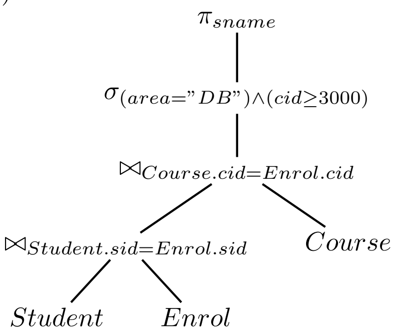
CS4224/CS5424 Quiz 2: Query Processing **Wk 3, Sem 1, 2023/24**

1. Consider the query Q on the following distributed database with three relations:

   - `Student` (<u>sid</u>, sname, major)
   - `Course` (<u>cid</u>, cname, area)
   - `Enrol` (<u>sid,cid</u>, grade)



Query Q

Each of the relations is partitioned into two fragments as follows:

- $S_1 = \pi_{sid,sname}(Student)$
- $S_2 = \pi_{sid,major}(Student)$
- $C_1 = \sigma_{cid\leq 1000}(Course)$
- $C_2 = \sigma_{cid>1000}(Course)$
- $E_1 = Enrol \ltimes_{cid} C_1$
- $E_2 = Enrol \ltimes_{cid} C_2$

Apply appropriate reduction techniques and rewriting rules to simplifiy the localized query of Q. Your final query should be optimized such that

1. all query expressions on fragments that do not contribute to the query results are eliminated, and
2. selections and projections are pushed down whenever possible to minimize the size of intermediate query results.

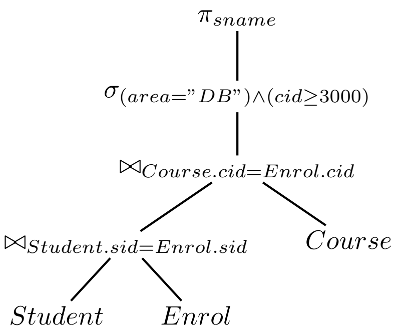
You may use the following extended version of Rule 5.2 (push-down of selection over join):

$$\sigma_p(R \bowtie_{p'} S) \equiv \sigma_{p_R}(R) \bowtie_{p'} \sigma_{p_S}(S), \text{ where}$$

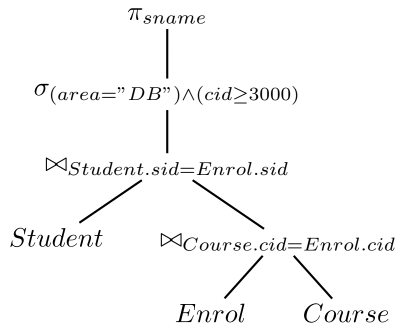
$$p = p_R \wedge p_S, \; attributes(p_R) \subseteq attributes(R), \text{ and } attributes(p_S) \subseteq attributes(S)$$

Solution:

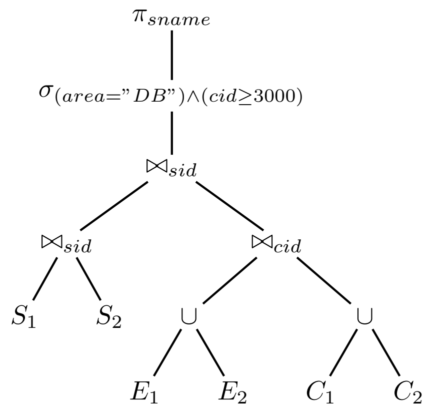
1. Input query



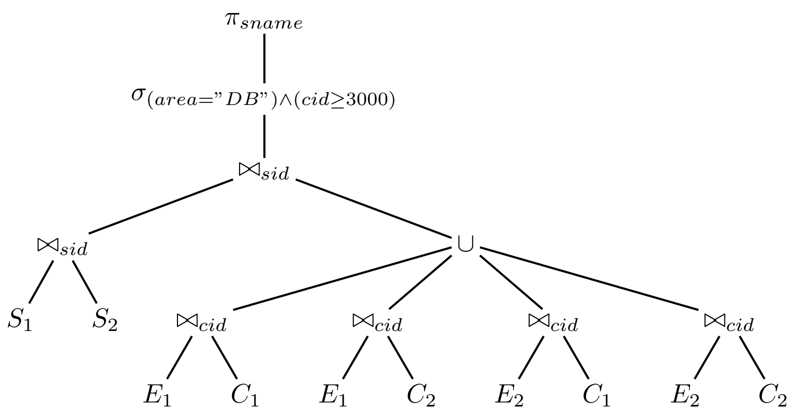
2. Rule 2.2 (Associativity of Join)



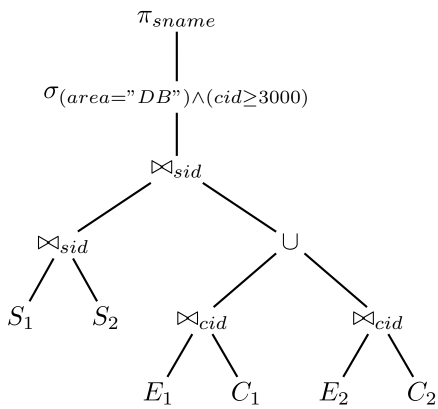
3. Localized query



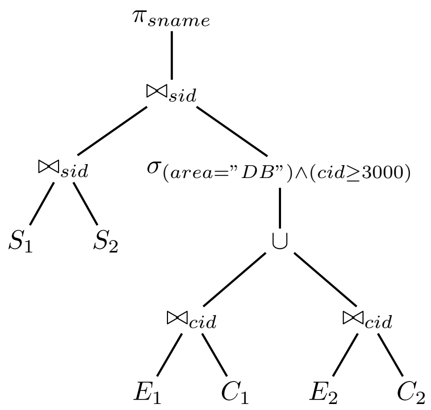
4. Distribute join over union



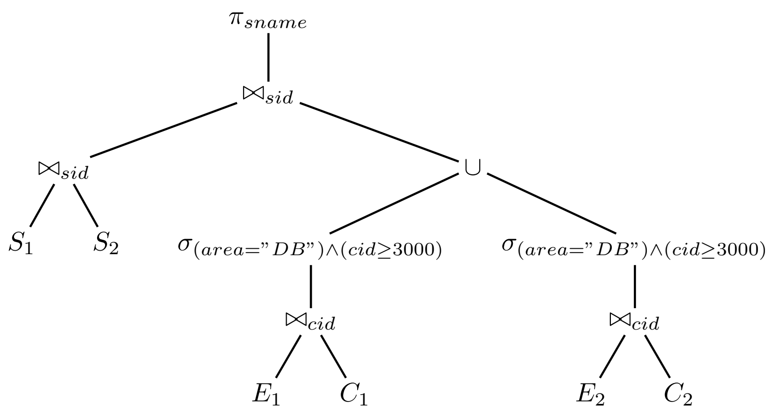
5. Rule 3 for derived fragmentation



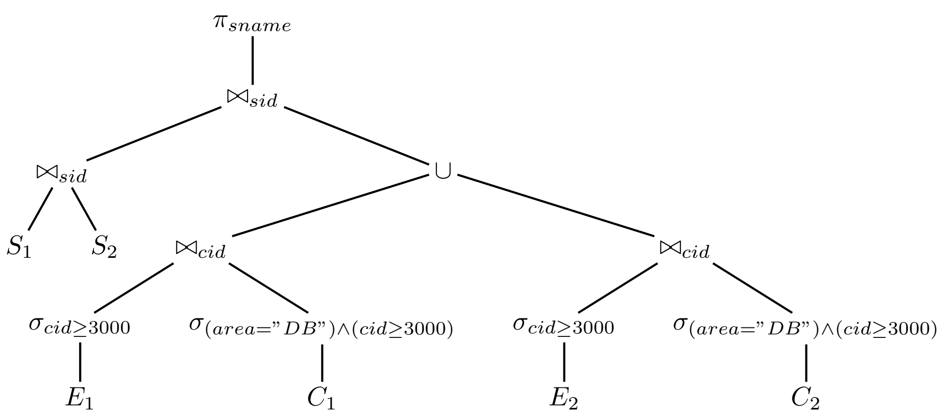
6. Rule 5.2 (Push-down of selection over join)



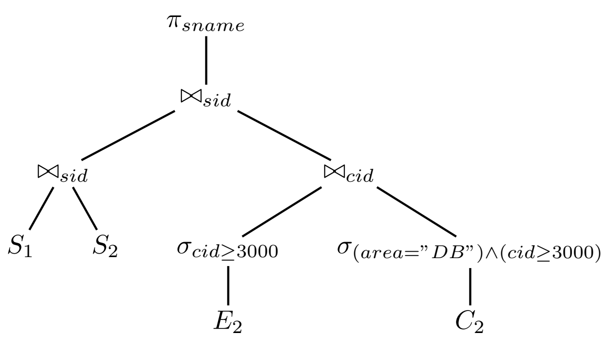
7. Rule 5.3 (Push-down of selection over union)



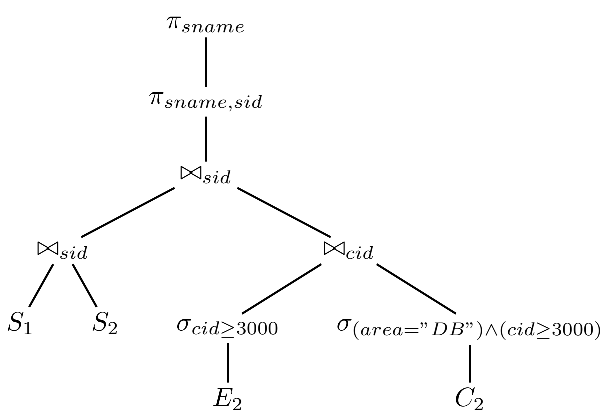
8. Extended version of Rule 5.2 (Push-down of selection over join)



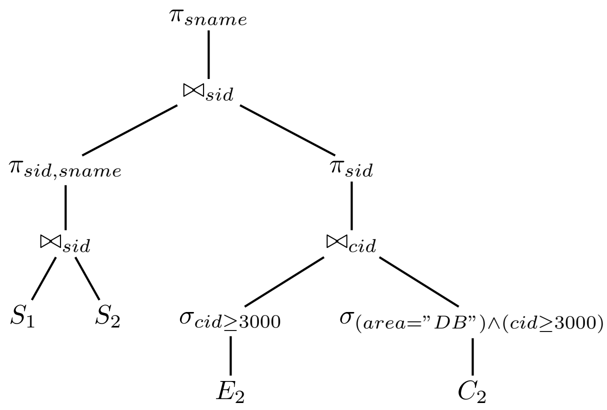
9. Reduction with selection for horizontal fragmentation



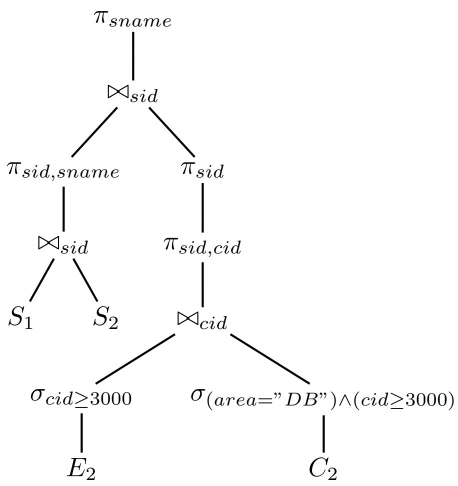
10. Rule 3.1 (Idempotence of projection)



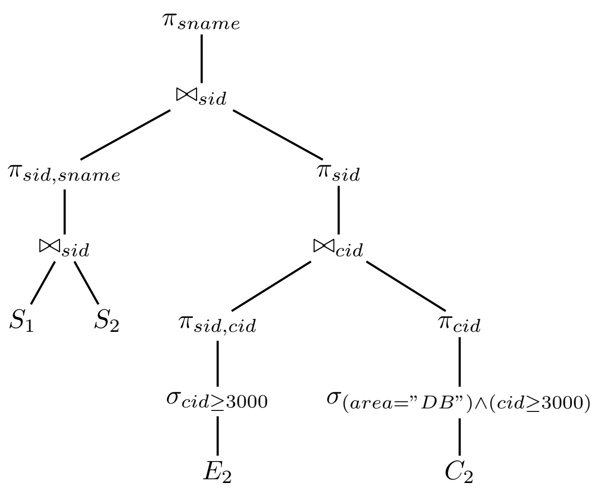
11. Rule 6.2 (push-down of projection over join)



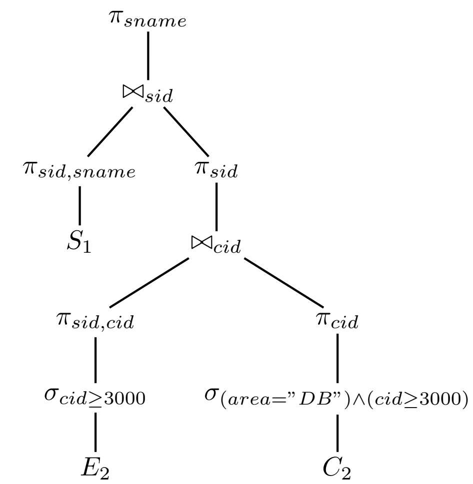
12. Rule 3.1 (Idempotence of projection)



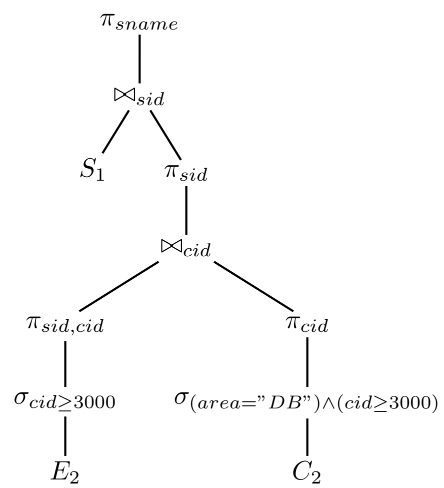
13. Rule 6.2 (push-down of projection over join)



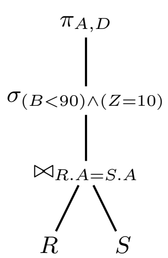
14. Rule 4 (reduction with vertical fragmentation)



15. Simplification (elimination of redundant projection on S_1)



2. Consider the following query Q on two relations, $R(\underline{A}, B, C, D, E)$ and $S(\underline{W}, X, Y, Z, A)$, where the key attribute of each relation is underlined, and $S.A$ is a foreign key of R consisting of only non-null values. Furthermore, R is partitioned into 4 fragments $\{R_{11}, R_{12}, R_2, R_3\}$ and S is partitioned into 4 fragments $\{S_{11}, S_{12}, S_{13}, S_2\}$.



Query Q

$S_1 = \sigma_{Y<600}(S)$
$S_2 = \sigma_{Y \geq 600}(S)$
$R_1 = \sigma_{B<20}(R)$
$R_2 = \sigma_{B \in [20,100)}(R)$
$R_3 = \sigma_{B \geq 100}(R)$
$R_{11} = \pi_{A,B,C}(R_1)$
$R_{12} = \pi_{A,D,E}(R_1)$
$S_{11} = S_1 \ltimes_A R_1$
$S_{12} = S_1 \ltimes_A R_2$
$S_{13} = S_1 \ltimes_A R_3$

Fragmentation of relations R & S

Write down all the relevant fragments that need to be accessed for evaluating query Q.

Solution:

$$\begin{aligned}
\pi_{A,D}(\sigma_{(B<90)\wedge(Z=10)}(R \bowtie_A S) &= \pi_{A,D}(\sigma_{B<90}(R) \bowtie_A \sigma_{Z=10}(S)) \\
&= \pi_{A,D}(\sigma_{B<90}(R)) \bowtie_A \pi_A(\sigma_{Z=10}(S)) \\
R &= (R_{11} \bowtie R_{12}) \cup R_2 \cup R_3 \\
\sigma_{B<90}(R) &= \sigma_{B<90}((R_{11} \bowtie R_{12}) \cup R_2 \cup R_3) \\
&= \sigma_{B<90}((R_{11} \bowtie R_{12}) \cup R_2) \\
&= (R_{11} \bowtie R_{12}) \cup \sigma_{B<90}(R_2) \\
\pi_{A,D}(\sigma_{B<90}(R)) &= \pi_{A,D}((R_{11} \bowtie R_{12}) \cup \sigma_{B<90}(R_2)) \\
&= \pi_{A,D}(R_{12} \cup \sigma_{B<90}(R_2)) \\
&= \pi_{A,D}(R_{12}) \cup \pi_{A,D}(\sigma_{B<90}(R_2))) \\
S &= S_{11} \cup S_{12} \cup S_{13} \cup S_2 \\
\pi_A(\sigma_{Z=10}(S)) &= \pi_A(\sigma_{Z=10}(S_{11} \cup S_{12} \cup S_{13} \cup S_2))
\end{aligned}$$

Since $S_{11} = S_1 \ltimes_A R_1$, we have $S_{11} \bowtie_A R_2 = S_{11} \bowtie_A R_3 = \emptyset$.
Since $S_{12} = S_1 \ltimes_A R_2$, we have $S_{12} \bowtie_A R_1 = S_{12} \bowtie_A R_3 = \emptyset$.
Since $S_{13} = S_1 \ltimes_A R_3$, we have $S_{13} \bowtie_A R_1 = S_{13} \bowtie_A R_2 = \emptyset$.

$$\begin{aligned}
\pi_{A,D}(\sigma_{B<90}(R)) \bowtie_A \pi_A(\sigma_{Z=10}(S)) = &(\pi_{A,D}(R_{12}) \bowtie_A \pi_A(\sigma_{Z=10}(S_{11} \cup S_2))) \cup \\
&(\pi_{A,D}(\sigma_{B<90}(R_2)) \bowtie_A \pi_A(\sigma_{Z=10}(S_{12} \cup S_2)))
\end{aligned}$$

Therefore, the relevant fragments are R_{12}, R_2, S_{11}, S_{12}, and S_2.

3. Consider a database consisting of three relations **R(A,B,X), T(C,D,X,Y), and U(E,F,Y)** that are distributed across three sites (S_1, S_2, and S_3).

   Assume the following:

   1. Attributes A, C, and E, are the key attributes of relations R, T, and U, respectively.
   2. All the attributes have the same integer domain.
   3. The three relations are **horizontally hash partitioned** using the same hash function into three fragments each: R_1,R_2,R_3 for relation R, T_1,T_2,T_3 for relation T, and U_1,U_2,U_3 for relation U.
   4. R is hash partitioned on attribute B.
   5. T is hash partitioned on attribute D.
   6. U is hash partitioned on attribute Y.
   7. The relation fragments are distributed as follows:
      (a) Site S_1 stores $\{R_1, T_1, U_1\}$,
      (b) Site S_2 stores $\{R_2, T_2, U_2\}$, and
      (c) Site S_3 stores $\{R_3, T_3, U_3\}$.
   8. Let size(X) denote the size (in GB) of a relation X or the result of a relational algebra expression X.
      (a) size(R) = 100 GB
      (b) size(T) = 240 GB
      (c) size(U) = 200 GB
      (d) size($R \bowtie_X T$) = 300 GB
      (e) size($T \bowtie_Y U$) = 300 GB

9. Whenever a fragment F_i at Site i is being repartitioned, none of the tuples in F_i get repartitioned to Site i; i.e., every tuple at Site i needs to be transmitted out of Site i.
10. For any query, the final query results can be distributed among the three sites in any manner; i.e., it is not necessary to store all the query results at a single site.

(a) Consider the query **Q1**: SELECT * FROM R, T WHERE R.X = T.X.
What is the **total communication cost** of the optimal query plan for **Q1** that minimizes the total communication cost (in terms of number of GB transmitted)?

(b) Consider the query **Q2**: SELECT * FROM T, U WHERE T.Y = U.Y.
What is the **total communication cost** of the optimal query plan for **Q2** that minimizes the total communication cost (in terms of number of GB transmitted)?

(c) Consider the query **Q3**: SELECT * FROM R, T, U WHERE R.X = T.X AND T.Y = U.Y.
What is the **total communication cost** of the optimal query plan for **Q3** that minimizes the total communication cost (in terms of number of GB transmitted)?

Solution:

(a)
- Neither of the tables (R & T) is partitioned on the join attribute X.
- The broadcast join (broadcast R) plan costs 2(100) = 200.
- The repartitioned join plan costs 100+240 = 340 which is more costly.
- The optimal plan is the broadcast join strategy by broadcasting R with a cost of 200.

(b)
- Only table U is partitioned on the join attribute Y.
- The directed join plan costs 240 (size of T).
- The broadcast join plan (broadcast U) costs 2(200) = 400.
- The optimal plan is the directed join strategy with a cost of 240.

(c)
- There are two possible join orderings for this query.
  - Plan 1: $(R \bowtie_X T) \bowtie_Y U$
  - Plan 2: $(T \bowtie_Y U) \bowtie_X R$
- Plan 1a: Compute $J1 = R \bowtie_X T$ using broadcast join (broadcast R) with a cost of 200 (from part a).
- Plan 1b: Two options to compute $J1 \bowtie_Y U$ since only U is partitioned on join attribute Y
  - Option 1: Directed join (repartition J1) costs 300.
  - Option 2: Broadcast join (broadcast U) costs 2(200) = 400.
  - The optimal plan for $J1 \bowtie_Y U$ is directed join strategy with a cost of 300.
- Thus, the optimal strategy for plan 1 is using broadcast join (broadcast R) to join R & T, followed by using directed join (repartition $J1 = R \bowtie_X T$) to join J1 & U with total cost = 200 + 300 = 500.
- Plan 2a: First compute $J2 = T \bowtie_Y U$ using directed join (repartition T) with a cost of 240 (from part b).
- Plan 2b: Two options to compute $J2 \bowtie_X R$ since neither J2 nor R is partitioned on join attribute X
  - Option 1: Repartitioned join costs 100 + 300 = 400
  - Option 2: Broadcast join (broadcast R) costs 2(100) = 200
  - The optimal plan for $J2 \bowtie_X R$ is broadcast join strategy (broadcast R) with a cost of 200.
- Thus, the optimal strategy for plan 2 is using directed join to join T & U, followed by using broadcast join (broadcast R) to join with R with total cost = 240 + 200 = 440
- Therefore, the optimal plan for Q3 is plan 2 with a cost of 440.