

Bases de Datos I

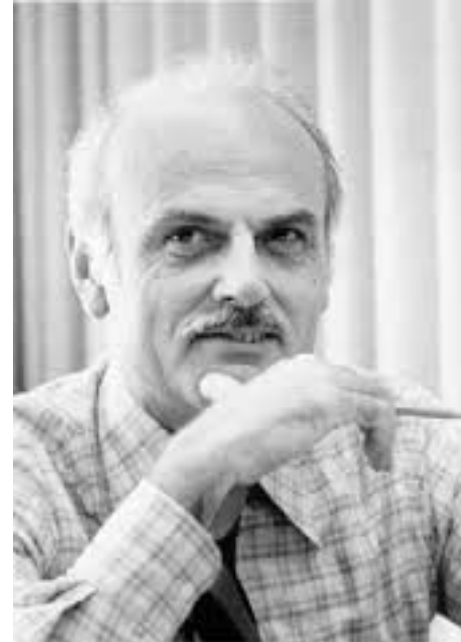
Unidad IV

Modelo Relacional



¿Cuándo comienza?

- **En 1970 Edgar Frank Codd, trabajando para el IBM Research Laboratory, publica:**
“A Relational Model of Data for Large Shared Data Banks”
- **Por su trabajo, ganó en 1981 el premio Turing ACM.**



¿Cuál es su objetivo?

- **Independencia física:**

Los cambios en el almacenamiento físico no deben modificar los programas que utilizan los usuarios para acceder a los datos.

La forma de almacenamiento no debe influir en su manipulación lógica.

¿Cuál es su objetivo?

- **Independencia lógica:**

Los cambios, eliminaciones y agregados en la descripción de los datos no debe influir en los programas que están accediendo a esos datos, siempre que los datos no hayan sido afectados por los cambios.

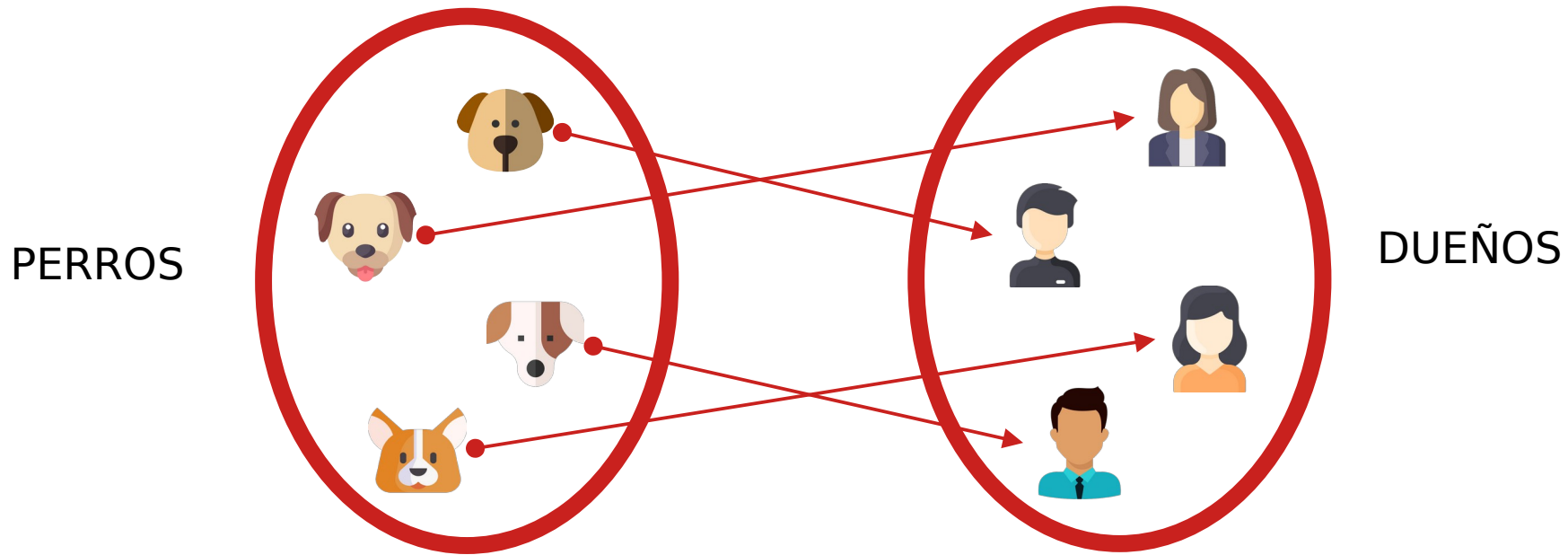
Bases y fundamentos

Este modelo fue muy difundido por su simplicidad y por estar sustentado en fundamentos matemáticos básicos.

- **Utiliza el concepto de relación matemática, y se basa en los principios de la teoría de conjuntos, la lógica de predicados y el álgebra.**
- **Explicita en términos amigables el contenido de la base de datos y permite la abstracción de los datos.**

Una Base de Datos relacional

- El modelo relacional, presenta una base de datos como una colección de relaciones matemáticas.



Seamos informales: ¿qué es una relación?

- Una relación puede pensarse como una tabla de valores o un arreglo bidimensional. Las filas representan un conjunto de valores relacionados entre sí, para modelar un hecho o entidad del mundo real. Las columnas de la tabla son la información definida sobre un determinado dominio, que se aplica a cada propiedad o atributo.

| atributo ₁ | atributo ₂ | ... | atributo _n |
|-----------------------|-----------------------|-----|-----------------------|
| valor _{1,1} | valor _{1,2} | ... | valor _{1,n} |
| ... | ... | ... | ... |
| valor _{m,1} | valor _{m,2} | ... | valor _{m,n} |

Formalizando: ¿qué es una relación?

- Una relación es un conjunto de tuplas.
- Una fila dentro del modelo relacional se denomina tupla, la cabecera de cada una de las “tablas” son los atributos y cada una de las tablas será una relación. Cada atributo tiene un tipo de dato, que se representa por el dominio.
- Cada relación esta compuesta por:
 - **Esquema** o intensión de la relación
 - **Contenido** o extensión de la relación

Esquema o intención

- **Un esquema de relación está constituido por un nombre y una lista de atributos, para describir la relación en sí.**
- **Dada una relación R, el esquema de la relación consiste en:**
 - Un **nombre** de relación **R**
 - Un **conjunto de atributos** $\{A_1, A_2, \dots, A_n\}$
- **Por ejemplo, el esquema de la relación PERSONA se escribe de la siguiente forma:**

PERSONA(dni, apellido, nombre).

Esquema o intención

- **Dado que el esquema de la relación posee un conjunto de atributos:**
 - Los **atributos** que componen el esquema de la relación, **no tiene un orden** necesario (en la práctica si).
 - El esquema **no** puede contener **atributos duplicados dentro de la misma relación** (mismo nombre).

Extensión o contenido

- **Dada una relación R con esquema $R(A_1, A_2, \dots, A_n)$, la extensión es un conjunto de tuplas.**

- **A la tupla t_i la escribimos:**

$$t_i = (v_1, v_2, \dots, v_n)$$

- **Por ejemplo, una tupla de la relación PERSONA la escribimos:**

(32111154, Ramos, Nadia Patricia).

Extensión o contenido

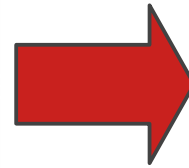
- **Dado que las relaciones son conjuntos:**
 - Las **tuplas** que componen la extensión de la misma, **no tienen orden**.
 - **No** pueden existir dos **tuplas iguales** en la extensión de la misma relación.
 - Cada componente de una tupla se referencia por el **nombre de su atributo** y no por su posición dentro de la misma.

Volvemos a la informalidad

| Cliente | | |
|------------|----------|--------|
| dni | apellido | nombre |
| 18.679.123 | Pérez | José |
| 22.498.517 | García | Roque |



Esquema o intensión



Extensión o contenido

Atributos y dominios

- Cada tupla t_i es, a su vez un conjunto de pares

$$t_i = \{(A_1:V_1), (A_2:V_2) \dots (A_n:V_n)\}$$

- Para cada par $(A_j:V_j)$, se cumple que V_j es un valor de dominio de A_j , o bien un valor especial denominado nulo.
- La extensión de una relación, sus tuplas, son un subconjunto del producto cartesiano entre los dominios de cada atributo de su esquema.

Dominio

- El dominio es el conjunto de posibles valores, que puede tener una tupla en el atributo correspondiente.
- Si en una tupla $t_i = (V_1, V_2, \dots, V_n)$, el valor V_j es un valor nulo, entonces el valor del atributo A_j es desconocido para la tupla t_i de la relación, o bien no es aplicable a esta tupla.

Dominio

- **Los dominios pueden ser de dos tipos:**
 - **predefinidos:** se corresponden a los tipos de datos que normalmente son proporcionados por los lenguajes de bases de datos, por ejemplo cadenas de caracteres o números enteros;
 - **definidos por el usuario:** el usuario define con más especificidad los valores que tomará el atributo.
- **Al definir el dominio, se están describiendo en forma directa algunas restricciones semánticas del modelo. Estas restricciones son los valores que puede tomar un objeto de la relación.**
- **Los dominios definidos por el usuario mínimamente deberán constar de un nombre de dominio y la especificación de los valores que lo conforman.**

Regla de integridad de dominio

- **Consiste en que un valor no nulo de un atributo A_i debe pertenecer al dominio del atributo A_i .**

Es decir, el valor de una tupla en el atributo A_i , debe pertenecer al dominio de A_i .

- **Especificación de los valores que puede tener cada atributo dentro de una determinada relación.**
- **$T_i[A_i]$ pertenece a $\text{dominio}(A_i)$.**

Regla de integridad de dominio

- **Los atributos pueden contener valores nulos o no permitirlos.**
- **Un valor nulo implica la no existencia de información. No refieren al valor 0 o cadenas vacías de caracteres.**
- **La existencia de atributos con valores nulos implica que su información no puede compararse con otros valores a excepción del nulo en sí mismo.**
- **El valor nulo sólo permite conocer la existencia o no de la información.**

Atributo

- **Un atributo A_i es el nombre del papel que ejerce un dominio D , en un esquema de relación.**
- **Representa una característica o propiedad de algún objeto o hecho.**
- **D es el dominio de A_i y se denota como $\text{dominio}(A_i)$**

Atributo

- **Un atributo A_i es el nombre del papel que ejerce un dominio D , en un esquema de relación.**
- **Representa una característica o propiedad de algún objeto o hecho.**
- **D es el dominio de A_i y se denota como $\text{dominio}(A_i)$**

Un ejemplo de dominio

- **Por ejemplo si tenemos la relación:**

`persona(dni, apellido, nombre, celular)`

- **El dominio del atributo dni es:**

$\text{dominio}(\text{dni}) = \{x/x \text{ es un número de documento}\}$

- **O sea:**

- El valor -1 no pertenece al dominio
- El valor 32111154 si pertenece

- **Entonces puede que una tupla contenga el valor 32111154 en el atributo dni, pero no puede existir una tupla con el valor -1**

Grado de una relación

- **El grado de una relación es el número de atributos que pertenecen a su esquema.**
- **Por ejemplo, el grado de la relación PERSONA(dni, apellido, nombre) es 3.**

Cardinalidad de una relación

- **La cardinalidad de una relación, es el número de tuplas que tiene su extensión.**
- **Por ejemplo, si tenemos en la relación PERSONA, y en su extensión a todos los habitantes de la Provincia de Tierra del Fuego, la cardinalidad de la misma serían 190.641 tuplas.**

Relaciones compatibles

- **Decimos que dos relaciones T y S son compatibles si:**
 - Tienen el mismo grado.
 - Se puede establecer una biyección (A_i, B_j) entre los atributos de la relación T y los atributos de la relación S, que hace corresponder a cada atributo A_i de T un atributo B_j de S, de modo que se cumple que $\text{dominio}(A_i) = \text{dominio}(B_j)$.



Un ejemplo de compatibilidad

- **Por ejemplo, son compatibles:**

- `persona(dni, apellido, nombre)`
- `empleado(documento, nombre, apellido)`
- Tienen el mismo grado (3) y la relación biyectiva sería $= \{(d,d),(n,n),(a,a)\}$

- **En cambio, no son compatibles:**

- `persona(documento, apellido, nombre)`
- `auto(patente, marca, modelo, fabricación)`
- No tienen el mismo grado ($3 \neq 4$)

Claves o llaves (keys)

- **Todos los datos que contiene una base de datos se debe poder identificar de alguna forma, para poder localizarlos sin lugar a ambigüedades.**
- **En el caso particular de las bases de datos que siguen el modelo relacional, para identificar los datos que la base de datos contiene, se utilizan claves.**
- **Una clave es un conjunto mínimo de atributos que determinan unívocamente a cada tupla de la relación.**

Regla de integridad de unicidad de clave

- **Establece que si el conjunto de atributos CP es clave de una relación R, entonces la extensión de R no puede tener en ningún momento dos tuplas con la misma combinación de valores para los atributos de CP.**

Más claves

- **Existen distintos tipos de claves**
 - Superclaves
 - Claves candidatas
 - Claves primarias
 - Claves secundarias o alternativas

Superclaves

Una superclave de una relación de esquema $R(A_1, A_2, \dots, A_n)$ es:

- **Un subconjunto de los atributos del esquema, tal que no puede haber dos tuplas en la extensión de la relación que tengan la misma combinación de valores para los atributos del subconjunto.**
- **Toda relación tiene, por lo menos, una superclave, que es la formada por todos los atributos de su esquema.**
- **Esto se debe a la propiedad que cumple toda relación de no tener tuplas repetidas.**

Claves candidatas

- **Una clave candidata de una relación R , es una superclave C de la relación que cumple que ningún subconjunto propio de C es superclave.**
- **Puesto que toda relación tiene por lo menos una superclave, podemos garantizar que toda relación tiene como mínimo una clave candidata.**

Claves primarias

- La clave primaria de una relación, es la clave candidata cuyos valores se utilizarán para identificar las tuplas de la relación.
- Si una relación solo tiene una clave candidata, entonces esta clave candidata debe ser también su clave primaria.
- Si una relación tiene más de una clave candidata, se elige alguna de ellas, siguiendo algún criterio (que puede ser simplicidad, claridad, eficiencia, etc) como clave primaria.
- Por convención la clave primaria se denota subrayando él o los atributos que la componen:

$R(\underline{A_1}, \underline{A_2}, \dots, A_i, \dots, A_n).$

Regla de integridad de entidad de clave primaria

- **Establece que si el conjunto de atributos CP es la clave primaria de una relación R, la extensión de R no puede tener ninguna tupla con algún valor nulo para alguno de los atributos de CP.**
- **Esto asegura que siempre puedan identificarse cada una de las tuplas dentro de la misma relación.**

Claves secundarias o alternativas

- Las claves candidatas no elegidas como primaria, se denominan claves secundarias o alternativas.
- La clave secundaria la denotaremos subrayando con traza y punto, él o los atributos que la componen:

$R(\underline{A_1}, \underline{A_2}, \dots, \underline{A_i}, \underline{A_{i+1}}, \dots, A_n)$

Ejemplo de claves

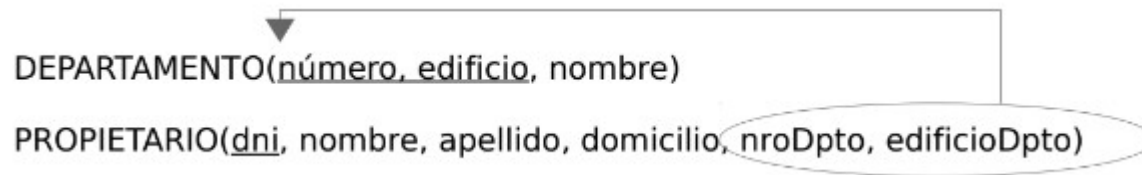
- **Por ejemplo, tenemos la relación:**
 - `persona(documento, apellido, nombre, cuil)`
- **Veamos algunas claves:**
 - `{documento, apellido, nombre}`, es superclave, porque **no** pueden haber dos tuplas con igualdad en estos atributos. ¿Porque no es candidata también?
 - `{documento}`, es **superclave** y **clave candidata**.
 - `{cuil}`, es **superclave** y **clave candidata**.
 - Además deberíamos elegir entre documento y cuil para que una sea **primaria**, por lo tanto la otra será **secundaria**.

¿Cómo vinculamos los datos?

- **En el mundo real pueden existir relaciones o vínculos entre los distintos hechos u objetos.**
- **En el modelo relacional, para reflejar este tipo de vínculos, tenemos la posibilidad de expresar conexiones entre las distintas tuplas de las relaciones.**
- **Estas vinculaciones se logran a través de las claves foráneas.**

Claves foráneas

- Para vincular los hechos o datos almacenados en la base de datos, utilizamos claves foráneas, ajenas o externas.
- Una clave foránea permite establecer una conexión entre las tuplas de dos relaciones no necesariamente distintas.
- La clave foránea es un atributo o un conjunto de atributos que conforman una clave primaria en otra relación y que indican un enlace lógico entre dos relaciones.



Claves foráneas

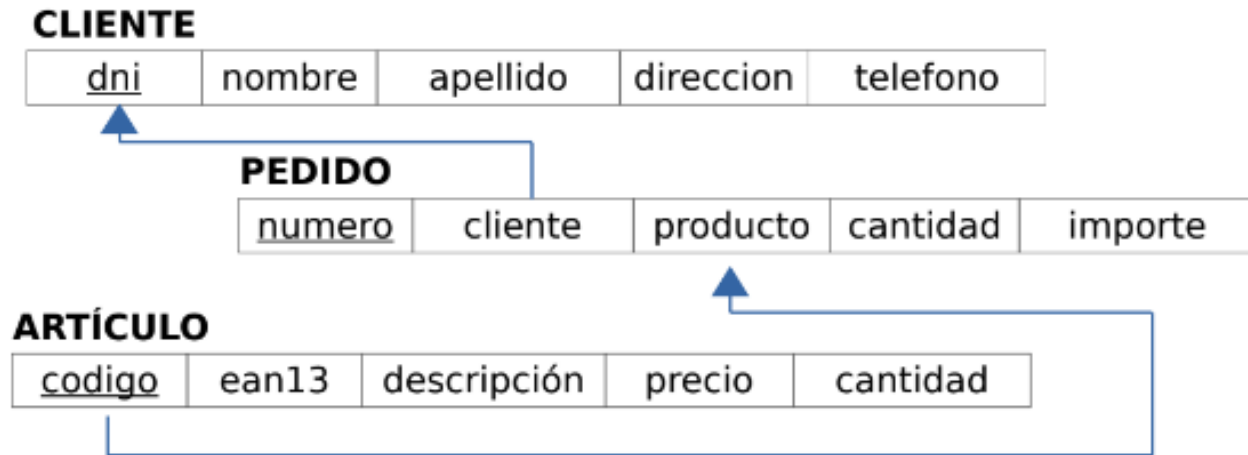
- **Una clave foránea de una relación R es un subconjunto de atributos del esquema de la relación, que denominamos CF y que cumple las siguientes condiciones:**
 - Existe una relación S (S no debe ser necesariamente diferente de R) que tiene por clave primaria CP.
 - Se cumple que, para toda tupla t de la extensión de R, los valores para CF de t son valores nulos o bien valores que coinciden con los valores para CP de alguna tupla s de S (Esto se conoce como **integridad referencial**).
 - La cantidad y los dominios de los atributos de la clave foránea deben coincidir con los dominios y la cantidad de los atributos de la clave primaria de la relación que referencian.

Claves foráneas

- **En el caso de que una tupla t de la extensión de R tenga valores para CF que coincidan con los valores para CP de una tupla s de S :**
 - decimos que t es una tupla que referencia s
 - s es una tupla que tiene una clave primaria referenciada por t .

Integridad referencial

- Esta restricción se establece entre dos relaciones, permitiendo mantener la consistencia entre las tuplas de ambas relaciones y debiendo constituir una tupla válida dentro de la relación referenciada.



Un ejemplo de clave foránea

- **Por ejemplo, tenemos dos relaciones:**

PERSONA(dni, apellido, nombre)

AUTO(patente, color, dueño)

- **La clave foránea dueño de la relación AUTO, referencia la clave primaria dni de la relación PERSONA**
- **La clave foránea dueño de la relación AUTO, referencia la relación PERSONA.**
- **La clave foránea la denotaremos subrayando con puntos.**

Un ejemplo de integridad referencial

- **Esto implica que, si existe una tupla de la relación AUTO:**
(AB397LR, Blanco, 32111154)
- **Debe existir necesariamente una tupla en la relación PERSONA que tenga 32111154 como clave primaria:**
(32111154, 'Ramos', 'Nadia Patricia')

Las vinculaciones

- **Con claves foráneas podemos establecer distintos tipos de vínculos:**
 - Vínculos uno a uno (1:1)
 - Vínculos uno a muchos (1:N)
 - Vínculos muchos a muchos (N:M)

Vínculos uno a uno (1:1)

¿Cómo hacemos para establecer un vínculo 1:1 entre relaciones?

- **Dadas dos relaciones R con claves primaria CR, y T con clave primaria CT.**
- **Agregamos una clave foránea en la cualquiera de las dos relaciones (no en ambas) que referencie la clave primaria de la otra.**
- **La clave foránea es clave secundaria.**

Ejemplo de vínculo uno a uno (1:1)

- **Entre la relación persona e historia clínica, hay un vínculo (1:1):**
 - persona(documento, apellido, nombre)
 - historia_clínica(numero, propietario, fojas)
 - **La clave primaria de la relación persona es documento.**
 - **La clave primaria de la relación historia_clínica es numero.**
 - **El atributo propietario, es clave foránea y clave candidata.**

Vínculo uno a muchos o viceversa (1:N)

¿Cómo hacemos para establecer un vínculo 1:N o N:1 entre relaciones?

- **Dadas dos relaciones R con claves primaria CR, y T con clave primaria CT.**
- **Agregamos una clave foránea en la relación del lado N, que referencie la clave primaria de la relación del lado 1.**

Ejemplo de vínculo uno a muchos o viceversa (1:N)

- **Entre la relación persona y mascota, hay un vínculo (1:N):**
 - persona(documento, apellido, nombre)
 - mascota(chip, nombre, propietario, raza)
 - **La clave primaria de la relación persona es documento.**
 - **La clave primaria de la relación mascota es chip.**
 - **El atributo propietario, es solo clave foránea.**

Vínculo muchos a muchos (N:M)

¿Cómo hacemos para establecer un vínculo N:M entre relaciones?

- **Dadas dos relaciones R con claves primaria CR, y T con clave primaria CT.**
- **Debemos generar una nueva relación S, que contenga una clave foránea que referencie la clave primaria de R y una clave foránea que referencie la clave primaria de T.**

Ejemplo de vínculo muchos a muchos (N:M)

- **Si entre la relación persona y mascota, hay un vínculo (N:M):**
 - persona(documento, apellido, nombre)
 - mascota(chip, nombre, raza)
 - dueño(persona, mascota)
 - **La clave primaria de la relación persona es documento.**
 - **La clave primaria de la relación mascota es chip.**
 - **La clave primaria de la relación dueño es persona y mascota, además de ser cada una clave foránea.**

Otra vez la integridad referencial

¿Que pasa en el caso de que una tupla referenciada quiera ser eliminada o modificada su clave primaria?

- **Distintas políticas de integridad referencial:**
 - **Restricción**
 - **Cascada**
 - **Anulación**

Restricción

- **La restricción en caso de borrado, consiste en no permitir borrar una tupla si tiene una clave primaria referenciada por alguna clave foránea.**
- **De forma similar, la restricción en caso de modificación consiste en no permitir modificar ningún atributo de la clave primaria de una tupla si tiene una clave primaria referenciada por alguna clave foránea.**

Ejemplo de restricción

- **Supongamos que tenemos el siguiente esquema:**
 - persona(documento, apellido, nombre)
 - mascota(chip, nombre, propietario, raza)
- **Si se desea eliminar una tupla de la relación persona que se encuentre referenciada (una persona que posee mascota), no se permitirá la eliminación.**

Cascada

- **La actualización en cascada en caso de borrado consiste en:**
 - permitir el borrado de una tupla t que tiene una clave primaria referenciada.
 - borrar todas las tuplas que referencian t .
- **La actualización en cascada en caso de modificación consiste en:**
 - permitir la modificación de atributos de la clave primaria de una tupla t que tiene una clave primaria referenciada.
 - modificar del mismo modo todas las claves foráneas de las tuplas que referencian t .

Ejemplo de Cascada

- **Supongamos que tenemos el siguiente esquema:**
 - persona(documento, apellido, nombre)
 - mascota(chip, nombre, propietario, raza)
- **Si se desea eliminar una tupla de la relación persona que se encuentre referenciada (una persona que posee mascota), se eliminarán además todas las mascotas.**

Anulación

- **La anulación en caso de borrado o modificación consiste en:**
 - **permitir el borrado o modificación de una tupla t que tiene una clave referenciada.**
 - **modificar todas las tuplas que referencian t , de modo que los atributos de la clave foránea correspondiente sean nulos.**

Anulación

- **Supongamos que tenemos el siguiente esquema:**
 - persona(documento, apellido, nombre)
 - mascota(chip, nombre, propietario, raza)
- **Si se desea eliminar una tupla de la relación persona que se encuentre referenciada (una persona que posee mascota), se establecerá la clave foránea de las mascotas a nulo (mascotas sin dueño).**

Bibliografía

- **Dolores Costal Costa. El modelo relacional y el álgebra relacional. UOC.**
- **Rafael Camps Paré y otros. Bases de datos. 2005. UOC.**
- **Gary W. Hansen y James V. Hansen. Diseño y Administración de Bases de Datos. Segunda Edición. Prentice Hall.**