

BASES DE DATOS TEMA 3. MODELO RELACIONAL

Contenidos generales

- * Conceptos del modelo relacional
- * Notación del modelo relacional
- * Lenguajes de consulta
 - Álgebra relacional
 - Cálculo relacional

Motivación

Esquema conceptual de una BD en modelo implementable
El modelo relacional es un modelo ampliamente extendido
Definición de un esquema para una BDR
Lenguajes formales de consulta para BDR

3.1. Orígenes del modelo relacional

Introducido por Codd en 1970

Su base es el concepto matemático de relación

Modelo sencillo y bastante extendido

Existen otros modelos más expresivos (p.e. OO)

Sin embargo, no se encuentran tan extendidos

Un modelo prometedor es el objeto-relacional

(Incorpora características de ambos modelos)

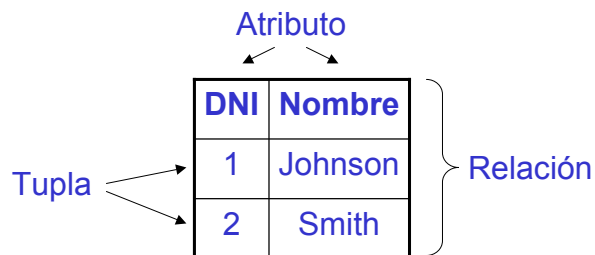
3.2. La estructura del modelo relacional

Base de datos como una colección de relaciones

Relaciones: Tablas formadas por filas y columnas

Fila: Conjunto de datos relativos a un hecho

Columna: Propiedad



3.2. La estructura del modelo relacional

Dominios, Tuplas, Atributos, Relaciones

Dominio: Conjunto de valores

Tupla: Lista de n valores ordenados con un dominio dado

Relación: Conjunto de tuplas

Atributo: Papel de un dominio en una relación

Dominio de un atributo: Valores permitidos para un atributo

nombreCli	dniCli	Domicilio
Johnson	1	La Reina nº7
Smith	2	Fragata azul nº8

Relaciones como subconjuntos del producto cartesiano de los dominios de sus atributos

Características de las relaciones: Tuplas sin orden,

Atributos ordenados, Valores en las tuplas, Interpretación

3.2. La estructura del modelo relacional

Notación del modelo relacional

Esquema de relación: $R(A_1, A_2, \dots, A_n)$

Tupla de grado n : $t = \langle v_1, v_2, \dots, v_n \rangle$

Valor de t para el atributo A_i : $t[A_i]$

Subtupla de t : $t[A_1, A_2, \dots, A_z]$

Ejemplo:

Esquema de relación: Clientes(nombreCli, dniCli, domicilio)

Tupla: $t = \langle \text{"Johnson"}, \text{"1"}, \text{"La Reina nº7"} \rangle$

Valor de tupla: $t[\text{nombreCli}]$

Subtupla: $t[\text{nombreCli}, \text{domicilio}]$

3.3. Restricciones relacionales y esquemas de BDR

- Restricciones de dominio: Valores atómicos, tipos estándar, intervalos, tipos enumerados, ...
- Restricciones de clave y de nulos
- Base de datos relacional
 - Compuestas por esquema relacional + restricciones

Esquema: Diseño lógico de la base de datos

Instancia: Datos de la BD en un instante determinado

Sucursales = (nombreSuc, ciudadSuc, activo)

Empleados = (nombreEmp, dniEmp, telefono, nombreSuc)

Cuentas = (numeroCta, saldo, nombreSuc)

Clientes = (nombreCli, dniCli, domicilio)

CtaCli = (numeroCta, dniCli)

Transacciones = (numeroCta, numeroTrans, fecha, importe)

3.3. Restricciones relacionales y esquemas de BDR

Esquema formado por varias relaciones

Una única relación contendría **información redundante**



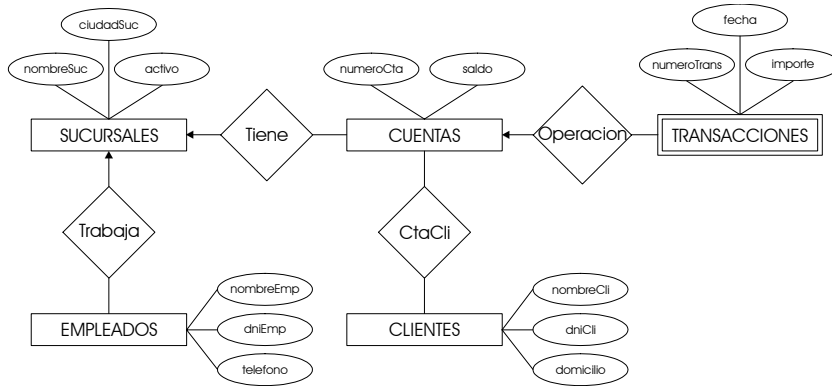
Possible source of anomalies

Todo = (nombreCli, dniCli, domicilio, numeroCta, saldo)

nombreCli	dniCli	Domicilio	NumeroCta	saldo
Johnson	1	La Reina nº7	1	10000
Johnson	1	La Reina nº7	2	20000
...				
Turner	4	Gibraltar español nº17	5	50000
Williams	5	Diamante S/N	5	50000
...				

3.3. Restricciones relacionales y esquemas de BDR

Diagrama E-R (conceptual) asociado



3.3. Restricciones relacionales y esquemas de BDR

Instancia para los ejemplos

nombreSuc	ciudadSuc	Activo
Castellana	Madrid	90000
Ganivet	Granada	21000
Paseo	Almería	17000
Zapillo	Almería	4000
Ronda	Almería	80000
Aduana	Roquetas	3000
Los Pinos	Huercal	37000
Sol	Madrid	71000

Sucursales

nombreEmp	dniEmp	telefono	NombreSuc
García	10	101010	Castellana
Torres	11	111111	Castellana
López	12	121212	Paseo
Villegas	13	131313	Paseo
Fernández	14	141414	Zapillo
Urrutia	15	151515	Sol

Empleados

numeroCta	saldo	nombreSuc
1	100	Castellana
2	200	Castellana
3	300	Paseo
4	400	Paseo
5	500	Zapillo
6	600	Sol

Cuentas

nombreCli	dniCli	Domicilio
Aranda	1	La Reina nº7
García	2	Fragata azul nº8
Hayes	3	Gibraltar español nº14
Turner	4	Gibraltar español nº17
Vilches	5	Diamante S/N
Lara	6	Gato negro nº13
Guerrero	7	Perro nº1

Clientes

numeroCta	numeroTrans	fecha	importe
1	1	10-10	+100
2	1	10-10	+300
2	2	11-10	-200
3	1	12-10	+300
4	1	12-10	+400
5	1	13-10	+500
6	1	13-10	+600

CtaCli

dniCli	numeroCta
1	1
1	2
2	3
3	4
4	5
5	5
6	5
7	6

Transacciones

3.3. Restricciones relacionales y esquemas de BDR

Integridad de entidades

Los valores de la clave primaria no pueden ser nulos

Integridad referencial

Tuplas que hagan referencia a tuplas existentes

Clave ajena o clave externa (Uso de líneas en los esquemas)

CE es clave externa en R1 si

* Los atributos de CE forman clave primaria en otra relación (p.e. R2).

(Un valor de CE en $t1 \in R1 \Rightarrow$ ese valor es CP en $t2 \in R2$ o es nulo en R1

* Dominio de CE en R1 = dominio de CP en R2

Otras restricciones (semánticas, de transición, ...)

3.4. Operaciones de actualización. Tratam. de restricciones

Operaciones en el modelo relacional

- Recuperación
- Actualización (Inserción, Eliminación, Modificación)

Inserción

- Inserta lista de valores de tupla o mediante una consulta que contiene los valores a insertar
- Rechazo si se violan las restricciones (dominio, clave, IR...

3.4. Operaciones de actualización. Tratam. de restricciones

Operaciones en el modelo relacional (y 2)

Eliminación

- Elimina tuplas de una relación
- Si se viola IR
 - Rechazar
 - Eliminar en cascada
 - Modificar si es posible los valores, sin violar IR

Actualizar

- Modifica valores de atributos de 1 o varias tuplas de acuerdo con una condición
- Mantener restricciones de dominio
- Especial cuidado al actualizar claves primarias y ajenas

3.5. Lenguajes de consulta

Lenguaje de consulta

Lenguaje para solicitar información a la BD

Tipos

- Procedimentales
- No procedimentales

Lenguajes de consulta formales para BDR

- Álgebra relacional (Procedimental)
- Cálculo relacional (No procedimental)

3.6. Algebra relacional (1)

Lenguaje de consulta procedimental

Operaciones de manipulación (consulta, inserción, elimin.)

Opera sobre relaciones y devuelve una relación

Operaciones fundamentales

- Selección (Selecciona tuplas)
- Proyección (Selecciona columnas)
- Producto cartesiano (Combina relaciones)
- Renombrar (Cambia el nombre a una relación)
- Unión (Realiza la unión de relaciones)
- Intersección (Realiza la intersección de relaciones)

3.6 Algebra relacional (2)

3.6.1. Operaciones fundamentales (1)

Selección

Selecciona las tuplas que satisfacen un predicado

Denotada por σ

Sintaxis: $\sigma_{\text{predicado}}(\text{relación})$

Ejemplo: Sucursales de la ciudad de Madrid

nombreSuc	ciudadSuc	Activo
Castellana	Madrid	90000
Ganivet	Granada	21000
Paseo	Almería	17000
Zapillo	Almería	4000
Ronda	Almería	80000
Aduana	Roquetas	3000
Los Pinos	Huerca	37000
Sol	Madrid	71000

sucursales

nombreSuc	ciudadSuc	Activo
Castellana	Madrid	90000
Sol	Madrid	71000

$\sigma_{\text{ciudadSuc}=\text{"Madrid"}}(\text{sucursales})$

3.6. Algebra relacional (3)

3.6.1. Operaciones fundamentales (2)

Selección (2)

Ej. Sucursales de Madrid con activo mayor de 800000

nombreSuc	ciudadSuc	Activo
Castellana	Madrid	90000
Ganivet	Granada	21000
Paseo	Almería	17000
Zapillo	Almería	4000
Ronda	Almería	80000
Aduana	Roquetas	3000
Los Pinos	Huerca	37000
Sol	Madrid	71000

nombreSuc	ciudadSuc	Activo
Castellana	Madrid	90000

$\sigma_{\text{ciudadSuc}=\text{"Madrid"} \wedge \text{activo}>800000}(\text{sucursales})$

sucursales

3.6. Algebra relacional (4)

3.6.1. Operaciones fundamentales (3)

Proyección

Devuelve una relación con ciertas columnas omitidas

No elimina duplicados

Denotada por Π

Sintaxis: $\Pi_{\text{columnas}}(\text{relación})$

Ejemplo: Nombre y teléfono de los empleados

nombreEmp	dniEmp	telefono	NombreSuc
García	10	101010	Castellana
Torres	11	111111	Castellana
López	12	121212	Paseo
Villegas	13	131313	Paseo
Fernández	14	141414	Zapillo
Urrutia	15	151515	Sol

nombreEmp	telefono
García	101010
Torres	111111
López	121212
Villegas	131313
Fernández	141414
Urrutia	151515

empleados

$\Pi_{\text{nombreEmp,telefono}}(\text{empleados})$

3.6. Algebra relacional (5)

3.6.1. Operaciones fundamentales (4)

Secuencias de operaciones

Aplicar operaciones sobre resultados de otras operaciones
Se debe a que el resultado de una operación es una relación
Util para selección-proyección

Ejemplo: Nombre y teléfono de empleados que trabajan en la sucursal Castellana

nombreEmp	dniEmp	telefono	NombreSuc
García	10	101010	Castellana
Torres	11	111111	Castellana
López	12	121212	Paseo
Villegas	13	131313	Paseo
Fernández	14	141414	Zapillo
Urrutia	15	151515	Sol

nombreEmp	telefono
García	101010
Torres	111111

empleados

$\Pi_{\text{nombreEmp,telefono}} (\sigma_{\text{nombreSuc}=\text{"Madrid"}}(\text{empleados}))$

3.6. Algebra relacional (6)

3.6.1. Operaciones fundamentales (5)

La operación unión (1)

Una nueva relación con todas y cada una de las tuplas de las relaciones que participan en la unión

$R \cup S$ incluye las tuplas que están en R o en S

Las tuplas repetidas son eliminadas

Las relaciones a unir han de ser compatibles



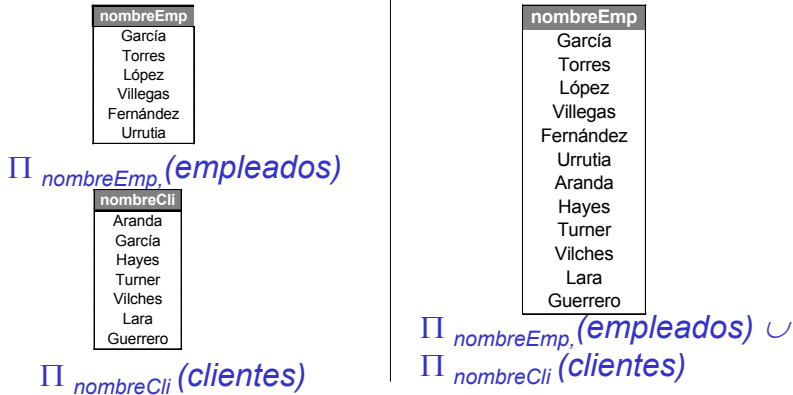
Igual número de atributos
Dominios iguales dos a dos

3.6. Algebra relacional (7)

3.6.1. Operaciones fundamentales (6)

La operación unión (2)

Ejemplo: Nombres que aparecen el banco



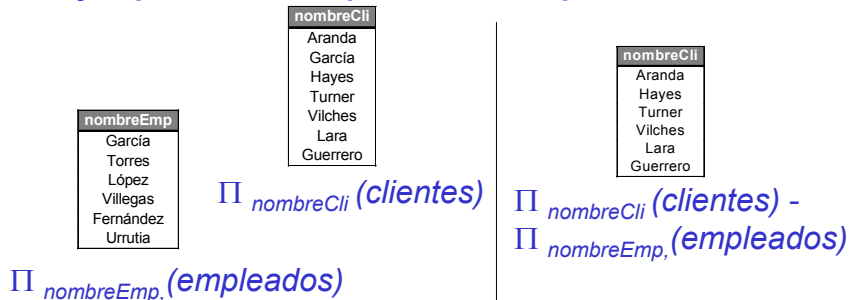
3.6. Algebra relacional (8)

3.6.1. Operaciones fundamentales (7)

La operación diferencia de conjuntos

Incluye tuplas que están en una relación pero no en la otra
 $R - S$ incluye las tuplas que están en R y no están en S

Ejemplo: Clientes que no son empleados



3.8. Algebra relacional (9)

3.6.1. Operaciones fundamentales (8)

La operación producto cartesiano (1)

Operación para combinar relaciones

$R \times S$ contiene todas las tuplas de R combinadas con todas las tuplas de S

Número de columnas de $R \times S = \text{Cols. de } R + \text{Cols. de } S$

Número de tuplas de $R \times S = \text{Tuplas de } R * \text{Tuplas de } S$

Para evitar ambigüedades: Usar sintaxis *tabla.columna*

3.6. Algebra relacional (10)

3.6.1. Operaciones fundamentales (9)

La operación producto cartesiano (2)

Empleados. nombreEmp	Empleados. dniEmp	Empleados. telefono	Empleados. nombreSuc	Sucursales.n ombreSuc	Sucursales. ciudadSuc	Sucursales. activo
García	10	101010	Castellana	Castellana	Madrid	90000
Torres	11	111111	Castellana	Castellana	Madrid	90000
López	12	121212	Paseo	Castellana	Madrid	90000
Villegas	13	131313	Paseo	Castellana	Madrid	90000
Fernández	14	141414	Zapillo	Castellana	Madrid	90000
Urrutia	15	151515	Sol	Castellana	Madrid	90000
García	10	101010	Castellana	Ganivet	Granada	21000
Torres	11	111111	Castellana	Zapillo	Almería	4000
López	12	121212	Paseo	Zapillo	Almería	4000
Villegas	13	131313	Paseo	Zapillo	Almería	4000
Fernández	14	141414	Zapillo	Zapillo	Almería	4000
Villegas	13	131313	Paseo	Sol	Madrid	71000
Fernández	14	141414	Zapillo	Sol	Madrid	71000
Urrutia	15	151515	Sol	Sol	Madrid	71000

3.6. Algebra relacional (11)

3.6.1. Operaciones fundamentales (10)

La operación producto cartesiano (3)

Selección de las tuplas relacionadas

$\sigma_{\text{empleados.nombreSuc} = \text{sucursales.nombreSuc}}$
(sucursales x empleados)

Empleados. nombreEmp	Empleados. dniEmp	Empleados. telefono	Empleados. nombreSuc	Sucursales. nombreSuc	Sucursales. ciudadSuc	Sucursales. activo
García	10	101010	Castellana	Castellana	Madrid	90000
Torres	11	111111	Castellana	Castellana	Madrid	90000
López	12	121212	Paseo	Castellana	Madrid	90000
Villegas	13	131313	Paseo	Castellana	Madrid	90000
Fernández	14	141414	Zapillo	Castellana	Madrid	90000
Urrutia	15	151515	Sol	Castellana	Madrid	90000
García	10	101010	Castellana	Ganivet	Granada	21000
Torres	11	111111	Castellana	Zapillo	Almería	4000
López	12	121212	Paseo	Zapillo	Almería	4000
Villegas	13	131313	Paseo	Zapillo	Almería	4000
Fernández	14	141414	Zapillo	Zapillo	Almería	4000
Villegas	13	131313	Paseo	Sol	Madrid	71000
Fernández	14	141414	Zapillo	Sol	Madrid	71000
Urrutia	15	151515	Sol	Sol	Madrid	71000

3.6. Algebra relacional (12)

3.6.1. Operaciones fundamentales (11)

La operación producto cartesiano (4)

Ejemplo: Nombres de empleados que trabajan en la ciudad de Madrid

nombreSuc	ciudadSuc	Activo
Castellana	Madrid	90000
Ganivet	Granada	21000
Paseo	Almería	17000
Zapillo	Almería	4000
Ronda	Almería	80000
Aduana	Roquetas	3000
Los Pinos	Huercal	37000
Sol	Madrid	71000

sucursales

nombreEmp	dniEmp	telefono	NombreSuc
García	10	101010	Castellana
Torres	11	111111	Castellana
López	12	121212	Paseo
Villegas	13	131313	Paseo
Fernández	14	141414	Zapillo
Urrutia	15	151515	Sol

empleados

Empleados.nombreEmp
García
Torres
Urrutia

$\Pi_{\text{empleados.nombreEmp}} (\sigma_{\text{sucursales.nombreSuc} = \text{empleados.nombreSuc}} (\sigma_{\text{sucursales.ciudadSuc} = \text{"Madrid"}} (\text{sucursales x empleados})))$

3.6. Algebra relacional (13)

3.6.1. Operaciones básicas (12)

La operación renombrar (1)

Cambia el nombre de una relación

Util para evitar ambigüedades cuando una relación aparece más de una vez en una consulta

$\rho_X(R)$ renombra la relación R como X

Así, podemos usar R o X para hacer ref. a la misma relación

3.6. Algebra relacional (14)

3.6.1. Operaciones fundamentales (13)

La operación renombrar (2)

Ejemplo: Empleados que trabajan en la misma sucursal que García

$\Pi_{empleados.nombreEmp}(\sigma_{empleados2.nombreSuc = empleados.nombreSuc}(empleados \times$

$\rho_{empleados2}(\Pi_{nombreSuc}(\sigma_{nombreEmp = "García"}(empleados))))))$

nombreEmp	dniEmp	telefono	NombreSuc
García	10	101010	Castellana
Torres	11	111111	Castellana
López	12	121212	Paseo
Villegas	13	131313	Paseo
Fernández	14	141414	Zapillo
Urrutia	15	151515	Sol

empleados

NombreSuc
Castellana

empleados2

nombreEmp
García
Torres

resultado

3.6. Algebra relacional (15)

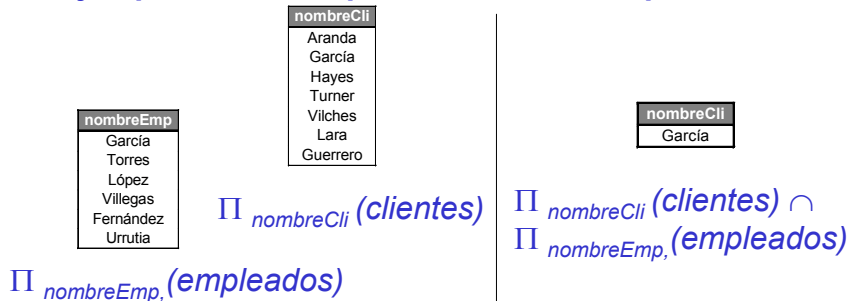
3.6.2. Otras operaciones (1)

La operación intersección de conjuntos

Incluye tuplas que están en todas las relaciones

$R \cap S$ incluye las tuplas que están en R y en S

Ejemplo: Clientes que también son empleados



3.6. Algebra relacional (16)

3.6.2. Otras operaciones (2)

La operación producto theta

Producto que permite especificar cualquier operación de combinación de relaciones

La condición es un subíndice de X ($=, <, <=, >, >=, <>$)

Ejemplo: Transacciones con importe superior a saldos

$(\Pi_{\text{numeroTrans}}(\text{cuentas} \times_{\text{cuentas.saldo} < \text{transacciones.importe}} \text{transacciones}))$

3.6. Algebra relacional (17)

3.6.2. Otras operaciones (3)

La operación producto natural (1)

Simplifica consultas que combinan varias relaciones

Es una reunión theta con el operador de igualdad para combinar los atributos comunes de las relaciones

No especifica explícitamente el predicado de combinación

Se representa como $|X|$ y la selección se hace en base a los campos comunes

Ejemplo: Nombres de clientes que tienen cuenta en Castellana

Como producto natural

$\Pi_{\text{nombreCli}} (\sigma_{\text{cuentas.nombreSuc} = \text{"Castellana"}} (\text{clientes} \times \text{ctacli} \times \text{cuentas}))$

3.6. Algebra relacional (18)

3.6.2. Otras operaciones (3)

La operación producto natural (2)

Como producto cartesiano

$\Pi_{\text{nombreCli}} (\sigma_{\text{cuentas.nombreSuc} = \text{"Castellana"} \wedge \text{cuentas.numeroCta} = \text{ctacli.numeroCta} \wedge \text{ctacli.dniCli} = \text{clientes.dniCli}} (\text{clientes} \times \text{ctacli} \times \text{cuentas}))$

nombreCli	dniCli	Domicilio
Aranda	1	La Reina nº7
García	2	Fragata azul nº8
Hayes	3	Gibraltar español nº14
Turner	4	Gibraltar español nº17
Vilches	5	Diamante S/N
Lara	6	Gato negro nº13
Guerrero	7	Perro nº1

Clientes

dniCli	numeroCta
1	1
1	2
2	3
3	4
4	5
5	5
6	5
7	6

CtaCli

numeroCta	saldo	nombreSuc
1	100	Castellana
2	200	Castellana
3	300	Paseo
4	400	Paseo
5	500	Zapillo
6	600	Sol

Cuentas

nombreCli
Aranda

Resultado

3.6. Algebra relacional (19)

3.6.2. Otras operaciones (4)

División

Util en algunas consultas que deben realizar comparaciones con “todas” las tuplas que cumplen una condición

Ejemplo: Clientes que tengan cuenta en todas las sucursales de Madrid

1. Clientes con alguna cuenta en Madrid

$\Pi_{dniCLI, ctaCli.numrCta}(\sigma_{ciudadSuc = \text{“Madrid”}}(sucursales \times clientes \times ctacli))$

2. Todas las sucursales de Madrid

$\Pi_{nombreSuc}(\sigma_{ciudadSuc = \text{“Madrid”}}(sucursales))$

3. El resultado es $1 \div 2$. En este caso no hay nadie.

3.6. Algebra relacional (20)

3.6.2. Otras operaciones (5)

La operación asignación

Crea una nueva relación a partir de otra

Se denota por el símbolo \leftarrow

Lo que se asigna puede ser una relación existente o el resultado de una operación

3.6. Algebra relacional (21)

3.6.2. Otras operaciones (6)

Extensiones del producto natural (1)

En $R \bowtie_x S$ sólo hay tuplas que tienen valores coincidentes en R y S

Existen operaciones binarias que devuelven todas las tuplas de una relación aunque no tengan valores coincidentes en la otra relación (**Uniones externas**)

- * **Unión externa izquierda (derecha)**
- * **Unión externa**

3.6. Algebra relacional (22)

3.6.2. Otras operaciones (7)

Extensiones del producto natural (2)

Unión externa izquierda. Left Join (1)

Operación que relaja el producto natural

En $R \bowtie_x S$ están todas las tuplas de R . Si en S hay tuplas con valores coincidentes, dichas tuplas también formarán parte de $R \bowtie_x S$. En caso contrario, serán completadas con valores nulos

3.6. Algebra relacional (23)

3.6.2. Otras operaciones (8)

Extensiones del producto natural (3)

Unión externa izquierda Left Join (2)

Ejemplo: Nombre de las sucursales de Almería junto con los datos de los empleados. Los datos de las sucursales son necesarios aunque no haya empleados

$\Pi_{sucursales.nombreSuc, empleados.nombreEmp}(\sigma_{ciudadSuc="Almería"}(sucursales) \times empleados)$

Sucursales.nombreSuc	Empleados.nombreEmp
Paseo	López
Paseo	Villegas
Zapillo	Fernández
Ronda	nulo

3.6. Algebra relacional (24)

3.6.2. Otras operaciones (9)

Extensiones del producto natural (4)

Unión externa derecha

Igual que Unión externa izquierda, salvo para la relación de la derecha

Denotada como $R \bowtie_r S$

3.6. Algebra relacional (25)

3.6.2. Otras operaciones (10)

Extensiones del producto natural (4)

Unión externa

Realiza la unión de dos relaciones no compatibles

Los atributos que no proceden se completan con valores nulos

Ejemplo: Nombre, DNI, Teléfono y Domicilio de todas las personas del sistema bancario

nombreEmp	dniEmp	telefono	domicilio
García	10	101010	nulo
Torres	11	111111	nulo
Urrutia	15	151515	nulo
Guerrero	7	777777	Perro nº1
Vilches	5	nulo	Diamante S/N
Lara	6	nulo	Gato negro nº13

3.7. Cálculo relacional de tuplas (1)

Lenguaje de consulta formal no procedimental

Permite construir expresiones equivalentes al A.R.

3.7.1. Variables de tupla y relaciones de rango

Expresiones en CR: Especifican variables de tupla

Cada variable se suele asociar a una relación (tabla)

tomando como valor cualquier tupla de la relación

Consulta: $\{t / P(t)\}$

Devuelve las tuplas que satisfacen un predicado

$t.A$ hace referencia al atributo A de la tupla t

Ejemplo: Sucursales de la ciudad de Madrid

$\{s / s \in \text{sucursal} \wedge s.\text{ciudadSuc} = \text{"Madrid"}\}$

Ejemplo: Nombres de sucursal de la ciudad de Madrid

$\{s.\text{nombreSuc} / s \in \text{sucursal} \wedge s.\text{ciudadSuc} = \text{"Madrid"}\}$

3.7. Cálculo relacional de tuplas (2)

3.7.2. Expresiones y fórmulas del CRT

Expresión general CRT: $\{t_1.A_1, \dots, t_n.A_n / P(t_1, \dots, t_m)\}$

t_1, \dots, t_m : Variables de tupla

A_1, \dots, A_n : Atributos de las relaciones t_i

P : Predicado del CRT

Fórmulas CRT constan de átomos

- $t_i \in R$, con t_i variable de tupla y R relación
- $t_i.A \text{ op } t_j.B$, donde op es $\{=, <, <=, >, >=, <>\}$
- $t_i.A \text{ op } c$, donde op es $\{=, <, <=, >, >=, <>\}$

Los átomos pueden conectarse con and, or y not

3.7. Cálculo relacional de tuplas (3)

3.7.3. Uso de cuantificadores

Universal (\forall): $(\forall t)(F)$ es verdadera si se cumple para todas las tuplas t

Existencial (\exists): $(\exists t)(F)$ es verdadera si hay al menos una tupla t para la que es cierta

Uso en combinación de relaciones

Implica trabajar con tuplas de relaciones diferentes

Ejemplo: Nombre de empleados que trabajan en Madrid

$\{e.nombreEmp / e \in empleados \wedge e.ciudadSuc="Madrid" \wedge \exists s \in sucursales (s.nombreSuc. = e.nombreSuc)\}$

3.7. Cálculo relacional de tuplas (4)

Operaciones de conjuntos

Unión

Disyunción de predicados

$\{p.nombre / p \in clientes \vee p \in empleados\}$

Intersección

Conjunción de predicados

$\{p.nombre / p \in clientes \wedge p \in empleados\}$

Diferencia

Conjunción de predicados con negación

$\{c.nombreCli / c \in clientes \wedge$

$\neg \exists e \in empleados (e.nombreEmp = c.nombreCli)\}$