

Cálculo Relacional de Tuplas

Bibliografía recomendada: Capítulo 4, Libro Ramakrishnan

Repaso

- Al igual que el Álgebra Relacional, el *Cálculo Relacional de Tuplas* (CRT) es un lenguaje de consulta asociado al Modelo Relacional (MR).
- CRT es un lenguaje declarativo: describe cuáles tuplas se deben devolver pero no cómo se calculan.
- Una consulta CRT tiene la forma $\{t / F(t)\}$, donde t es una tupla (variable libre) y F es la fórmula que describe a la tupla t . La única variable libre en F debe ser t .
Ejemplos:
 $\{t / t \in \mathbf{Actor}\}$ devuelve a todos los actores.
 $\{t / \exists a (a \in \mathbf{Actor} \wedge t.\text{nombreActor} = a.\text{nombreActor})\}$ devuelve los nombres de todos los actores.
- Las consultas CRT se aplican sobre instancias de relación y también devuelven una instancia de relación (el conjunto de tuplas para las cuales la fórmula devuelve verdadero).
- Los atributos que tendrán las tuplas devueltas son solamente los que aparecen alguna vez en la consulta.

Ejemplo:

$\{t / \exists a (a \in \mathbf{Actor} \wedge t.\text{nombreActor} = a.\text{nombreActor} \wedge t.\text{edad} = a.\text{edad})\}$
devuelve los nombres y edades de todos los actores.

- El lenguaje para escribir fórmulas es un subconjunto de la *Lógica de Primer Orden*.
- Es posible definir consultas que devuelvan infinitos resultados (consultas *inseguras*).

Ejemplo:

$\{t / \neg(t \in \mathbf{Actor})\}$

- Si dejamos afuera a las consultas *inseguras*, el poder expresivo de AR y CRT es el mismo.

Ejercicios

Actor(idActor, nombreActor, edad)
Serie(idSerie, nombreSerie, idGénero, añoInicio, añoFin)
Género(idGénero, nombreGénero)
Canal(idCanal, nombreCanal)
Participa_En(idActor, idSerie)
Transmite(idCanal, idSerie)

1. Listar los nombres de los actores mayores de 30 años que participan en la serie “Friends”.

$$\{t / \exists a, s, p (a \in \mathbf{Actor} \wedge a.edad > 30 \wedge s \in \mathbf{Serie} \wedge s.nombreSerie = \text{“Friends”} \wedge p \in \mathbf{Participa_En} \wedge p.idActor = a.idActor \wedge p.idSerie = s.idSerie \wedge t.nombreActor = a.nombreActor)\}$$

Notar que la única variable libre es t y que estará compuesta solamente por el nombre del actor ya que en la fórmula solamente se hace referencia al atributo *nombreActor* de t .

2. Listar los nombres de los canales que transmiten todas las series de comedia.

$$\{t / \exists c (c \in \mathbf{Canal} \wedge t.nombreCanal = c.nombreCanal \wedge \forall s (esSerieDeComedia(s) \Rightarrow \exists tra (tra \in \mathbf{Transmite} \wedge tra.idCanal = c.idCanal \wedge tra.idSerie = s.idSerie))\})\}$$

Definimos el predicado *esSerieDeComedia* como:

$$esSerieDeComedia(s) = s \in \mathbf{Serie} \wedge \exists g (g \in \mathbf{Género} \wedge g.idGénero = s.idGénero \wedge g.nombreGénero = \text{“Comedia”})$$

3. Listar los nombres de los actores mayores de 30 años que participan en la serie “Friends” y que además participaron alguna vez en series que comenzaron luego del año 2000.

$$\{t / \exists a (a \in \mathbf{Actor} \wedge mayorDe30yEnFriends(a) \wedge algunaVezEnSerieDesp2000(a) \wedge t.nombreActor = a.nombreActor)\}$$
$$mayorDe30yEnFriends(a) = a.edad > 30 \wedge \exists s, p (s \in \mathbf{Serie} \wedge s.nombreSerie = \text{“Friends”} \wedge p \in \mathbf{Participa_En} \wedge p.idActor = a.idActor \wedge p.idSerie = s.idSerie)$$
$$algunaVezEnSerieDesp2000(a) = \exists s, p (s \in \mathbf{Serie} \wedge s.añoInicio > 2000 \wedge p \in \mathbf{Participa_En} \wedge p.idActor = a.idActor \wedge p.idSerie = s.idSerie)$$

4. Listar los nombres de los actores mayores de 30 años que participan en la serie “Friends” y que además **nunca** participaron en series que comenzaron luego del año 2000.

$$\{t / \exists a (a \in \mathbf{Actor} \wedge mayorDe30yEnFriends(a) \wedge nuncaEnSerieDesp2000(a) \wedge t.nombreActor = a.nombreActor)\}$$

Para *mayorDe30yEnFriends(a)* usamos el mismo del ejercicio anterior.

$nuncaEnSerieDesp2000(a) = \neg \exists s (s \in \mathbf{Serie} \wedge s.añoInicio > 2000 \wedge \exists p (p \in \mathbf{Participa_En} \wedge p.idActor = a.idActor \wedge p.idSerie = s.idSerie))$

5. Listar los ids de las series que comenzaron más recientemente.

$\{t / \exists s (s \in \mathbf{Serie} \wedge t.idSerie = s.idSerie \wedge inicioMásReciente(s))\}$

$inicioMásReciente(s) = \neg \exists s' (s' \in \mathbf{Serie} \wedge s'.añoInicio > s.añoInicio)$

6. Listar los ids de los actores que participaron en al menos 2 series.

$\{t / \exists a (a \in \mathbf{Actor} \wedge t.idActor = a.idActor \wedge participóEnAlMenos2Series(a))\}$

$participóEnAlMenos2Series(a) = \exists p, p' (p \in \mathbf{Participa_En} \wedge p' \in \mathbf{Participa_En} \wedge p \neq p' \wedge p.idActor = a.idActor \wedge p'.idActor = a.idActor)$

Ejercicios pendientes

(Para que cada uno lo piense solo)

- Resolver el último ejercicio suponiendo que piden devolver los que cumplen “exactamente 2”.
- Comparar cada ejercicio de CRT con su versión en AR y analizar en qué lenguaje cree que es más sencillo definir las consultas.