Indexes

or

Row reference information

Last few pages of any book

Concept of an Index

An index contains a pointer for each table row (rowid)

To access the data rows quickly

Different Index scans

Unique scan	1 record
Range Scan	More than 1 record
skip scan	
fast full scan	count (*)
index-join	more than 1 table


Bitmap Index

BITMAP INDEX

For column that has a low cardinality
few distinct values

Examples *Male / female*
Yes / No
Good / bad
Colours

- *Reverse Key Index*

Function based index:

```
Create index emp_idx on emp
(UPPER(ename));
```

Which columns to be indexed

Columns that are often found in the WHERE clause

Columns that are used to join tables.

If query will return less % of the rows

Columns not to Index

Columns that are constantly updated.

Columns that contain a lot of null values.

Columns that have a poor distribution of data

When Oracle does not use index

Not equal operator

< >

Functions on the where clause

When using IS NULL , IS NOT NULL

When Full table scan is cheaper

When Oracle does not use index

Non-unique index

Poor Clustering factor

No statistics

Skewness Problem

Finding index usage

ALTER INDEX <index> MONITORING USAGE. ;

ALTER INDEX <index> NOMONITORING usage ;

Views for Indexes

User_indexes
User_ind_columns
V$object_usage
Index_stats

Views for Indexes

dbms_stats.gather_table_stats(user, 'EMP', cascade=>true);

Analyze index name validate structure ;

Views for Indexes

User_indexes
User_ind_columns
V$object_usage
Index_stats

Index Skip Scan

Index Maintenance

Alter index emp_idx rebuild ;

Why Should I Rebuild the Index

Why should I rebuild Index

Oracle does not delete Index entries of Records that are deleted by the user.

Performance will become worse as oracle may choose Full table scan instead of Index scan

So we rebuild to reclaim the space occupied by the deleted rows

Benefits of Rebuilding Index

Space occupied by the deleted rows is reclaimed

Performance will become better as Oracle Chooses Index scan after rebuild .

When Should I Rebuild Index

Height is > 4

Del_lf_rows is > 20 %

Poor clustering factor

Checking Clustering Factor

Execute dbms_stats.gather_table_stats(scott, 'EMP', cascade=>true);

Cascade is

Gather_table_stats + Gather_index_stats
In a single query

```
Select   blocks,  num_rows    from
        user_tables
Where    table_name = 'BTREETABLE';
```

```
Select  index_name,  blevel,  clustering_factor
    from        user_indexes
where table_name = 'BTREETABLE'
```

Blevel is Height

Blevel ---Btree level

"Good" CF --- If CF is closer to Blocks in table

"Bad" CF --- If CF is closer to Rows in table