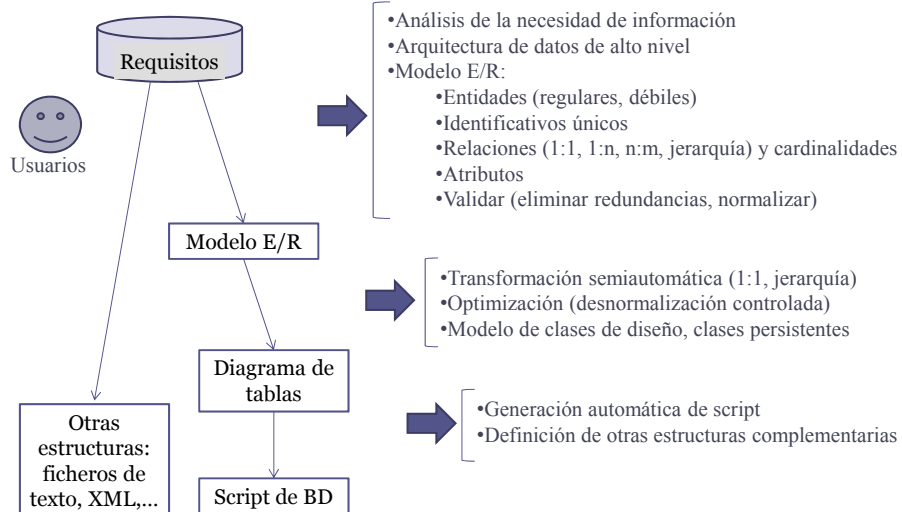




Modelo de datos

Ingeniería del Software Avanzada
Técnicas de análisis y diseño

Proceso



Modelo E/R

- **Entidades**
 - Objeto sobre el que se desea almacenar información
 - Todas las ocurrencias de una entidad tienen los mismos atributos (información que las caracteriza) y se distinguen entre ellas:
 - Identificador único (conjunto mínimo de atributos que las identifica)
 - Débiles, su existencia depende de otra (posible dependencia en identificación)
- **Relaciones**
 - Correspondencia entre entidades
 - Cardinalidad:
 - Diferentes tipos de relación indican cardinalidad máxima
 - Importancia como reglas de negocio la cardinalidad mínima
 - n:m con posibilidad de atributos
 - Jerarquía (total, disjunta, no disjunta)



Modelo E/R – Explicación adicional

Débiles: Ciudad/Código Postal, si se elimina una ciudad, desaparecen sus códigos postales.

Débiles con dependencia en identificación: Líneas de pedido. Si se elimina un pedido desaparecen sus líneas de pedido. Las líneas de pedido se llaman con el código del pedido más un secuencial.

Jerarquía total: todos los empleados son administrativos o profesores. **Disjunta:** no hay empleados que sean administrativos y profesores a la vez. **No disjunta:** un empleado puede ser comercial y también directivo.

Redundancia de relaciones: mirar triángulos. Un coche está matriculado en una ciudad y solo en una y una persona está empadronada en una ciudad y solo en una. Si un coche tiene un solo propietario y debe estar matriculado en la ciudad de su propietario, entonces la relación coche/ciudad sobra. Si un coche puede tener más de un propietario o matricularse en una ciudad distinta a la de su propietario, entonces no sobra.

Diagrama de tablas

- **Transformación**
 - Entidad, tabla; Identificador único, clave; Atributo, campo; Relaciones, claves externas o tablas
 - Decisiones de diseño:
 - Relaciones de jerarquía: unir subtipos a supertipo, sin uniones, unir supertipo a subtipos
 - 1:1, extender claves externas en una u otra dirección (o en las dos)
- **Optimización**
 - Objetivo: reestructurar el modelo físico de datos para mejorar la eficiencia del sistema
 - Desnormalización controlada del modelo físico para reducir o simplificar el número de accesos a la base de datos:
 - Elementos redundantes (atributos, relaciones y tablas)
 - Elementos calculados (atributos)
- **Consistencia**
 - Entre el modelo de clases (clases persistentes) y las tablas, sus campos y sus relaciones

Diagrama de tablas - Explicación adicional

Paso E/R a Tablas - Jerarquía: tabla empleados (una tabla con un campo tipo que indique el tipo de empleado), tablas profesores y administrativos (dos tablas), tablas profesores, administrativos y empleados (esta última con los campos comunes a los dos). Depende de la cantidad de información común y propia, de las relaciones con otras tablas (si son desde el supertipo o desde algún subtipo), de la forma normal de acceso, de la privacidad de la información,...

Paso E/R a Tablas - 1:1: empleado/despacho. Lo habitual es usar la cardinalidad mínima (todos los despachos son de alguien pero hay muchos empleados sin despacho, luego la clave externa se pone en despacho, se evita que sea null). Se puede poner clave externa en las dos, mejor accesos pero cuidado con información redundante.

Desnormalización: incluir la edad, además de la fecha de nacimiento. Incluir una tabla de empleados/cliente, con las horas trabajadas por cada empleado en cada cliente, además de las tablas empleado/proyecto (con las horas trabajadas de cada empleado en cada proyecto) y proyecto/cliente.

Estructura física

- La arquitectura del sistema de alto nivel determina la arquitectura de datos (bases de datos relacionales, ficheros, etc.)
- En fase de diseño se determina la tecnología y herramientas (ficheros xml, postgresql, etc.)
- Las herramientas CASE permiten obtener automáticamente el script de definición de base de datos (según la tecnología) a partir del diagrama de tablas
- Otras opciones de estructuras de datos (fichero de texto, xml, etc.) deben definirse y generarse