

1 Apuntes ferramentas CASE

BASES DE DATOS

Modelo Relacional

1.1 Sumario

- 1 INTRODUCCIÓN
- 2 MODELO RELACIONAL
 - ♦ 2.1 ESTRUCTURA
 - ♦ 2.2 RESTRICIÓN
 - ♦ 2.3 OPERACIÓN
- 3 English Vocabulary

2 INTRODUCCIÓN

- **SXBD:** (Sistema Xestor de Bases de Datos) é unha colección de datos (xeralmente chamada base de datos) **interrelacionados** e un conxunto de programas para acceder a ese datos.
 - ♦ O seu obxectivo é ofrecer un entorno cómodo, eficiente e que poida ser usado para almacenar e extraer información.
- **Problemas** que presentaban as aplicacións de xestión de datos nos anos 60:
 - ♦ **Dificultade de acceso ós datos:** só se podía acceder da forma prevista polo programa.
 - ♦ **Redundancia e inconsistencia:** os arquivos presentaban distintos formatos e a información sobre un mesmo obxecto podía estar replicada en varios lugares, o que ocasionaba problemas de actualizacións. Este problema aínda existe hoxe pero é debido ó factor humano.
 - ♦ **Illamento dos datos:** pois ó estar este en varios arquivos e ter distintos formatos resulta difícil realizar novos programas para acceder ós datos.
 - ♦ **Concurrencia:** en moitos casos é preciso o acceso a unha mesma información de forma simultánea, nestes casos ó remate das operacións a información debe ser consistente.
 - ♦ **Seguridade:** tódolos usuarios podían acceder a toda a información, dándose casos de violación da intimidade.
- **Obxectivos:**
 - ♦ **Independencia física:** é a independencia entre as estruturas de almacenamento e a estrutura que teñen os datos no mundo real.
 - ♦ **Independencia lóxica:** é a independencia entre os datos vistos polas aplicacións ? usuarios finais e a estrutura dos datos.
 - ♦ **Programas non orientados a procedementos:** programas nos que se indique ?que datos se desexan consultar? esquecéndose de describir a forma en que hai que acceder a eles.
 - ♦ **Coherencia dos datos:** os datos teñen que ser lóxicos e coherentes, véxase, unha persoa non pode ter unha idade de 356 anos.
 - ♦ **Compartición dos datos:** varias aplicación poden acceder os datos como se cada unha fose a única que o está a realizar, débense establecer mecanismos de control de concurrencia.
 - ♦ **Seguridade:** cada usuario debe acceder ós datos para os que está autorizado.
 - ♦ **Funcións de respaldo e recuperación:** establecer procedementos capaces de recuperar o sistema cando este se caia.

3 MODELO RELACIONAL

- Como todo modelo ten unha estrutura, unhas restricións e un conxunto de operacións.

3.1 ESTRUCTURA

- - ♦ Unha base de datos é un conxunto de **relacións** ou **táboas**. Cada táboa identifícase por un **nome** e unha lista de **atributos** (as columnas da táboa) que describen un conxunto de entidades do mundo real. Cada fila (tupla) dunha táboa representa unha entidade do mundo real (P.e. persoa).
 - ♦ **Dominio:** conxunto finito, homoxéneo e definido de valores que pode tomar cada atributo (Columna). $\text{dom}(A_i)$.
 - ♦ **Esquema dunha relación - intensión:** a intensión/esquema dunha relación R é o conxunto de atributos que a forman, independentemente da orde na que estea.

- ◊ Representase $R(A_1, \dots, A_n)$ ou $r[[\text{Image:}]] \text{ dom}(A_1) \times \dots \times \text{dom}(A_n)$
- ◊ Ten un aspecto **estático**, pois salvo que se modifique a definición do esquema este invariable ó longo do tempo.
- ♦ **Instancia dunha relación ? extensión:** denotado $r(R)$ é o conxunto de tuplas, tal que cada tupla é unha lista ordenada de n valores de v_1 a v_n
 - ◊ $r = \{t_1, \dots, t_k\}$ $t_j = \langle v_1, \dots, v_n \rangle$ onde $v_i \neq ?$ $i \neq n$ é un elemento de $\text{dom}(A_i)$
 - ◊ Son os datos que hai na táboa nun momento dado, o recheo, o contido da relación.
 - ◊ É dinámica, pois hai insercións, borrados e actualizacións dos datos da táboa ó longo do tempo.
- ♦ **Gráo:** número de atributos que ten o esquema.
- ♦ **Cardinalidade:** número de filas que ten a relación nun momento dado.
- ♦ Cada valor dun atributo para unha entidade dada (fila, tupla) debe ser **atómico** (un único valor) ou **nulo** (que se descoñece o seu valor).

3.2 RESTRICIÓNS

- ♦ **De dominio:** o valor de cada atributo debe ser atómico e pertencente o dominio dese atributo. $\text{dom}(\text{Idade})$: números naturais < 150
- ♦ **De verificación (CHECK):** os valores dun atributo deben verificar algunha condición: p. e. Idade para conducir autobuses 21 ? Idade ? 65.
- ♦ **De chave:** non pode haber tuplas repetidas, estas deben diferenciarse en ó menos no valor dun atributo. Existirá un subconxuntos de atributos que varíen dunha tupla a outra.
 - ◊ **Superchave, SK:** é un conxunto de atributos $SK = \{A_1, \dots, A_k\}[[\text{Image:}]] R$, sendo $R = \{A_1, \dots, A_n\} / [[\text{Image:}]] t_1, t_2$ se $t_1[SK] = t_2[SK] \Rightarrow t_1 = t_2$, isto é, $(t_1[R] = t_2[R])$
 - ◊ **Atributos chave ou superchave:** conxunto de atributos que toman unha COMBINACIÓN de valores distinta para cada entidade. P.e.:
 - NIF, Nome:
 - 11111111A, Ana
 - 11111111B, Ana
 - son dúas combinacións distintas.
 - N_Empr, NIF, Nome, Apel1:
 - 3, 11111111A, Ana, Ruíz
 - 4, 11111111B, Ana, Ruíz
 - Sou outra combinación que toma valores distintos.
 - Permite *identificar/distinguir* unha entidade de calquera outra do conxunto de entidades da entidade tipo.
 - ◊ **Chave:** é unha superchave mínima, isto é:

$$K = \{A_1, \dots, A_n\} [[\text{Image:}]] R / [[\text{Image:}]] t_1, t_2 \text{ se } t_1[K] = t_2[K] \Rightarrow t_1 = t_2, \text{ e } [[\text{Image:}]] X [[\text{Image:}]] K / [[\text{Image:}]] t_1, t_2 \text{ se } t_1[K-X] = t_2[K-X] \Rightarrow t_1 = t_2$$

- ♦ **Chaves candidatas:** son aquelas superchaves que teñen un número mínimo de atributos. Poden ser varias e cada unha delas pode estar composta por varios atributos. P.e, no caso que nos ocupa as chaves candidatas serían dúas, cada unha delas composta por un único atributo:
 - $\{N_empr\}$, con esta chave pódese identificar calquera entidade.
 - $\{NIF\}$, con esta o mesmo, pois ninguén vai ter NIFs iguais, nin números de empregado.
 - Alguén podería pensar que outra chave candidata podería estar composta por $\{\text{Nome}, \text{Apel1}, \text{Apel2}\}$, pero que pasaría no caso de que houberse dous empregados co mesmo nome e apelidos?, como se faría a distinción entre elas?.
- ◊ **Chave primaria ou principal:** é a chave seleccionada polo analista, entre tódalas candidatas, para identificar a cada unha das entidades. P.e.: N_Emp
- ◊ **Chave alternativa:** cada unha das chaves candidatas que non foron seleccionadas como primaria. P.e.: NIF.
- ◊ **Toda táboa debe ter unha chave primaria.**
- ♦ **De integridade de entidade:** todo atributo que compón a chave primaria non pode conter valores nulos (Por aquilo de que non poden estar repetidas).
- ♦ **De integridade referencial:** os valores que pode tomar unha chave foránea (allea, estranxeira) deben existir na chave primaria da táboa correspondente ou ser nulos.
 - ◊ **Chave foránea (allea, estranxeira):** conxunto de atributos dunha táboa que son chave primaria noutra táboa.
 - ◊ Os valores dunha chave foránea non poden facer referencia a algo que non existe na táboa á que se referencia.
 - ◊ **Comportamento ante a modificación/borrado** dos valores da **chave principal**, distínguense catro casos:
 - **Borrado/Modificación en cascada (CASCADE):** o borrado dunha tupla, ou a modificación dos valores da chave principal dunha tupla dunha relación pai, ocasiona o borrado (modificación) de tódalas tuplas relacionadas na relación filla.

- **Borrado/Modificación restrinxida (RESTRICTED):** se existen tuplas na relación filla relacionadas coa tupla da relación pai sobre a que se desexa realizar a operación, entón non se deixara levar a cabo.
- **Borrado/Modificación con posta a nulos (SET NULL):** o borrado dunha tupla, ou a modificación dos valores da chave principal dunha tupla dunha relación pai, fai que os atributos da chave foránea da relación filla tomen valores nulos, sempre e cando os teñan permitidos.
- **Borrado/Modificación con posta a valores por defecto (SET DEFAULT):** o borrado dunha tupla, ou a modificación dos valores da chave principal dunha tupla dunha relación pai, fai que os atributos da chave foránea da relación filla tomen os valores por defecto, sempre e cando existan tuplas na táboa pai con eses valores por defecto na chave primaria.

◊ **Tipos de Integridade Referencial.**

- **Total.-** O valor dunha chave foránea debe existir exactamente igual na chave primaria referenciada (ningún atributo pode tomar o valor nulo) ou ben ser nula (en tódolos seus atributos). Cando a chave foránea está composta por un só atributo, esta é a condición que debe cumprir.
- **Parcial.-** se a chave foránea está composta por varios atributos, permítese que parte dos atributos da chave foránea tomen valor nulo pero o valor dos demais atributos da chave foránea debe existir nos atributos correspondentes da chave primaria referenciada.
- **Débil.-** ó igual que no caso anterior, pero os atributos da chave foránea con valores distintos de nulo non teñen porque ter os mesmos valores nos atributos correspondentes da chave primaria referenciada. A integridade referencial só se comprobará cando tódolos atributos que conforman a chave foránea tomen valores distintos de nulo.

- ♦ **De desencadeadores (Triggers):** é un pequeno programa que se executa (dispara) ante operacións de actualización (inserción, borrado e modificación). Por exemplo se modificamos a cantidade en **stock** dun produto é esta baixa do **stock mínimo** entón poñer o atributo **repoñer** a true.

◊ If stock < stock_minimo then repoñer=true (executarase cada vez que se modifique o atributo stock).

- ♦ **De dependencias entre datos:** son as dependencias funcionais que se estudaran máis adiante.

3.3 OPERACIÓNS

- ♦ Baséase na alxebra relacional e verase nas transparencias creadas por Mónica García Constenla.

4 English Vocabulary

- SQL
 - ♦ DDL and DML
- Entity-Relationship Model (ERM)
- Extended Entity-Relationship Model (EER)
- Kristopher test