## Applies to:

Oracle Database - Enterprise Edition - Version 8.0.3.0 and later  
 Information in this document applies to any platform.  
 \*\*\*Checked for relevance on 11-Jan-2016\*\*\*

## Goal

 The purpose of this article is to provide more understanding on the space usage of tables when using LOBS.

## Solution

A select from the BYTES column in DBA\_SEGMENTS for the table shows the table segment but does not include LOB (CLOB or BLOB) segments sizes.  
  
 To caluclate the total size for the table and the associated LOBS segments a sum of the following must occur:  
    
 the bytes for the table         => from dba\_segments  
 +  
 the bytes for the LOB segments  => from dba\_lobs and dba\_segments where segment\_type is LOBSEGMENT  
 +  
 the bytes for the LOB Index (Lob Locator) = from dba\_indexes and dba\_segments

The way to accomplish this is to create a SQL file with the following contents (between the lines):

==================================================

ACCEPT SCHEMA PROMPT 'Table Owner: '  
 ACCEPT TABNAME PROMPT 'Table Name:  '  
 SELECT  
  (SELECT SUM(S.BYTES)                                                                                                 -- The Table Segment size  
   FROM DBA\_SEGMENTS S  
   WHERE S.OWNER = UPPER('&SCHEMA') AND  
        (S.SEGMENT\_NAME = UPPER('&TABNAME'))) +  
  (SELECT SUM(S.BYTES)                                                                                                 -- The Lob Segment Size  
   FROM DBA\_SEGMENTS S, DBA\_LOBS L  
   WHERE S.OWNER = UPPER('&SCHEMA') AND  
        (L.SEGMENT\_NAME = S.SEGMENT\_NAME AND L.TABLE\_NAME = UPPER('&TABNAME') AND L.OWNER = UPPER('&SCHEMA'))) +  
  (SELECT SUM(S.BYTES)                                                                                                 -- The Lob Index size  
   FROM DBA\_SEGMENTS S, DBA\_INDEXES I  
   WHERE S.OWNER = UPPER('&SCHEMA') AND  
        (I.INDEX\_NAME = S.SEGMENT\_NAME AND I.TABLE\_NAME = UPPER('&TABNAME') AND INDEX\_TYPE = 'LOB' AND I.OWNER = UPPER('&SCHEMA')))  
   "TOTAL TABLE SIZE"  
 FROM DUAL;

==================================================

**CASE STUDY** (on UNIX)

$vi lob\_table\_size.sql  
  
 ACCEPT SCHEMA PROMPT 'Table Owner: '  
 ACCEPT TABNAME PROMPT 'Table Name:  '  
 SELECT  
  (SELECT SUM(S.BYTES)                                                                                                 -- The table segment size  
   FROM DBA\_SEGMENTS S  
   WHERE S.OWNER = UPPER('&SCHEMA') AND  
        (S.SEGMENT\_NAME = UPPER('&TABNAME'))) +  
  (SELECT SUM(S.BYTES)                                                                                                 -- The Lob Segment Size  
   FROM DBA\_SEGMENTS S, DBA\_LOBS L  
   WHERE S.OWNER = UPPER('&SCHEMA') AND  
        (L.SEGMENT\_NAME = S.SEGMENT\_NAME AND L.TABLE\_NAME = UPPER('&TABNAME') AND L.OWNER = UPPER('&SCHEMA'))) +  
  (SELECT SUM(S.BYTES)                                                                                                 -- The Lob Index size  
   FROM DBA\_SEGMENTS S, DBA\_INDEXES I  
   WHERE S.OWNER = UPPER('&SCHEMA') AND  
        (I.INDEX\_NAME = S.SEGMENT\_NAME AND I.TABLE\_NAME = UPPER('&TABNAME') AND INDEX\_TYPE = 'LOB' AND I.OWNER = UPPER('&SCHEMA')))  
   "TOTAL TABLE SIZE"  
 FROM DUAL;  
  
 :wq  
  
 $ sqlplus / as sysdba;  
  
 create user test identified by test;  
 grant DBA to test;  
 alter user test default tablespace users;  
 connect test/test;  
  
-- CREATE THE TABLE FOR THE TEST  
  
 CREATE TABLE TEST\_TABLE (a number, b CLOB, c CLOB)  
 LOB(b) STORE AS SECUREFILE (DEDUPLICATE retention none CACHE)  
LOB(c) STORE AS SECUREFILE (DEDUPLICATE retention none CACHE);  
  
-- INSERT ROWS INTO THE TABLE  
  
 declare  
      i number;  
      my\_insert\_b clob;  
      my\_insert\_c clob;  
 begin  
      for i in 1..5000 loop  
           my\_insert\_b := my\_insert\_b||'b';  
          my\_insert\_c := my\_insert\_c||'c';  
      end loop;  
      for i in 1..1000 loop  
           insert into TEST\_TABLE values (i,to\_char(i)||my\_insert\_b, my\_insert\_c);  
      end loop;  
      commit;  
 end;  
 /  
  
-- EXAMINE THE SEGMENTS CREATED BY THE TABLE AND ITS DEPENDENT SEGMENTS  
  
 COLUMN SEGMENT\_NAME FORMAT A32  
 COLUMN COLUMN\_NAME FORMAT A12  
  
 SELECT SEGMENT\_NAME, SEGMENT\_TYPE, BYTES FROM USER\_SEGMENTS;  
  
 -- SEGMENT\_NAME                     SEGMENT\_TYPE            BYTES  
 -- -------------------------------- ------------------ ----------  
 -- TEST\_TABLE                       TABLE                  262144  
 -- SYS\_IL0000069158C00002$$         LOBINDEX               589824  
 -- SYS\_IL0000069158C00003$$         LOBINDEX                65536  
 -- SYS\_LOB0000069158C00002$$        LOBSEGMENT           19070976  
 -- SYS\_LOB0000069158C00003$$        LOBSEGMENT            1245184  
  
 SELECT SUM(BYTES) FROM USER\_SEGMENTS;  
  
 -- SUM(BYTES)  
 -- ----------  
 --   21233664

**-- EXAMINE WHICH LOB SEGMENT BELONGS TO WHICH COLUMN IN TEST\_TABLE**

SELECT COLUMN\_NAME, SEGMENT\_NAME  
 FROM DBA\_LOBS  
 WHERE OWNER = 'TEST' AND TABLE\_NAME = 'TEST\_TABLE'

-- COLUMN\_NAME  SEGMENT\_NAME  
 -- ------------ --------------------------------  
 -- B            SYS\_LOB0000069158C00002$$  
 -- C            SYS\_LOB0000069158C00003$$  
  
 -- THE LOB INDEXES CAN BE ASSOCIATED WITH A COLUMN BY THE NAME OF THE INDEX WITH REGARD TO THE NAME OF THE LOB SEGMENT  
  
  
  
-- EXECUTE THE SCRIPT TO SUM THE LOB STORAGE FOR THE TABLE  
  
 run lob\_table\_size.sql

-- Table Owner: test  
 -- Table Name:  test\_table  
 -- old   4:   WHERE S.OWNER = UPPER('&SCHEMA') AND  
 -- new   4:   WHERE S.OWNER = UPPER('test') AND  
 -- old   5:        (S.SEGMENT\_NAME = UPPER('&TABNAME'))) +  
 -- new   5:        (S.SEGMENT\_NAME = UPPER('test\_table'))) +  
 -- old   8:   WHERE S.OWNER = UPPER('&SCHEMA') AND  
 -- new   8:   WHERE S.OWNER = UPPER('test') AND  
 -- old   9:        (L.SEGMENT\_NAME = S.SEGMENT\_NAME AND L.TABLE\_NAME = UPPER('&TABNAME') AND L.OWNER = UPPER('&SCHEMA'))) +  
 -- new   9:        (L.SEGMENT\_NAME = S.SEGMENT\_NAME AND L.TABLE\_NAME = UPPER('test\_table') AND L.OWNER = UPPER('test'))) +  
 -- old  12:   WHERE S.OWNER = UPPER('&SCHEMA') AND  
 -- new  12:   WHERE S.OWNER = UPPER('test') AND  
 -- old  13:        (I.INDEX\_NAME = S.SEGMENT\_NAME AND I.TABLE\_NAME = UPPER('&TABNAME') AND INDEX\_TYPE = 'LOB' AND I.OWNER = UPPER('&SCHEMA')))  
 -- new  13:        (I.INDEX\_NAME = S.SEGMENT\_NAME AND I.TABLE\_NAME = UPPER('test\_table') AND INDEX\_TYPE = 'LOB' AND I.OWNER = UPPER('test')))  
  
 -- TOTAL TABLE SIZE  
 -- ----------------  
 --         21233664

Explanation  
 -----------  
  
 LOBs over approximately 4000 bytes in length are stored "out of line" These LOBs are assigned to a special LOB segment that is separate from the table segment.

--把&SCHEMA替换成实际的用户

--把&TABNAME替换成实际的表名