Query Processing kiynne api data ekk retrieve krnn query ekk gahuvama ek database enginerun venne komada kiyla bln ekai.

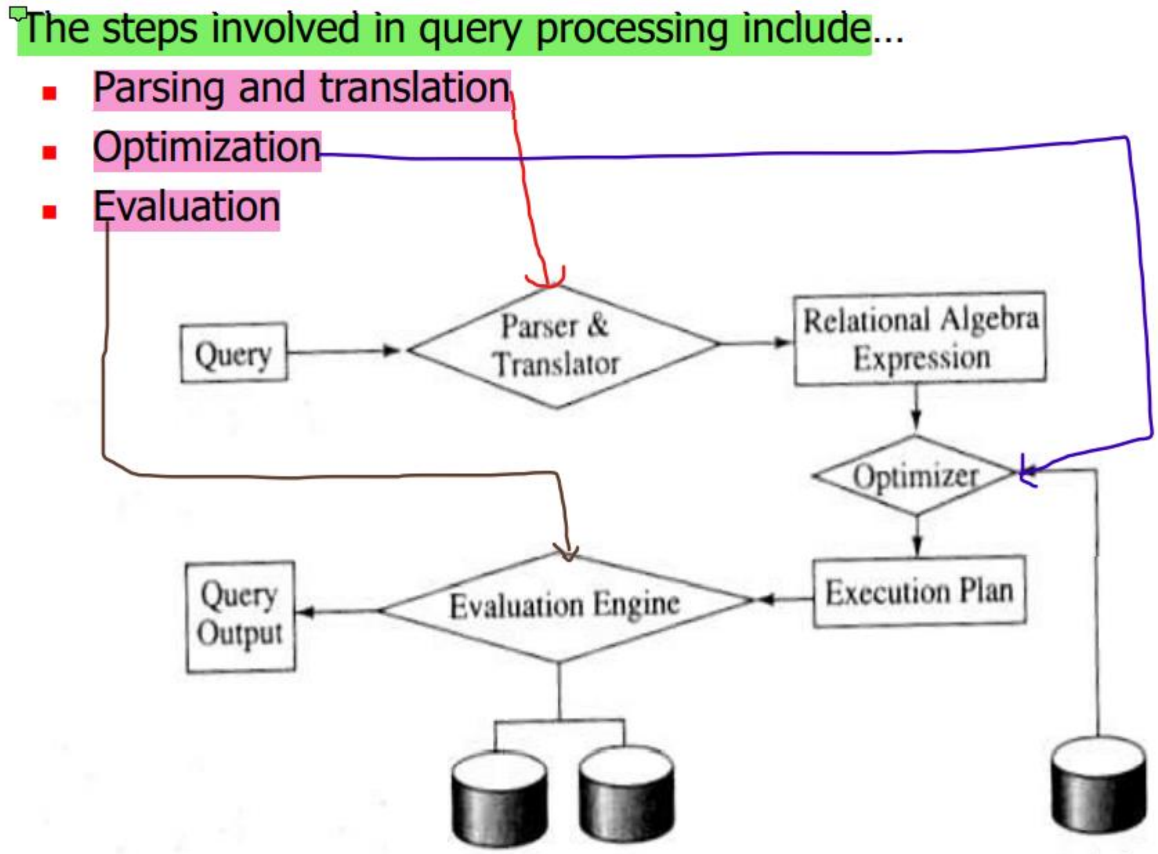
query-> Relational Algebra Expression->Execution Plan->Output

Eka sidda venne memai

Query -> Relational Algebra expression (Mekata use krnne parse and translator use krl)

Relational Algebra expression -> Execution plan (Meka krnne optimizer use krla)

Execution plan -> output (Meka krnne Evaluation engine use krla)



The steps involved in query processing include…

- Parsing and translation
- Optimization
- Evaluation

1. Parse and translator

Query ekk Relational algebra expression ekkt convert krnn tama meka use krnne.

EX: -
SELECT s.sname
FROM S, SP
WHERE S.sno = SP.sno AND
SP.pno = ‘P2’

Table names

1 algebra: π s.name (σ s.sno = sp. Sno and sp.pnp = ‘P2’ (s × sp))

2 algebra: π s.sname ((s) s.sno = sp.sno (σ sp.pno = ‘P2’ (sp)))

Me vge ek query ekkt apita algebra godak hadann puluvn.

Meva Algebra operators ve

Me mark eken kiynne **join the table** krnn kiyn ek-> ×

Me select operation icon eken kiynne **apply the condition** -> σ

Me icon eken kiynne **take out the result** kiyn eka -> π

2. Optimization

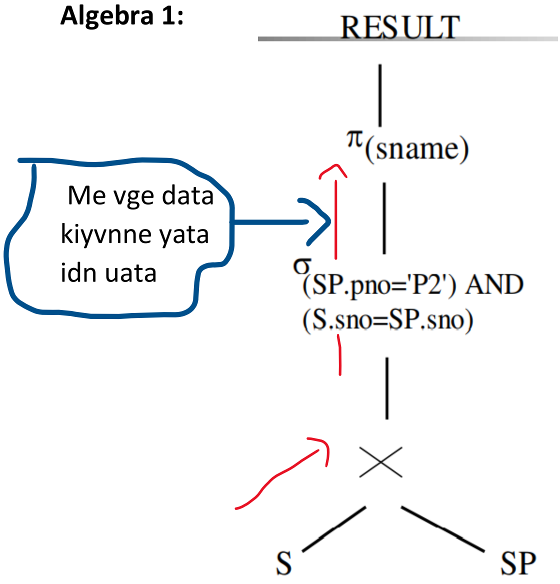
Meke krnne ihata hadagattu algebra vlin best execution plan ek (Hodama eka) toragann eka. Based on the cost we can decide what is the best one. Then me cost ek find krn hati api pahalin blmu.

Attatam database serves ekt time ek yanne hard disk eken data memaory ekt adinn(reading data) ha memory ek data hard disk ekt write krnn(Writing data).

Then I/O cost high nm performance is down, and I/O cost is law performance is high

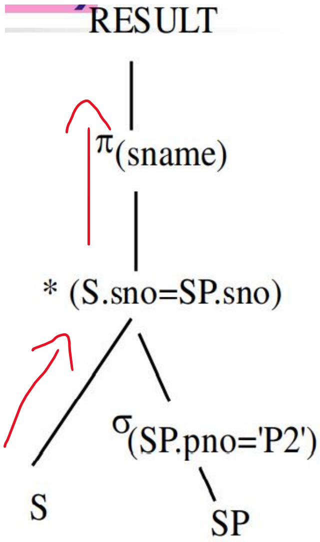
Ihata liypu algebra deke query tree ek pahat tiyenne. Mekt api **Query tree** or **Execution plan** kiyll kiynva

Algebra 1:



-

Algebra: 2



Dn api krnna oni me plan deken mkkd best ek kiyla bln eka.

Dn api blmu meke cost ek estimate krn hati. Assume

S (table) has 100 pages,

SP (table) has 10000 pages,

Algebra 1: Cost = 100*10000 = 1,000,000 read (Meke kiynne me vge seen ekk. Api assume kara S

1,000,000 write

1,000,000 read

Approximate Total cost = 3,000,000

table eke pages 100 ha SP table eke pages 100000 tiyei kiyla. Then Firstly apit hard disk eken data tika memory ekt adinn oni. eyt issellama harddisk eke data tika read krnna 1,000,000 cost ekk ynva and join krddi condition ek satisfied ev write krnn 1,000,000 cost ekk ynva Then mehi condition 2t tiyei ekt data aaai read krnn oni ekt tava 1,000,000 cost ekk ynva. Then ektuva 3,000,000)

Algebra 2: Approximate Total Cost = 10,000+100 = 10,100

Meke krnne issellam SP table eke data tika memro ekt gannav. Yet 10000 cost ekk ynva. E aragen ek join krnn kalin condition ekk blnva. Assume e condition ek satisfied krnne 50 record kiyla. Then e 50 record tika tama memry ek write krnne. Ek small amount ekk nisa api liynne na. meanwhile S table eke data tika adinva eke tiyenne pages 100. Then Total = 10100. Actually, meke enna oni 50*100. But api mehi ganne approximate cost ekk

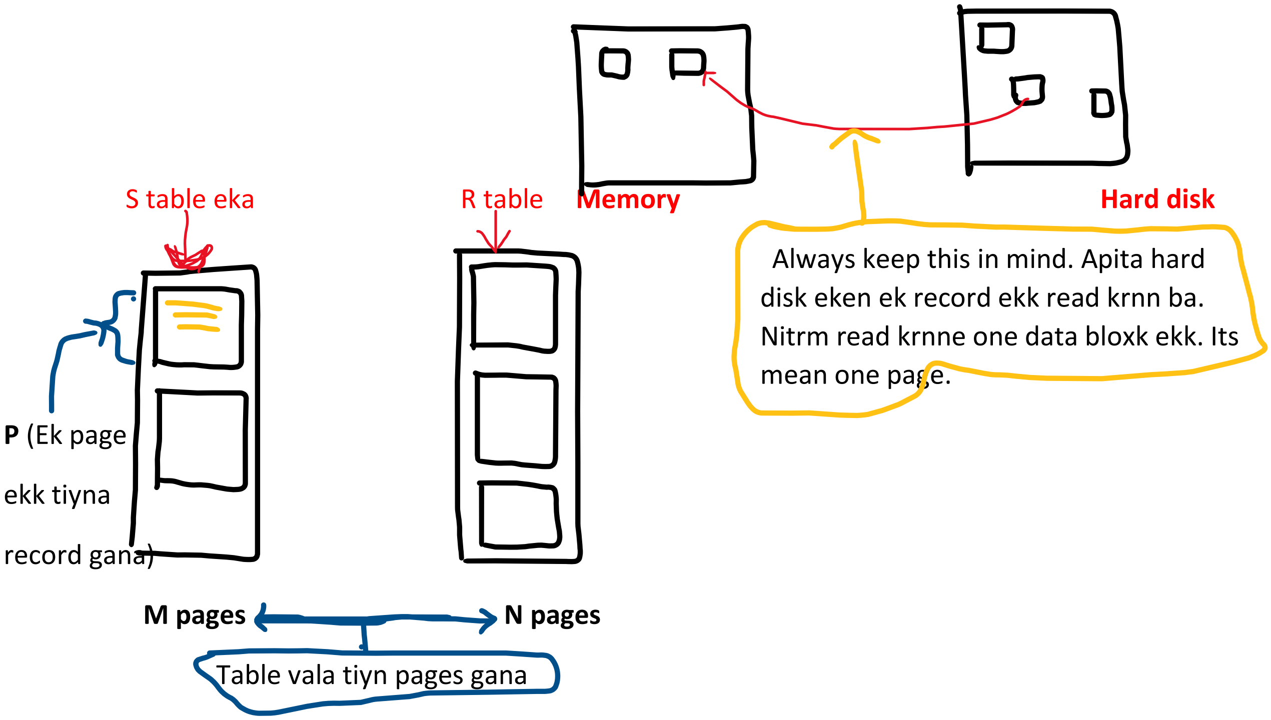
<u>Then, best one is plan 2</u>

Api metana idn kata krnne Joining algorithm kihipayak gana. Inbuild algorithm godak tiyenva. Evgen kihipayak api mehi kata krnva.

1. Simple Nested Loops Join

Meke venne Assume S and R kiyla table deck tiyenava. Then apita ganna oni s.id = r.aid record tika. Ehidi mema join algorithm ek krnne Firstly take the first record of R and check with S table records. The if the condition is satisfied join them. Then aaai R table eke ilaga ek aran S eke okkoma record ekk check krnva. Mema R table eke records okkoma ekin ek aran S table record ekk check krl ek satisfied nm join krnva. Meka thama me Simple nested loops join algorithm eke venne. Then api blmu meke cost ek komada calculate krnne kiyla.

Api meke assume krnva 1 page ekk = 1 disk ekk kiyla



Always keep this in mind. Apita hard disk eken ek record ekk read krnn ba. Nitrm read krnne one data bloxk ekk. Its mean one page.

Cost SNLJ (Single Nested Loops Join) = M + (N * (M*P))

Mehi S eke hama record ekkm aran R ekt ekk match karanava. Then S pages gana gannavan. S page eke gann hama record ekk ganatam R ek fully read krnva. Then S table eke tiyen record gana = M*P. Eke ganatam R table ek pages gana read venva. Then R table ek read vena vara gana = N*(M*P)

Assume Ihata table vala s table eke 5000 pages ha R table eke 20000 page tiyei. And S table eke 50 ek page ekk 50 tuples tiyei. Then,

Cost SNLJ = 5000 + (20000 * (5000 * 50)))

2. Page Oriented Nested Loop Join

Simple nested loop join ekei Page Oriented nested loop join ekei tiyn difference ek nm Simple ek krnne ek record ekk aran ek anit table eke record ekk check krn eka. But page oriented ek krnne single recoed ekk aran check krnva venuvata ek para one Page ekk tiyna okkoma record aran eva anit table ekt ekk compare krl join krna eka.

Cost PONL = M + (N * M)

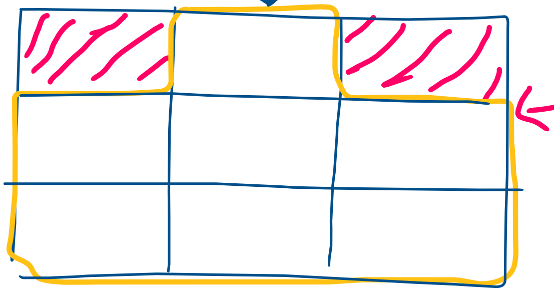
Meke krl tiyenne S table eke page gana aran, Passe R table ek S table eke pages ganat read venva. (Bcz ek para ganne ek page ekk nisa.) Then apit R table ek read ven vara gana N*M lesa gann puluvn

3. Block Nested Loop Join

Block ekk kiynne ste of pages (page kihipayak) kiyn eka. Then mehi krnne Single record ekk Ho Single page ekk record okkoma gannav venuvata ekapara page kihipayak aran(It may be 5 pages) aran anit table ekt ekk check krl join krna ekai

Cost BNL = M + (N * Num of blocks in R table)

Api blmu block size ek gann hati. Me joining operation ek venn Buffer pool ek atule. Ek memory eke tiyenne. Eke frame tiyenne. Ek frame ekk ek page ekk store krgnn puluvn. Then Assume Buffer eke frame 9k tiyenva. Then eken ek frame ekk api ain krnn oni R table eken ena page ek save krgnn. And tava ekk ven krnn oni joining eke result ek tiyagann. Then frame gana B lesa gattot Block size = B-2 venva. Eya pahata adala penvala tiyenne



Block number ek B-2

Then, Number of blocks = [M/ b-2]. Kotu varahan damme nattan enne decimal value. That's why put [].

Cost BNL = M + (N * [M/B-2])

4. Index Nested Loop Join

Meke krnne mema seen ekk. Attatam api me venkm inner table eke record okkom kiyuveane. Ape scenario ekt anuva R table eke. But meke api okkoma record read krnne na. Assume Apita S and R join krnn oni me condition ek use krl. S.id = R.sid. Then api meke krnne R table eke tiyena sid column ekt index ekk hada gannav. Then assume api S table eke id = 10 record ek find krnva kiyla. Etkota venne api hadagattu index eke 10 tiyeida bll tiyei nm ekt adala record ek hoyagannav(rid use krl). Then apita mehidi inner table eke okkoma record kiyvnn avashya na

Index cost ek depend venva index ek type ek matha. Ek B+tree ekk nm cost = 2-4 and ek Hash ekk nm cost = 1.2 ve

Cost INDJ = M + ((index cost) * (Num of record in S))

M + (( ) + (P*M))

Depend on the index type

5. Sorting Merge Join

Meke sorting algorithm gana iassarahata krnva qwa. Meke saralavam venne assume apit S and R table join krgnn oni S.id = R.sid column use krl. Then mehi issellama krnne S table ek id column ekt anuva sort krgnn eka ha R table ek sid column ekt anuva join krgnn eka. Then passe check krl join krnva.

Cost SMJ = (Sorting Cost) + (Merging Cost)

(Sorting Cost) + (M+N)

Mek gana iassarahat krnva. Komada sorting ekk cost ek calculate krnne kiyla

S table eke pages ha R table eke page vala ektuva