```
*** Create a table and populate with the ID column initially set to allow
nulls

SQL> CREATE TABLE no_nulls (id NUMBER);

Table created.

SQL> INSERT INTO no_nulls SELECT rownum FROM dual CONNECT BY level <=10000;

10000 rows created.

SQL> COMMIT;

Commit complete.


*** Create an index on the ID column



SQL> CREATE INDEX no_nulls_id_i ON no_nulls(id);

Index created.

SQL> exec dbms_stats.gather_table_stats(ownname=>NULL, tabname=>'NO_NULLS',
estimate_percent=> NULL, method_opt=> 'FOR ALL COLUMNS SIZE 1');

PL/SQL procedure successfully completed.

SQL> set autotrace on


*** Search for all rows where the ID is NULL. There are none, but the CBO
can't know this as the column allows NULLs

*** Therefore a FTS must be performed


SQL> SELECT * FROM no_nulls WHERE id IS NULL;

no rows selected

Execution Plan
----------------------------------------------------------
   0      SELECT STATEMENT Optimizer=CHOOSE (Cost=7 Card=1 Bytes=3)
   1    0   TABLE ACCESS (FULL) OF 'NO_NULLS' (Cost=7 Card=1 Bytes=3)




Statistics
----------------------------------------------------------
          0  recursive calls
          0  db block gets
         23  consistent gets
          0  physical reads
          0  redo size
        152  bytes sent via SQL*Net to client
        235  bytes received via SQL*Net from client
          1  SQL*Net roundtrips to/from client
          0  sorts (memory)
```

```
          0  sorts (disk)
          0  rows processed


*** Now change the column to be NOT NULL and validated. The column can no
longer possibly have any NULL values, it's impossible.


SQL> ALTER TABLE no_nulls MODIFY id NOT NULL;

Table altered.


*** Now search for NULLs and the CBO uses the index to confirm there indeed
are no nulls. Much more efficient.


SQL> SELECT * FROM no_nulls WHERE id IS NULL;

no rows selected


Execution Plan
----------------------------------------------------------
   0      SELECT STATEMENT Optimizer=CHOOSE (Cost=2 Card=1 Bytes=3)
   1    0   INDEX (RANGE SCAN) OF 'NO_NULLS_ID_I' (NON-UNIQUE) (Cost=2
            Card=1 Bytes=3)



Statistics
----------------------------------------------------------
          0  recursive calls
          0  db block gets
          2  consistent gets
          0  physical reads
          0  redo size
        152  bytes sent via SQL*Net to client
        235  bytes received via SQL*Net from client
          1  SQL*Net roundtrips to/from client
          0  sorts (memory)
          0  sorts (disk)
          0  rows processed


*** Now reset the constraint to NOT NULL but this time with NOVALIDATE


SQL> ALTER TABLE no_nulls MODIFY id NULL;

Table altered.

SQL> ALTER TABLE no_nulls MODIFY id NOT NULL NOVALIDATE;

Table altered.


*** The CBO can no longer guarantee no NULL values and so must perform a
FTS
```

```
SQL> SELECT * FROM no_nulls WHERE id IS NULL;

no rows selected

Execution Plan
----------------------------------------------------------
   0      SELECT STATEMENT Optimizer=CHOOSE (Cost=7 Card=1 Bytes=3)
   1    0   TABLE ACCESS (FULL) OF 'NO_NULLS' (Cost=7 Card=1 Bytes=3)



Statistics
----------------------------------------------------------
          0  recursive calls
          0  db block gets
         23  consistent gets
          0  physical reads
          0  redo size
        152  bytes sent via SQL*Net to client
        235  bytes received via SQL*Net from client
          1  SQL*Net roundtrips to/from client
          0  sorts (memory)
          0  sorts (disk)
          0  rows processed


*** Now validate the constraint but drop the index


SQL> ALTER TABLE no_nulls MODIFY id NULL;

Table altered.

SQL> ALTER TABLE no_nulls MODIFY id NOT NULL VALIDATE;

Table altered.


SQL> DROP INDEX no_nulls_id_i;

Index dropped.


*** The CBO requires the index to bypass the FTS in 9i. No index, a FTS is
used even though the CBO knows no data can match the criteria


SQL> SELECT * FROM no_nulls WHERE id IS NULL;

no rows selected

Execution Plan
----------------------------------------------------------
   0      SELECT STATEMENT Optimizer=CHOOSE (Cost=7 Card=1 Bytes=3)
   1    0   TABLE ACCESS (FULL) OF 'NO_NULLS' (Cost=7 Card=1 Bytes=3)
```

```
Statistics
----------------------------------------------------------
          0  recursive calls
          0  db block gets
         23  consistent gets
          0  physical reads
          0  redo size
        152  bytes sent via SQL*Net to client
        235  bytes received via SQL*Net from client
          1  SQL*Net roundtrips to/from client
          0  sorts (memory)
          0  sorts (disk)
          0  rows processed
```