Database Systems, CSCI 4380-01
Homework # 1 Answers
Due Thursday February 4, 2016 at 11:59:59 PM

Homework Statement. Suppose you are given the following database that tracks all UFO related events reported by different users of a database.

```
Users(userid, email, name)
Sightings(eventid, userid, status, date, time, state, city, country, description,
          shape, color, size, numpeople, visualangle, fromloc, toloc, casereview)
Abductions(eventid, userid, city, state, country, date, numpeople, description,
           casereview)
CropCircles(eventid, userid, city, state, country, picture, radius, date)
Behaviors(eventid, behavior)
```

Note example behaviors: hovered, orb-like, changed speed, discontinuous trajectory

```
Tags(eventid, tag)
```

Note example tags: eurowheel, bird, near railway, affected electricity

Note: All date fields are formatted as `mon-day-year`, e.g. `01-31-2016`. You can assume that you can check if a date time `X` comes after another date time `Y` by checking whether `X > Y`. Time is formatted similarly, e.g. `08:00:00` and can be compared using $>$ as date.

Each relation stores a UFO based event (sightings, abductions and cropcircles) reported by different users. An event with the same eventid can be of many types, for example both a UFO sighting and an abduction, or a crop circle. The remaining attributes will be explained as they come in specific queries.

Write the following queries using relational algebra:

Part 1. The following queries only need a single SELECT (σ), followed by a PROJECT (π) and RENAMING (ρ) as necessary:

(a) Return the countries in which there was a UFO sighting in 2015 with a shape "equilateral triangle", color "black" involving 2 or more people.

Answer.

$$\Pi_{country}(\sigma_C\, Sightings)$$

where C is given by

shape='equilateral triangle' and color='black' and numpeople $\geq$ *2 and date* $\geq$ *'01-01-2015' and date* $\leq$ *'12-31-2015'*

(b) Return the city and state in which a crop circle is seen in "United States" since 2010 (inclusive) or a radius of at least 10.

Answer.

$$\Pi_{city,state}(\sigma_{country='United\ States'\ and\ date \geq\ '01\text{-}01\text{-}2010'\ or\ radius \geq\ 10}\, CropCircles)$$

(c) Return the id of all events involving a behavior of "hovering".

Answer.

$$\Pi_{eventid}\,(\sigma_{behavior=\text{`}hovering\text{'}}, Behaviors)$$

Part 2. The following queries combine SELECT (σ), SET operations ($\cap, \cup, -$), PROJECTION (π) and RENAMING (ρ) as necessary:

(a) Return the id of all users who have reported either a sighting or an abduction.

Answer.

$$(\Pi_{userid}\, Sightings) \cup (\Pi_{userid}\, Abductions)$$

(b) Return all countries that have had sightings but no abductions and no crop circles.

Answer.

$$(\Pi_{country}\, Sightings) - ((\Pi_{country}\, Abductions) \cup (\Pi_{country}\, CropCircles))$$

or

$$((\Pi_{country}\, Sightings) - (\Pi_{country}\, Abductions)) - (\Pi_{country}\, CropCircles)$$

(c) Return the id all events involving both an abduction and a crop circle in the same state and country.

Answer.

$$\Pi_{eventid}((\Pi_{eventid,state,country}\, Abduction) \cap (\Pi_{eventid,state,country}\, CropCircles))$$

(d) Return the id of all events involving an abduction and a crop circle occurring in different locations (i.e. not the same state and country). They could be in the same country and different states or even different countries.

Answer.

$$\begin{aligned} R1 &= \Pi_{eventid}((\Pi_{eventid,state,country}\, Abduction) \cap (\Pi_{eventid,state,country}\, CropCircles)) \\ R2 &= (\Pi_{eventid}\, Abduction) \cap (\Pi_{eventid}\, CropCircles) \\ R3 &= R2 - R1 \end{aligned}$$

R2 is all events with both a crop circle and an abduction. So, R3 is the answer to this question.

Part 3. The following queries combine SELECT (σ) statements with a JOIN ($\bowtie$) (or CARTESIAN PRODUCT), followed by a PROJECT (π) and RENAMING (ρ) as necessary:

(a) Return the name of all users who have reported a sighting in "Lake Okobogee, Iowa, United States" (city, state and country respectively).

Answer.

$$\Pi_{name}(Users*(\sigma_{city=\text{`}Lake\ Okobogee\text{'}\ and\ state=\text{`}Iowa\text{'}\ and\ country=\text{`}United\ States\text{'}}, Sightings))$$

(b) Return the name of all users who have reported a crop circle in 2015 with tag "bird".

Answer.

$$\Pi_{name}(\sigma_{tag='bird' \text{ and } date \geq '01\text{-}01\text{-}2015' \text{ and } date \leq '12\text{-}31\text{-}2015'}(Users * CropCircles * Tags))$$

(c) Return the name of all users who have reported a sighting and an abduction on the same date.

Answer. (of course you can use any of theta or regular join, Cartesian product in your answer, as long as it is done correctly.)

$$R1(e1, u1, city1, state1, country1, date1, nump1, d1, cr1) = Abductions$$
$$R2 = Users * (R1 \bowtie_{userid=u1 \text{ and } date=d1} Sightings)$$
$$R3 = \Pi_{name} R2$$

Part 4. Freeform, you decide which combination is needed. Any relational algebra operator is fine.

(a) Find sighting events with tags "bird" or "eurowheel", involving a "round" or "oval" white object with "hovered" behavior. Return the state, city, country and description of these events.

Answer.

$$\begin{aligned} R1 &= \sigma_{tag='Bird' \text{ or } tag='Eurowheel'} Tags \\ R2 &= \sigma_{color='white' \text{ and } (shape='round' \text{ or } shape='oval')} Sightings \\ R3 &= \sigma_{behavior='hovered'} Behaviors \\ R4 &= \Pi_{state,city,country,description} (R1 * R2 * R3) \end{aligned}$$

Note: While technically you can do the "OR" statements by individual queries unioned, you should get used to distinguishing between when you need UNION and when OR is sufficient.

(b) Find pair of events (with different ids) involving sightings in the same state, city, country, date, involving at least 2 people, occurring before "17:00" that are reported by at least two different users. Return the pair of event ids.

Answer.

$$R1 = \Pi_{eventid,state,city,country,date,userid}(\sigma_{numpeople \geq 2 \text{ and } time \geq '17:00'} Sightings$$
$$R2(e2, state, city, country, date, u2) = R1$$
$$R3 = \Pi_{eventid,e2} (\sigma_{userid <> u2} (R1 * R2))$$

Note: You really do not need to check that eventids are different because if the userid is different, then eventid has to be different. Why? Given an event, you only can store a single userid (because event id is key). So, two different event ids must come from different events. Of course, you can use any version of Join or Cartesian project you wish here.

(c) **Bonus/optional.** Find pairs of events e1, e2 such that e2 has all the behaviors reported for e1 (in the Behaviors relation). Return the event ids for e1 and e2.

Note: If you really want to challenge yourself, try also restricting it to only the sighting events and require e1 and e2 to have exactly the same set of behaviors. This part is not part of the homework, but feel free to show me your solutions for discussion.

Answer.

Ok, this will take some explaining.

We will construct two main relations first:

All(e1,e2,b1) where e1 and e2 are two different events. The relation will contain a tuple (e1,e2,b1) if b1 is a behavior for e1. In other words, if e2 had the same behaviors as e1, it would have all the tuples in this relation.

Common(e1,e2,b1) where b1 is all behaviors e1 and e2 have in common.

Suppose now we compute: **Left=All-Common**. A tuple e1,e2 will be in Left only if there is at least one behavior of e1 that e2 does not have. These are exactly the events we do not want. So, we will need to subtract them from all the event pairs to find the answer to this query.

$$
\begin{aligned}
B1(e1, b1) &= Behaviors \\
B2(e2, b2) &= Behaviors \\
All &= \Pi_{e1,e2,b1} \left(B1 \bowtie_{e1<>e2} B2\right) \\
Common &= \Pi_{e1,e2,b1} \left(B1 \bowtie_{e1<>e2 \text{ and } b1=b2} B2\right) \\
Left &= \Pi_{e1,e2} \left(All - Common\right) \\
Events &= \Pi_{e1,e2} All \\
Answer &= Events - Left
\end{aligned}
$$

See this with an example of behaviors table:

e1	b1
1	a
2	a
2	b
3	a
3	b

Behaviors

e1	e2	b1
1	2	a
1	3	a
2	1	a
2	3	a
2	1	b
2	3	b
3	1	a
3	2	a
3	1	b
3	2	b

All

e1	e2	b1
1	2	a
1	3	a
2	1	a
2	3	a
2	3	b
3	1	a
3	2	a

Common

e1	e2
2	1
3	1

Left

e1	e2
1	2
1	3
2	3
3	2

Answer

So, 2 and 3 have all behaviors for 1 (just a), and 3 and 2 are have the behaviors for each other (a and b). However, 1 does not have all the behaviors for 2 and 3 (as it is missing b).

Now, finding pairs e1,e2 with the same exact behaviors is easy. Just join Answer with each other to find both pairs (AllIdentical below):

$$
\begin{aligned}
Answer2(e3, e4) &= Answer \\
AllIdentical &= \Pi_{e1,e2} \left(Answer \bowtie_{e1=e4 \text{ and } e2=e3} Answer2\right)
\end{aligned}
$$