|  |  |  |
| --- | --- | --- |
| **Causes and Solutions for ora-600 [kdsgrp1] (Doc ID 1332252.1)** | [IMG_256](https://support.oracle.com/epmos/faces/DocumentDisplay?_afrLoop=342439845004377%26id=1332252.1%26_afrWindowMode=0%26_adf.ctrl-state=blwukjzpj_4%20/o%20To%20Bottom)  [To Bottom](https://support.oracle.com/epmos/faces/DocumentDisplay?_afrLoop=342439845004377&id=1332252.1&_afrWindowMode=0&_adf.ctrl-state=blwukjzpj_4 \\o To Bottom) | IMG_257 |

IMG_258

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **In this Document**   |  |  | | --- | --- | |  | [Purpose](https://support.oracle.com/epmos/faces/DocumentDisplay?_afrLoop=342439845004377&id=1332252.1&_afrWindowMode=0&_adf.ctrl-state=blwukjzpj_4 \\l PURPOSE) |  |  |  | | --- | --- | |  | [Troubleshooting Steps](https://support.oracle.com/epmos/faces/DocumentDisplay?_afrLoop=342439845004377&id=1332252.1&_afrWindowMode=0&_adf.ctrl-state=blwukjzpj_4 \\l TRBLSHOOT) |  |  |  | | --- | --- | |  | [References](https://support.oracle.com/epmos/faces/DocumentDisplay?_afrLoop=342439845004377&id=1332252.1&_afrWindowMode=0&_adf.ctrl-state=blwukjzpj_4 \\l REF) |     **APPLIES TO:**  Oracle Database - Enterprise Edition - Version 10.2.0.4 and later  Information in this document applies to any platform.  **PURPOSE**  This document discusses the ora-600 [kdsgrp1] error, its possible causes and the work around solutions that can be tried.  **TROUBLESHOOTING STEPS**  The ora-600 [kdsgrp1] error is thrown when a fetch operation fails to find the expected row. The error is hit in memory and so may be a memory only error or an error that results from corruption on disk.  This error may indicate (but is not restricted to) any of the following conditions:   * Lost writes * Parallel DML issues * Index corruption * Data block corruption * Consistent read [CR] issues * Buffer cache corruption   A full list of known issues is given in  [Note 285586.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&sourceId=1332252.1&id=285586.1) - ORA-600 [kdsgrp1]  Each bug has a short description that indicates the circumstances where it is hit. The bug list can be shortened by selecting your database release to show only those issues that may affect you.  This issue may be intermittent or it may persist until the underlying disk level corruption is fixed. Intermittent issues are likely to be memory based (however intermittent access to the corruption can be confused with intermittent memory issues).  **Common Work Around Solutions**  If the issue is in memory only we can try to immediately resolve the issue by flushing the buffer cache but remember to consider the performance impact on production systems:  alter system flush buffer\_cache;  If we have an intermittent consistent read issue we can try disabling rowCR which is an optimization to reduce consistent-read rollbacks during queries by setting \_row\_cr=FALSE in the initialization files. However, this could lead to performance degradation of queries. Please check the ratio of the two statistics "RowCR hits"/"RowCR attempts" to determine whether the workaround is to be used.  If this is a result of index corruption then we can drop and rebuild the index. Note that this will require a maintenance window on production systems.  **Root Cause Determination**  Now lets look at how we discover the root cause of the problem: the first step in finding the root cause of this issue is to inspect the generated trace file. The ora-600 will generate both a trace file in the trace directory and an incident file under the incident id within the incident directory.  The top part of the trace file tells us the SQL that was being run when the error was hit:  ----- Current SQL Statement for this session (sql\_id=9mamr7xn4wg7x) -----    This immediately shows us the data objects that were accessed. Searching the trace file for the text string 'Plan Table' will locate the SQL execution plan that is dumped within this trace file. For a persistent issue this allows us to determine which indexes have been accessed and so identify indexes that should be validated to check for block corruption:  SQL> analyze index scott.pk\_dept validate structure online;  Index analyzed.  An other approach we can take is to use the file and block information contained in the trace file. At the top of the trace file we will find information on the block where the corruption was found:  \*\*\* SESSION ID:(3202.5644) 2011-03-19 04:12:16.910  row 07c7c8c7.a continuation at  file# 31 block# 510151 slot 11 not found  This information can be used to identify the object details in dba\_extents:  Select owner, segment\_name, segment\_type, partition\_name,tablespace\_name  From dba\_extents  Where relative\_fno = <file id>  And <block#> between block\_id and (block\_id+blocks-1);  We can then validate this object, for example a table and all it's indexes:  Analyze table scott.dept validate structure cascade online;  Remember that we may be dealing with a permanent corruption that is not located in the object blocks themselves. Examples of this include:   * Dictionary corruption issue from transportable tablespace operations: check dba\_tablespaces to see if the tablespace has been plugged in. * Lost writes in ASM diskgroup mirrors - most likely to be seen when there is heavy IO and disk resync activity. To check this run dbms\_diskgroup.checkfile to detect mirror discrepancies   If analyze reports no corruption then check if there are any chained rows on the table. If these exist then we may have an undetected corruption and the issue should reproduce whenever the SQL is run. Exporting the table will also detect this issue.  If analyze and exporting the table (in the presence of chained rows) both report no errors then this should be considered a consistent read issue.  Once you understand the nature of the problem you can review the list of known bugs and determine which one matches your condition. If you cannot determine which issue is affecting you then open a service request with Oracle Support and upload the RDBMS and ASM (if applicable)instance alert logs for all nodes, any trace and incident files generated and a full description of the nature of the problem.  **REFERENCES**  [NOTE:285586.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&sourceId=1332252.1&id=285586.1) - ORA-600 [kdsgrp1]  [NOTE:28814.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&sourceId=1332252.1&id=28814.1) - Handling Oracle Block Corruptions  [NOTE:411.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&sourceId=1332252.1&id=411.1) - ADR Different Methods to Create IPS Package |