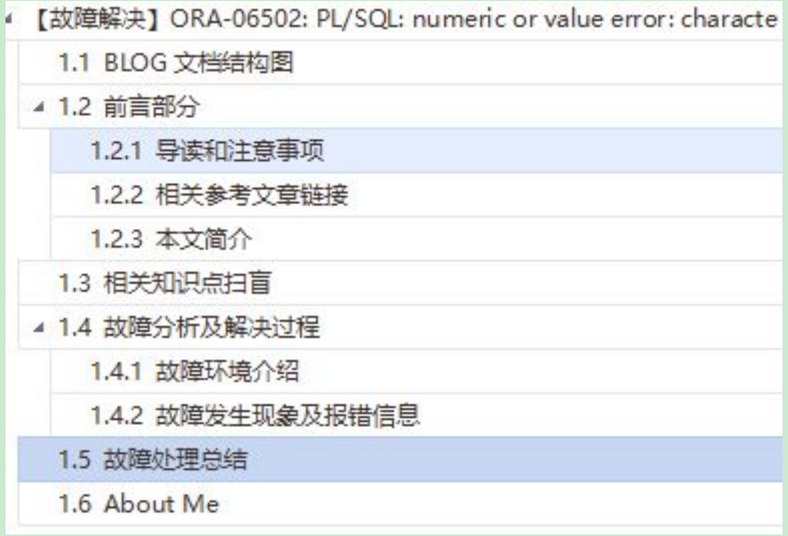
【故障解决】enq: PS - contention

1.1 BLOG 文档结构图



【故障解决】ORA-06502: PL/SQL: numeric or value error: characte
1.1 BLOG 文档结构图
1.2 前言部分
1.2.1 导读和注意事项
1.2.2 相关参考文章链接
1.2.3 本文简介
1.3 相关知识点扫盲
1.4 故障分析及解决过程
1.4.1 故障环境介绍
1.4.2 故障发生现象及报错信息
1.5 故障处理总结
1.6 About Me

1.2 前言部分

1.2.1 导读和注意事项

各位技术爱好者，看完本文后，你可以掌握如下的技能，也可以学到一些其它你所不知道的知识，~O(∩_∩)O~：

① 等待事件 enq: PS - contention 的解决办法

② 一般等待事件的解决办法

Tips：

① 若文章代码格式有错乱，推荐使用 QQ 或 360 浏览器，也可以下载 pdf 格式的文档来查看，pdf 文档下载地址：http://yunpan.cn/cdEQedhCs2kFz（提取码：ed9b）

② 本篇 BLOG 中代码部分需要特别关注的地方我都用黄色背景和红色字体来表示，比如下边的例子中，thread 1 的最大归档日志号为 33，thread 2 的最大归档日志号为 43 是需要特别关注的地方，命令一般使用粉红颜色标注，注释一般采用蓝色字体表示。

```
  List of Archived Logs in backup set 11
  Thrd Seq     Low SCN    Low Time            Next SCN   Next Time
  ---- ------- ---------- ------------------- ---------- ---------
  1    32      1621589    2015-05-29 11:09:52 1625242    2015-05-29 11:15:48
  1    33      1625242    2015-05-29 11:15:48 1625293    2015-05-29 11:15:58
  2    42      1613951    2015-05-29 10:41:18 1625245    2015-05-29 11:15:49
  2    43      1625245    2015-05-29 11:15:49 1625253    2015-05-29 11:15:53



[ZFXDESKDB1:root]:/>lsvg -o
T_XDESK_APP1_vg
rootvg
[ZFXDESKDB1:root]:/>
[ZFXDESKDB1:root]:/>lsvg rootvg


====》 2097152*512/1024/1024/1024=1G
```

本文如有错误或不完善的地方请大家多多指正，ITPUB 留言或 QQ 皆可，您的批评指正是我写作的最大动力。

1.2.2 相关参考文章链接

1.2.3 本文简介

1.3 相关知识点扫盲

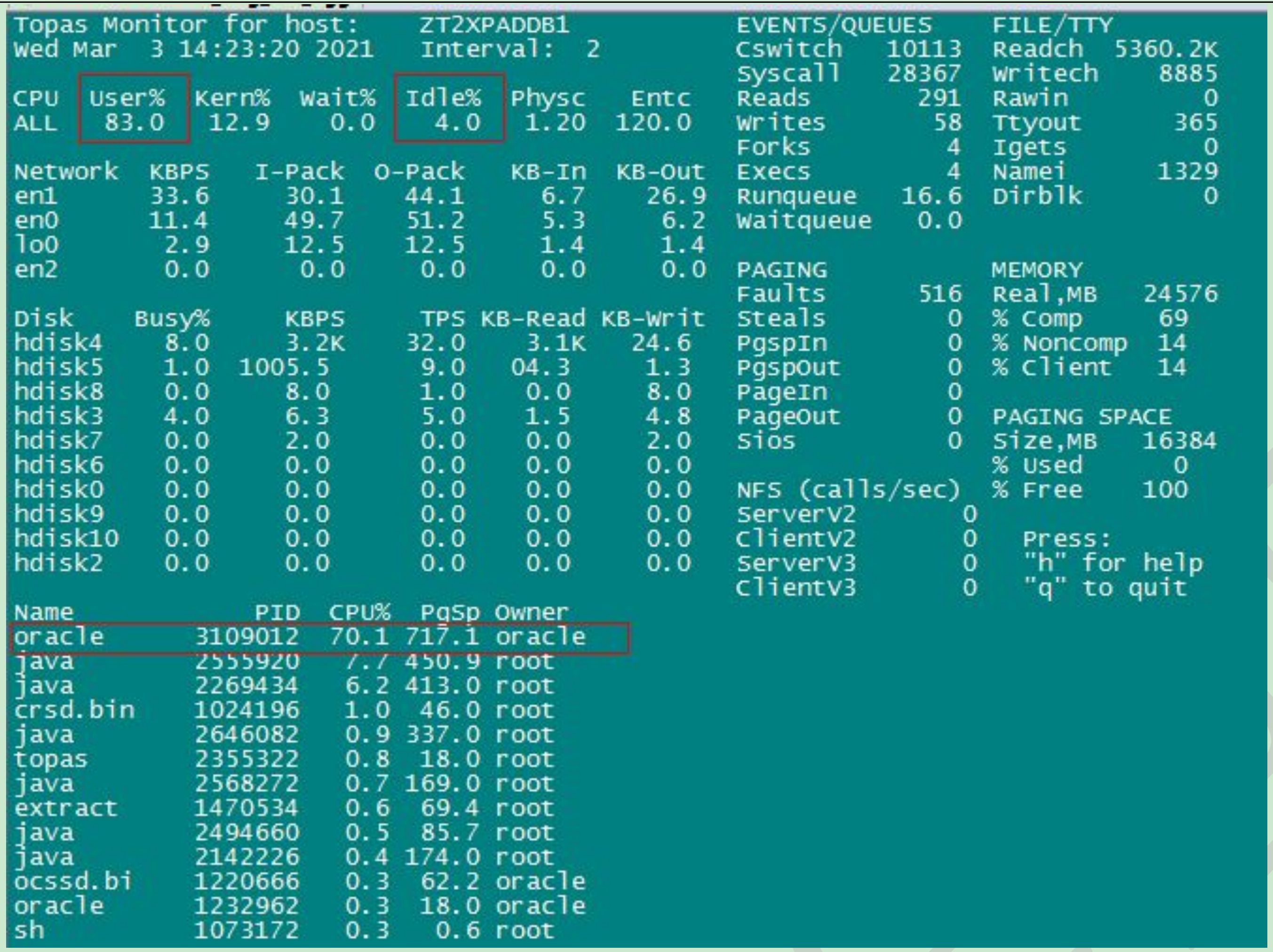
1.4 故障分析及解决过程

1.4.1 故障环境介绍

项目	source db
db 类型	RAC
db version	10.2.0.4.0
db 存储	RAW
ORACLE_SID	XXX
db_name	XXX
主机 IP 地址：	XXX
OS 版本及 kernel 版本	AIX 5.3.0.0
OS hostname	XXX

1.4.2 故障发生现象及报错信息

开发人员反馈数据库很慢，让帮忙查查原因，那首当其冲的就是看主机的情况了，主机是 AIX 系统，采用 TOPAS 查看主机的情况，如下图，从图中可以看出的确有一个 oracle 的进程非常占用 CPU 资源：

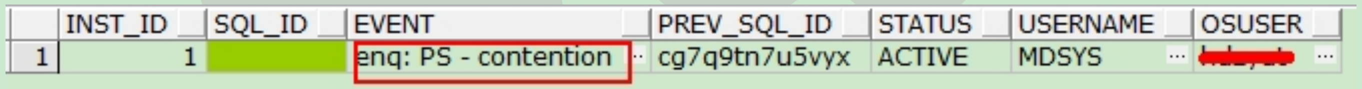


```
Topas Monitor for host:    ZT2XPADDB1          EVENTS/QUEUES    FILE/TTY
Wed Mar  3 14:23:20 2021   Interval:  2        Cswitch   10113  Readch  5360.2K
                                               Syscall   28367  Writech    8885
CPU  User%  Kern%  Wait%  Idle%  Physc  Entc   Reads       291  Rawin         0
ALL   83.0   12.9    0.0    4.0   1.20 120.0   Writes       58  Ttyout      365
                                               Forks         4  Igets         0
Network  KBPS   I-Pack  O-Pack   KB-In  KB-Out Execs         4  Namei      1329
en1      33.6     30.1    44.1     6.7    26.9 Runqueue   16.6  Dirblk        0
en0      11.4     49.7    51.2     5.3     6.2 Waitqueue   0.0
lo0       2.9     12.5    12.5     1.4     1.4
en2       0.0      0.0     0.0     0.0     0.0 PAGING           MEMORY
                                               Faults      516  Real,MB   24576
Disk    Busy%     KBPS     TPS KB-Read KB-Writ Steals        0  % Comp     69
hdisk4    8.0     3.2K    32.0    3.1K   24.6  PgspIn        0  % Noncomp  14
hdisk5    1.0  1005.5      9.0   04.3     1.3  PgspOut       0  % Client   14
hdisk8    0.0      8.0     1.0    0.0     8.0  PageIn        0
hdisk3    4.0      6.3     5.0    1.5     4.8  PageOut       0  PAGING SPACE
hdisk7    0.0      2.0     0.0    0.0     2.0  Sios          0  Size,MB   16384
hdisk6    0.0      0.0     0.0    0.0     0.0                   % Used      0
hdisk0    0.0      0.0     0.0    0.0     0.0  NFS (calls/sec)  % Free    100
hdisk9    0.0      0.0     0.0    0.0     0.0  ServerV2       0
hdisk10   0.0      0.0     0.0    0.0     0.0  ClientV2       0   Press:
hdisk2    0.0      0.0     0.0    0.0     0.0  ServerV3       0   "h" for help
                                               ClientV3       0   "q" to quit
Name            PID  CPU%  PgSp Owner
oracle      3109012  70.1 717.1 oracle
java        2555920   7.7 450.9 root
java        2269434   6.2 413.0 root
crsd.bin    1024196   1.0  46.0 root
java        2646082   0.9 337.0 root
topas       2355322   0.8  18.0 root
java        2568272   0.7 169.0 root
extract     1470534   0.6  69.4 root
java        2494660   0.5  85.7 root
java        2142226   0.4 174.0 root
ocssd.bi    1220666   0.3  62.2 oracle
oracle      1232962   0.3  18.0 oracle
sh          1073172   0.3   0.6 root
```

1.4.3 故障分析及解决过程

根据 os 的进程号到数据库中查看相关的会话：

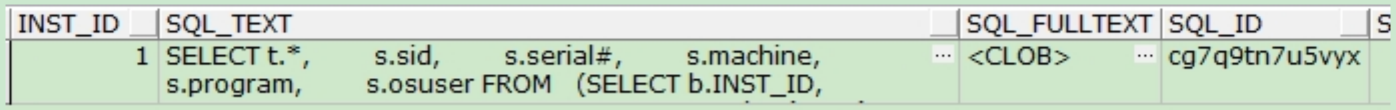
```
SELECT a.INST_ID, a.SQL_ID, a.EVENT, a.PREV_SQL_ID, a.STATUS,a.USERNAME,a.OSUSER
  FROM gv$session a, gv$process b
 WHERE a.PADDR = b.ADDR
   and b.SPID = 3109012;
```



	INST_ID	SQL_ID	EVENT	PREV_SQL_ID	STATUS	USERNAME	OSUSER
1	1		enq: PS - contention	cg7q9tn7u5vyx	ACTIVE	MDSYS	[redacted]

可以看到该会话的等待事件是 enq: PS - contention，并且有相关的 SQL 和 OSUSER，可以联系到当时的开发人员，据说已经跑了 1 个小时了，我们先来看看具体的 sql 内容：

```
SELECT *
  FROM gv$sqlarea a
 WHERE a.SQL_ID = 'cg7q9tn7u5vyx'
   and a.INST_ID = 1;
```



INST_ID	SQL_TEXT	SQL_FULLTEXT	SQL_ID	S
1	SELECT t.*, s.sid, s.serial#, s.machine, s.program, s.osuser FROM (SELECT b.INST_ID,	<CLOB>	cg7q9tn7u5vyx	

SQL 文本 copy 出来：

```
SELECT t.*, s.sid, s.serial#, s.machine, s.program, s.osuser
  FROM (SELECT b.INST_ID,
             c.USERNAME,
             a.event,
             to_char(a.cnt) AS seconds,
             a.sql_id,
             dbms_lob.substr(b.sql_fulltext, 100, 1) sqltext
         FROM (SELECT rownum rn, t.*
                FROM (SELECT s.INST_ID,
                           decode(s.session_state,
                                 'WAITING',
                                 s.event,
                                 'Cpu + Wait For Cpu') Event,
                           s.sql_id,
                           s.user_id,
                           COUNT(*) CNT
                       FROM gv$active_session_history s
                      WHERE sample_time > SYSDATE - 30 / 1440
                      GROUP BY INST_ID,
                             s.user_id,
                             decode(s.session_state,
                                   'WAITING',
                                   s.event,
                                   'Cpu + Wait For Cpu'),
                             s.sql_id
                      ORDER BY CNT DESC) t
               WHERE rownum < 20) a,
             gv$sqlarea b,
             dba_users c
        WHERE a.sql_id = b.sql_id
          AND a.user_id = c.user_id
          AND a.INST_ID = b.INST_ID
        ORDER BY CNT DESC) t,
      gv$session s
 WHERE t.sql_id = s.sql_id(+)
   AND t.INST_ID = s.INST_ID(+)
 ORDER BY t.INST_ID
```

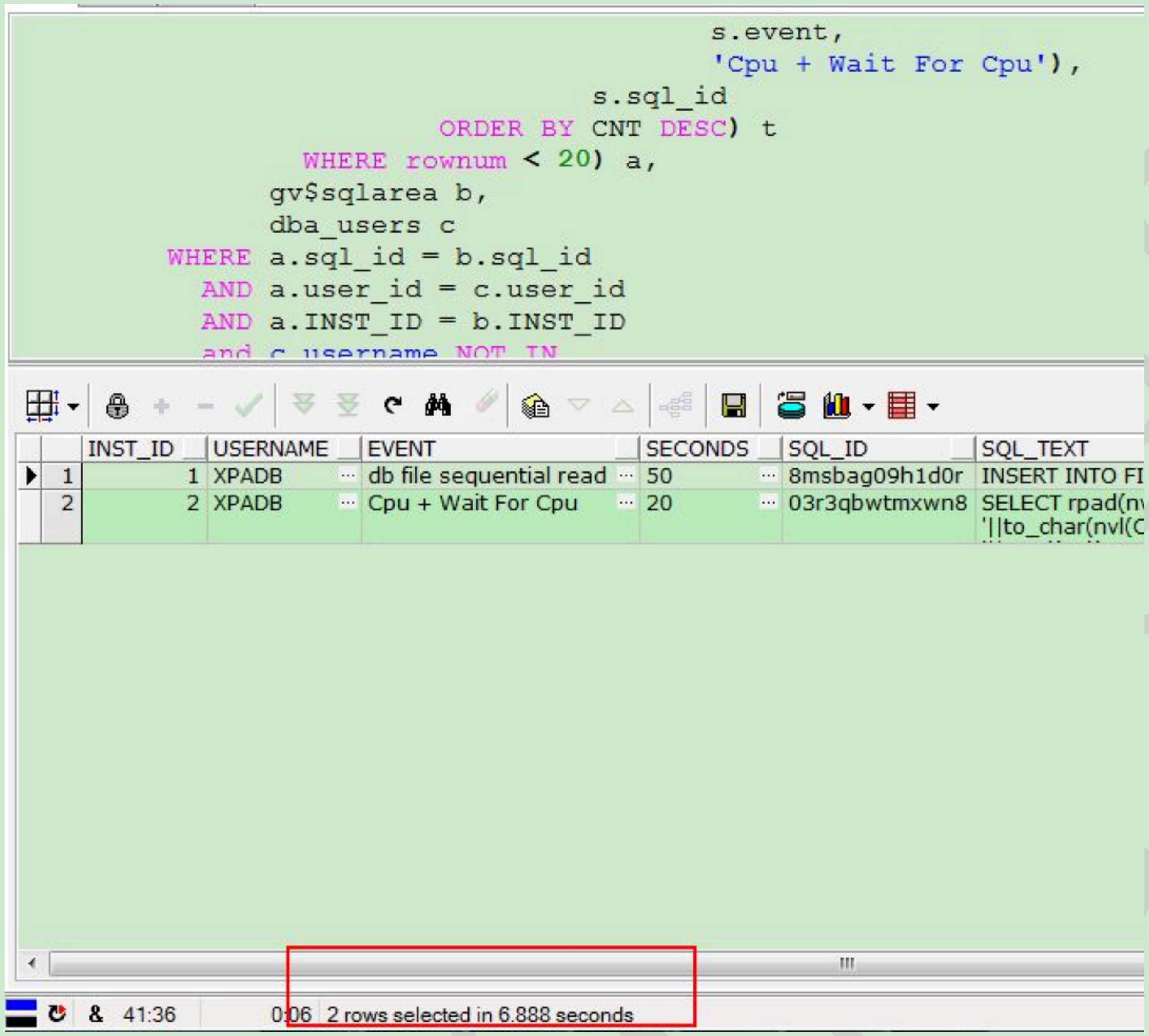
从文本中可以看出该 sql 查询的是数据字典，估计是从网上 copy 过来的，以哥多年的开发经验瞅了一眼就发现一个特殊的地方 `dbms_lob.substr(b.sql_fulltext, 100, 1)`，这类 clob 类型的都比较耗费资源，因为比较忙就不深入的分析了，简单看了下把该句修改为 b.SQL_TEXT，满足要求即可，没有必要去查询 clob。

简单修改后：

```
SELECT t.*, s.sid, s.serial#, s.machine, s.program, s.osuser
  FROM (SELECT b.INST_ID,
             c.USERNAME,
             a.event,
             to_char(a.cnt) AS seconds,
             a.sql_id,
             --dbms_lob.substr(b.sql_fulltext, 100, 1) sqltext ,
             b.SQL_TEXT
         FROM (SELECT rownum rn, t.*
                FROM (SELECT s.INST_ID,
                           decode(s.session_state,
                                 'WAITING',
                                 s.event,
                                 'Cpu + Wait For Cpu') Event,
                           s.sql_id,
                           s.user_id,
                           COUNT(*) CNT
                       FROM gv$active_session_history s
                      WHERE sample_time > SYSDATE - 30 / 1440
                      GROUP BY INST_ID,
                             s.user_id,
                             decode(s.session_state,
                                   'WAITING',
                                   s.event,
                                   'Cpu + Wait For Cpu'),
                             s.sql_id
                      ORDER BY CNT DESC) t
               WHERE rownum < 20) a,
             gv$sqlarea b,
             dba_users c
        WHERE a.sql_id = b.sql_id
          AND a.user_id = c.user_id
          AND a.INST_ID = b.INST_ID
        ORDER BY CNT DESC) t,
      gv$session s
 WHERE t.sql_id = s.sql_id(+)
```

```
   AND t.INST_ID = s.INST_ID(+)
 ORDER BY t.INST_ID;
```

执行一下：



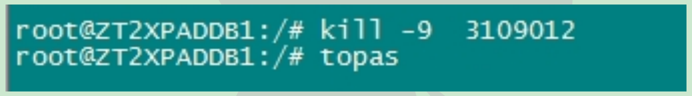
```
                                       s.event,
                                       'Cpu + Wait For Cpu'),
                                s.sql_id
                        ORDER BY CNT DESC) t
                WHERE rownum < 20) a,
              gv$sqlarea b,
              dba_users c
        WHERE a.sql_id = b.sql_id
          AND a.user_id = c.user_id
          AND a.INST_ID = b.INST_ID
          and c.username NOT IN
```

	INST_ID	USERNAME	EVENT	SECONDS	SQL_ID	SQL_TEXT
1	1	XPADB	db file sequential read	50	8msbag09h1d0r	INSERT INTO FI
2	2	XPADB	Cpu + Wait For Cpu	20	03r3qbwtmxwn8	SELECT rpad(n\ '||to_char(nvl(C

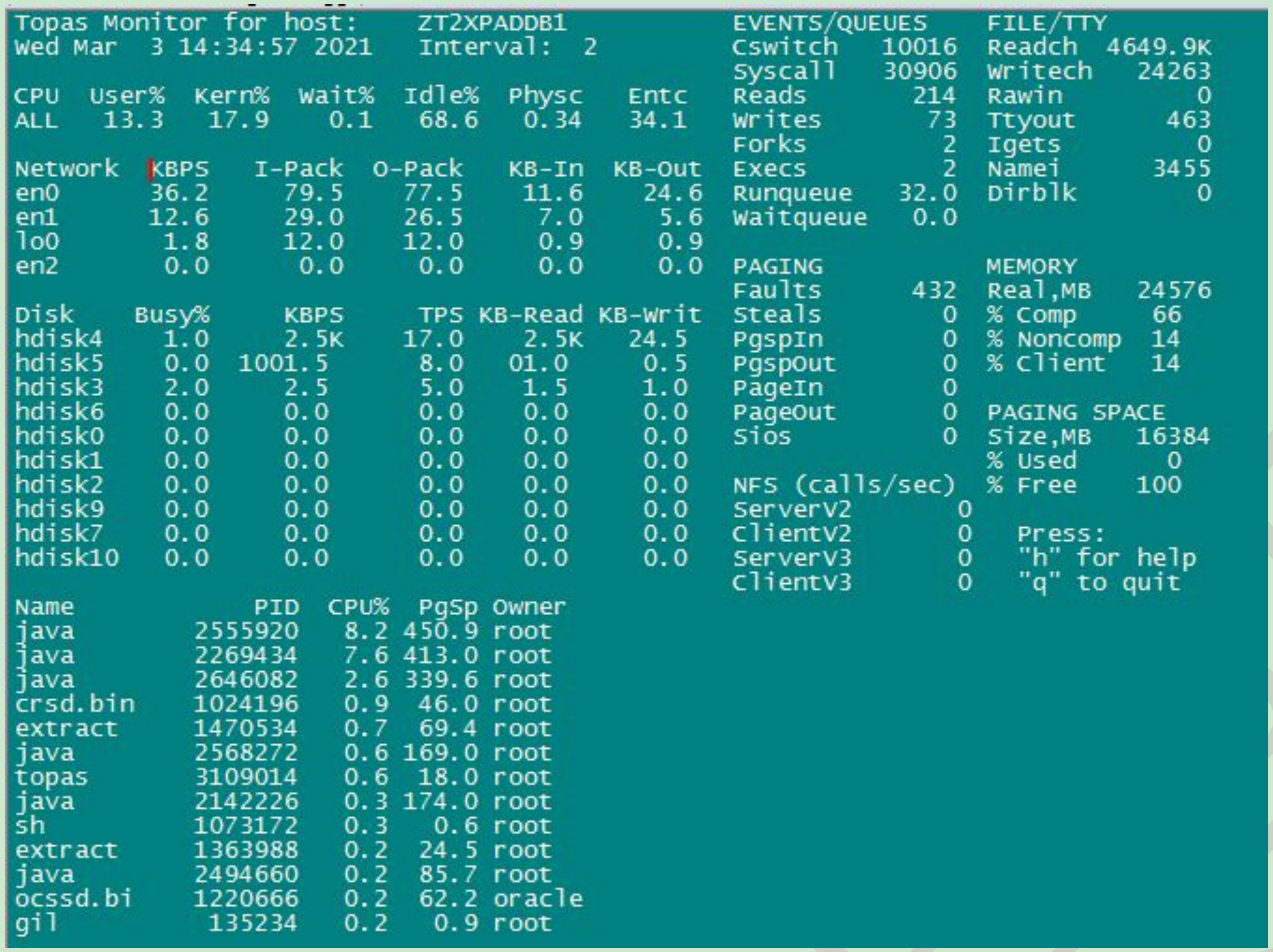
41:36　0:06　2 rows selected in 6.888 seconds

效率还是可以的，从之前的 1 个小时没有跑出来到现在的 6 秒，还是很不错的，主要是需要找出 SQL 中的瓶颈部分，这个就需要经验和多读书、多看报。少吃零食多睡觉了。^_^

下来问了下开发人员说可以停掉的，那我就 kill 掉了，kill 掉后主机的情况如下：



```
root@ZT2XPADDB1:/# kill -9  3109012
root@ZT2XPADDB1:/# topas
```



```
Topas Monitor for host:    ZT2XPADDB1           EVENTS/QUEUES    FILE/TTY
Wed Mar  3 14:34:57 2021   Interval:  2         Cswitch    10016 Readch  4649.9K
                                                Syscall    30906 Writech   24263
CPU  User%  Kern%  Wait%  Idle%  Physc   Entc   Reads        214 Rawin         0
ALL   13.3   17.9    0.1   68.6   0.34   34.1   Writes        73 Ttyout      463
                                                Forks          2 Igets         0
Network  KBPS   I-Pack  O-Pack   KB-In  KB-Out  Execs          2 Namei      3455
en0       36.2     79.5    77.5    11.6    24.6  Runqueue    32.0 Dirblk        0
en1       12.6     29.0    26.5     7.0     5.6  Waitqueue    0.0
lo0        1.8     12.0    12.0     0.9     0.9
en2        0.0      0.0     0.0     0.0     0.0  PAGING           MEMORY
                                                Faults       432 Real,MB   24576
Disk    Busy%     KBPS     TPS KB-Read KB-Writ  Steals         0 % Comp     66
hdisk4    1.0     2.5K    17.0    2.5K   24.5   PgspIn         0 % Noncomp  14
hdisk5    0.0  1001.5      8.0    01.0     0.5   PgspOut        0 % Client   14
hdisk3    2.0     2.5      5.0     1.5     1.0   PageIn         0
hdisk6    0.0     0.0      0.0     0.0     0.0   PageOut        0 PAGING SPACE
hdisk0    0.0     0.0      0.0     0.0     0.0   Sios           0 Size,MB   16384
hdisk1    0.0     0.0      0.0     0.0     0.0                     % Used      0
hdisk2    0.0     0.0      0.0     0.0     0.0   NFS (calls/sec)  % Free    100
hdisk9    0.0     0.0      0.0     0.0     0.0   ServerV2         0
hdisk7    0.0     0.0      0.0     0.0     0.0   ClientV2         0  Press:
hdisk10   0.0     0.0      0.0     0.0     0.0   ServerV3         0  "h" for help
                                                ClientV3         0  "q" to quit
Name            PID  CPU%  PgSp Owner
java        2555920   8.2 450.9 root
java        2269434   7.6 413.0 root
java        2646082   2.6 339.6 root
crsd.bin    1024196   0.9  46.0 root
extract     1470534   0.7  69.4 root
java        2568272   0.6 169.0 root
topas       3109014   0.6  18.0 root
java        2142226   0.3 174.0 root
sh          1073172   0.3   0.6 root
extract     1363988   0.2  24.5 root
java        2494660   0.2  85.7 root
ocssd.bi    1220666   0.2  62.2 oracle
gil          135234   0.2   0.9 root
```

1.4.3.1 metalink 解释

参考：Metalink： Bug 5476091

Description
If a session is waiting on a mutex wait (eg: 'cursor: pin X')
then interrupts to the session are ignored.
eg: Ctrl-C does not have any effect.
This issue can show up as a deadlock in a Parallel Query
between the QC (Query coordinator) and one of its slaves
with the QC waiting on "enq: PS - contention" deadlocked
against the slave holding the requested PS enqueue.

Bug 5476091 - Ctrl-C ignored for sessions waiting for mutexes / Deadlock with "enq: PS" - superceded (文档 ID 5476091.8)

Bug 5476091 Ctrl-C ignored for sessions waiting for mutexes / Deadlock with "enq: PS" - superceded
This note gives a brief overview of bug 5476091.
The content was last updated on: 21-JUL-2015
Click here for det*ails of* each *of the sections below.*
Affects:

Product (*Component*)	Oracle Server (Rdbms)
Range of versions *believed* to be affected	Versions BELOW 11.2
Versions *confirmed* as being affected	• 10.2.0.4 • 10.2.0.3
Platforms affected	Generic (all / most platforms affected)

Note that this fix has been superseded by the fix in Bug:10214450

Fixed:

This fix has been **superseded** - please see the fixed version information for Bug:10214450 . The box below only shows versions where the code change/s for 5476091 are first included - those versions may not contain the later improved fix.

The fix for 5476091 is first included in	• 11.2.0.1 (Base Release) • 11.1.0.7 (Server Patch Set) • 10.2.0.5 (Server Patch Set)

Symptoms:

- Deadlock
- Hang (Process Hang)
- Mutex Contention
- Waits for "cursor: pin X"
- Waits for "enq: PS - contention"

Related To:

- Parallel Query (PQO)

Description

If a session is waiting on a mutex wait (eg: 'cursor: pin X')
then interrupts to the session are ignored.
eg: Ctrl-C does not have any effect.

This issue can show up as a deadlock in a Parallel Query
between the QC (Query coordinator) and one of its slaves
with the QC waiting on "enq: PS - contention" deadlocked
against the slave holding the requested PS enqueue.

Note:

This fix is superceded by the fix in bug 10214450

> **Please note:** The above is a summary description only. Actual symptoms can vary. Matching to any symptoms here does not confirm that you are encountering this problem. For questions about this bug please consult Oracle Support.

References

Bug:5476091 (This link will only work for PUBLISHED bugs)

Note:245840.1 Information on the sections in this article

1.5 故障处理总结

到此所有的处理算是基本完毕，过程很简单，但是不同的场景处理方式有很多种，我们应该学会灵活变通。

1.6 About Me

....

本文作者：小麦苗，只专注于数据库的技术，更注重技术的运用

ITPUB BLOG：http://blog.itpub.net/26736162

本文地址：http://blog.itpub.net/26736162/viewspace-1985380/

本文pdf版：http://yunpan.cn/cdEQedhCs2kFz （提取码：ed9b）

QQ：642808185 若加 QQ 请注明您所正在读的文章标题

于 2016-01-28 10:00~ 2016-01-28 19:00 在中行完成

<版权所有，文章允许转载，但须以链接方式注明源地址，否则追究法律责任!>

....