*** In this example, the ALBUM_ID in the BIG_DWH_TABLE_2 has a very bad Clustering Factor

```
SQL> SELECT index_name, leaf_blocks, clustering_factor FROM user_indexes WHERE index_name
like 'BIG_DWH%ALBUM_ID_I';

INDEX_NAME                     LEAF_BLOCKS CLUSTERING_FACTOR
------------------------------ ----------- -----------------
BIG_DWH_2_ALBUM_ID_I                  2090            989933
BIG_DWH_ALBUM_ID_I                    2090              4948


SQL> SELECT * from big_dwh_table_2 WHERE album_id IS NOT NULL ORDER BY album_id;

1000000 rows selected.


Execution Plan
----------------------------------------------------------
Plan hash value: 1154224976

-----------------------------------------------------------------------------------------
| Id  | Operation          | Name            | Rows  | Bytes |TempSpc| Cost (%CPU)| Time     |
-----------------------------------------------------------------------------------------
|   0 | SELECT STATEMENT   |                 |  1000K|    28M|       |  9300   (1)| 00:01:52 |
|   1 |  SORT ORDER BY     |                 |  1000K|    28M|    91M|  9300   (1)| 00:01:52 |
|*  2 |   TABLE ACCESS FULL| BIG_DWH_TABLE_2 |  1000K|    28M|       |  1112   (2)| 00:00:14 |
-----------------------------------------------------------------------------------------
```

*** With the bad CF, Oracle picked the FTS when retrieving all rows from the table, where previously with the BIG_DWH_TABLE example, it had used the index

```
SQL> SELECT * from big_dwh_table_2 WHERE album_id BETWEEN 1 AND 1000 ORDER BY album_id;

100000 rows selected.


Execution Plan
----------------------------------------------------------
Plan hash value: 1154224976

-----------------------------------------------------------------------------------------
| Id  | Operation          | Name            | Rows  | Bytes |TempSpc| Cost (%CPU)| Time     |
-----------------------------------------------------------------------------------------
|   0 | SELECT STATEMENT   |                 |   100K|  2929K|       |  1928   (2)| 00:00:24 |
|   1 |  SORT ORDER BY     |                 |   100K|  2929K|  9432K|  1928   (2)| 00:00:24 |
|*  2 |   TABLE ACCESS FULL| BIG_DWH_TABLE_2 |   100K|  2929K|       |  1108   (2)| 00:00:14 |
-----------------------------------------------------------------------------------------
```

*** Oracle still fancies the FTS, even when selecting just 10% of the rows with the poor CF

```
SQL> SELECT * from big_dwh_table_2 WHERE album_id BETWEEN 1 AND 100 ORDER BY album_id;

10000 rows selected.


Execution Plan
----------------------------------------------------------
Plan hash value: 1154224976

-----------------------------------------------------------------------------------------
| Id  | Operation          | Name            | Rows  | Bytes |TempSpc| Cost (%CPU)| Time     |
-----------------------------------------------------------------------------------------
|   0 | SELECT STATEMENT   |                 | 10001 |   292K|       |  1192   (2)| 00:00:15 |
|   1 |  SORT ORDER BY     |                 | 10001 |   292K|   952K|  1192   (2)| 00:00:15 |
|*  2 |   TABLE ACCESS FULL| BIG_DWH_TABLE_2 | 10001 |   292K|       |  1107   (2)| 00:00:14 |
-----------------------------------------------------------------------------------------
```

*** Still fancies the FTS with just 1% of data ...

```
SQL> SELECT * from big_dwh_table_2 WHERE album_id BETWEEN 1 AND 15 ORDER BY album_id;

1500 rows selected.


Execution Plan
----------------------------------------------------------
Plan hash value: 1154224976

-------------------------------------------------------------------------------------
| Id  | Operation          | Name            | Rows  | Bytes | Cost (%CPU)| Time     |
-------------------------------------------------------------------------------------
|   0 | SELECT STATEMENT   |                 |  1500 | 45000 |  1108   (2)| 00:00:14 |
|   1 |  SORT ORDER BY     |                 |  1500 | 45000 |  1108   (2)| 00:00:14 |
|*  2 |   TABLE ACCESS FULL| BIG_DWH_TABLE_2 |  1500 | 45000 |  1107   (2)| 00:00:14 |
-------------------------------------------------------------------------------------


*** Still fancies the FTS with just 0.15% of data ...




SQL> SELECT * from big_dwh_table_2 WHERE album_id BETWEEN 1 AND 11 ORDER BY album_id;

1100 rows selected.


Execution Plan
----------------------------------------------------------
Plan hash value: 1165252589

-----------------------------------------------------------------------------------------------------
| Id  | Operation                    | Name                  | Rows  | Bytes | Cost (%CPU)| Time     |
-----------------------------------------------------------------------------------------------------
|   0 | SELECT STATEMENT             |                       |  1100 | 33000 |  1096   (1)| 00:00:14|
|   1 |  TABLE ACCESS BY INDEX ROWID| BIG_DWH_TABLE_2       |  1100 | 33000 |  1096   (1)| 00:00:14|
|*  2 |   INDEX RANGE SCAN          | BIG_DWH_2_ALBUM_ID_I  |  1100 |       |     5   (0)| 00:00:01|
-----------------------------------------------------------------------------------------------------


*** It wasn’t until just 0.11% of data was retrieved that the CBO used the index to retrieve
the data and eliminate the sort operation
```