75.15 / 75.28 / 95.05 - Base de Datos

Álgebra y Cálculos Relacionales

Mariano Beiró

Dpto. de Computación - Facultad de Ingeniería (UBA)

Temas

Introducción

Lenguajes de manipulación de datos (DML)

- Para interactuar con un modelo es necesario utilizar un lenguaje.
- Los lenguajes que permiten extraer información de un modelo de datos se denominan lenguajes de manipulación de datos, o DML *(Data Manipulation Languages)*.

Procedurales

Lenguajes

Declarativos

- Los lenguajes procedurales indican un procedimiento a seguir, utilizando operaciones que indican cómo manipular las datos.
- Los lenguajes declarativos indican qué resultado se quiere obtener, sin especificar cómo hacerlo.
- Los lenguajes procedurales se consideran de más bajo nivel.

Introducción

Lenguajes del modelo relacional

- En el modelo relacional existen distintos DML's, algunos de ellos formales y algunos prácticos.
- El lenguaje práctico más conocido es SQL *(Structured Query Language)*, y es declarativo.
- Los lenguajes formales del modelo relacional son:
  - El álgebra relacional (procedural).
  - El cálculo relacional (declarativo).

Álgebra Relacional

Características

- Es un lenguaje procedural.
- Propuesto por E. Codd en 1970, se lo considera parte integral del modelo relacional.
- Su utilidad radica en que:
  - Provee un marco formal de operaciones para el modelo relacional.
  - Se emplea como base para optimizar la ejecución de consultas.
- El álgebra relacional especifica los procedimientos de consulta de datos a partir de un conjunto de operaciones.
- Una operación -en el contexto del modelo relacional- es una función cuyos operandos son una o más relaciones, y cuyo resultado es también una relación.

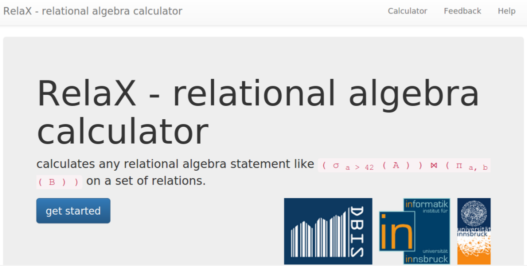
$$O : R_1 \times R_2 \times ... \times R_n \to S$$

- La aridad es la cantidad de operandos que toma una operación.
- Las operaciones del álgebra relacional pueden combinarse entre ellas para formar una expresión.

Álgebra Relacional

Recursos utilizados

- 2022 World Cup Dataset
- RelaX - Relational algebra calculator
  `http://dbis-uibk.github.io/relax/`

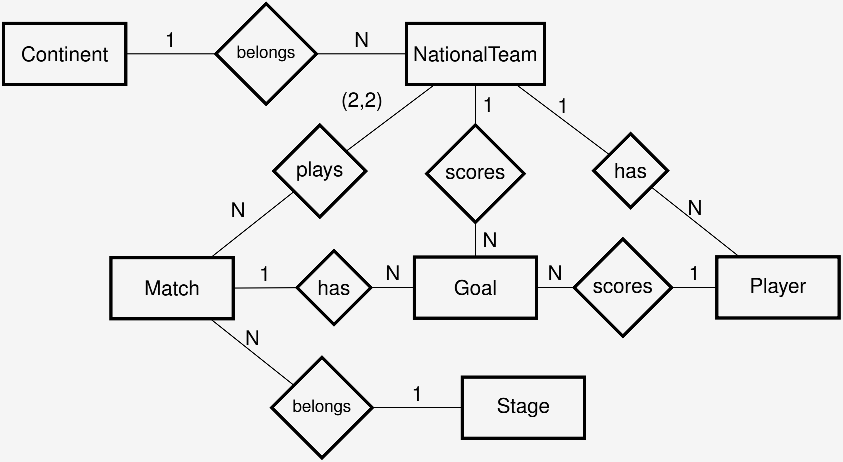


Link con dataset precargado: `https://dbis-uibk.github.io/relax/calc/gist/9e403afbfdf8c707a3a649c2425723f8`.

Álgebra Relacional

Recursos utilizados: 2022 World Cup Dataset

- Modelo ER simplificado (sin atributos):



Álgebra Relacional

Recursos utilizados: 2022 World Cup Dataset

- *Esquema de base de datos relacional*:
  - Continents(id, name)
    (3, ‘Europe’)
  - NationalTeams(short_name, name, group, continent)
    (‘ESP’, ‘Spain’, ‘E’, 3)
  - Matches(id, home, away, datetime, stage)
    (23, ‘ARG’, ‘MEX’, ’2022-11-26 16:00:00’, 1)
  - Players(id, name, birth_year, playing_position, local_club, national_team)
    (184, ‘Emiliano Martínez’, 1992, ‘GK’, ‘Aston Villa’, ‘ARG’)
  - Scores(id, match_id, team_id, player_id, minute, score_type)
    (1, 1, 1, 8, 55, 1) (Corresponde a la entidad Goal).
  - Stages(id, name)
    (3, ‘Quarter-final’)
  - Asumiremos que “name” es siempre clave candidata.
  - Los tipos de score son: 1-normal; 2-penal; 3-gol en contra (se asigna al equipo contrario); 4-gol en serie de penales.

Selección

- El operador de selección (σ) es un operador unario.

$$\sigma_{cond} : R \to S$$

- Dada una relación $R(A_1, A_2, ..., A_n)$ y una condición que se aplica a cada tupla de R, $\sigma_{cond}(R)$ selecciona aquellas tuplas de R para las cuales la condición es verdadera.

NationalTeams

short_name	name	group	continent
ARG	Argentina	C	5
AUS	Australia	D	2
BEL	Belgium	F	3
BRA	Brazil	G	5
CMR	Cameroon	G	1

$$\downarrow \sigma_{group='G'}(\text{NationalTeams})$$

short_name	name	group	continent
BRA	Brazil	G	5
CMR	Cameroon	G	1

Selección

Condiciones

- Utilizaremos *c*ondiciones atómicas de la forma:
  - $A_i \odot A_j$
  - $A_i \odot c$, con $c \in dom(A_i)$
- En donde $\odot$ debe ser un operador de comparación:
  - $=, \neq$
  - $>, \geq, <, \leq$ (sólo para atributos cuyos dominios están ordenados)
- Una condición se construye combinando condiciones atómicas con los operadores lógicos `and` ($\wedge$), `or` ($\vee$) y `not` ($\neg$).

Ejemplo: World Cup 2022

Seleccionar aquellos jugadores del mundial que pertenecen al club local “Barcelona” y que nacieron antes del 2000.

Respuesta

$\sigma_{(local\_club = \text{``}Barcelona\text{''}) \wedge (birth\_year < 2000)}(\text{Players})$

Proyección

- El operador de proyección (π) es también un operador unario.

$$\pi_L : R \rightarrow S$$

- Dada una relación $R(A_1, A_2, ..., A_n)$ y una lista de atributos $L = (L_1, L_2, ..., L_k)$, con $L_i \in (A_1, A_2, ..., A_n)$, $\pi_L(R)$ devuelve una relación cuyas tuplas representan los posibles valores de los atributos de L en R.
- Podemos pensar que lo que hace es *proyectar* cada tupla de R a un espacio de menor dimensión en que sólo se conservan los atributos que están en L.

ArgPlayers

id	name	local_club
551	Rodrigo De Paul	Atlético Madrid
615	Thiago Almada	Atlanta Utd
674	Ángel Correa	Atlético Madrid
675	Ángel Di María	Juventus

$\xrightarrow{\pi_{local\_club}(\text{ArgPlayers})}$

local_club
Atlético Madrid
Atlanda Utd
Juventus

Proyección

- El *orden* de los atributos en la relación resultado es el mismo orden en que figuran en *L*.
- El operador de proyección siempre *remueve tuplas duplicadas*, ya que su resultado debe ser también una relación válida.

Ejemplo: World Cup 2022

Liste las posiciones de juego de los jugadores.

Respuesta

$\pi_{playing\_position}(Players)$

Secuencias de operaciones. Asignación ($\leftarrow$)

- ¿Cómo listamos los nombres de los países del grupo G?

NationalTeams

short_name	name	group	continent
AUS	Australia	D	2
BEL	Belgium	F	3
BRA	Brazil	G	5
CMR	Cameroon	G	1
...	...	...	...

$$Temp \leftarrow \sigma_{group='G'}(\text{NationalTeams})$$
$$Selecciones\_GrupoG \leftarrow \pi_{name}(\text{Temp})$$

- Podemos también hacerlo en un único paso:

$$Selecciones\_GrupoG \leftarrow \pi_{name}(\sigma_{group='G'}(\text{NationalTeams}))$$

Selecciones_GrupoG

name
Brazil
Cameroon
...

Redenominación

- El operador de redenominación (ρ) permite modificar los nombres de los atributos de una relación y/o el de la relación misma.
- Nos permite preparar el resultado para la realización de una operación posterior.
- Dada una relación $R(A_1, A_2, ..., A_n)$, un nuevo nombre de relación S y una lista de n nombres de atributo $(B_1, B_2, ..., B_n)$, $\rho_{S(B_1,B_2,...,B_n)}(R)$ produce una relación de nombre S y atributos $(B_1, B_2, ..., B_n)$ cuyas tuplas coinciden con las tuplas de R.
- $\rho_S(R)$ sólo cambia el nombre de la relación R por S.

ArgPlayers

name	local_club
Rodrigo De Paul	Atlético Madrid
Thiago Almada	Atlanta Utd
Ángel Correa	Atlético Madrid
Ángel Di María	Juventus

$$\xrightarrow{\rho_{Argentinos(nombre,club\_local)}(\text{ArgPlayers})}$$

Argentinos

nombre	club_local
Rodrigo De Paul	Atlético Madrid
Thiago Almada	Atlanta Utd
Ángel Correa	Atlético Madrid
Ángel Di María	Juventus

Operaciones de conjuntos

Unión

- Dadas dos relaciones $R(A_1, A_2, ..., A_n)$ y $S(B_1, B_2, ..., B_n)$, la unión $R \cup S$ es una relación que contiene a todas las tuplas de R y de S.
- Es necesario que R y S tengan el mismo grado.
- Además, para calcular $R \cup S$ las relaciones R y S deben coincidir en sus atributos en lo que respecta al dominio. Es decir, $dom(A_i) = dom(B_i)$. Esta condición se denomina compatibilidad de unión o compatibilidad de tipo.
- Por convención, en la relación resultado el listado de atributos coincide con el de R: $(A_1, A_2, ..., A_n)$.

Normal_Scorers

id	name
364	Kim Young-gwon
372	Kudus Mohammed
377	Kylian Mbappé
388	Lionel Messi

Penalty_Scorers

id	name
269	Ismaila Sarr
377	Kylian Mbappé
388	Lionel Messi

$\xrightarrow{Normal\_Scorers \cup Penalty\_Scorers}$

id	name
269	Ismaila Sarr
364	Kim Young-gwon
372	Kudus Mohammed
377	Kylian Mbappé
388	Lionel Messi

Operaciones de conjuntos

Intersección

- Dadas dos relaciones $R(A_1, A_2, ..., A_n)$ y $S(B_1, B_2, ..., B_n)$, la intersección $R \cap S$ conserva las tuplas que se encuentran presentes tanto en R como en S.
- R y S deben tener el mismo grado.
- Al igual que la unión, la intersección requiere compatibilidad de tipo.
- El listado de atributos de la relación resultado será $(A_1, A_2, ..., A_n)$.

Normal_Scorers

id	name
364	Kim Young-gwon
372	Kudus Mohammed
377	Kylian Mbappé
388	Lionel Messi

Penalty_Scorers

id	name
269	Ismaila Sarr
377	Kylian Mbappé
388	Lionel Messi

$\xrightarrow{Normal\_Scorers \cap Penalty\_Scorers}$

id	name
377	Kylian Mbappé
388	Lionel Messi

Operaciones de conjuntos

Diferencia

- Dadas dos relaciones $R(A_1, A_2, ..., A_n)$ y $S(B_1, B_2, ..., B_n)$, la diferencia $R - S$ conserva sólo aquellas tuplas de R que no pertenecen a S.
- R y S deben tener el mismo grado.
- También requiere compatibilidad de tipo.
- El listado de atributos de la relación resultado será $(A_1, A_2, ..., A_n)$.

Normal_Scorers

id	name
364	Kim Young-gwon
372	Kudus Mohammed
377	Kylian Mbappé
388	Lionel Messi

Penalty_Scorers

id	name
269	Ismaila Sarr
377	Kylian Mbappé
388	Lionel Messi

$$\xrightarrow{Normal\_Scorers - Penalty\_Scorers}$$

id	name
364	Kim Young-gwon
372	Kudus Mohammed

- Las tres operaciones de conjuntos que definimos son operaciones binarias.

Producto cartesiano

- Dadas dos relaciones $R(A_1, A_2, ..., A_n)$ y $S(B_1, B_2, ..., B_m)$, el producto cartesiano $R \times S$ produce una nueva relación T cuyas tuplas son *todas* aquellas de la forma $(t_1, t_2, ..., t_n, t_{n+1}, t_{n+2}, ..., t_{n+m})$, con $(t_1, t_2, ..., t_n) \in R$ y $(t_{n+1}, t_{n+2}, ..., t_{n+m}) \in S$.
- El esquema de la relación resultante T es $(A_1, A_2, ..., A_n, B_1, B_2, ..., B_m)$. Salvo...
- ... si algún atributo A_i tiene el mismo nombre que un atributo B_j. En ese caso, la convención será que en el resultado los atributos se llamarán "$R.A_i$" y "$S.B_j$". En el caso de estar calculando $R \times R$, llamaremos a los atributos $R1.A_i$ y $R2.A_i$.
- Aunque generalmente debe ser acompañada de alguna selección para reducir las combinaciones del resultado.
- El producto cartesiano no requiere compatibilidad de tipos.

Producto cartesiano

NationalTeams

short_name	name	group
ARG	Argentina	C
AUS	Australia	D
BEL	Belgium	F
BRA	Brazil	G
...	...	...

Players

id	name	national_team
227	Gonzalo Montiel	ARG
353	Kevin De Bruyne	BEL
...	...	...

$\downarrow$ NationalTeams $\times$ Players

short_name	NationalTeams.name	group	id	Players.name	national_team
ARG	Argentina	C	227	Gonzalo Montiel	ARG
ARG	Argentina	C	353	Kevin De Bruyne	BEL
AUS	Australia	D	227	Gonzalo Montiel	ARG
AUS	Australia	D	353	Kevin De Bruyne	BEL
BEL	Belgium	F	227	Gonzalo Montiel	ARG
BEL	Belgium	F	353	Kevin De Bruyne	BEL
BRA	Brazil	G	227	Gonzalo Montiel	ARG
BRA	Brazil	G	353	Kevin De Bruyne	BEL
...	...	...	...	...	...

Producto cartesiano

- NationalTeams(short_name, name, group)
- Players(id, name, national_team)

- Cómo hacemos para obtener las tuplas que representan la pertenencia de un jugador a una selección nacional?

NationalTeams

short_name	name	group
ARG	Argentina	C
AUS	Australia	D
BEL	Belgium	F
BRA	Brazil	G
...	...	...

Players

id	name	national_team
227	Gonzalo Montiel	ARG
353	Kevin De Bruyne	BEL
...	...	...

$$\downarrow \sigma_{short\_name=national\_team}(\text{NationalTeams} \times \text{Players})$$

short_name	NationalTeams.name	group	id	Players.name	national_team
ARG	Argentina	C	227	Gonzalo Montiel	ARG
BEL	Belgium	F	353	Kevin De Bruyne	BEL
...	...	...	...	...	...

Producto cartesiano

Ejemplo: World Cup 2022

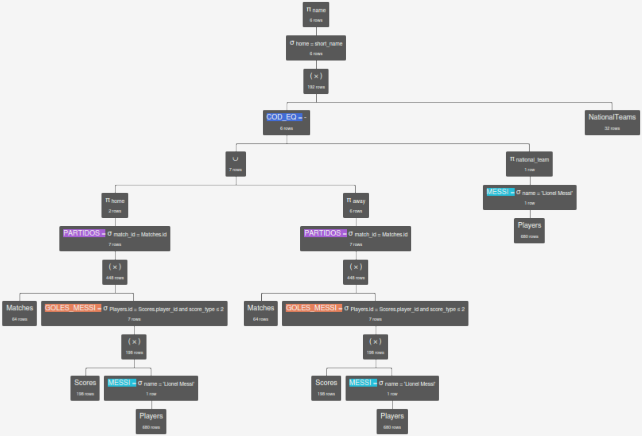
Liste los nombres de países a los que Lionel Messi les convirtió un gol en tiempo reglamentario.

Respuesta

$MESSI \leftarrow \sigma_{name='LionelMessi'}(Players)$
$GOLES\_MESSI \leftarrow \sigma_{Players.id=Scores.player\_id \wedge score\_type<=2}(Scores \times MESSI)$
$PARTIDOS \leftarrow (\sigma_{match\_id=Matches.id}(Matches \times GOLES\_MESSI))$
$COD\_EQ \leftarrow \pi_{home}(PARTIDOS) \cup \pi_{away}(PARTIDOS) - \pi_{national\_team}(MESSI)$
$\pi_{name}(\sigma_{home=short\_name}(COD\_EQ \times NationalTeams))$

Árboles de consulta

- Para cada expresión del álgebra relacional se puede construir un árbol de consulta que representa el orden de ejecución.
- Para el ejemplo anterior sobre el producto cartesiano:



Junta

- La operación de junta combina un producto cartesiano con una selección. Dadas dos relaciones $R(A_1, A_2, ..., A_n)$ y $S(B_1, B_2, ..., B_m)$ y una condición, la junta $R \bowtie_{cond} S$ selecciona del producto cartesiano $R \times S$ las tuplas que cumplen la condición.
- No se admite cualquier tipo de condición de selección, sino sólo la conjunción de operaciones atómicas que incluyen columnas de ambas relaciones, es decir, de la forma:
  - $A_i \odot B_j$

  En donde $\odot$ debe ser un operador de comparación:
  - $=, \neq$
  - $>, \geq, <, \leq$ (sólo para atributos cuyos dominios están ordenados)

  Una condición se construye entonces combinando operaciones atómicas con el operador lógico `and` ($\wedge$).

Junta

- Ahora la combinación de NationalTeams y Players se hace mucho más sencilla:

NationalTeams

short_name	name	group
ARG	Argentina	C
AUS	Australia	D
BEL	Belgium	F
BRA	Brazil	G
...	...	...

Players

id	name	national_team
227	Gonzalo Montiel	ARG
353	Kevin De Bruyne	BEL
...	...	...

$$\downarrow \text{(NationalTeams} \bowtie_{short\_name=national\_team} \text{Players)}$$

short_name	NationalTeams.name	group	id	Players.name	national_team
ARG	Argentina	C	227	Gonzalo Montiel	ARG
BEL	Belgium	F	353	Kevin De Bruyne	BEL
...	...	...	...	...	...

Junta

Ejemplo: World Cup 2022

Obtenga el listado de los nombres de los jugadores de la Selección Argentina.

Respuesta

$PLAYER\_TEAM \leftarrow NationalTeams \bowtie_{short\_name=national\_team} Players$

$\pi_{Player.name}(\sigma_{NationalTeams.name=\text{``}Argentina\text{''}}(PLAYER\_TEAM))$

Junta

Tipos particulares de junta

- El caso más general de operación de junta también se denomina junta theta (*theta join*).
- Cuando la junta sólo utiliza comparaciones de igualdad en sus condiciones atómicas, se denomina junta por igual (*equijoin*).
- En la junta por igual, el resultado dispondrá de pares de atributos distintos que poseerán información redundante. Para librarse de uno de ellos, se define la junta natural.

Junta

Junta Natural

- Para realizar una junta natural entre dos relaciones en reemplazo de una junta por igual, las mismas deben estar preparadas de manera que los pares de atributos (A_i, B_j) de cada condición atómica tengan el mismo nombre en una y otra relación. El resultado dispondrá de uno sólo de los atributos, conservando su nombre.
- La junta natural entre dos relaciones R y S se simboliza $R * S$.
- ¡Atención! En la junta natural no se especifican las condiciones, por lo tanto todo par de atributos de igual nombre en una y otra relación será comparado por igual en la condición de selección implícita.
- Los atributos comparados en una junta se denominan atributos de junta.

Junta

Junta Natural: Ejemplo

Ejemplo: RENAPER

Personas(DNI, nombre, género, fecha_nacimiento)
HijoDe(DNI_padre, DNI_hijo)
CasadaCon(DNI1, DNI2, fecha_matrimonio)

Liste a todos los hijos de "Abraham Simpson" (suponga que no hay dos personas con ese nombre).

Respuesta

$PADRE \leftarrow \rho_{DNI\_padre}(\pi_{DNI}(\sigma_{nombre=\text{“Abraham Simpson”}}(Personas)))$
$HIJOS \leftarrow \rho_{DNI\_hijo,nombre}(\pi_{DNI,nombre}(Personas))$
$\pi_{DNI\_hijo,nombre}$ (PADRE * HijoDe * HIJOS)

División

- Esta vez, primero el ejemplo...
- Nos interesa saber qué alumnos aprobaron los 3 TPs.

NOTAS

alumno	TP	nota
Pedro	1	7
Pedro	3	2
Juan	1	3
Juan	2	6
Juan	3	8
Walter	1	4
Walter	2	9
Walter	3	8

$\rightarrow$

APROBADOS

alumno	TP
Pedro	1
Juan	2
Juan	3
Walter	1
Walter	2
Walter	3

REQUISITOS

TP
1
2
3

$\downarrow$ (Aprobados $\div$ Requisitos)

alumno
Walter

División

- Es una operación inversa al producto cartesiano.
- Partimos de una relación $R(A_1, A_2, ..., A_n)$ y una relación $S(B_1, B_2, ..., B_m)$ cuyos atributos están incluídos en los de R.
- Llamaremos $A = \{A_1, A_2, ..., A_n\}$ y $B = \{B_1, B_2, ..., B_m\}$. Entonces $B \subset A$.
- Llamaremos $Y = A - B$.
- Se define entonces la división $R \div S$ como la relación $T(Y)$ que contiene todas las tuplas t que cumplen que:
  1. t pertenece a $\pi_Y(R)$.
  2. Para cada tupla $t_S \in S$ existe una tupla $t_R \in R$ tal que $t_R[Y] = t$ y $t_R[B] = t_S$.
- Propiedad: T es la relación de mayor cardinalidad posible contenida en $\pi_Y(R)$ y que cumple que $T * S \subset R$.

División

Ejemplo: Tenistas

Tenistas(nombre_tenista, país, altura, diestro)
('Novak Djokovic', 'Serbia', 1.88, True)
Torneos(nombre_torneo, tipo_torneo)
('Abierto de Australia', 'Grand Slam')
Campeones(nombre_tenista, nombre_torneo, modalidad, año)
('Juan Martín del Potro', 'Torneo de Estocolmo', 'Single', 2016)

Liste a aquellos tenistas que hayan ganado todos los torneos de tipo "Grand Slam" existentes al menos una vez.

Respuesta

$TORNEOS\_GRAND\_SLAM \leftarrow \pi_{nombre\_torneo}(\sigma_{tipo\_torneo=\text{``Grand Slam''}}(Torneos))$
$\pi_{nombre\_tenista,nombre\_torneo}(Campeones) \div TORNEOS\_GRAND\_SLAM$

Conjuntos completos de operadores

- Hemos definido una serie de operadores básicos del álgebra relacional: $\sigma, \pi, \rho, \cup, \cap, -, \times, \bowtie, *, \div$.
- Sin embargo, existen subconjuntos de ellos que tienen la misma capacidad de expresión que todo el conjunto.
- A dichos subconjuntos se los denomina conjuntos completos de operadores.
- $\{\sigma, \pi, \rho, \cup, -, \times\}$ forman un conjunto completo de operadores.

¿Cómo se demuestra?

Mostrando que cada uno de los operadores restantes puede construirse a partir de estos seis.

Operaciones adicionales

- Existen operaciones frecuentes de bases de datos que no pueden ser expresadas en el álgebra relacional básica.
- Se han propuesto numerosos operadores para extender el álgebra relacional, entre ellos:
  - La proyección generalizada.
  - La agregación.
  - La junta externa.
- Sólo presentaremos aquí la *junta externa*.

Junta externa

- Supongamos que quisieramos juntar la tabla de Scores con la de Matches, pero sin que desaparezca ningún partido.

Matches

id	home	away	datetime
5	DEN	TUN	2022-11-22 10:00:00
6	MEX	POL	2022-11-22 13:00:00
7	FRA	AUS	2022-11-22 16:00:00

Scores

id	match_id	team_id	player_id	minute	score_type
17	7	FRA	16	27	1
18	7	FRA	502	32	1
19	7	FRA	377	68	1
20	7	FRA	502	71	1
21	7	AUS	132	9	1

$$\downarrow (\text{Matches} \;⟕_{Matches.id=match\_id}\; \text{Scores})$$

Matches.id	home	away	datetime	Scores.id	match_id	team_id	player_id	minute	score_type
5	DEN	TUN	2022-11-22...						
6	MEX	POL	2022-11-22...						
7	FRA	AUS	2022-11-22...	17	7	FRA	16	27	1
7	FRA	AUS	2022-11-22...	18	7	FRA	502	32	1
7	FRA	AUS	2022-11-22...	19	7	FRA	377	68	1
7	FRA	AUS	2022-11-22...	20	7	FRA	502	71	1
7	FRA	AUS	2022-11-22...	21	7	AUS	132	9	1

- El resultado muestra cada gol junto con el partido en que ocurrió, mostrando también los partidos en que no hubo goles.

Junta externa

- La junta externa evita que eso suceda, asegurando que las tuplas de una o ambas relaciones estén presentes en el resultado, aún cuando no puedan combinarse con ninguna tupla de la otra.
- Existen 3 tipos de junta externa:
  - Junta externa izquierda ($R \ltimes\!\!\!\bowtie S$)
  - Junta externa derecha ($R \bowtie\!\!\!\rtimes S$)
  - Junta externa completa ($R ⟗ S$)
- Dadas dos relaciones $R(A_1, A_2, ..., A_n)$ y $S(B_1, B_2, ..., B_m)$ y una condición, la junta externa $R[⟕, ⟖, ⟗]_{cond} S$ selecciona del producto $R \times S$ las tuplas que cumplen la condición, y añade...
  - ...una tupla $(t[A_1, t[A_2], ..., t[A_n], NULL, NULL, ..., NULL)$ de dimensión $n + m$ por cada tupla de $t \in R$ que no se encuentra en la proyección sobre $(A_1, A_2, ..., A_n)$ (Junta externa izquierda, ⟕).
  - ...una tupla $(NULL, NULL, ..., NULL, t[B_1, t[B_2], ..., t[B_m])$ de dimensión $n + m$ por cada tupla de $t \in S$ que no se encuentra en la proyección sobre $(B_1, B_2, ..., B_m)$ (Junta externa derecha, ⟖).
  - ...ambos tipos de tuplas descriptos (Junta externa completa, ⟗).

Ejercicios: World Cup 2022

Ejercicio 1

Liste el nombre de los continentes que no fueron representados por ningún equipo en los cuartos de final del Mundial.

Ejercicio 2

Liste el nombre de los jugadores que marcaron al menos 3 goles durante el Mundial.

Ejercicio 3

Liste el nombre y selección nacional de el/los jugadores más jóvenes del Mundial.

Cálculo Relacional

Características

- Es un lenguaje declarativo, de más alto nivel que el álgebra relacional.
- Al no ser procedural, no especifica un orden de operaciones a realizar.
- Está basado en la lógica de predicados.
- Presenta dos variantes:
  - El cálculo relacional de tuplas.
  - El cálculo relacional de dominios.
- El lenguaje SQL está inspirado en el cálculo relacional de tuplas.

Cálculo Relacional

Proposiciones, predicados y relaciones

- Proposiciones:
  - Rafael Nadal ganó el torneo de Roland Garros en 2009.
  - Gabriela Sabatini ganó el Abierto de Estados Unidos en 1990.
  - Andy Murray ganó el Campeonato de Wimbledon en 2016.
- Un conjunto de proposiciones que tienen la misma estructura puede tipificarse a través de un predicado.
  - [tenista] ganó [torneo] en [año]
- Un predicado es una función cuyo resultado es un valor de verdad: Verdadero (V) ó Falso (F)
  - TenistaCampeón(nombre_tenista, nombre_torneo, año)
  - Entonces:
    - TenistaCampeón(Rafael Nadal, Roland Garros, 2011) = V
    - TenistaCampeón(Juan Martín del Potro, Roland Garros, 2011) = F
- En el cálculo relacional, los esquemas de relación pueden pensarse como predicados.
  - Las bases de datos sólo almacenan proposiciones verdaderas.
  - Ergo, cada tupla t de una relación R predica que $R(t) = V$.

Cálculo Relacional

Lógica de predicados de primer orden

- La lógica de predicados de primer orden se basa en:
  - Predicados: Son funciones de una o más variables cuyo resultado es un valor de verdad (V ó F).
    - Ejemplo: $p(m, n)$.
  - Operaciones entre predicados.
    - $\wedge, \vee, \neg, \rightarrow$
    - Combinando los predicados con operaciones se obtienen predicados más complejos
    - Ejemplo: $(p(m, n) \wedge \neg q(m)) \vee q(n)$.
  - Cuantificadores de variables.
    - Cuantificador universal: $(\forall m)q(m)$. Es verdadero si para cualquier valor de m el predicado $q(m)$ es verdadero.
    - Cuantificador existencial: $(\exists m)q(m)$. Es verdadero si existe al menos un valor de m para el cual el predicado $q(m)$ es verdadero.

Cálculo Relacional de Tuplas

Predicados y operaciones

- En el cálculo relacional de tuplas las variables representan tuplas.
- Un predicado simple es una función de una tupla o de atributos de tuplas, cuyo resultado es un valor de verdad (V ó F). Se admiten como predicados simples:
  - $R(t)$, en donde R es una relación
  - $t_1.A_i \odot t_2.A_j$
  - $t.A_i \odot c$, con $c \in dom(A_i)$
  - En donde $\odot$ debe ser un operador de comparación:
    - $=, \neq$
    - $>, \geq, <, \leq$ (sólo para atributos cuyos dominios están ordenados)
- Las operaciones entre predicados admitidas son $\wedge, \vee, \neg$.

Una expresión del cálculo relacional de tuplas tiene la forma:
$\{t_1.A_{11}, t_1.A_{12}, ..., t_1.A_{1k_1}, ..., t_n.A_{nk_n} | p(t_1, t_2, ..., t_n, t_{n+1}, ..., t_{n+m})\}$,
en donde p es un predicado válido. $\{t_1, t_2, ..., t_n\}$ deben ser *variables libres*, y $\{t_{n+1}, t_{n+2}, ..., t_{n+m}\}$ deben ser *variables ligadas*.

Cálculo Relacional de Tuplas

Ejemplos

Ejemplo: World Cup 2022

Liste los nombres de los países que jugaron el Mundial 2022.

Respuesta

$\{n.name | NationalTeams(n)\}$

Ejemplo: World Cup 2022

Liste los nombres de los jugadores nacidos antes de 1980.

Respuesta

$\{p.name | Players(p) \wedge p.birth\_year < 1980\}$

Cálculo Relacional de Tuplas

Cuantificadores

- Pero, ¿cómo hacemos si queremos listar a los jugadores que hicieron *algún* gol durante el mundial?
- Necesitamos de los cuantificadores.
  - Cuantificador universal: $(\forall t)p(t)$. Es verdadero si para cualquier tupla t el predicado $p(t)$ es verdadero.
  - Cuantificador existencial: $(\exists t)p(t)$. Es verdadero si existe al menos una tupla t para la cual el predicado $p(t)$ es verdadero.
- El listado de los nombres de los jugadores que hicieron goles se obtiene como:

Respuesta

$$\{p.name | Players(p) \wedge (\exists s)(Scores(s) \wedge s.player\_id = p.id)\}$$

Cálculo Relacional de Tuplas

Cuantificadores

- Atención! Una variable que fue cuantificada no puede aparecer seleccionada en el lado izquierdo de la barra (|), y toda variable que aparece sólo en el lado derecho debe estar cuantificada.
  - Las variables que fueron cuantificadas son variables ligadas.
  - Las variables que no fueron cuantificadas son variables libres.
- Reiteramos:

Una expresión del cálculo relacional de tuplas tiene la forma: $\{t_1.A_{11}, t_1.A_{12}, ..., t_1.A_{1k_1}, ..., t_n.A_{nk_n} | p(t_1, t_2, ..., t_n, t_{n+1}, ..., t_{n+m})\}$, en donde p es un predicado válido. $\{t_1, t_2, ..., t_n\}$ deben ser *variables libres*, y $\{t_{n+1}, t_{n+2}, ..., t_{n+m}\}$ deben ser *variables ligadas*.

Cálculo Relacional de Tuplas

Cuantificadores

Ejemplo: World Cup 2022

Liste los nombres de los jugadores de la Selección Española.

Respuesta

$$\{p.name | Players(p) \wedge (\exists n)(NationalTeams(n) \wedge$$

$$n.short\_name = p.national\_team \wedge n.name = \text{``}Spain\text{''})\}$$

Cálculo Relacional de Tuplas

Cuantificadores

Ejemplo: World Cup 2022

Liste el nombre del jugador más anciano del Mundial.

Respuesta

$$\{p.name | Players(p) \wedge (\forall \theta)(\neg Players(\theta) \vee \theta.birth\_year \geq p.birth\_year)\}$$

- Observemos que el cuantificador $\forall \theta$ necesita típicamente de una negación dentro de su expresión, para restringir el universo de θ's sobre los que requerimos que la expresión sea verdadera. De lo contrario, el resultado estará vacío.

Cálculo Relacional de Tuplas

Expresiones seguras

- No toda expresión válida del cálculo de tuplas es una expresión segura *(safe expression)*.
- Por ejemplo, la expresión...

$$\{p.name | \neg Players(p)\}$$

- ... no es una expresión segura. Producirá una cantidad infinita de tuplas con valores como “safsq” o 57.
- Una expresión segura es aquella que garantiza formalmente que producirá una cantidad finita de tuplas.
- Puede probarse que ésto es equivalente a garantizar que los valores de los atributos del resultado son parte del *dominio de la expresión*.

Cálculo Relacional de Tuplas

Expresiones seguras

- Ejemplos:
  - $\{p_1.nombre | (\exists p_2)(Personas(p_2) \wedge p_2.edad = p_1.edad)\}$
    - ✗ Expresión no segura
    - Probablemente queríamos $\{p_1.nombre | Personas(p_1) \wedge (\exists p_2)(Personas(p_2) \wedge p_2.edad = p_1.edad)\}$
  - $\{p_1.nombre | Empleados(p_1) \wedge (\nexists p_2)(Empleados(p_2) \wedge p_2.sueldo > p_1.sueldo)\}$
    - ✓ Expresión segura
  - $\{t.nombre | \neg((Clientes(t) \wedge Proveedores(t))\}$
    - ✗ Expresión no segura
    - Probablemente queríamos $\{t.nombre | (Clientes(t) \vee Proveedores(t)) \wedge \neg(Clientes(t) \wedge Proveedores(t))\}$
- Recomendación: Cuidado cuando usamos cuantificadores ó negamos predicados!

Cálculo Relacional de Tuplas

Ejercicio: Tenistas

Tenistas(nombre_tenista, país, altura, diestro)
('Novak Djokovic', 'Serbia', 1.88, True)
Torneos(nombre_torneo, tipo_torneo)
('Abierto de Australia', 'Grand Slam')
Campeones(nombre_tenista, nombre_torneo, modalidad, año)
('Juan Martín del Potro', 'Torneo de Estocolmo', 'Single', 2016)

Liste los nombres de los tenistas que ganaron todos los torneos de Grand Slam.

Respuesta

$$\{c.nombre\_tenista \mid Campeones(c) \wedge (\forall t)(\neg Torneos(t) \vee t.tipo\_torneo \neq \text{"Grand Slam"} \vee (\exists c_2)(Campeones(c_2) \wedge c_2.nombre\_tenista = c.nombre\_tenista \wedge c_2.nombre\_torneo = t.nombre\_torneo))\}$$

Cálculo Relacional de Dominios

Predicados, operaciones y cuantificadores

- En el cálculo relacional de dominios las variables representan dominios, es decir que hacen referencia a los atributos.
- Un predicado simple es una función de un conjunto de dominios, cuyo resultado es un valor de verdad (V ó F). Se admiten como predicados simples:
  - $R(x_1, x_2, ..., x_n)$, en donde $R(A_1, A_2, ..., A_n)$ es una relación
  - $x_i \odot x_j$
  - $x_i \odot c$, con $c \in dom(A_i)$
  - En donde $\odot$ debe ser un operador de comparación:
    - $=, \neq$
    - $>, \geq, <, \leq$ (sólo para atributos cuyos dominios están ordenados)
- Las operaciones entre predicados admitidas son $\wedge, \vee, \neg$.
- Se utilizan los cuantificadores con las mismas reglas que en el CRT.

Cálculo Relacional de Dominios

Predicados y operaciones

Una expresión del cálculo relacional de dominios tiene la forma:
$\{x_1, x_2, ..., x_n | p(x_1, x_2, ..., x_n, x_{n+1}, ..., x_{n+m})\}$,
en donde p es un predicado válido. $\{x_1, x_2, ..., x_n\}$ deben ser *variables libres*, y $\{x_{n+1}, x_{n+2}, ..., x_{n+m}\}$ deben ser *variables ligadas*.

Cálculo Relacional de Dominios

Ejemplos

Ejemplo: World Cup 2022

Liste los nombres de los países que jugaron el Mundial 2022.

Respuesta

$\{n | (\exists s)(\exists g)(\exists c)(NationalTeams(s, n, g, c))\}$

Ejemplo: World Cup 2022

Liste los nombres de los jugadores nacidos antes de 1980.

Respuesta

$\{n | (\exists i)(\exists b)(\exists p)(\exists l)(\exists t)(Player(i, n, b, p, l, t) \wedge b < 1980)\}$

Cálculo Relacional de Dominios

Ejemplos

Ejemplo: RENAPER

Personas(DNI, nombre, género, fecha_nacimiento)
HijoDe(DNI_padre, DNI_hijo)
CasadaCon(DNI1, DNI2, fecha_matrimonio)

Liste a todos los hijos de "Abraham Simpson" (suponga que no hay dos personas con ese nombre).

Respuesta

$$\{\ h, n_1 | (\exists d_1)(\exists g_1)(\exists f_1)(\exists d_2)(\exists n_2)(\exists g_2)(\exists f_2)(\exists p)$$
$$(\text{Personas}(d_1, n_1, g_1, f_1) \wedge \text{Personas}(d_2, n_2, g_2, f_2)$$
$$\wedge \text{HijoDe}(p, h) \wedge n_2 = \text{``Abraham Simpson''} \wedge h = d_1 \wedge p = d2)\}$$

Completitud Relacional

- E. Codd demostró la equivalencia entre el álgebra relacional básica y el cálculo relacional[1].
- Esta equivalencia implica que ambos lenguajes tienen el mismo poder expresivo.
  - Toda consulta expresable a través del cálculo relacional es también expresable en el álgebra relacional básica y viceversa.

> **¿Cómo se demuestra esta equivalencia?**
>
> Mostrando que cada uno de los operadores del álgebra relacional básica es expresable a través del cálculo relacional, y que una expresión genérica segura del cálculo relacional es expresable utilizando los operadores del álgebra relacional básica.

- A su vez, se dice que un lenguaje es relacionalmente completo cuando tiene la misma capacidad expresiva que el cálculo relacional.
  - El álgebra relacional básica es relacionalmente completa.

[1] Restringido a expresiones seguras.

Bibliografía

[ELM16] Fundamentals of Database Systems, 7th Edition.
R. Elmasri, S. Navathe, 2016.
Capítulo 8

[SILB19] Database System Concepts, 7th Edition.
A. Silberschatz, H. Korth, S. Sudarshan, 2019.
Capítulo 2.5 y 2.6, Capítulo 27 (online)

[CONN15] Database Systems, a Practical Approach to Design, Implementation and Management, 6th Edition.
T. Connolly, C. Begg, 2015.
Capítulo 5