```
*** Create a simple little table, with 1M rows

SQL> CREATE TABLE bowie_test AS SELECT rownum id, 'Bowie' text FROM dual
CONNECT BY LEVEL <=1000000;

Table created.



*** Create a default 8K block size on the very well clustered ID column



SQL> CREATE INDEX bowie_test_8k_i ON bowie_test(id);

Index created.

SQL> exec dbms_stats.gather_table_stats(ownname=>null,
tabname=>'BOWIE_TEST', cascade=>true, estimate_percent=>null,
method_opt=>'FOR ALL COLUMNS SIZE 1');

PL/SQL procedure successfully completed.

SQL> ANALYZE INDEX bowie_test_8k_i validate structure;

Index analyzed.

SQL> SELECT name, height, br_blks, lf_blks, lf_rows FROM index_stats;

NAME                               HEIGHT    BR_BLKS    LF_BLKS    LF_ROWS

------------------------------ ---------- ---------- ---------- ----------

BOWIE_TEST_8K_I                         3          5       2226    1000000



*** Notice the index has a height of 3

*** Flush buffer cache to make the index work as hard as possible



SQL> alter system flush buffer_cache;

System altered.



*** In other session 2, keep an eye on the current values of a few session
statistics for the first session



SQL> select n.name, s.value from v$sesstat s, v$statname n where
s.statistic# = n.statistic# and s.sid = 136 and (n.name = 'CPU used by
this session' or n.name = 'consistent gets' or n.name = 'physical reads');



NAME                                                       VALUE

----------------------------------------------------- ----------
```

```
CPU used by this session                                            5995
consistent gets                                                  4015434
physical reads                                                     17463
```

*** Back in session 1, run the following PL/SQL which will basically read the whole table, a one row scan at a time

```
SQL> set timing on
SQL> declare
  2  v_id  number;
  3  v_text char(5);
  4  begin
  5    for i in 1..1000000 loop
  6      select id, text into v_id, v_text from bowie_test where id = i;
  7   end loop;
  8  end;
  9  /
PL/SQL procedure successfully completed.
Elapsed: 00:00:59.83
```

*** Note it took just under 1 minute to complete

*** Back in Session 2, recapture the session stats to see how they've changed

```
SQL> select n.name, s.value from v$sesstat s, v$statname n where s.statistic# = n.statistic# and s.sid = 136 and (n.name = 'CPU used by this session' or n.name = 'consistent gets' or n.name = 'physical reads');

NAME                                                    VALUE
---------------------------------------------- ----------
CPU used by this session                             11092 (+ 50.97 secs)
consistent gets                                    8017659 (+ 4,002,225)
physical reads                                       21903 (+ 4,440)
```

```
*** Note that out of the minute elapsed, nearly 51 seconds was CPU
related. It performed just on 4 CRs per execution as expected and just the
400 physical I/Os


*** Back in session 1, a second run with data cached


SQL> /

PL/SQL procedure successfully completed.

Elapsed: 00:00:53.53


*** This time, a little faster at just under 54 seconds

*** Back in session 2

SQL> select n.name, s.value from v$sesstat s, v$statname n where
s.statistic# = n.statistic# and s.sid = 136 and (n.name = 'CPU used by
this session' or n.name = 'consistent gets' or n.name = 'physical reads');


NAME                                                  VALUE

-------------------------------------------- ----------

CPU used by this session                              16072 (+ 49.80 secs)

consistent gets                                    12019884 (+ 4,002,225)

physical reads                                        21903 (0)

*** CPU just a touch lower, CRs the same and no physical I/Os


*****************************


*** Repeat exercise, but this time with an index in a larger block
tablespace ....


SQL> DROP INDEX bowie_test_8K_i;

Index dropped.


*** The index is now in a 16K block tablespace


SQL> CREATE INDEX bowie_test_16k_i ON bowie_test(id) TABLESPACE ts_16k;

Index created.
```

```
SQL> ANALYZE INDEX bowie_test_16k_i validate structure;

Index analyzed.


SQL> SELECT name, height, br_blks, lf_blks, lf_rows FROM index_stats;

NAME                               HEIGHT    BR_BLKS    LF_BLKS    LF_ROWS

------------------------------ ---------- ---------- ---------- ----------

BOWIE_TEST_16K_I                        2          1       1099    1000000


*** Note in this particular example, we have managed to rebuild the index
so that the height has indeed been reduced.

*** Hopefully, performance will improve as a result ...

SQL> alter system flush buffer_cache;

System altered.


*** Back in Session 2

SQL> select n.name, s.value from v$sesstat s, v$statname n where
s.statistic# = n.statistic# and s.sid = 136 and (n.name = 'CPU used by
this session' or n.name = 'consistent gets' or n.name = 'physical reads');


NAME                                          VALUE

---------------------------------------- ----------

CPU used by this session                      16259

consistent gets                            12025095

physical reads                                26261


SQL> declare

  2  v_id  number;

  3  v_text char(5);

  4  begin

  5    for i in 1..1000000 loop

  6      select id, text into v_id, v_text from bowie_test where id = i;

  7   end loop;

  8  end;

  9  /
```

```
PL/SQL procedure successfully completed.

Elapsed: 00:01:02.69
```

*** We notice that performance hasn't actually improved as we had hoped. Performance in this particular instance has actually gone a little worse ...

```
SQL> select n.name, s.value from v$sesstat s, v$statname n where
s.statistic# = n.statistic# and s.sid = 136 and (n.name = 'CPU used by
this session' or n.name = 'consistent gets' or n.name = 'physical reads');


NAME                                          VALUE

---------------------------------------- ----------

CPU used by this session                      21381 (+ 51.22 secs)

consistent gets                            15026193 (+ 3,001,098)

physical reads                                29574 (+ 3,313)
```

*** Note that CPU has actually increased a little even though both CRs and PIOs have reduced.

*** Back in session 1, second run (with data cached)

```
SQL> /

PL/SQL procedure successfully completed.

Elapsed: 00:00:55.64
```

*** This time things have improved but it's still worse than the equivalent smaller block run ...

```
SQL> select n.name, s.value from v$sesstat s, v$statname n where
s.statistic# = n.statistic# and s.sid = 136 and (n.name = 'CPU used by
this session' or n.name = 'consistent gets' or n.name = 'physical reads');
```

```
NAME                                          VALUE
---------------------------------------- ----------
CPU used by this session                      26456 (+ 50.75 secs)
consistent gets                            18027291 (+ 3,001,098)
physical reads                                29574 (0)
```

*** CPU has dropped but it's still more than the CPU used by the second run with the index in a smaller block

*** Although the differences were not substantial, the smaller block index outperformed the larger block index in this specific instance even though the larger block index only has a height of 2 ...