*** create a table, populate it with some rows and create PK with a
default Unique index to police it

SQL> CREATE TABLE ziggy (id NUMBER, name VARCHAR2(30));
Table created.

SQL> INSERT into ziggy SELECT rownum, 'Bowie' FROM dual CONNECT BY level <=1000;
1000 rows created.

SQL> COMMIT;
Commit complete.

SQL> ALTER TABLE ziggy ADD PRIMARY KEY (id);
Table altered.

SQL> exec dbms_stats.gather_table_stats(ownname=>NULL, tabname=>'ZIGGY',
estimate_percent=> NULL, method_opt=> 'FOR ALL COLUMNS SIZE 1');
PL/SQL procedure successfully completed.

*** In one session, run the following a couple of times to ensure no recursive SQL:

SQL> select * from ziggy where id = 10;

ID   NAME
---------- -----------------------------
10   Bowie

*** In other session, run the following (where SID = the sid of the other session) before and after an execution of the select statement in the other session.

SQL> select n.name, s.value from v$sesstat s, v$statname n where s.statistic# = n.statistic# and s.sid = 123 and n.name like 'consistent%';

NAME
---------- -----------------------------
consistent gets 17703
consistent gets from cache 17703
consistent gets - examination 10536
consistent gets direct 0
consistent changes 99

NAME
---------- -----------------------------
-
consistent gets
17706
consistent gets from cache
17706
consistent gets - examination
10539
consistent gets direct
0
consistent changes
99

*** Note that consistent gets increases by 3, consistent gets from cache increases by 3, consistent gets - examination increases by 3 (1 for the index root block, 1 for the index leaf block and 1 for the table block).
*** That's a total of 3 CRs and 3 latches (as all CRs are examinations which only require 1 latch)

*** Now the same thing but with a non-unique index

SQL> ALTER TABLE ziggy DROP PRIMARY KEY;
Table altered.

SQL> ALTER TABLE ziggy ADD PRIMARY KEY (id) USING INDEX
  2  (CREATE INDEX ziggy_id_i ON ziggy(id));
Table altered.

SQL> exec dbms_stats.gather_table_stats(ownname=>NULL, tabname=>'ZIGGY',
  estimate_percent=> NULL, method_opt=> 'FOR ALL COLUMNS SIZE 1');
PL/SQL procedure successfully completed.

*** In one session, run the following a couple of times to ensure no recursive SQL:

SQL> select * from ziggy where id = 10;

    ID   NAME
---------- ------------------------
    10   Bowie

*** In other session, run the following (where SID = the sid of the other session) before and after an execution of the select statement in the other session.

SQL> select n.name, s.value from v$sesstat s, v$statname n where
  s.statistic# = n.statistic# and s.sid = 123 and n.name like 'consistent%';

| NAME   | VALUE       |
|--------------------------|
| consistent gets          | 18504       |
| consistent gets from cache| 18504       |
| consistent gets - examination | 10949      |
consistent gets direct
0
consistent changes
113

NAME
VALUE

<table>
<thead>
<tr>
<th>NAME</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>consistent gets</td>
<td>18508</td>
</tr>
<tr>
<td>consistent gets from cache</td>
<td>18508</td>
</tr>
<tr>
<td>consistent gets - examination</td>
<td>10950</td>
</tr>
<tr>
<td>consistent gets direct</td>
<td>0</td>
</tr>
<tr>
<td>consistent changes</td>
<td>113</td>
</tr>
</tbody>
</table>

*** Note that consistent gets increases by 4 (not 3), consistent gets from cache increases by 4 (not 3), consistent gets - examination increases by only 1 (not 3).

*** In summary, only the root block is acquired via a 1 latch examination CR, the other 3 CRs are "full" 2 latch gets which is a total of 7 latch hits

*** That's 3 latches for the Unique Index and 7 latches for the Non-Unique Index