**Changing between 32-bit and 64-bit Word Sizes (Doc ID 62290.1)**

SCOPE & APPLICATION  
 -------------------  
  
 This document is created to provide all the details for changing word size from  
 32bit to 64bit. This document is a "cut/paste" of applicable sections from the   
 Oracle9i Database Migration guide (A96530-02), to quickly provide the needed   
 details and steps to change the word-size.   
  
 This note is applicable to Oracle 8.0.x, Oracle8i, Oracle9i and Oracle10g.  
  
 LIMITATIONS OF USE  
 ------------------  
 This note is not applicable for:  
  
 - databases having JVM installed in an Oracle8i environment, or   
 - Oracle Applications installed in an Oracle8i environment  
 - databases using native compilation.  This assumes that PL/SQL is set to interpreted.  
    
 To migrate these types of database, please check [Note:183649.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&amp;sourceId=62290.1&amp;id=183649.1)   
  
 CHANGING WORD-SIZE  
 ------------------  
 You can change the word-size of your Oracle database server during a migration,  
 upgrade, or downgrade operation. A change in word-size includes the following   
 scenarios:   
  
 You have 32-bit Oracle software installed on 64-bit hardware and want to   
 change to 64-bit Oracle software.   
  
 You have 64-bit Oracle software installed on 64-bit hardware and want to   
 change to 32-bit Oracle software.   
  
 If you are changing word-size during a migration, upgrade, or downgrade   
 operation then no additional action is required. The word-size is changed   
 automatically during any of these operations. However, if you want to change   
 the word-size within the same major release, then follow the instructions in  
 "Changing the Word-Size of Your Current Release" below. For example, if you   
 have the 32-bit version of Oracle release 9.0.1 and you want to switch to the  
 64-bit version of Oracle release 9.0.1, then you must complete this procedure.   
 The following information applies if you are changing your   
 hardware from 32-bit to 64-bit or from 64-bit to 32-bit:   
  
 If you want to change your hardware wordsize, then you should be able to switch   
 from 32-bit hardware to 64-bit hardware and still use your existing  
 32-bit Oracle software without encountering any problems, except on Linux  
 systems (32-bit Oracle on 64-bit Linux is not supported).  Always check to be  
 sure the combination is certified to run Oracle before proceeding with any   
 changes.  
  
 If you want to change your hardware from 64-bit to 32-bit, then you   
 must first change your Oracle software to 32-bit software before  
 changing your hardware wordsize.   
  
 The on-disk format for database data, redo, and undo is identical for the   
 32-bit and 64-bit installations of Oracle. The only internal structural   
 differences between the 32-bit and 64-bit Oracle installations are the   
 following:   
  
 The compiled format of PL/SQL is different. The instructions for how and   
 when to recompile PL/SQL are provided in the appropriate chapters of  
 the Migration book. The storage format of user-defined types is based on the   
 release of Oracle that created the database. The existing storage format will   
 be converted to the correct format transparently when necessary. User-defined   
 types include object types, REFs, varrays, and nested tables.  
  
 Note: For Oracle 9.2   
  
 In the first release of the migration guide it is said that changing the   
 wordsize during upgrade or migration is not supported.  This is incorrect  
 a documentation bug has been logged for this.  Bug 2590998 explains the  
 error in the documentation.  This has been fixed in the second release of   
 Oracle 9I release 2 (9.2) Migration guide where it is correctly written  
 that changing wordsize during the migration or the upgrade is supported.  
  
 It is recomended to apply the latest patchset BEFORE the wordsize conversion.  
 This would avoid some bugs and also some steps in this note during the wordsize  
 conversion, like [Bug:1867501](https://support.oracle.com/epmos/faces/BugDisplay?parent=DOCUMENT&amp;sourceId=62290.1&amp;id=1867501) and [Bug:1926809](https://support.oracle.com/epmos/faces/BugDisplay?parent=DOCUMENT&amp;sourceId=62290.1&amp;id=1926809).  
  
 CONSIDERATIONS BEFORE PROCEEDING WITH THE ACTIONS BELOW  
 -------------------------------------------------------  
 1) It is necessary to reload OLAP when converting word size due to its dependency   
 on plsql as documented in Note 386990.1.   
  
 2) Normally an upgrade to a newer release will automatically take care of  
 a word size change from 32-bit to 64-bit. However, upgrading 10gR1 to 10gR2 is   
 an exception.  
  
 Please refer to  
 Oracle Database Upgrade Guide  
 10g Release 2 (10.2)  
 Part Number B14238-01  
 [http://download.oracle.com/docs/cd/B19306\_01/server.102/b14238/intro.htm#i1008703](http://download.oracle.com/docs/cd/B19306_01/server.102/b14238/intro.htm" \l "i1008703) Converting Databases to 64-bit Oracle Database Software  
  
 If you are installing 64-bit Oracle Database 10g software but were previously   
 using a 32-bit Oracle Database installation, then the databases will automatically   
 be converted to 64-bit during the upgrade to Oracle Database 10g except when   
 upgrading from Release 1 (10.1) to Release 2 (10.2).  
  
 Note:  
 The process is not automatic for the release 1 to release 2 upgrade, but is   
 automatic for all other upgrades. This is because the utlip.sql script is not   
 run during the release 1 to release 2 upgrade to invalidate all PL/SQL objects.   
 You must run the utlip.sql script with the database in UPGRADE / MIGRATE mode   
 as the last step in the release 10.1 environment, before upgrading to   
 release 10.2.  
  
 3) Bug 5079213: ORA-6544 [56319] DURING UPGRADE FROM 10.1.0.5 32BIT TO 10.2.0.2 64BIT  
  -- For patch upgrades that are changing word size, utlip.sql must be run   
     manually as it is not automatically run as part of the upgrade.  
  
 CHANGING THE WORD-SIZE OF YOUR CURRENT RELEASE  
 ----------------------------------------------  
  
 The instructions in this section guide you through changing the word-size of   
 your current release (switching from 32-bit software to 64-bit software or  
 vice versa).   
  
 Complete the following steps to change the word-size of your current release:   
  
 1. Start SQL\*Plus.   
  
 2. Connect to the database instance AS SYSDBA.   
  
 3. Run SHUTDOWN IMMEDIATE on the database:   
  
    SQL> SHUTDOWN IMMEDIATE  
  
    Issue the command for all instances if you are running Oracle Parallel  
    Server.  
  
 =============================================================================  
    Note:   
  
    NCHAR columns in user tables are not changed during the upgrade.   
    To change NCHAR columns in user tables, see "Upgrade User NCHAR   
    Columns" in the Migration guide.   
  
 =============================================================================  
  
 4. Perform a full backup of the database (optional, but highly recommended)  
  
    See Also:   
  
    Oracle9i User-Managed Backup and Recovery Guide for more information.   
  
 5. If you are using the same Oracle home for your current release and the   
    release to which you are switching, then deinstall your current release  
    using the Oracle Installer. You do not need to deinstall your current   
    release if you are using separate Oracle home directories.   
  
 6. If you currently have a 32-bit installation, then install the 64-bit   
    version of the same release. Or, if you currently have a 64-bit   
    installation, then install the 32-bit version of the same release.   
  
 =============================================================================  
    Note:   
  
    Installation and deinstallation are operating system-specific. For  
    installation and deinstallation instructions, see your  
    Oracle9i operating system-specific installation documentation and   
    the Oracle9i README for your operating system.   
  
    Installation documentation can also be found at technet.oracle.com  
    
 =============================================================================  
  
 7. Copy configuration files to a location outside of the old Oracle home:   
  
    a. If your initialization parameter file resides within the old   
       environment's Oracle home, then copy it to a location outside of the   
       old environment's Oracle home. The initialization parameter file can   
       reside anywhere you wish, but it should not reside in the old  
       environment's Oracle home after you switch to the new release.   
  
    b. If your initialization parameter file has an IFILE (include file)   
       entry and the file specified in the IFILE entry resides within the   
       old environment's Oracle home, then copy the file specified by the   
       IFILE entry to a location outside of the old environment's Oracle   
       home.  The file specified in the IFILE entry has additional   
       initialization parameters. After you copy this file, edit the IFILE   
       entry in the initialization parameter file to point to its new   
       location.   
  
    c. If you have a password file that resides within the old Oracle home,   
       then move or copy the password file to the Oracle9i Oracle home.  
       The name and location of the password file are operating   
       system-specific; for example, on UNIX operating systems, the default  
       password file is ORACLE\_HOME/dbs/orapwsid, but on Windows platforms,   
       the default password file is ORACLE\_HOME\database\pwdsid.ora.   
       In both cases, sid is your Oracle instance ID.   
  
 =============================================================================  
       Note:   
  
       For Oracle9i Real Application Clusters, perform this step on   
       all nodes. Also, if your initdb\_name.ora file resides within   
       the old environment's Oracle home, then move or copy the   
       initdb\_name.ora file to a location outside of the old   
       environment's Oracle home.   
  
 =============================================================================  
  
 8. Change your environment to point at the new 64Bit ORACLE\_HOME.  
  
      Note: Check with platform specific documentation if other env variables   
            need to be changed e.g. LD\_LIBRARY\_PATH   
  
 9. If you are changing the wordsize of an Oracle 8.0, Oracle8i or Oracle9i 9.0.x database  
    then please make the following changes in the 64-bit ORACLE\_HOME/dbs   
    init$ORACLE\_SID.ora file to prepare for the wordsize change:  
  
     aq\_tm\_processes=0     
     job\_queue\_processes=0  
     \_system\_trig\_enabled= false  
  
     Changing the first two parameters will avoid the problems detailed in   
     Bug 1421476 and Bug 1816609  
  
     The last parameter should be set to FALSE for scripts which perform   
     dictionary operations as the objects on which the triggers depend may   
     become invalid or be dropped, causing the triggers to fail and thus   
     preventing the scripts from running successfully.   
  
     See note 149948.1 'IMPORTANT: Set "\_SYSTEM\_TRIG\_ENABLED=FALSE" When   
     Upgrading / Downgrading / Applying Patch Sets' for more info.  
  
    If you are changing the wordsize of an Oracle9i 9.2.0.x or Oracle10g  
    database, go to step 10.  
  
 10. When changing wordsize from a 32-bit Oracle version to a 64-bit Oracle  
     version, Oracle recommends doubling the size of parameters such as:  
  
     SHARED\_POOL\_SIZE  
     SHARED\_POOL\_RESERVED\_SIZE  
     LARGE\_POOL\_SIZE  
  
     This is mainly due to an increase in the size of internal data structures.  
     For an in-depth explanation of this, please see note 209766.1  
     'Memory Requirements of Databases Migrated from 32-bit to 64-bit'  
  
 11. At a system prompt, change to the ORACLE\_HOME/rdbms/admin directory.   
  
 12. Start SQL\*Plus.   
  
 13. Connect to the database instance AS SYSDBA.   
  
 14. If you are changing the wordsize of an Oracle 8.0, Oracle8i or Oracle9i 9.0.x database,  
     run STARTUP RESTRICT:   
  
     SQL> STARTUP RESTRICT  
  
     You may need to use the PFILE option to specify the location of your   
     initialization parameter file.   
  
     If you are changing the wordsize of an Oracle9i 9.2.0.x database, run STARTUP MIGRATE:  
  
     SQL> STARTUP MIGRATE  
  
     If you are changing the wordsize of an Oracle10g database, run STARTUP UPGRADE:  
  
     SQL> STARTUP UPGRADE  
  
 15. Set the system to spool results to a log file for later verification of   
     success:   
  
     SQL> SPOOL catoutw.log  
  
     If you want to see the output of the script you will run on your screen,   
     then you can also issue a SET ECHO ON statement:   
  
     SQL> SET ECHO ON  
  
  
 16. Run utlirp.sql:   
  
     SQL> @$ORACLE\_HOME/rdbms/admin/utlirp.sql  
  
     The utlirp.sql script recompiles existing PL/SQL modules in the format   
     required by the new database. If the version does not include a call to   
     utlrp, then you must manually run utlrp.sql to recompile invalid objects.  
     This script first alters certain dictionary tables. Then, it reloads   
     package STANDARD and DBMS\_STANDARD, which are necessary for using PL/SQL.   
     Finally, it triggers a recompile of all PL/SQL modules, such as packages,   
     procedures, types, and so on.   
  
    ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++  
    Additional Actions for Java:  
  
 When migrating a database from 32 to 64bit (or vice versa) additional actions  
 are required for java.  In theory the format of java shared data objects (SRO)  
 is not compatible between 32 and 64 bit and so these objects need to be dropped  
 and regenerated.  In practice it may be the case prior to release 11 such  
 objects could interoperate but if so this would only be by chance and should  
 not be relied upon.  
  
 The steps to do the regeneration are as follows.  These should be done  
 immediately before running utlirp.  They may take several minutes to complete.  
 They must be done connected as SYS.  
  
 begin  
   update obj$ set status=5 where obj#=(select obj# from obj$,javasnm$   
     where owner#=0 and type#=29 and short(+)=name and   
     nvl(longdbcs,name)='oracle/aurora/rdbms/Compiler');  
   commit;  
   declare  
     cursor C1 is select  
        'DROP JAVA DATA "' || u.name || '"."' || o.name || '"'  
        from obj$ o,user$ u where o.type#=56 and u.user#=o.owner#;  
    
     ddl\_statement varchar2(200);  
     iterations number;  
     previous\_iterations number;  
     loop\_count number;  
     my\_err     number;  
   begin  
     previous\_iterations := 10000000;  
     loop  
       -- To make sure we eventually stop, pick a max number of iterations  
       select count(\*) into iterations from obj$ where type#=56;  
       exit when iterations=0 or iterations >= previous\_iterations;  
       previous\_iterations := iterations;  
       loop\_count := 0;  
       open C1;  
       loop  
         begin  
           fetch C1 into ddl\_statement;  
           exit when C1%NOTFOUND or loop\_count > iterations;  
         exception when others then  
            my\_err := sqlcode;  
            if my\_err = -1555 then -- snapshot too old, re-execute fetch query  
              exit;  
            else  
              raise;  
            end if;  
         end;  
         initjvmaux.exec(ddl\_statement);  
         loop\_count := loop\_count + 1;  
       end loop;  
       close C1;  
     end loop;  
   end;  
   commit;  
   initjvmaux.drp('delete from java$policy$shared$table');  
   update obj$ set status=1 where obj#=(select obj# from obj$,javasnm$   
     where owner#=0 and type#=29 and short(+)=name and   
     nvl(longdbcs,name)='oracle/aurora/rdbms/Compiler');  
   commit;  
 end;  
 /  
  
 create or replace java system  
 /  
  
  
    ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++  
  
  
 17. Locate the version you are migrating from below, and execute the appropriate  
     script:  
  
     - If you are migrating an Oracle 8.0, Oracle8i or Oracle 9i 9.0.x database,  
     run the following script:  
  
     SQL> @$ORACLE\_HOME/rdbms/admin/catalog.sql  
  
     - If you are migrating an Oracle9i 9.2.0.x database, run the following   
     script:  
  
     SQL> @$ORACLE\_HOME/rdbms/admin/catpatch.sql  
  
     - If you are migrating an Oracle10g 10.1.0.x or 10.2.0.x database, run the   
     following script:  
  
     SQL> @$ORACLE\_HOME/rdbms/admin/catupgrd.sql  
  
 =============================================================================  
  
     Note:   
  
     If the patchset level is not being changed (for example, you are  
     migrating a 9.2.0.8 32-bit database to 9.2.0.8 64-bit) then there is no  
     need to run the $ORACLE\_HOME/rdbms/admin/catpatch.sql script or the  
     $ORACLE\_HOME/rdbms/admin/catupgrd.sql script because the data dictionary   
     is already at the correct level.  
  
 =============================================================================  
  
 18. Check the validity of the DBMS\_STANDARD package:  
  
     SQL> select status from dba\_objects  
     where object\_name='DBMS\_STANDARD'  
     and object\_type='PACKAGE'  
     and owner='SYS';  
  
 19. If the package is invalid, recompile it:  
  
     SQL> alter package dbms\_standard compile;  
  
 20. If you are changing the wordsize of an Oracle 8.0, Oracle8i or Oracle 9i 9.0.x database,  
     run the following script:  
  
     SQL> @$ORACLE\_HOME/rdbms/admin/catproc.sql  
  
     If you are changing the wordsize of an Oracle9i 9.2.0.x or Oracle10g  database, no other  
     script needs to be run.  
  
 21. Run the following SQL statement to check for invalid objects:  
  
     SQL> select owner, object\_name, object\_type from dba\_objects  
     where status <> 'VALID';  
  
 22. Turn off the spooling of script results to the log file:   
  
     SQL> SPOOL OFF  
  
     Then, check the spool file and verify that the packages and procedures   
     compiled successfully. You named the spool file in Step 15; the suggested   
     name was catoutw.log. Correct any problems you find in this file (for  
     example, compile any invalid objects)  
  
     If you specified SET ECHO ON, then you may want to SET ECHO OFF now:   
  
     SQL> SET ECHO OFF  
  
 23. If you are changing the wordsize of an Oracle 8.0, Oracle8i or Oracle9i 9.0.x database,  
     disable the restriction on sessions:   
  
     SQL> ALTER SYSTEM DISABLE RESTRICTED SESSION;  
  
 24. Shutdown the database. If you are changing the wordsize of an Oracle 8.0, Oracle8i or   
     Oracle9i 9.0.x database, remove the following parameter from init.ora  
  
     aq\_tm\_processes=0  
     job\_queue\_processes=0  
     \_system\_trig\_enabled=false  
  
  
 The word-size of your database is now changed.   
  
 You can open the database for normal use.  
  
 RELATED DOCUMENTS  
 -----------------  
  
  
[Note:214242.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&amp;sourceId=62290.1&amp;id=214242.1) ORA-600 [17069] while running utlirp.sql converting to 8.1.7.4 64-Bit  
[Note 565773.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&amp;sourceId=62290.1&amp;id=565773.1) Invalid Objects After Removing OLAP or Migration of a  Database to 64 Bit  
[Note 341880.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&amp;sourceId=62290.1&amp;id=341880.1) How to Convert a 32-bit Database to 64-bit Database on Linux  
[Note 752986.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&amp;sourceId=62290.1&amp;id=752986.1) Database Migration With OS Upgrade On Windows Platform   
[Note 757245.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&amp;sourceId=62290.1&amp;id=757245.1) Can you / How to Upgrade RDBMS and Convert From 32-bit to 64-bit Binaries Directly on Linux or Windows based Intel Platforms Using the Database Upgrade Assistant (DBUA)  
[Note 548978.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&amp;sourceId=62290.1&amp;id=548978.1) How To Change Oracle 11g Wordsize from 32-bit to 64-bit.  
  
 Bug 5079213: ORA-6544 [56319] DURING UPGRADE FROM 10.1.0.5 32BIT TO 10.2.0.2 64BIT  
  -- For patch upgrades that are changing word size, utlip.sql must be run   
     manually as it is not automatically run as part of the upgrade.  
  
 Oracle 9i Database Migration Release 2 (9.2) Part Number A96530-01 (HTML) -  
    <http://download.oracle.com/docs/cd/B10501_01/server.920/a96530/toc.htm> Oracle 9i Database Migration Release 1 (9.0.1) Part Number A90191-02 (HTML) -  
    <http://download.oracle.com/docs/cd/A91202_01/901_doc/server.901/a90191/toc.htm> Oracle8i Migration Release 3 (8.1.7) Part Number A86632-01 (HTML) -  
    <http://download.oracle.com/docs/cd/A87860_01/doc/server.817/a86632/toc.htm> Oracle8 Migration Release 8.0 Part Number A58243-01 (HTML) -  
    <http://download.oracle.com/docs/cd/A64702_01/doc/server.805/a58243/toc.htm> Oracle Documentation Master Index -  
    <http://www.oracle.com/technology/documentation/index.html>