

## Tema 2: Estática Relacional

La información cuanto más restringida es más valiosa.

- El modelos relacionan se basa en la noción matemática de relación,  $R: N \times N$ :
  - Se pueden describir mediante una función,  $R=\{(x, y) / y=x^2\}$
  - O se pueden enumerar las relaciones,  $R=\{(1,1), (2,4), \dots\}$
- **Dominio:** Conjunto de valores de la misma naturaleza. Puede tener restricciones.
- **Relación:** Subconjunto de productos cartesianos, se pueden dar entre elementos del mismo dominio o entre elementos de distinto dominios.
  - Nombre x Nombre. Persona: Nombre x Edad x Altura x Teléfono
- **Atributo:** Propiedad común entre los elementos de una relación, y se define sobre un dominio. Ej: edad se define sobre el dominio de los enteros entre 0 y 120 años.
  - **Definición Intensiva:** Definición invariable, no cambia con el tiempo. Se representa como: etiqueta(atrib1, atrib2,...)
  - **Definición Extensiva:** Definición variable, para referirnos uno de los miembros en un momento del tiempo. Se representa en una tabla, cada individuo en una fila y como cabecera estaría la definición intensiva y el nombre del atributo.

Dominio y Relación son independientes / Relación y Atributo dependientes.
- **Esquema de una relación:** Asociación de atributos que caracteriza y distingue a los miembros de una relación. relación  $\rightarrow$  etiqueta(atrib1, atrib2, atrib3,...)
  - Persona (Nombre, DNI, Teléfono)
    - **Grado:** Numero de atributos, es estático.
    - **Cardinalidad:** Numero de tuplas, numero de filas en una tabla, dinámico.
- **Restricciones Inherentes:**
  - El orden de las tuplas no es significativo (filas), ni tampoco el de los atributos dentro de la tupla(columnas). Lo importante es que definido el orden, todos los atributos los almacenemos en el mismo orden.
  - No hay dos tuplas iguales, no completamente. Deben identificar individuos.
  - Cada atributo toma un solo valor del dominio dentro de una tupla.
  - **Integridad de Entidad:** Si algo existe es identificable, si no es identificable no existe. Y ese identificador no puede tomar valor nulo.
  - **Integridad Referencial:** Todo valor referido o lo referenciado existe, debe apuntar al dato y no puede dejar de hacerlo o se rompe la integridad.
- **Representación y notación :** Se usa camelCase y snake\_case, preferiblemente la ultima. Los nombre deben ser representativos y no se pueden repetir en el mismo ámbito las etiquetas.
  - **Definición intensiva** (es la que utilizaremos): grafo relacional.
  - **Definición extensiva:** representación tabular(en una tabla).
  - Los dominios tienen etiqueta en plural. Ej: Personas, Teléfonos...
  - Los atributos tienen etiqueta en singular(ya que solo hay un valor). Ej:DNI, nombre, teléfono...
- **Ocurrencia o Tupla:** Asociación de valores para un individuo.  $\langle \text{val1}, \text{val2}, \text{val3}, \dots \rangle$   
*ej: <Jorge, Rodríguez, 02592568, S>*

- **Atributos opcionales:** No tienen porqué tener valor, cuando esto ocurre toma el valor null, 0 o “. El dominio incluye el nulo. Se indica con un \*. Esto se da cuando:
  - Es desconocido por el usuario. No la conoce todavía.
  - No aplicable. No siempre se tiene por los que puede dejarse en blanco.
  - Es desconocido por mi. No se ha sido dado al sistema.
- **Atributo obligatorio:** Cuando no se acepta el valor nulo, no se puede dejar en blanco, el dominio no contiene al nulo.
- **Clave:** Atributo o conjunto de atributos con función definida. Ej: nombre, para ordenar alfabéticamente.
- **Superclave:** Atributo que es capaz de identificar aún individuo. Siempre hay una, aunque no sea mínima, ya que las tuplas son únicas. No se mantiene para todos los tipos de relaciones. Lo seguirá siendo aunque añadamos mas atributos, pero no será mínima. Es mínima cuando al eliminar cualquier atributo deja de ser superclave. NO puede ser una atributo opcional.
- **Clave ajena:** Atributo o conjunto de atributos de una relación referenciante utilizados para apuntar o vincular cada fila con una fila de otra relación, la relación referenciada. El atributo referenciante, apunta a una superclave, se indica con una flecha desde la referenciante(puede ser opcional) a la referenciada(obligatoria). Si es múltiple se puede desdoblar para no equivocarse. Si la referenciante apunta a un valor este debe existir para mantener la integridad.
- **Tipo de relaciones entre esquemas:**
  - **1 a 1:** Correspondencia biunívoca. La clave ajena puede localizarse en cualquiera de los dos. Fuerte—>Débil. La más débil será la más volátil o por existencia. Se pondría fusionar ambos esquemas en uno sólo.
  - **1 a n:** Correspondencia múltiple(1 del primero se relaciona con varias del segundo), los hijos son los que contienen la clave del padre. Los hijos apuntan al padre. La clave ajena esta en la relación con múltiples tuplas (llave)
  - **m a n:** Surge de un problema de diseño, se resuelve añadiendo una nueva relación, una relación intermedia. Y los atributos de la relación intermedia que han llevado a su creación son superclaves de la relación.  
dl\_club\_persona(..., ..., ...)
- **Clave candidata:** Superclave mínima, si eliminamos una atributo cambia el orden de los elementos, se elige la más corta o que más aparece como primaria.
- **Restricciones Semánticas:**
  - **de Rechazo:** Rechaza las operaciones sobre los datos que rompen la restricción semántica o no contempladas.
    - Simple: Solo afecta a un elemento relacional, dominio, relación o tupla.
    - Aserción: Afectan a varias tablas, filas, ... aunque no esté operando en ella. En la practica no es viable, es teórico, en la practica se usa disparadores.
  - **Clave primaria:** Clave candidata privilegiada. Se indica subrayando con una línea continua. NO puede ser opcional.
  - **Clave alternativa:** El resto de claves candidatas. Se indica subrayando con una línea discontinua. NO puede ser opcional. Si son varios atributos con una llave.
  - **Obligatoriedad (NOT NULL)**

- **Reglas de Integridad Referencial:** Corrección automática para evitar que se pierda la integridad referencial.  $On Delete \Rightarrow D \_\_$   $On Update \Rightarrow U \_\_$

$R_1$   
↓ D-U  
 $R_2$

◦ **Restrict(R):** No se lleva a cabo si tiene hijos, aborta la operación y el usuario decide que hacer.

◦ **No Action(NA):** Se borra y después comprueba si tiene hijos, si los tiene se restaura el padre, pero si no los tiene se deja borrado.

◦ **Cascade(C):** Si borro o modifico el padre, le ocurrirá lo mismo a los hijos.

◦ **Set Null(SN):** Si es opcional el atributo, cuando se borra el padre los hijos toman el valor nulo.

◦ **Set Default(SD):** Se pone el valor por defecto en los hijos cuando se borra el padre.

**Fases del Diseño relacional:** En las 3 primeras debe estar toda la semántica.

- Diseño, hacer el grafo.
- Implementación, hacer la tablas.
- Incorporación semántica, hacer la restricciones mediante disparadores.
- Elementos externos.

- **Documentación:** Sirve para completar el grafo con comentarios acerca de la cobertura semántica de la solución propuesta.

- Los supuestos semánticos explícitos, son los requisitos aportados por el cliente/usuario.

- **Los supuestos semánticos explícitos no contemplados:** Me lo han pedido y no ha sido posible hacerlos. Se enumeran y explican casos que no han sido posible implementarlos en el diseño propuesto, y se puede proponer una solución para ellos.

- **Los supuestos semánticos implícitos:** No me lo han especificado y lo he hecho. Cuando el diseño necesita mas información que la proporcionada por el cliente, y se implementa en el diseño y afecta a la cobertura. Hay que indicar todo lo que añado y restringe el diseño.

- **Dependencia funcional  $x \rightarrow y$ :** y es dependiente de x, si dado el valor del antecedente(x) el consecuente(y) es inequívoco. Ejem: DNI  $\rightarrow$  nombre.

Propiedades (axiomas de Armstrong):  $\forall x, y, z$  atributos de la relación R

- **Reflexiva:**  $\{x, y\} \rightarrow x$
- **Aumentativa:** si  $x \rightarrow y$ , entonces  $\{x, z\} \rightarrow \{y, z\}$
- **Transitiva:** si  $x \rightarrow y \wedge y \rightarrow z$ , entonces  $x \rightarrow z$

Reglas derivadas:

- **Unión:** si  $x \rightarrow y \wedge x \rightarrow z$ , entonces  $x \rightarrow \{y, z\}$
- **Descomposición:** si  $x \rightarrow \{y, z\}$ , entonces  $x \rightarrow y \wedge x \rightarrow z$
- **Composición:** si  $x \rightarrow z \wedge y \rightarrow w$ , entonces  $\{x, y\} \rightarrow \{z, w\}$
- **Pseudotransitividad:** si  $x \rightarrow y \wedge yz \rightarrow w$ , entonces  $\{x, z\} \rightarrow w$

Re-definición del concepto de **superclave**:

- **superclave:** x es superclave sii  $\forall y \in R$  se verifica  $x \rightarrow y$
- **clave candidata:** x es candidata sii es superclave y  $\neg \exists z \subset x$  que sea superclave.
- **atributos primos:** los que forman parte de una clave candidata.

No se puede hacer d/u en R2 si exist algun R1  
Si se d/u R2 se transt a R1  
Si se hace en R2 se pone nulo en R1

- **Forma Normal:** Sirve para validar que el diseño es correcto, evitando redundancias o defectos, cada vez son mas restrictivas. Estudiaremos las 4 principales, el resto se usan muy poco.
  - **1NF:**
    - Todos los atributos son atómicos (indivisibles).
    - Los atributos tienen un solo valor, monovaluados.
    - Tiene clave primaria, superclave mínima no nula.
    - Todas las tuplas con la misma cantidad de atributos, grado fijo, los puede haber nulos.
    - No importa el orden de las filas y columnas.
  - **2NF:** Está en 1NF y no existen dependencias parciales de una clave candidata. Un subconjunto de la clave primaria tiene una relación funcional sobre algún otro atributo no primo, que no pertenece a ninguna clave. Se resuelve creando una nueva relación .
  - **3NF:** Está en 2NF y no existen dependencias transitivas entre la clave y no primos, que no pertenece a ninguna clave. Que haya una dependencia de un atributo que no es primo con otro que tampoco lo es. Se resuelve con una nueva relación que recoja ambos atributos.
  - **BCNF:** Esta en 2FN y no existen dependencias transitivas. No restringe a que no sean primos. No todas las 3NF están en BCNF.