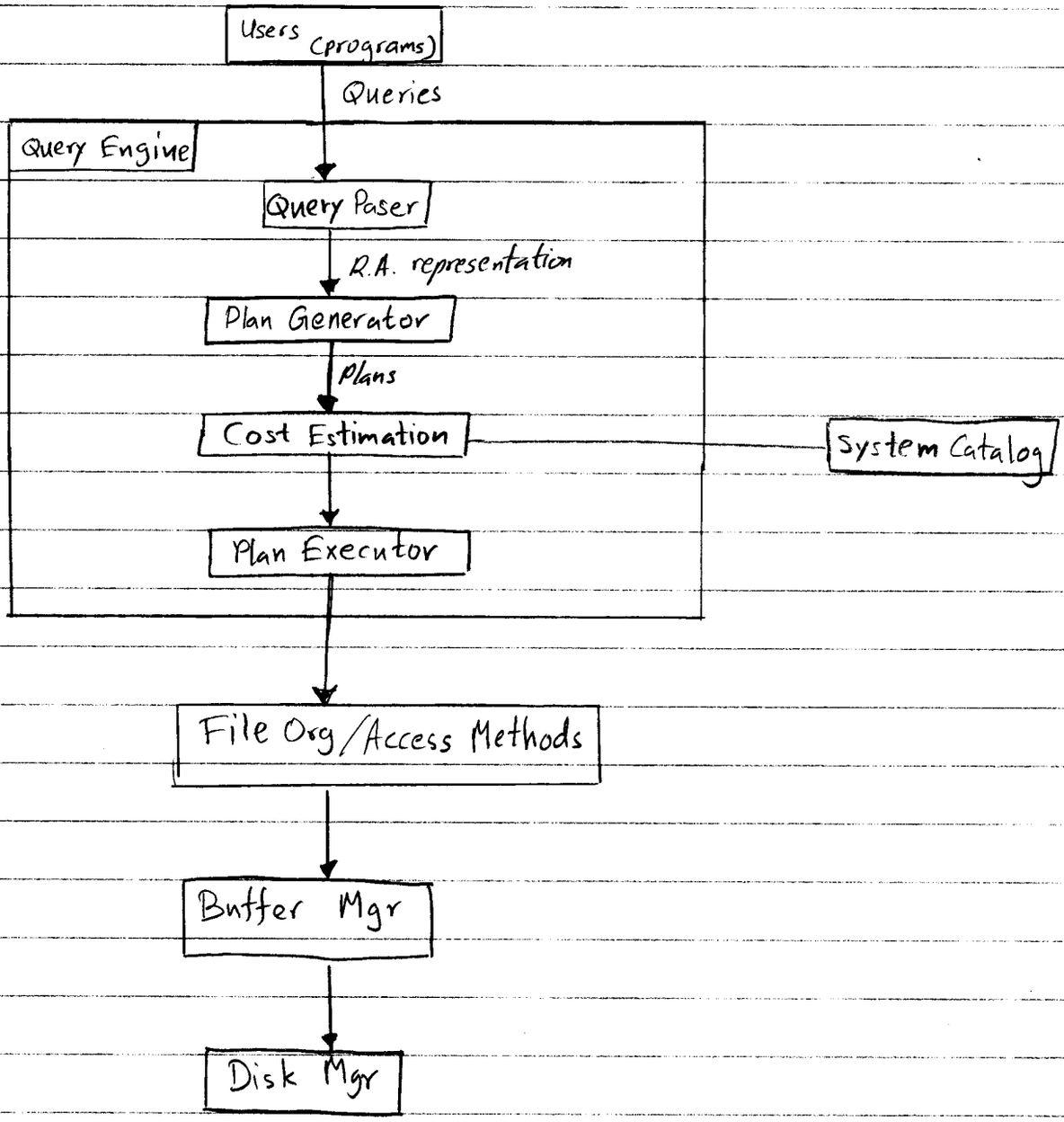
03/11

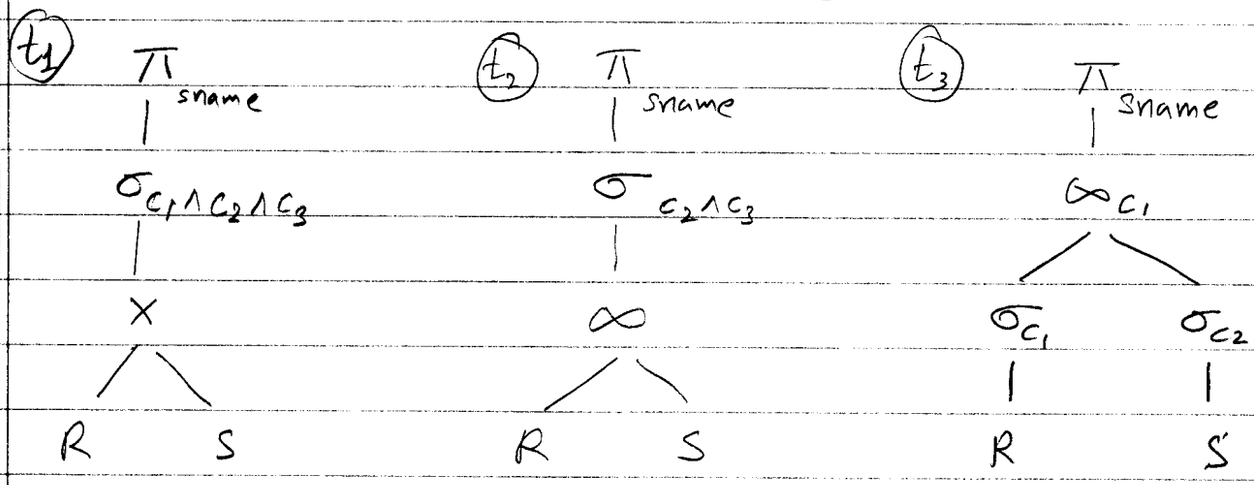


Most of the time we call

Query Optimizer = { Plane Generator + Cost Estimator

```
SELECT  S.sname
FROM    Reservation R, Sailor S
WHERE   R.sid = S.sid AND R.bid = 100 AND S.rating > 5;
```

$$\pi_{S.sname}\left(\sigma_{\underbrace{R.sid=S.sid}_{c_1} \wedge \underbrace{R.bid=100}_{c_2} \wedge \underbrace{S.rating>5}_{c_3}} (R \times S)\right)$$



Plan: a R.A. tree augmented with algorithm choices and processing operations for intermediate results.

Plan Generator

1. See R.A. tree - set of equivalence R.A. trees
2. Augmentation

   putting detail on how the operation are going to be performed.
3. Cost Estimator

   give the augmented trees the cost estimates in the system catalog.

System Catalog

bookkeeping information about the current database.

1. information about database design
2. information about support data, index, views
3. statistics.

system catalog is a most likely collection of DB tables.

example: · pg_index

pg_attributes

Table	Attribute	Type
R	bid	int
R	sid	int
⋮	⋮	⋮
S	sid	int
S	rating	int
⋮	⋮	⋮

pg_statistic

Operator Processing Algorithm

1 $\sigma_{condition}(S)$

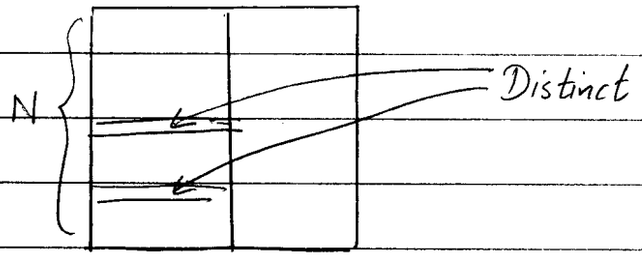
Access Path: a method for retriving tuples from a table.

1) File Scan
2) Index-based
   - 2.1) tree-based
   - 2.2) hash-based

example: index: <a,b,c> and condition: <a,b,c,d,e> is OK for both tree-based and hash-based index. But condition: <a,b> is not OK for hash-based index.

Objective: Find most selective access path (i.e., fewest I/O's)

2. Π condition (S)



Easy: but duplicate elimination may be required

1) sorting $O(N \log N)$

2) hashing ← prefered way.

3. Joins

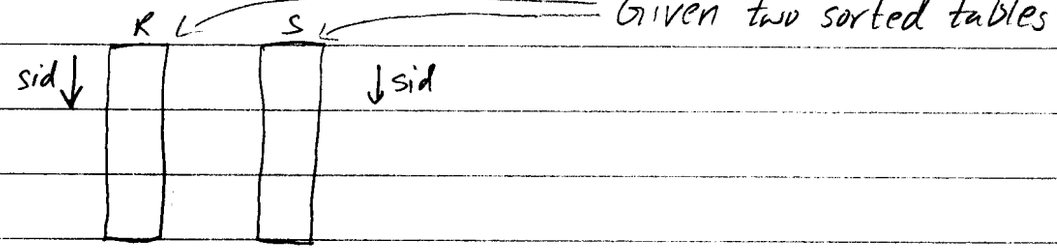
1) Nested-loop (NLJ) $(R \bowtie S)$ — M pages, N pages — Cost = M×N

```
for (r in R)
    for (s in S)
        if (condition == true)
            output tuple
```

2) indexed-NLJ output tuple — constant.

only one loop, so cost is $M \times C$

3) sort-merge join



cost with sorting: $M + N + N\log(N) + M\log(M)$

example: consider t_1, t_2, t_3 from page 2

assume R.bid = 100 and S.rating > 5

⇓ 200 pages ⇓ 20 pages

then

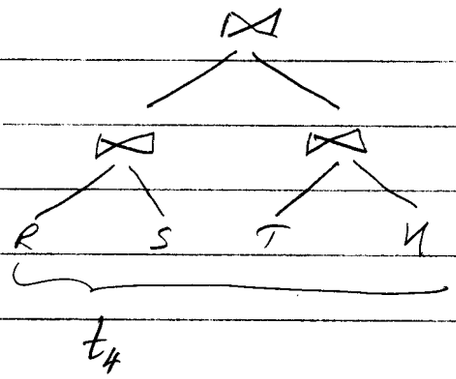
t_1 Cost :

t_2 Cost : $M \times N$

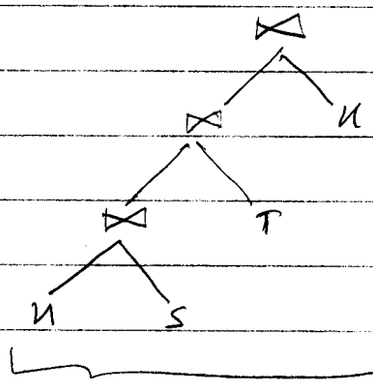
t_3 Cost : $M + N + \frac{M}{200} + \frac{N}{20}$

example:

Consider: $R \bowtie S \bowtie T \bowtie U$



t_4



↳ called pipeline processing is better than t_4