where and .

* 1. Query:

select c.cname,

case when exists(

select 1

from worksFor w, hasManager m

where w.cname = c.cname and w.salary > 55000

and m.eid = w.pid and m.mid in (

select k.pid2

from Knows k

where k.pid1 = w.pid

)

) then '1'

else '0'

end as t

from Company c;

Simplifying,

select c.cname, '1' as t

from Company c

where exists (

select 1

from worksFor w, hasManager m, Knows k

where k.pid1 = w.pid and k.pid2 = m.mid

and m.eid = w.pid and w.cname = c.cname and w.salary > 55000

)

union

select c.cname, '0' as t

from Company c

where not exists (

select 1

from worksFor w, hasManager m, Knows k

where k.pid1 = w.pid and k.pid2 = m.mid

and m.eid = w.pid and w.cname = c.cname and w.salary > 55000

);

Converting it to RA SQL:

select distinct q1.cname, 1 as t

from (

select c.\*

from Company c, worksFor w, hasManager hm, Knows k

where k.pid1 = w.pid and k.pid2 = hm.mid

and hm.eid = w.pid and w.cname = c.cname and w.salary > 55000

) q1

union

select distinct q2.cname, 0 as t

from (

select c.\*

from Company c

except

select c.\*

from Company c, worksFor w, hasManager hm, Knows k

where k.pid1 = w.pid and k.pid2 = hm.mid

and hm.eid = w.pid and w.cname = c.cname and w.salary > 55000

) q2;

* 1. RA expression:
  2. exists (union)

Translating the **exists** in where clause:

select L1 (r1, …, rn)

from (R1 r1, … Rn rn), (S1 s1, …, S1 sm)

where C1 (r1, …, rn) and C2 (s1, …, sm, r1, …, rn)

union

select L1 (r1, …, rn)

from (R1 r1, … Rn rn), (S1 s1, …, S1 sm)

where C1 (r1, …, rn) and C3 (s1, …, sm, r1, …, rn)

RA expression:

* 1. exists (intersect)

select L1(r1,...,rn)

from R1 r1, ..., Rn rn

where C1(r1,...,rn) and exists (select s1.\*, s2.\*, ..., sm.\*

from S1 s1,..., S1 sm

where C2(s1,...,sm, r1,...,rn)

intersect

select s1.\*, s2.\*, ..., sm.\*

from S1 s1,..., S1 sm

where C3(s1,...,sm, r1,...,rn));

Translating **exists** in where clause:

select distinct L1q(r1,...,rn)

from (

select L1(r1,...,rn)

from R1 r1, ..., Rn rn, S1 s1,..., S1 sm

where C1(r1,...,rn) and C2(s1,...,sm, r1,...,rn)

intersect

select L1(r1,...,rn)

from R1 r1, ..., Rn rn, S1 s1,..., S1 sm

where C1(r1,...,rn) and C3(s1,...,sm, r1,...,rn)

) q;

RA expression:

* 1. exists (except)

select distinct L1(r1,...,rn)

from R1 r1, ..., Rn rn

where C1(r1,...,rn) and exists (select s1.\*, s2.\*, ..., sm.\*

from S1 s1,..., S1 sm

where C2(s1,...,sm, r1,...,rn)

except

select s1.\*, s2.\*, ..., sm.\*

from S1 s1,..., S1 sm

where C3(s1,...,sm, r1,...,rn));

Translating the exists in where clause:

select distinct L1q(r1,...,rn)

from (

select L1(r1,...,rn)

from R1 r1, ..., Rn rn, S1 s1,..., S1 sm

where C1(r1,...,rn) and C2(s1,...,sm, r1,...,rn)

except

select L1(r1,...,rn)

from R1 r1, ..., Rn rn, S1 s1,..., S1 sm

where C1(r1,...,rn) and C3(s1,...,sm, r1,...,rn)

) q;

RA expression:

* 1. not exists (union)  
       
     select L1(r1,...,rn)

from R1 r1, ..., Rn rn

where C1(r1,...,rn) and not exists (select s1.\*, s2.\*, ..., sm.\*

from S1 s1,..., S1 sm

where C2(s1,...,sm, r1,...,rn)

union

select s1.\*, s2.\*, ..., sm.\*

from S1 s1,..., S1 sm

where C3(s1,...,sm, r1,...,rn))

Translating the **not exists** in the where clause:

select distinct L1q(r1,...,rn)

from (

select L1(r1,...,rn)

from R1 r1, ..., Rn rn

where C1(r1,...,rn)

except

select L1(r1,...,rn)

from R1 r1, ..., Rn rn, S1 s1,..., S1 sm

where C1 (r1, …, rn) and C2 (s1, …, sm, r1, …, rn)

union

select L1 (r1, …, rn)

from (R1 r1, … Rn rn), (S1 s1, …, S1 sm)

where C1 (r1, …, rn) and C3 (s1, …, sm, r1, …, rn)

) q;

RA Expression:

* 1. not exists (intersect)

select L1(r1,...,rn)

from R1 r1, ..., Rn rn

where C1(r1,...,rn) and not exists (select s1.\*, s2.\*, ..., sm.\*

from S1 s1,..., S1 sm

where C2(s1,...,sm, r1,...,rn)

intersect

select s1.\*, s2.\*, ..., sm.\*

from S1 s1,..., S1 sm

where C3(s1,...,sm, r1,...,rn))

Translating the **not exists** in the where clause:

select distinct L1q2(r1,...,rn)

from (

select L1(r1,...,rn)

from R1 r1, ..., Rn rn

where C1(r1,...,rn)

except

select distinct L1q1(r1,...,rn)

from (

select L1(r1,...,rn)

from R1 r1, ..., Rn rn, S1 s1,..., S1 sm

where C1(r1,...,rn) and C2(s1,...,sm, r1,...,rn)

intersect

select L1(r1,...,rn)

from R1 r1, ..., Rn rn, S1 s1,..., S1 sm

where C1(r1,...,rn) and C3(s1,...,sm, r1,...,rn)

) q1

) q2

RA expression:

* 1. not exists (except)  
       
     select L1(r1,...,rn)

from R1 r1, ..., Rn rn

where C1(r1,...,rn) and not exists (select s1.\*, s2.\*, ..., sm.\*

from S1 s1,..., S1 sm

where C2(s1,...,sm, r1,...,rn)

except

select s1.\*, s2.\*, ..., sm.\*

from S1 s1,..., S1 sm

where C3(s1,...,sm, r1,...,rn))

Translating the **not exists** in the where clause:

select distinct L1q2(r1,...,rn)

from (

select L1(r1,...,rn)

from R1 r1, ..., Rn rn

where C1(r1,...,rn)

except

select distinct L1q1(r1,...,rn)

from (

select L1(r1,...,rn)

from R1 r1, ..., Rn rn, S1 s1,..., S1 sm

where C1(r1,...,rn) and C2(s1,...,sm, r1,...,rn)

except

select L1(r1,...,rn)

from R1 r1, ..., Rn rn, S1 s1,..., S1 sm

where C1(r1,...,rn) and C3(s1,...,sm, r1,...,rn)

) q1

) q2

RA Expression:

1. To prove,

πa,d(R ⋈c=d S) = πa,d(πa,c(R) ⋈c=d πd(S))

Given, R (a, b, c) and S (d, e)

Consider three attributes (a, b, c) that belong to R and (d) that belong to S and c = d.

*Attribute, c exists in R and d in S, such that the join condition is satisfied.*

1. The RA SQL query can be expressed in standard notation as below:
2. (c1) Pushing projections over Joins,

(c2) Using semi-joins rule,

Therefore, the optimized RA expression is:

1. The RA SQL query can be expressed in standard notation as below:
2. (c1) Pushing selection over joins,

(c2) Converting natural joins to Semi-joins,

1. The RA SQL query can be expressed in standard notation as below:
2. (c1) Eliminating attributes,

(c2) Pushing selections over joins,

1. The RA SQL query can be expressed in standard notation as below:
2. (c1) Pushing selections over joins

(c2) Using semi-joins

9.

1. The RA SQL query can be expressed in standard notation as below:
2. (c1) Cascading selections

(c2) Attribute elimination,