Benchmarking

# Results

By using the V$SQL Table we were able to benchmark the average time spent on an individual query. We tested two queries: a query that selected all tuples, and a query with a where clause. Our table had 4999 entries. It contained a primary key attribute as well as a high cardinality attribute (SALARY) and a low cardinality attribute (GENDER). Each of these queries were ran and benchmarked with both the SALARY and GENDER attributes having both B-Tree and Bitmap indexes. We also did each query without an index as a control. The results are below in milliseconds.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Milliseconds* | Benchmark | High Cardinality Index | High Cardinality  Bitmap Index | Low Cardinality Index | Low Cardinality Bitmap Index |
| SELECT ALL | 7351 | 3890 | 3863 | 3800 | 3661 |
| WHERE | 6820 | 3356 | 48495 | 4315 | 3347 |

# Queries

/\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* BENCHMARK \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT \* FROM EMPLOYEE\_TEST\_DATA;

SELECT AVG(ELAPSED\_TIME) from v$sql

WHERE SQL\_TEXT like 'SELECT \* FROM EMPLOYEE\_TEST\_DATA';

/\* The average of ten queries with no index comes to 7351ms. \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* SELECT WITH DIFF CARDINALITY \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

select \* FROM EMPLOYEE\_TEST\_DATA;

SELECT AVG(ELAPSED\_TIME) from v$sql

WHERE SQL\_TEXT like 'select \* FROM EMPLOYEE\_TEST\_DATA';

CREATE INDEX salary\_idx ON EMPLOYEE\_TEST\_DATA(SALARY);

SELECT AVG(ELAPSED\_TIME) from v$sql

WHERE SQL\_TEXT like 'select \* FROM EMPLOYEE\_TEST\_DATA';

/\* The average of ten queries with an B+-Tree index on high cardinality SALARY

comes to 3890ms. \*/

DROP INDEX salary\_idx;

CREATE INDEX gender\_idx ON EMPLOYEE\_TEST\_DATA(GENDER);

select \* From EMPLOYEE\_TEST\_DATA;

SELECT AVG(ELAPSED\_TIME) from v$sql

WHERE SQL\_TEXT like 'select \* From EMPLOYEE\_TEST\_DATA';

/\* The average of ten queries with B+-Tree index on low cardinality GENDER

comes to 3800ms. \*/

CREATE BITMAP INDEX salary\_idx ON EMPLOYEE\_TEST\_DATA(SALARY);

Select \* From EMPLOYEE\_TEST\_DATA;

SELECT AVG(ELAPSED\_TIME) from v$sql

WHERE SQL\_TEXT like 'Select \* From EMPLOYEE\_TEST\_DATA';

/\* The average of ten queries with an Bitmap index on high cardinality SALARY

comes to 3863ms. \*/

DROP INDEX salary\_idx;

CREATE BITMAP INDEX gender\_idx ON EMPLOYEE\_TEST\_DATA(GENDER);

Select \* FROM EMPLOYEE\_TEST\_DATA;

SELECT AVG(ELAPSED\_TIME) from v$sql

WHERE SQL\_TEXT like 'Select \* FROM EMPLOYEE\_TEST\_DATA';

/\* The average of ten queries with Bitmap index on low cardinality GENDER

comes to 3661ms. \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* WHERE WITH HIGH CARDINALITY \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

DROP INDEX gender\_idx;

SELECT \* FROM EMPLOYEE\_TEST\_DATA where SALARY = 95049.27;

SELECT AVG(ELAPSED\_TIME) from v$sql

WHERE SQL\_TEXT like 'SELECT \* FROM EMPLOYEE\_TEST\_DATA where SALARY = 95049.27';

/\* With no index the average of 10 queries is 6820ms. \*/

CREATE INDEX salary\_idx on EMPLOYEE\_TEST\_DATA(SALARY);

select \* FROM EMPLOYEE\_TEST\_DATA where SALARY = 95049.27;

SELECT AVG(ELAPSED\_TIME) from v$sql

WHERE SQL\_TEXT like 'select \* FROM EMPLOYEE\_TEST\_DATA where SALARY = 95049.27';

/\* With a B-Tree index the average of 10 queries is 3356ms; \*/

DROP INDEX salary\_idx;

CREATE BITMAP INDEX salary\_idx ON EMPLOYEE\_TEST\_DATA(SALARY);

select \* from EMPLOYEE\_TEST\_DATA where SALARY = 95049.27;

SELECT AVG(ELAPSED\_TIME) from v$sql

WHERE SQL\_TEXT like 'select \* from EMPLOYEE\_TEST\_DATA where SALARY = 95049.27';

/\* With a Bitmap index the average of 10 queries is 48494ms. \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* WHERE with low cardinality \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

DROP INDEX salary\_idx;

CREATE INDEX gender\_idx on EMPLOYEE\_TEST\_DATA(GENDER);

SELECT \* FROM EMPLOYEE\_TEST\_DATA where GENDER like 'Male';

SELECT AVG(ELAPSED\_TIME) from v$sql

WHERE SQL\_TEXT like 'SELECT \* FROM EMPLOYEE\_TEST\_DATA where GENDER like ''Male''';

/\* With a B-Tree index the average of 10 queries is 4315ms. \*/

DROP INDEX gender\_idx;

CREATE BITMAP INDEX gender\_idx ON EMPLOYEE\_TEST\_DATA(GENDER);

select \* from EMPLOYEE\_TEST\_DATA where GENDER like 'Male';

SELECT AVG(ELAPSED\_TIME) from v$sql

WHERE SQL\_TEXT like 'select \* from EMPLOYEE\_TEST\_DATA where GENDER like ''Male''';

/\* With a Bitmap index the average of 10 queries is 3347ms; \*/