Fall 2022 B461 Assignment 5
Query translation and Optimization

Chinmayee Mundhe, Aravind Sheru, Muazzam Siddiqui

Released: October 31, 2022
Due: November 14, 2022

1 Introduction

The goals for this assignment are to

1. Translate the pure SQL queries to RA SQL queries;
2. Convert RA SQL to RA expression;
3. Use rewrite rules to optimize an RA expression;
4. Convert RA expression to RA SQL;

To turn in your assignment, you will need to upload to Canvas the following files zipped into one :

- `assignment5.sql`
- `assignment5.txt`
- `assignment5.pdf`

The `assignment5.sql` contains the necessary SQL statements that solve the problems in this assignment. The `assignment1.sql` file must be such that the AI's can run it in their PostgreSQL environment.

The `assignment5.txt` file contains the results of running your queries.

The `assignment5.pdf` file contains the solutions for RA notation.

For the problems in this assignment we will use the following database schema:[1]

```
Westerosi(wid, wname, wlocation)
House(hname,kingdom)
Skill(skill)
OfHouse(wid, hname, wages)
HouseAllyRegion(hname,region)
WesterosiSkill(wid,skill)
Predecessor(succid,predid)
Knows(wid1,wid2)
```

In this database[2] we maintain a set of Westerosis[3] (`Westerosi`), a set of Houses (`House`), and a set of skills (`Skill`). The `wname` attribute in `Westerosi` is the name of the resident of Westeros.

The `wlocation` attribute in `Westerosi` specifies the area in which the person is currently stationed. The `hname` attribute in `House` is the name of a House in Westeros.

The `kingdom` attribute in `House` is the name of the location wherein the lord of the house resides. The `skill` attribute in `Skill` is the name of a skill possessed by Westerosis.

A Westerosi can be of at most one House. This information is maintained in the `OfHouse` relation. (We permit that a Westerosi does not belong to any House.) The `wages` attribute in `OfHouse` specifies the wages made by the Westerosi.

The `region` attribute in `HouseAllyRegion` indicates a region in which the house has allies. (Houses may have allies in multiple regions.)

A Westerosi can have multiple skills. This information is maintained in the `WesterosiSkill` relation. A skill can be the skill of multiple Westerosis. (A Westerosi may not have any skills, and a skill may have no Westerosis with that skill.)

A pair (s, p) in `Predecessor` indicates that a Westerosi (successor) s has a Westerosi p as one of his or her predecessors. We permit that a successor has multiple predecessors and that a predecessor may be succeeded by multiple successors. (It is possible that a Westerosi has no predecessor and that a Westerosi is not a predecessor.) We further require that a Westerosi and his or her predecessors must belong to the same House.

The relation `Knows` maintains a set of pairs (w_1, w_2) where w_1 and w_2 are wids of Westerosis. The pair (w_1, w_2) indicates that the person with wid w_1

[1] The primary key, which may consist of one or more attributes, of each of these relations is underlined.

[2] The values of the database are inspired by a popular series - Game of Thrones just to make the course a little fun. We in no way bear responsibility for any spoilers or faults in the storyline/theories based on these values. So kindly humor us and have just as fun with making the queries as we do in asking for them!

[3] Residents of Westeros

knows the person with wid w_2. We do not assume that the relation Knows is symmetric: it is possible that (w_1, w_2) is in the relation but that (w_2, w_1) is not.

The domain for the attributes wid, wages, succid, and predid is integer. The domain for all other attributes is text.

We assume the following foreign key constraints:

- wid is a foreign key in OfHouse referencing the primary key wid in Westerosi;
- hname is a foreign key in OfHouse referencing the primary key hname in House;
- hname is a foreign key in HouseAllyRegion referencing the primary key hname in House;
- wid is a foreign key in WesterosiSkill referencing the primary key wid in Westerosi;
- skill is a foreign key in WesterosiSkill referencing the primary key skill in Skill;
- succid is a foreign key in Predecessor referencing the primary key wid in Westerosi; and
- predid is a foreign key in Predecessor referencing the primary key wid in Westerosi;
- wid1 is a foreign key in Knows referencing the primary key wid in Westerosi; and
- wid2 is a foreign key in Knows referencing the primary key wid in Westerosi

The file data.sql contains the data supplied for this assignment.

2 Translating Pure SQL Queries to RA SQL queries and RA Optimization

Create a database in PostgreSQL that stores the data provided in the `data.sql` file.

Good News! - There are only 5 questions for this assignment.

Bad News! - There are many additional steps while solving each question. For each question you will follow these steps -

- STEP 1 - Translate each given SQL query to the respective **RA SQL query** using the rules taught in Week 9 and Week 10.
- STEP 2 - Convert the above translated **RA SQL query** to the corresponding **RA expression**.
- STEP 3 - Optimize the above **RA expression** using the optimization rules taught in Week 11 and show at least 2 significant optimization steps using **RA expressions**.
- STEP 4 - Finally convert the above **Optimized RA expression** to the corresponding **Optimized RA SQL query**. This must be carried out in Word/Latex for each rewrite rule applied step-by-step. That's all!

1. Translate and Optimize the following SQL query to RA SQL:

```
SELECT W.WID, W.WNAME
      FROM WESTEROSI W, PREDECESSOR P
      WHERE W.WID = P.SUCCID
      AND W.WID not in
           (SELECT W.WID
                FROM WESTEROSI W,
                PREDECESSOR P,
                OFHOUSE OH1,
                OFHOUSE OH2
                WHERE W.WID = P.SUCCID
                     AND OH1.WID = P.SUCCID
                     AND OH2.WID = P.PREDID
                     AND OH1.WAGES <= OH2.WAGES );
```

2. Translate and Optimize the following SQL query to RA SQL:

```
SELECT H.HNAME, H.KINGDOM
       FROM HOUSE H
       WHERE H.HNAME in
             (SELECT OH.HNAME
                   FROM OFHOUSE OH
                   WHERE OH.WAGES < 60000
                   AND OH.WID = SOME
                   (SELECT WS.WID
                         FROM WESTEROSISKILL WS
                         WHERE WS.SKILL = 'Archery'));
```

3. Translate and Optimize the following SQL query to RA SQL:

```
SELECT       DISTINCT W.WID
FROM         WESTEROSI W
WHERE        W.WLOCATION = 'Winterfell'
AND EXISTS   (SELECT 1
             FROM OFHOUSE H, WESTEROSISKILL W1
             WHERE W.WID = H.WID AND W.WID = W1.WID
             AND H.WAGES = 50000 AND NOT W1.SKILL = 'Swordsmanship');
```

4. Translate and Optimize the following SQL query to RA SQL:

```
SELECT            W.WID
FROM              WESTEROSI W
WHERE             NOT EXISTS (SELECT 1
                  FROM HOUSEALLYREGION HA
                  WHERE HA.REGION = 'IronIslands' AND
HA.HNAME NOT IN   (SELECT H.HNAME
                  FROM OFHOUSE H
                  WHERE H.WID = W.WID
AND H.WID in      (SELECT WS.WID
                  FROM WESTEROSISKILL WS
                  WHERE WS.SKILL = 'Archery')));
```

5. Formulate a Pure SQL query for the following statement and further Translate and Optimize it to RA SQL:

   - Find the wname and wlocation of each westerosi whose wages are strictly greater than 50000, and belongs to a house which has a kingdom in KingsLanding and has some skill.