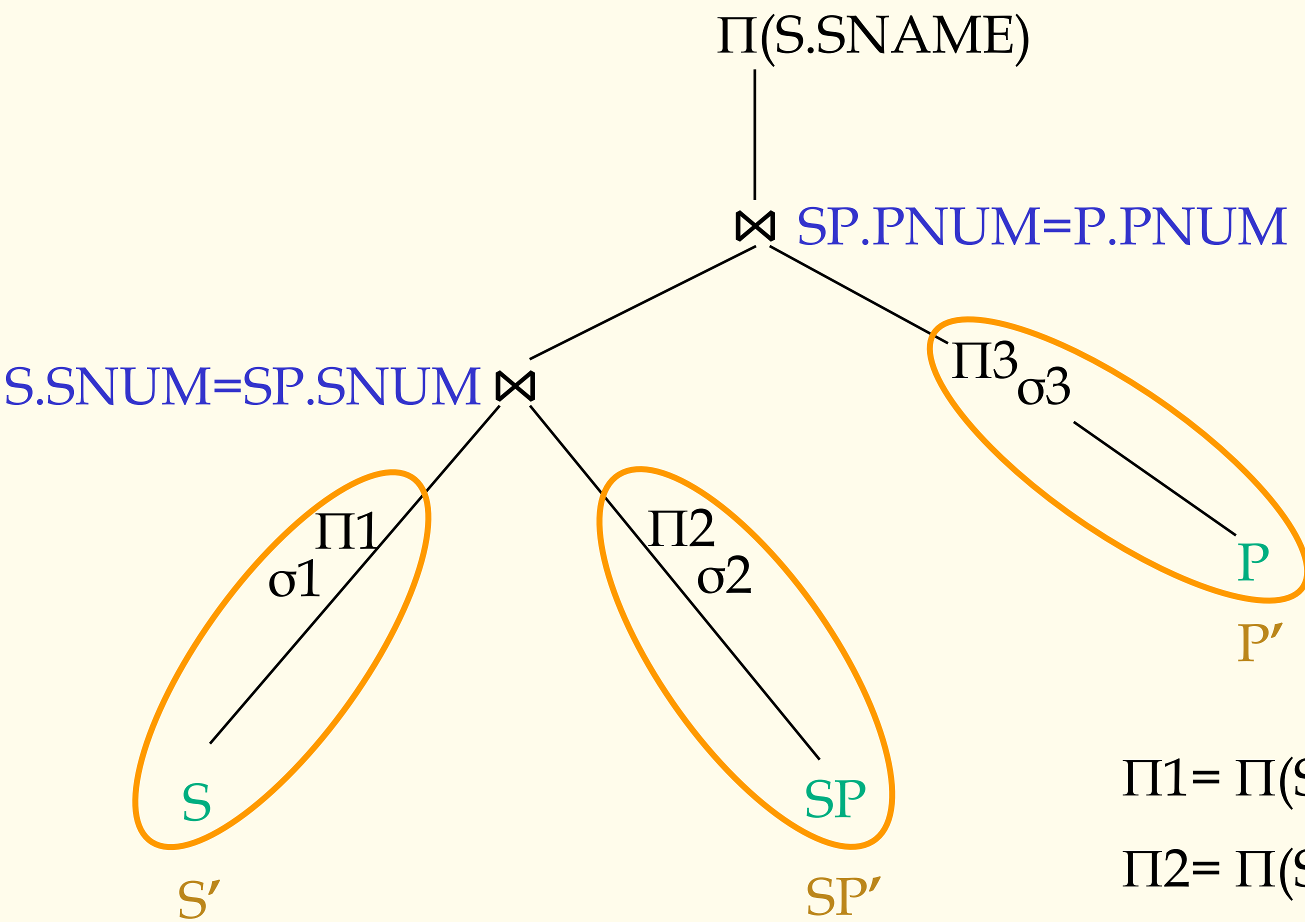
After equivalent transform (Algebra optimization) :

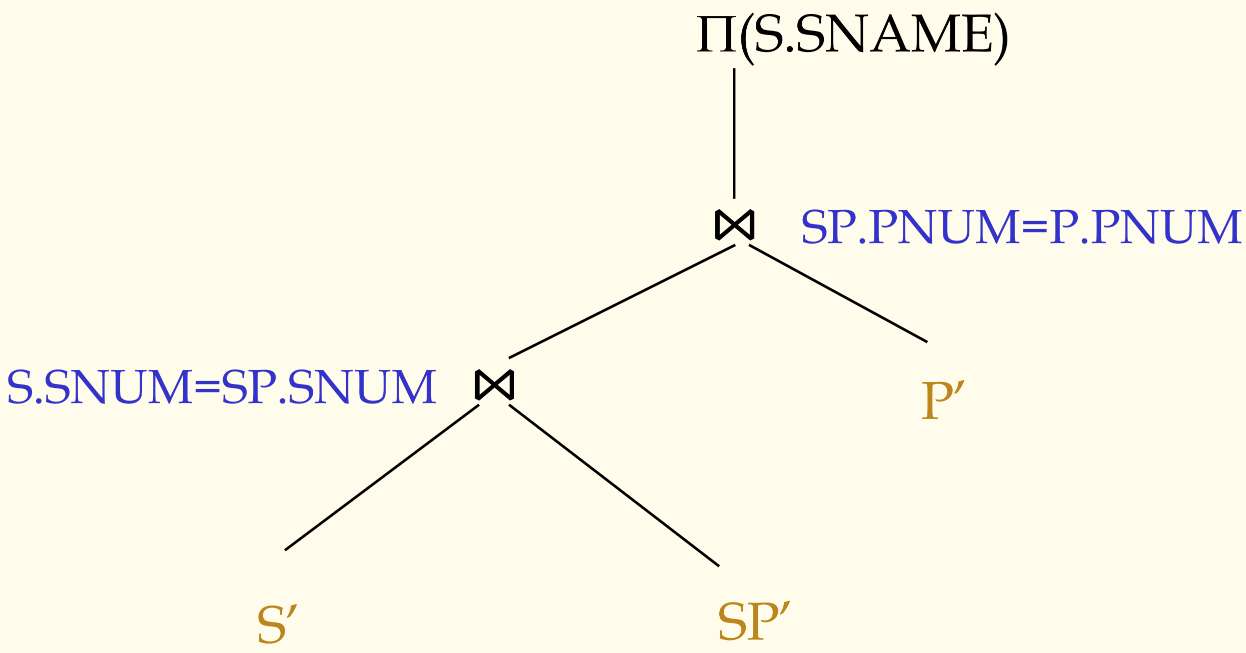


Π1= Π(S.SNUM,S.SNAME)

Π2= Π(SP.SNUM,SP.PNUE)

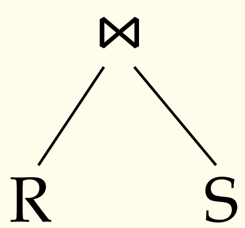
Π3= Π(P.PNUM)

The result of equivalent transform



The operation optimization of the tree :

- Decide the order of two joins
- For every join operation, there are many computing method:



Nest Loop, R as O, S as I
Nest Loop, S as O, R as I
Nest Loop, use possible index on R or S
Merge Scan
Hash Join
⋮

The goal of query optimization is to select a “good” solution from so many possible execution strategies. So it is a complex task.

4.4.2 The Equivalent Transform of a Query

That is so called algebra optimization. It takes a series of transform on original query expression, and transform it into an equivalent, most effective form to be executed.

For example: $\Pi_{NAME,DEPT}\sigma_{DEPT=15}(EMP) \equiv \sigma_{DEPT=15}\,\Pi_{NAME,DEPT}\,(EMP)$

(1) Query tree

For example: $\Pi_{SNUM}\sigma_{AREA='NORTH'}(SUPPLY \bowtie_{DEPTNUM} DEPT)$

Π(SNUM)

σ AREA = ‘North’

$\bowtie$ DEPTNUM=DEPTNUM

Supply DEPT

Leaves: relations

Middle nodes: unary/binary operations

Leaves $\rightarrow$ root: the executing order of operations

(2) The equivalent transform rules of relational algebra

1) Exchange rule of ⋈/×: $E1 \times E2 \equiv E2 \times E1$
2) Combination rule of ⋈/×: $E1\times(E2\times E3)\equiv(E1\times E2)\times E3$
3) Cluster rule of Π: $\Pi_{A1\ldots An}(\Pi_{B1\ldots Bm}(E))\equiv\Pi_{A1\ldots An}(E)$, legal when $A_1\ldots A_n$ is the sub set of $\{B_1\ldots B_m\}$
4) Cluster rule of σ: $\sigma_{F1}(\sigma_{F2}(E))\equiv\sigma_{F1\wedge F2}(E)$
5) Exchange rule of σ and Π: $\sigma_F(\Pi_{A1\ldots An}(E))\equiv\Pi_{A1\ldots An}(\sigma_F(E))$ if F includes attributes $B_1\ldots B_m$ which don't belong to $A_1\ldots A_n$, then $\Pi_{A1\ldots An}(\sigma_F(E)) \equiv \Pi_{A1\ldots An}\,\sigma_F(\Pi_{A1\ldots An\,,\ B1\ldots Bm}(E))$
6) If the attributes in F are all the attributes in E1, then

   $\sigma_F(E1\times E2) \equiv \sigma_F(E1)\times E2$

- if F in the form of F1∧F2, and there are only E1's attributes in F1, and there are only E2's attributes in F2, then $\sigma_F(E1 \times E2) \equiv \sigma_{F1}(E1) \times \sigma_{F2}(E2)$
- if F in the form of F1∧F2, and there are only E1's attributes in F1, while F2 includes the attributes both in E1 and E2, then $\sigma_F(E1 \times E2) \equiv \sigma_{F2}(\sigma_{F1}(E1) \times E2)$

7) $\sigma_F(E1 \cup E2) \equiv \sigma_F(E1) \cup \sigma_F(E2)$
8) $\sigma_F(E1 - E2) \equiv \sigma_F(E1) - \sigma_F(E2)$
9) Suppose $A_1 \ldots A_n$ is a set of attributes, in which $B_1 \ldots B_m$ are E1's attributes, and $C_1 \ldots C_k$ are E2's attributes, then

   $\Pi_{A1 \ldots An}(E1 \times E2) \equiv \Pi_{B1 \ldots Bm}(E1) \times \Pi_{C1 \ldots Ck}(E2)$
10) $\Pi_{A1 \ldots An}(E1 \cup E2) \equiv \Pi_{A1 \ldots An}(E1) \cup \Pi_{A1 \ldots An}(E2)$