1 RA to DRC and SQL

Consider the following RA expression:

\[
\pi_{\text{Country}} ( \\
\sigma_{((\text{Country} = \text{Country}_1 \land \text{Code}_2 = \text{Country}_2) \lor (\text{Country} = \text{Country}_2 \land \text{Code}_2 = \text{Country}_1))} \land \text{River} = \text{River}_2 (\\
\text{geoSource} \times \text{borders} \times \rho_{(\text{River}_2, \text{Code}_2, \text{P}_2)} (\text{geoEstuary})))
\]

1. Explain briefly but precisely in natural language what this RA expression returns independently of the actual data stored in the Mondial database.

2. Convert the RA expression into a DRC expression and a SQL query. Write down the number of rows returned by the SQL query.

3. Consider a database instance in which there exists a country for which all rivers that originate in the country also have their estuary in the same country. Is it possible according to the relational schema that the RA expression returns this country? Explain.

Solution:

1. Natural language: The codes of all countries for which at least one river that originates in this country has its estuary in a neighboring country.

2. DRC expression:

\[
\{\text{Code} \mid \exists \text{River}, \text{Code}_2 (\text{geoSource}(\text{River}, \text{Code},_1) \\
\land (\text{borders}(\text{Code}, \text{Code}_2,_) \lor \text{borders}(\text{Code}_2, \text{Code},_1)) \\
\land \text{geoEstuary}(\text{River}, \text{Code}_2, _1))\}
\]

SQL query:

```sql
SELECT DISTINCT s.Country
FROM geoEstuary e, borders b, geoSource s
OR (e.Country = b.Country2 AND s.Country = b.Country1)) \\
AND e.River = s.River;
```

The query returns 39 rows when executed on the Mondial database.

3. Yes. Based on the relational schema this can happen in two ways.

   (a) The relational schema allows that a country is a neighboring country to itself. For instance, the tuple (’D’, ’D’, 3621) can be inserted into the borders relation according to the Mondial relational schema. Then, the RA expression returns this country.
(b) The relational schema allows that a river’s estuary can be located in two different countries. If one of the countries is the country itself and the second one a neighboring country, then the RA expression returns this country.

2 DRC to RA and SQL

Consider the following DRC expression:

\[
\{ \text{Code} | \text{country}(\text{Code}, \ldots) \\
\land \forall \text{Prov} (\text{province}(\text{Prov}, \text{Code}, \ldots) \Rightarrow \text{geoMountain}(\_ , \text{Code}, \text{Prov})) \} 
\]

1. Explain briefly but precisely in natural language what this DRC expression returns independently of the actual data stored in the Mondial database.

2. Convert the DRC expression into an RA expression and a SQL query. Write down the number of rows returned by the SQL query.

3. Explain briefly but precisely in natural language what a modified DRC expression that contains a conjunction instead of an implication returns independently of the actual data stored in the Mondial database.

*Solution:*

1. Natural language: The codes of all countries where at least one mountain is located in each of the country’s provinces.

2. RA expression:

\[
\pi_{\text{Code}}(\text{country}) \\
- \pi_{\text{Country}}(\pi_{\text{Name}}, \pi_{\text{Country}}(\text{province}) \\
- \pi_{\text{Province}}, \pi_{\text{Country}}(\text{geoMountain}))
\]

SQL query:

```sql
SELECT Code
FROM country
WHERE Code NOT IN (SELECT p.Country
FROM (SELECT Name, Country FROM province
EXCEPT
SELECT Province, Country FROM geoMountain)
) p);
```

The query returns 51 rows when executed on the Mondial database.

3. The result is undefined since the DRC expression is not domain-independent.
3 SQL to RA and DRC

1. Consider the following SQL query:

\[
\text{SELECT } b.\text{Country1}, \text{SUM}(b.\text{Length}) \\
\text{FROM} (\text{SELECT } \text{Country1}, \text{Length} \text{ FROM borders} \\
\text{UNION ALL} \\
\text{SELECT } \text{Country2}, \text{Length} \text{ FROM borders} \\
) \text{ b} \\
\text{GROUP BY } b.\text{Country1};
\]

(a) Explain briefly but precisely in natural language what this SQL query returns independently of the actual data stored in the Mondial database. Assume that a country cannot be a neighboring country to itself.

(b) Can the SQL query be converted into a DRC expression? If yes, give the expression, if no, explain why not.

(c) Convert the SQL query into an RA expression.

Solution:

(a) Natural language: The code and total border lengths of all countries with at least one border.

(b) No. The sum aggregation cannot be expressed in DRC as the number of summands is unbound. For instance, country A might have 4 neighboring countries whereas country B might have 7 neighboring countries. Then, it is not possible to enumerate the number of summands independently of the number of neighboring countries.

(c) RA expression:

\[
\begin{align*}
\text{Country1} & \cup_{\text{sum}(\text{Length})} (\pi \text{Country1} \text{, Country2} \text{, Length} (\text{borders})) \\
\end{align*}
\]

2. Consider the following SQL query:

\[
\text{SELECT } c.\text{Name}, c.\text{Code} \\
\text{FROM country } c, \text{ geoSource } g \\
\text{WHERE } c.\text{Code} = g.\text{Country};
\]

(a) Explain briefly but precisely in natural language what this SQL query returns independently of the actual data stored in the Mondial database.

(b) Is it possible to convert the SQL query into an equivalent RA and/or DRC expression? Equivalent means that the result is the same on all possible database instances. If yes, give the expression(s). If no, explain why not.

Solution:

(a) Natural language: The names and codes of all countries in which at least one river originates.
2. No, it is not possible to formulate equivalent DRC and RA expressions. The reason is that the attribute *Country* is not the primary key of the relation *geoSource* (it is only part of the primary key). Thus, there can be database instances in which several rivers originate in the same country. Then, the SQL query returns after the projection as many (Name, Code) tuples for this country as rivers originate in the country. As DRC and RA eliminate duplicates, their results would contain the (Name, Code) tuple only once after the projection.