Contention of the SYSTEM Rollback Segment May Occur Due to an Abrupt Increase in Transaction Processing (Doc ID 2515969.1)

In this Document

  Symptoms
  Cause
  Solution

APPLIES TO:

Oracle Database - Enterprise Edition - Version 11.2.0.4 and later
Information in this document applies to any platform.

SYMPTOMS

Contention of the SYSTEM rollback segment may occur due to an abrupt increase in transaction processing.
Looking at the information of Active Session History, we can see that the blocker session is waiting at buffer busy waits
and the object is class = # 15 (undo segment header).

```
SAMPLE_TIME          EVENT                   P1TEXT  P1   P2TEXT   P2    P3TEXT   P3
----------------     ----------------------  ------- ---  -------- ----- -------  ---
2019/1/1 00:15:16    buffer busy waits       file#   1    block#   231   class#   1
2019/1/1 00:15:16    buffer busy waits       file#   1    block#   228   class#   1
2019/1/1 00:15:16    buffer busy waits       file#   1    block#   228   class#   1
2019/1/1 00:15:16    buffer busy waits       file#   1    block#   231   class#   1
2019/1/1 00:15:16    buffer busy waits       file#   1    block#   231   class#   1
2019/1/1 00:15:16    gc buffer busy release  file#   1    block#   128   class#   15  << Blocker
Session
```

CAUSE

If the number of concurrent transactions has suddenly increased and the number of UNDO segments that are in the status
of ONLINE runs short, the system changes the status of the OFFLINE status UNDO segment to ONLINE, and that undo
segment is used.

When changing the UNDO segment from OFFLINE to ONLINE, the process of updating the internal table UNDO$ is done,
using the SYSTEM rollback segment. When the number of transactions rapidly increases,
Waiting for the undo segment header in the SYSTEM rollback segment may occur due to updating of UNDO$ and
transaction processing may be kept waiting.

SOLUTION

It is possible to suppress the conflict of undo segmentation online by setting the following.

  · Set hidden parameter "_rollback_segment_count"

By specifying the hidden parameter "_rollback_segment_count", you can set the minimum number of segments to be
ONLINE at instance startup.

```
alter system set "_rollback_segment_count"=<minimum_number>;
```

Didn't find what you are looking for?