

Advance Concepts in DBMS

By

Anand Joshi

Indexes (Indices)

- Primary mechanism to get performance improved on a database.
- Persistent data structure, stored in database.
- Indexes are used underneath by the query execution engine.

	Animal	Stock	...
1	Cat	2	...
2	Dog	5	...
3	Cow	1	...
4	Dog	9	...
5	Cat	2	...
6	Cat	8	...
7	Cow	6	...

Farmville

```
SELECT Animal , Stock  
From Farmville  
WHERE Animal = 'Cow'
```

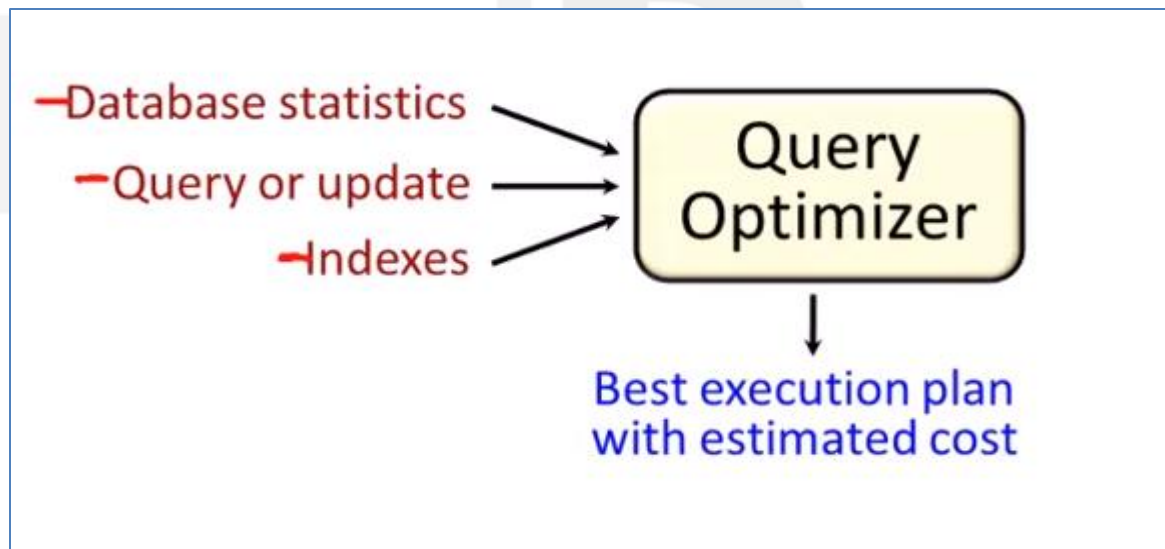
- It will get the result by looping through the animal column and matching the required string with the other entries.
- Isn't that time consuming ?
- Imagine how long it would take if you had 20 million tuples !
- And the worst part what if such queries got frequent ?
- The solution is quite simple and elegant !
- Index the column on which matching condition takes place !

- Index is quite similar to the index of your book.
- The following index will be made.

Animals	index
Cow	3, 7
Dog	2, 4
Cat	1, 5, 6

- **CREATE INDEX INDEX_NAME ON farmville(Animal)**
- Similarly indexes can be made on any other columns too.
- Underlying datastructures are
 - Balanced trees (B , B+ Trees).
 - Hash tables.

- Many DBMS build indexes automatically on Primary Key or Candidate keys.
- Choosing an appropriate index is very crucial as it can greatly affect your server response.
- There are tools available to find best design for your database.



Constraints

- Impose restrictions on allowable data, beyond those imposed by structure and type.
- Used to catch errors while entering a data.
- E.g. CET Score ≤ 200 etc

Triggers

- “ When an **event** occurs , check **condition** , if true , do **action** “
- E.g. enrollment > 200 is an event , when it becomes true , reject all the new applications.
- E.g. aggregate > 99% is an event , when it is true , hire the candidate
- Why do we need triggers ?
 - Move monitoring logic from apps into DBMS
 - To enforce constraints.

Referential Integrity

DEP_ID	DEP_NAME	DEP_HEAD
D001	Marketing	John S.
D002	Accounts	Tony D.

DEPARTMENT

EMP_ID	NAME	ADDRESS	DEPT_ID
E001	Robert	Shanghai	D001
E002	Polly W.	Beijing	D001
E003	David J.	Nanjing	D002
E004	Nelson G.	Shanghai	D002

EMPLOYEE

- What happens if we delete row with id D001 from DEPARTMENT table ?
- The entries in EMPLOYEE table pointing to D001 will be dangling !
- In this case we say that it has lost Referential Integrity.

- Referential Integrity is also called Foreign Key Constraint.
- Solution to maintain referential integrity.
 - Restrict (default)
 - SET NULL (in reference tables)
 - Cascade (delete/update reference tuples)

Views

- Virtual tables.
- Hide some data from the user.
- Views lie at the top (view level) of abstraction layers.
- Syntax :

CREATE VIEW Vname(A1,A2.....An) AS <normal sql query>

- They help in speeding up your queries.

Denormalisation

Order_no	Product_id	qty
101	P1	2
102	P3	1
103	P1	1
104	P2	3
105	P2	2

Orders

Product_id	description	Cost
P1	superheroes	20
P2	Toy Guns	10
P3	Soft toys	12

Products

- The shopkeeper uses following query to calculate the total sales of his shop.

```
SELECT SUM( (cost*qty) + ( 0.10*cost*qty ) )  
FROM Orders INNER JOIN Product  
ON Orders.Product_Id = Products.Product_id
```

- The preceding query includes 10% sales tax in product cost too.
- The above query will be very slow for larger databases as the JOIN process would be very convolved.
- Let us see the magic of Denormalisation here !

- Denormalised table

Order_no	Product_id	Qty	Product cost	tax	Order cost
101	P1	2	40	8	88
102	P3	1	12	1.2	13.2
103	P1	1	20	2	22
104	P2	3	30	9	99
105	P2	2	20	4	44

- Now the query will be

```
SELECT SUM ( Order_Cost )  
FROM Orders
```

Summary

- Constraints are used to impose restriction on your Database.
- Referential Integrity is also called Foreign Key Constraint.
- Triggers get activated only when certain event becomes true. They are used to impose constraints.
- There are 3 ways to speed up your query processing
 - Indexes
 - Denormalisation
 - Views

Contact Info

- trainers@finaldesk.com
- rishabh@finaldesk.com
- nilesh@finaldesk.com
- jignesh@finaldesk.com
- yash@finaldesk.com
- anand@finaldesk.com