

等待事件类

“TX,TM,DX” 锁应急处理

现象描述:

数据库大量锁异常等待，系统资源消耗高，cpu 负载高 （针对大量TX,TM,DX等类型的锁造成的大量异常等待）

影响因素:

多个事务争用造成。

解决方法

以下语句列出是谁造成了阻塞

```
column event format a30
column sess format a20
set linesize 250
set pagesize 0
break on id1 skip 1
select decode(request,0,'Holder:', 'Waiter:') || s.inst_id || ':' || s.sid||','|| s.serial# sess,
       id1, id2, lmode, request, l.type, ctime, s.username,s.sql_id, s.event
-- ,s.service_name
from gv$lock l, gv$session s
where (id1, id2, l.type) in
      (select id1, id2, type from gv$lock where request>0
       )
and l.sid=s.sid
and l.inst_id=s.inst_id
order by id1, ctime desc, request
/
```

按照这个语句多查询几次，如果 Holder 不变，则 KILL 掉。操作前记录相关日志

“Latch free” 应急处理

现象描述:

数据库大量 latch free 等待，系统资源消耗高，cpu 负载接近 100%

影响因素:

解决方法:

1.手动执行 hang 查杀脚本:

/usr/bin/ksh /home/oracle/auto_hang_analyze.sh

观察几分钟，如果 latch free 等待事件一直没有下降，则执行第二步。

2.查询当前 active 的会话模块:

```
select username,machine,count(*) from v$session where status='ACTIVE' having count(*)>6  
group by username,machine order by 1;
```

将会话数量过多的模块通知开发商，让他们切换部分业务到另外一个节点

然后进行系统资源监控和数据库监控

“Cache buffer chains” 应急处理

现象描述:

数据库大量 cache buffer chains 等待，系统资源消耗高，cpu 负载高

影响因素:

A.低效的 SQL 语句是发生 cache buffers chains(热块争用)，锁存器争用的最重要原因。

B.多个进程同时扫描大范围的索引或表时，可能广泛引发 cache buffers chains 锁存器争用。

C.应用程序打开执行相同的低效率 SQL 语句的多个并发会话，并且这些 SQL 语句都设法得到相同的数据集，这种情景十分普遍。

解决方法:

1.查看 latch: cache buffers chains 事件相关的会话信息;

```
select sid,username,machine,program,p1raw,sql_id,logon_time,last_call_et from v$session where event='latch: cache buffers chains';
```

使用 ora 命令 ora get_kill_sh &sql_id &username 进行查杀.

查杀后记录该 sql 语句,丢给相应的开发商处理

2、查看哪个 SQL 执行的次数最多

```
select sql_id,count(*) from v$session where event='latch: cache buffers chains' group by sql_id order by 2;
```

“Library cache lock” 应急处理

现象描述:

数据库大量 library cache lock 等待, 系统资源消耗高, cpu 的 idle 为 0

影响因素:

library cache lock 出现的情况比较复杂, 例如:

A、大量对某个对象访问;

B、shared pool 有问题;

解决方法:

1、看看是不是某条 SQL 引起

```
select sql_id,count(*) from v$session where event='library cache lock' group by sql_id order by 2;
```

然后分析 SQL 中的对象和执行计划等, 再跟开发商确认, 用 ora get_kill_sh 进行杀

2、shared pool 的内部结构造成, 再开一个窗口用 topas 监控系统资源, 然后清理 shared pool

alter system flush shared pool; (该操作需要向直属领导确认)

“gc buffer busy” 应急处理

现象描述:

一般的现象为 CPU 较高, IO 较忙, 处理方法与 cache buffer chains 应急处理一样

影响因素:

gc buffer busy 出现在 RAC 中, 出现概率并不高, 因为 BOSS 是对业务做了分离的, 是由于多节点同时大量访问某些数据块引起的

解决方法:

1. 查看 latch: cache buffers chains 事件相关的会话信息;

```
select sid,username,machine,program,p1raw,sql_id,logon_time,last_call_et from v$session where event='gc buffer busy';
```

使用 ora 命令 ora get_kill_sh &sql_id &username 进行查杀.

查杀后记录该 sql 语句, 丢给相应的开发商处理

2、查看哪个 SQL 执行的次数最多

```
select sql_id,count(*) from v$session where event=' gc buffer busy ' group by sql_id order by 2;
```

“cursor: pin S wait on X” 事件应急处理

现象描述:

影响因素:

一般包含以下几种:

- 1、常见硬解析
- 2、High Version Counts
- 3、BUG
- 4、shared_pool 空间不足 (告警日志报 ora-3136)

解决方法:

1、查找等待事件的阻塞者：

```
select p2raw,to_number(substr(to_char(rawtohex(p2raw)),1,8),'XXXXXXXX') sid from
v$session where event = 'cursor: pin S wait on X';
```

2、查看阻塞者在做什么：

```
select sid,serial#,SQL_ID,BLOCKING_SESSION,BLOCKING_SESSION_STATUS,EVENT
from v$session where SID=31;
```

3、根据阻塞者的 SQL 分析产生原因。

针对第四中情况,增加 shared_pool 大小:

```
Show parameter shared_pool_size
```

```
Alter system set shared_pool_size=9g scope=memory sid='dgdb1';
```

“latch: undo global data” 事件应急处理

现象描述：

一个大事务对某个表进行 DML 操作，使用大量 undo 空间。大量并发语句发起对这个表的操作，由于一致性读，需要使用 undo 记录进行回滚，产生 latch:undo global data 等待，cpu 使用率上升

影响因素：

一般包含以下几种：

1、大事务对某个表进行 DML 操作

解决方法：

1、查找 session 使用 undo 量的 SQL：

```
SELECT r.name rbs,
       nvl(s.username, 'None') oracle_user,
       s.osuser client_user,
       p.username unix_user,
       s.sid,
       s.serial#,
       p.spid unix_pid,
       t.used_ublk * TO_NUMBER(x.value) / 1024 / 1024 as undo_mb ,
       TO_CHAR(s.logon_time, 'mm/dd/yy hh24:mi:ss') as login_time,
       TO_CHAR(sysdate - (s.last_call_et) / 86400, 'mm/dd/yy hh24:mi:ss') as last_txn,
```

```
t.START_TIME transaction_starttime
FROM v$process      p,
     v$rollname      r,
     v$session        s,
     v$transaction t,
     v$parameter x
WHERE s.taddr = t.addr
      AND s.paddr = p.addr
      AND r.usn = t.xidusn(+)
      AND x.name = 'db_block_size'
ORDER by undo_mb desc
/
```

2、大事务对数据库和应用影响还不大得情况下，可以采取的方法：

- a.查找 v\$session_longops，评估是让事务进行还是 Kill 发起大事务的 session 各自的代价，选择其中一个代价较低的方式。
- b.如果是选择 kill 掉 session，可以开启并发回滚事务的特性，加快事务回滚。

3、大量并发语句，大量'latch:undo global data'等待，应用已经无法响应，CPU 使用 90% 以上的情况：

- a.此时不管是采用何种回滚特性（并发回滚、单进程回滚），由于已经没有 cpu 资源，回滚都非常耗时。
- b.联系应用确认是否可以空表暂时代替，如果可以，可以再 kill 掉 session 后，将表 rename 掉，重新建一种空表，让应用临时使用。
- c.后续使用分批提交的方式，将源表数据回插空表。
- d.如不能空表代替，则只能暂停应用，kill 掉等待 session，cpu 恢复正常后并发回滚，或建空表回插数据。

4、事件处理完毕后，对发起大事务的程序发给应用侧修改，如果是个人发起，则加强培训。

“enq:US-content” or 回滚表空间使用过度事件应急处理

现象描述：

影响因素：

一般包含以下几种：

- 1、回滚表空间使用过度，session 发起新事务查找回滚段时需要排队等待
- 2、Oracle Bug

解决方法：

1、查找 undo 表空间使用情况的 SQL：

```
select b.tablespace_name,
```

```

nvl(used_undo,0) "USED_UNDO(M)",
total_undo "Total_undo(M)",
trunc(nvl(used_undo,0) / total_undo * 100, 2) || '%' used_PCT
from (select nvl(sum(bytes / 1024 / 1024), 0) used_undo, tablespace_name
      from dba_undo_extents
      where status in ('ACTIVE','UNEXPIRED')
      group by tablespace_name) a,
(select tablespace_name, sum(bytes / 1024 / 1024) total_undo
      from dba_data_files
      where tablespace_name in
        (select value
          from v$spparameter
          where name = 'undo_tablespace'
          and (sid = (select instance_name from v$instance) or
              sid = '*'))
      group by tablespace_name) b
where a.tablespace_name (+)= b.tablespace_name
/
```

- 2、回滚表空间中大部分都是 ACTIVE extent 的情况
 - a.查询是哪些 session 引起，用“二十”中提供的 SQL
 - b.临时新增 undo 数据文件解决
 - c.事后根据查找到的 session 对提交应用侧修改程序。个人引起则加强培训
- 3、回滚表空间中大部分都是 UNEXPIRED extent 的情况
 - a.减小 undo_retention 值
 - b.如果 undo_retention 的值正常，则新增 undo 数据文件，加大 undo 表空间
- 4、回滚表空间使用情况正常
 - a.查看 undo 相关的隐含参数是否设置正确。

“gc buffer busy” 问题

现象描述:

环境: SunOS 5.9(64bit, sparc)+RAC+oracle 10.2.0.3

场景: 话务性能库不断报出 gc buffer busy，系统资源消耗较高。

影响因素:

以下几点可导致此事件:

A、Hot Blocks;

B、低效率的查询;

解决方法:

1、查询事件相关用户, sql

```
select sid,username,machine,program,p1raw,sql_id,logon_time,last_call_et from v$session  
where event='gc buffer busy';
```

2、根据 sql_id,username 查询出 sql, 分析执行效率, 与厂商协商解决。

```
select sql_text from v$session join v$sqlarea where sql_id = 'xx';
```

资源消耗类

合同系统响应缓慢

现象描述:

环境: Solaris+oracle 9.2.0.8.0

场景: 数据文件(文件系统)所在的磁盘 I/O 繁忙程度一直处在 100%,根目录所在的磁盘使用率超过 85%。

影响因素:

Solaris 文件系统根目录使用率超过 80% 以上后, 磁盘 I/O 性能会受到影响

解决方法

1.清理根目录, 使使用率低于 80%。

2.将数据文件分别打散到不同的磁盘上

高资源消耗进程应急处理

现象描述:

某个 oracle process CPU 使用率非常高。

某个 oracle process MEM 使用率非常高。

影响因素:

暂无

解决方法:

1、使用 TOPAS 观察哪个进程 CPU 使用率高，找出相关进程号，通过以下命令定位数据库 SID 号. `Select sid,sql_id,event,status from v$session where paddr in (select addr from v$process where spid=&进程号);`

2、使用以下命令查看 oracle 会话使用内存超过 100M 的用户

`set line 180`

`col MACHINE for a10`

`col PROGRAM for a25`

`col USERNAME for a15`

```
select      s.sid,      s.serial#,      s.username,      s.machine,      s.program,s.process,      s.sql_id,
p.pga_alloc_mem/1048576 size_m, p.spid
  from v$session s, v$process p where s.paddr=p.addr
        and p.pga_alloc_mem>104857600 order by 7 desc;
```

3、对相关进程和会话进行分析，决定是否 kill.

IO 非常高

现象描述:

数据库登陆缓慢，或者根本不可用，无法登陆数据库进行查询等操作。表空间满或者文件系统满了一般情况下不会发生，因为我们有监控告警。这里主要是针对异常 sql 引起数据库 hang 的情况

影响因素:

分析 IO 高的原因，例如：

- A、大量的并行；
- B、长事务；
- C、物理读高

解决方法:

在操作系统使用命令: `ps -ef|grep LOCAL=NO |awk '{print $2}' |xargs kill -9`
`kill` 所有非本地进程,然后检查系统资源状态, 检查数据库状态

无线城市现网数据库第一个节点内存使用率高

现象描述:

环境: Aix+RAC+oracle 11.2.0.1

场景: 无线城市现网数据库第一个节点内存使用率一直保持在 95% 以上, 且时不时由于内存使用过高和使用 swap 导致实例或者主机重启

影响因素:

过多的空闲会话连接数导致, 平均连接数为 1800 个以上, 且大部分连接不释放, 会持续增加, 消耗过多的 PGA 资源引起内存不足

解决方法:

开发商华为调整应用连接池参数。

PGA 使用过大

现象描述:

影响 SQL 执行的效率

影响因素:

PGA 使用过大

解决方法:

1、查询当前 PGA 使用大小:

```
select sum(pga_alloc_mem)/1048576/1024 size_gb  
from v$sqlprocess ;
```

2.查询使用 PGA 较大的具体进程

例如以下语句可以查出具体占用内存大于 100m 的进程信息:(例如)

```
set line 180  
col MACHINE for a10  
col PROGRAM for a25  
col USERNAME for a15  
select s.sid, s.serial#, s.username, s.machine, s.program,s.process, s.sql_id,  
p.pga_alloc_mem/1048576 size_m, p.spid  
from v$sqlsession s, v$sqlprocess p where s.paddr=p.addr  
and p.pga_alloc_mem>104857600 order by 7 desc;
```

3.咨询开发商可以删除语句

```
alter system kill session 'sid,serial#';
```

CPU 使用过高应急处理

现象描述:

usr%使用率达到 90 以上

影响因素:

CPU 使用过高,一般表现在以下几点:

- A、不良 SQL 造成的大量等待事件
- B、大量的短连接造成 CPU 负载高,BOSS 外围系统曾出现过。
- C、BUG

解决方法:

1、当 CPU 出现高负载的时候,首先我们要检查当前的数据库里是否有大量异常等待,例如: latch free, library cache lock/pin 等待事件。

```
select event,count(*),wait _class from v$session group by event,wait_class order by 2;
```

如果有,根据相关等待事件分析问题。也可以通过 HANG 分析,进行阻塞源头会话定位。

2、当 CPU 出现高负载的时候,检查发现当前数据库并无任何异常等待事件,我们就要参考平时的 CPU 使用率指标,然后通过会话、事务量来衡量。

3、当 CPU 出现高负载的时候,检查发现当前数据库并无任何异常等待事件,当前活动 SQL 语句与平时差别很大,我们可以关闭监听,检查是否由于连接造成的。

无线城市第二节点 CPU 消耗 100%

现象描述:

环境: Aix+RAC+ORACLE 11.2.0.1.0

现象: 无线城市现网数据库第二节点 CPU 使用率由正常的 40%上升到 100%,且一直保持 100%:

1、使用 topas 查看各进程使用 CPU 情况,发现没有非常消耗 CPU 的进程,但是前 20 的进程都消耗 2%的 CPU

2、查看数据库等待事件,发现等待事件都为 cursor: mutex S,且有 40 个,同时发现 SQL 语句类似为 insert into im_loginfo(logid,mobnum,reqtime,re

```
sptime,interfaceid,reqsysid,respysid,reqmsg,respmsg,transactionid,resultcode,nodeip)
values( seqlog.nextval,:1,:2,:3,:4,:5,:6,:7,:8,:9,:10,:11)
```

3、持续一段时间后,节点实例自动重启。

影响因素:

绑定变量绑定失效引起的。

解决方法:

怀疑是 bug,已提 SR 由 Oracle 确认.

管线系统资源使用 CPU 低、内存高问题

现象描述:

环境: SunOS 5.10(64bit, sparc)+RAC+oracle 10.2.0.3

场景: 管线系统反应高峰时期进行应用操作很慢, 查看系统后,CPU 使用率比较低, 但内存的使用率达到 95% 以上, 剩余内存只有 600M 左右, 在 AWR 报告中也没有看到明显的性能问题。

Tasks: 1317 total, 3 running, 1314 sleeping, 0 stopped, 0 zombie

Cpu(s): 23.3%us, 2.0%sy, 0.0%ni, 50.6%id, 23.2%wa, 0.1%hi, 0.7%si, 0.0%st

Mem: 65974424k total, 62568640k used, 3405784k free, 165220k buffers

影响因素:

经过分析为原因 SGA 参数的设置不合理,取消自动管理的方式。

分析思路:

注释: 这里需要明确分析的思路, 比如, 内存的调整频率, 或者 CPU 的波动, 如果有相关检查语句, 最好可以补充进去。

解决方法:

1. 取消内存的自动管理方式。
2. 重新调整 SGA 大小

大表索引被删除导致 CPU 100%的应急处理

现象描述:

CPU 使用率 90% 以上

影响因素:

一般包含以下几种:

- 1、大表上索引被删除, 部分 SQL 全表扫描导致资源集聚消耗

解决方法:

- 1、查找某个实例上等待 session 的信息, 定位低效 SQL:

```
select to_char(a.logon_time,'yyyy-mm-dd hh24:mi') logon_time,  
       a.sql_id,
```

```
a.event,  
a.username,  
a.osuser,  
a.process,  
a.machine,  
a.program,a.module,  
b.sql_text,  
b.LAST_LOAD_TIME,  
to_char(b.last_active_time,'yyyy-mm-dd hh24:mi:ss') last_active_time,  
c.owner,c.object_name,  
a.last_call_et,  
a.sid,a.SQL_CHILD_NUMBER,  
c.object_type,p.PGA_ALLOC_MEM,a.p1,a.p2,a.p3,  
'kill -9 ||p.spid killstr  
from v$session a, v$sql b, dba_objects c,v$process p  
where a.wait_class <> 'Idle' and a.status='ACTIVE' and p.addr=a.paddr  
and a.sql_id = b.sql_id(+)  
and a.sql_child_number = b.CHILD_NUMBER(+)  
and a.row_wait_obj# = c.object_id(+)  
and a.type='USER'  
order by a.sql_id,a.event;
```

2、非 rac 环境

- a.部分大表索引删除，导致全表扫描，cpu 100%
- b.此时数据库基本处于 hang 状态，应用已经无法响应
- c.停止相关应用，kill 掉这些等待 session，新建索引（parallel nologging）
- d.如果无法停止应用，停止数据库的监听，kill 掉等待 session，新建索引（parallel nologging）

3、rac 环境

- a.部分大表索引删除，导致全表扫描，cpu 100%
 - b.此时该节点基本处于 hang 状态，应用已经无法响应
 - c.停止相关应用，kill 掉这些等待 session，新建索引（parallel nologging）
 - d.如果无法停止应用，停止所有节点的监听，kill 掉等待 session，新建索引（parallel nologging）
-

网络连接类

网投数据库应用无法连进数据库

现象描述:

环境: SUSE Linux Enterprise Server 10+oracle 10.2.0.1

场景: 网投数据库, 应用人员反映数据库, 新的连接无法进入数据库, 至使应用无法使用。检查数据库连接数, 监听均属正常状态。同时发现有查询数据库视图的语句占用大量连接未释放。

影响因素:

下列因素可影响新连接进入:

A. 数据库监听异常

B. 数据库连接数达到最大值

解决方法:

1. 检查监听状态:

`lsnrctl status`

2. 检查连接数:

`show parameter session;`

`select username,status from v$session;`

连接数未达上限。查询数据状态数据库还有大量状态为 `INACTIVE` 的可用连接, 连接数也在最大值范围内, 但新的连接无法进入。只能从应用方来入手, tomcat 连接池最大值设置为 300。后来东信调整应用程序连接机制, 问题解决。

分析思路:

注释: 这里可以简要的描述连接池的基本连接机制, 如何发现查询数据库的占大量连接的语句, 处理问题的思路很重要。

连接不上数据库

现象描述:

数据库无法连接

影响因素:

客户报连接不上数据库，例如：

- 1、连接数达到上限；
- 2、监听有问题；

解决方法:

1、看看连接数

show parameter session;

select count(*) from v\$session;

若我们用 sqlplus / as sysdba 也无法连接，则先 KILL 掉几个 LOCAL=NO 的会话，然后再进去分析是哪个模块的连接较多，然后通知开发商处理

ps -ef | grep LOCAL=NO --找出 PID，KILL 掉几个

select machine,program,count(*) from v\$session group by machine,program order by 3; --找出较多的模块

2、查看监听

lsnrctl status

看看状态是不是 ready，或者 reload 一次，看看故障是否能恢复

ESB 库客户端不定时断开连接

现象描述:

环境：Aix+RAC+ORACLE 11.2.0.2.0

现象：PL/SQL 等开发工具连接数据库后，会不定时的断开连接，断开后界面无反应，经过长时间才能重连，影响开发工作。根据现场观察到的故障现象：

1、检查数据库的各项相关设置，包括系统超时限制、用户资源限制和网络超时限制等方面，均未发现异常。

2、在不同网段进行模拟，发现所有网段均有断开连接现象发生。

3、经过长时间和全方面的模拟测试，发现断开规律为 idle 超过 10 分钟的数据库连接会被自动断开。

分析思路:

注释： [这里最好配上测试的过程。](#)

影响因素:

一般情况下，为网络、防火墙策略限制。

解决方法:

重新修改网管防火墙相关限制项。

X 通过监听无法连接数据库

现象描述:

环境: Redhat+RAC+ORACLE 11.2.0.1.0

现象: 数据库工作状态正常, 监听器工作状态也正常。并且, 监听服务也已经注册了。但即使在服务器本地, 通过监听无法访问数据库。通过 tns 访问数据库时, 总是报 lost connect 的错误信息。直接登陆时没有问题。

现象分析:

这个问题可能是 oracle 的一个 bug, 在排除了常用的解决办法办法之后, 可以考虑修改 \$ORACLE_HOME/bin/oracle 文件的权限。

但这个问题在许多数据库并无法重现, 在遇到这样的问题时, 可以尝试这样解决一下。

解决方法:

将 \$ORACLE_HOME/bin/oracle 的权限更改为 6571

空间管理类

表空间故障应急处理

现象描述:

场景一: 在 RAC 环境下进行表空间扩容(添加数据文件)时, 只在一个节点上对数据文件建立了软连接, 另一个节点没有建立软连接。

场景二: 在 RAC 环境下进行表空间扩容(添加数据文件)时, 两个节点都没有建立软连接, 只在一个节点的本地文件系统添加了数据文件, 或者添加数据文件时有空格等特殊字符

场景三：不小心将其他环境的裸设备加到当前的环境中。（绝不允许出现此类错误）

影响因素:

一般情况下，都属于人为错误.

解决方法:

（场景一）解决方法:

- 1、 将两个节点数据文件改为离线状态
`alter database datafile 'XXX' offline;`
- 2、 在问题节点对数据文件建立软连接
`ln -s 裸设备 数据文件`
- 3、 在问题节点恢复数据文件
`recover datafile 'XXX';`
- 4、 将数据文件改为在线状态
`alter database datafile 'XXX' online;`
- 5、 确认数据库告警日志无报错。

（场景二）解决方法:

- 1、 将问题节点数据文件改为离线状态
`alter database datafile 'XXX' offline;`
- 2、 在各节点对数据文件建立软连接
`ln -s 裸设备 数据文件`
- 3、 通过 `ALTER DATABASE CREATE DATAFILE ‘源文件’ AS ‘目标文件’; copy` 数据文件至目标位置
`ALTER DATABASE CREATE DATAFILE '源文件' AS '目标文件';`
- 4、 恢复数据文件
`recover datafile '目标文件';`
- 5、 将数据文件改为在线状态
`alter database datafile '目标文件' online;`
- 6、 将错误的本地数据文件移到其他路径，避免“/oracle”文件系统使用比率达到告警值。
- 7、 确认数据库告警日志无报错。

（场景三）解决方法:

- 1、 除了恢复，没有太好的方法。需要备份、归档都在。

SYSAUX 表空间爆满没有存储扩容应急处理

现象描述:

如果 SYSAUX 表空间不可用时，数据库的核心功能还是可以继续运行的。只是些存放在 SYSAUX 表空间里的功能收到限制，比如 OEM。

影响因素:

一般包含以下几种:

- A、业务表被放在 SYSAUX 表空间
- B、一些辅助表的数据量太大

解决方法:

检查 SYSAUX 中的对象，是哪个对象占用了大量空间.检查时业务表还是辅助表，确认该对象是否可以清理.

```
col segment_name format a30
col segment_type format a30
set linesize 300
select OWNER,SEGMENT_NAME,SEGMENT_TYPE,BYTES/1024/1024/1024 GB
from dba_segments
where TABLESPACE_NAME='SYSAUX'
order by 4;
```

文件系统使用率达到或超过 95%应急处理

现象描述:

使用 df 命令查看文件系统使用率达到 95%

影响因素:

备份文件、dump 文件、trace 文件等大量产生造成.

解决方法:

检查相关文件系统下面的文件大小:

例如: 查看/u01 目录下每个子目录的大小

`du -sh /u01/*`

根据需求进行清理.

临时表空间过度事件应急处理

现象描述:

临时表空间使用过度, 需要使用临时表空间的 SQL 运行缓慢

影响因素:

一般包含以下几种:

- 1、部分 SQL 大量使用临时段

解决方法:

- 1、查找单节点 temp 表空间使用情况的 SQL:

```
SELECT A.tablespace_name tablespace,
       D.mb_total,
       SUM(A.used_blocks * D.block_size) / 1024 / 1024 mb_used,
       D.mb_total - SUM(A.used_blocks * D.block_size) / 1024 / 1024 mb_free
FROM v$sort_segment A,
     (SELECT B.name, C.block_size, SUM(C.bytes) / 1024 / 1024 mb_total
      FROM v$tablespace B, v$tempfile C
      WHERE B.ts# = C.ts#
      GROUP BY B.name, C.block_size) D
WHERE A.tablespace_name = D.name
GROUP by A.tablespace_name, D.mb_total;
```

- 2、查询 session 使用 temp 段的情况:

```
SELECT S.sid || ',' || S.serial# sid_serial,
       S.username,
       S.osuser,
       P.spid,
       S.module,
       S.program,
       s.sql_id
       SUM(T.blocks) * TBS.block_size / 1024   mb_used,
       T.tablespace,
       COUNT(*) sort_ops
FROM v$sort_usage T, v$session S, dba_tablespaces TBS, v$process P
WHERE T.session_addr = S.saddr
      AND S.paddr = P.addr
      AND T.tablespace = TBS.tablespace_name
```

```
GROUP BY S.sid,  
        S.serial#,  
        S.username,  
        S.osuser,  
        P.spid,  
        S.module,  
        S.program,  
        TBS.block_size,  
        T.tablespace  
ORDER BY sid_serial;
```

- 3、如果是某些 session 使用 temp 段过度引起
 - a. 联系应用停止相关程序 or kill 掉这些 session
- 4、如果 session 使用 temp 情况都属正常情况
 - b. 加大临时表空间

网分后台 undo 表空间无法自动扩展

现象描述:

环境: Aix+RAC+oracle 10.2.0.3.0

场景: 网分数据库节点一, 后台 alter 日志报: **ORA-1628: max # extents reached for rollback segment** 错误信息, 使数据库的调度任务无法完成。

影响因素:

回滚段的 maxextens 数达到参数设置的最大值, 造成不能分配下一个 extent

分析思路:

注释: 这里需要将如何看 maxextens 的查询方法记录上。

解决方法

1. 新建立一个表空间 undo2, 修改节点 1 的 undo 表空间为 undo2 后问题排除。
2. 修改增大 undo 的 maxextents 参数值

绿色坐席 undo 表空间无法扩展

现象描述:

环境: Linux+ oracle 11.2.0.1.0

场景：在向数据库插入一条记录时，数据库频繁出现 **ORA-30036: unable to extend segment by 8 in undo tablespace 'UNDOTBS1'** 的报错信息，无法插入记录。

影响因素：

产生该错误，通常是由于当前的 UNDO 表空间很小，并且在同时有一个很大的事务存在。并且，在 Oracle 看来，这种提示并不是一种异常。因此，您无法在该 ORA 报错的同时在 alert 文件中看到相关的信息，你只能通过相应的 trace 文件来看到这类的信息。

当数据库产生上述信息时，在 alert 中会留下这样的信息：

kdidelleafblks -- encountered error 30036. See trace file for details

相应的 trace 文件(vmdb_j000_3784.trc)中，会有这样的描述：

ORA-30036 DIAGNOSTIC

This diagnostic information is dumped to trace file at
most once every 24 hours, it does not indicate any error.

ORA-30036 happens when trying to extend undo segment _SYSSMU12_710107609\$ (usn=12)
by 8 blocks

Reason: Cannot Wrap, Fail to Steal

Current undo tablespace UNDOTBS1 (tsn=2)

undo tablespace current size 262144 blks, maxsize 262144 blks, fixed sized

Undo Retention (reactive):568, Max Query Length:1383

Parameter Undo Retention:900, Tuned Undo Retention:568, High threshold Undo Retention:-2
autotune:1

Retention Guarantee FALSE

Current Time is 1351615156

同时，在该 trace 文件的 Info about Undo Segments in this undo tablespace 信息列表中，将会很容易找到当前存在的大事务。如：

Undo Segment 1: HW Ext=2, 1st Active Ext=-1, Total Extents=3

Extent Info of Undo Segment 1

ExtID	Status	Size	CmtTime
0	Active	7	1351614703
1	Active	8	1351614760
2	Unexpired	128	1351614760
2	Unexp wmq1	128	1351614760

=====

_SYSSMU1_2861643892\$ (usn 1) is INUSE (act=16, une=128, exp=0)
(unewmq1=128, expwmql=0)

解决方法：

1、通过分析上述 trace 文件，可以找到当前存在的大事务，对该事务进行适当分解，减少对 UNDO 表空间的资源需求量。

2、增大 UNDO 表空间的大小。

紧急措施：

1、增大 UNDO 表空间

网分后台库 temporary 段暴涨

现象描述:

环境: AIX Version 6.1+RAC+oracle 10.2.0.3

场景: 网分后台库表空间 tbs_tmp_data 出现暴涨, 一小时大约增长 30-40G. 经查询, 发现 tbs_tmp_data 表空间出现大量的 SEGMENT_NAME 类似于 35.75187 这样的 temporary 段。

影响因素:

以下操作会占用大量的 temporary:

- 1、 用户执行 imp/exp 导入导出操作时, 会使用大量的 temporary 段
- 2、 用户在 rebuild index 时
- 3、 执行 create table as 语句时
- 4、 移动用户下的数据到别的表空间时

解决方法:

找出产生 temporary 段的用户:

```
SQL> select OWNER,SEGMENT_NAME,SEGMENT_TYPE,BYTES/1024/1024 from
dba_segments      where      TABLESPACE_NAME=upper('tbs_tmp_data')      and
SEGMENT_TYPE=upper('temporary');
```

找出操作:

```
SQL> select sid from v$session where username='ETL' and STATUS='ACTIVE' and program in
('IMP','EXP','IMPDP','EXPDP');
```

```
SQL> select p.spid,to_char(SQL_FULLTEXT) from v$sqlarea a,v$session s,v$process p where
p.addr=s.paddr and s.sql_id=a.sql_id and s.username='ETL' and s.STATUS='ACTIVE' and
(SQL_FULLTEXT like '%alter%' or SQL_FULLTEXT like '%create table%as select%');
```

我们也可以上面的方法找出导致 temporary 段暴涨的原因, 跟开发商沟通, 可以选择 kill 掉 sid 对应的操作系统进程。尽量减少在业务高峰期进行这些操作。

数据泵进程挂起导致磁盘爆满

现象描述:

环境: SunOS 5.10+RAC+ORACLE 10.2.0.3

场景: 管线系统数据库, 由于系统磁盘空间涨到 100%, 数据库无法继续使用。

影响因素:

检查发现磁盘空间一直在增长, 但是数据库没有用户做操作。最后检查有一个数据库

expdp 进程在 running，导出的文件使磁盘空间一直增长

解决方法:

select job_name,state from dba_datapump_jobs;找出 running 的 job_name,
expdp 用 attach 参数连入数据库，输入命令: kill_job，即杀掉 expdp 进程，磁盘空间停止增长，清理不必要的数据库，重启库恢复正常。

无法启动类

短信网关 RAC 数据库 CRS 无法正常启动

现象描述:

环境: redhat linux +RAC+oracle 11.1.0.7

场景: 短信网关一套数据库，在作了十八大安全相关整改后，数据库 CRS 无法正常启动。

影响因素:

影响 crs 正常启动的因素较多，不能正常启动，数据库集群无法使用。但检查 crs 各项配置均正常。

解决方法:

检查 crs 日志发现报系统权限相关的错误，再次确认在安全整改过程中，/etc/security 目录的权限被更改为 700。正常应该为 755 权限,执行 `chmod 755 /etc/security` 再进行执行 `crsctl start crs` 可正常启动。

网分后台数据库无法正常启动

现象描述:

环境: Aix+RAC+oracle 10.2.0.3.0

场景: 当天进行数据库存储扩容，扩容操作后，数据库无法正常启动。

影响因素:

IBM 工程师对存储挂载时，挂载失败。存储出错导致数据库无法启动。

解决方法:

后经过重新挂载存储，数据库恢复正常。

分析思路:

注释: 这里可能需要描述如何去查询挂载失败。

珠海无线城市 RAC 数据库 CRS 无法正常启动

现象描述:

环境: redhat linux +RAC+oracle 10.2.0.4

场景: 珠海无线城市一套数据库，在作了十八大安全相关整改后，数据库 CRS 无法正常启来。

影响因素:

影响 crs 正常启动的因素较多，不能正常启动，数据库集群无法使用。但检查 crs 各项配置均正常。

解决方法:

检查 crs 日志发现报系统权限相关的错误，再次确认在安全整改过程中，/etc/security 目录的权限被更改为 700。正常应该为 755 权限,执行 `chmod 755 /etc/security` 再进行执行 `crsctl start crs` 可正常启动。

ORACLE GC OMS 无法启动

现象描述:

环境: OS Version: Windows 2003 x86, GC Version: 11.0.1.0, WebLogic Version: 10.3.2.0

Oracle DB Version: 11.2.0.1

场景: OMS 服务在主机启动以后无法起来 OMS 日志中有一些报错信息:
weblogic.diagnostics.lifecycle.DiagnosticComponentLifecycleException: weblogic.store.PersistentStoreException: [Store:280020]There was an error while reading from the log file
Caused by: java.io.IOException: Error reading from file, 已到文件结尾, errno=38

影响因素:

oms 主机所在的 39.189 机器此次故障前一周,出现过因 GC 安装路径所在磁盘空间耗尽导致 OMS 宕机的,当时清理空间后,重新启动 OMS,正常启动.但主机重启后,OMS 无法启动.

根据搜索的一些信息·当 OMS 日志所在目录爆满一次后，下次重新启动时日志将无法写入。

解决方法:

- 1.E:\app\Middleware\oms11g\BIN>emctl stop oms -all #停止后才能删除 DAT 文件
 - 2.E:\app\gc_inst\user_projects\domains\GCDomain\servers\EMGC_OMS1\data\store\diagnostics 目录下的 DAT 文件备份后删除
 3. E:\app\Middleware\oms11g\BIN>emctl start oms
- 之后，OMS 可正常启动，问题解决。

短信中心报表库节点 2CRS 服务无法启动

现象描述:

环境: Aix 6.1+RAC+ORACLE 11.1.0.7.0

现象: 短信中心惠州 MO 数据库共享 DS4800 存储控制器故障,需要更换存储控制器,在更换 IBM 存储控制器时,IBM 工程师把数据库把 ASM 磁盘与操作系统磁盘之前映射关系清除了,更换控制器之前没有保存映射关系,导致节点 2 CRS 服务资源无法正常启动。

影响因素:

1. 在 开 启 节 点 2 CRS 服 务 时 , 报 错 .
Oracle Cluster Registry initialization failed accessing Oracle Cluster Registry device: PROC-26: Error while accessing the physical storage Operating System error [Device busy] [16]
2. 检查 CRS 日志和操作系统日志,CRS 日志与操作系统日志均无记录. 从开启 CRS 服务时所报的错上看,初步可以判断出,是 OCR DISK 和 VOTING DISK 出现问题. 与华为工程师沟通,让他检查节点 2 系统磁盘与 OCR DISK ,VOTING DISK 对应关系和权限是否正确. 对应关系和权限都正确.
3. 既然 OCR DISK 和 VOTING DISK 映射和权限都没问题.但是肯定一点是 OCR 和 VOTING DISK 问题. Oracle Cluster Registry initialization failed accessing Oracle Cluster Registry device: PROC-26: Error while accessing the physical storage Operating System error [Device busy] [16] 在开启节点 2 CRS 时,为什么会报磁盘在 busy 的错误呢? 难道是 OCR DISK 和 VOTING DISK 是独占方式? 节点 2 无法使用? 这个是我的一点猜测.
4. 测试我的猜测,关闭节点 1 CRS, 开启节点 2 CRS. 接着再开启节点 1 CRS. 这时发现节点 2 CRS 能正常开启,但是节点 1 CRS 无法正常,在开启节点 1 时, 报的错和上面一样,是磁盘模式出现问题.
5. 检查两节点 OCR DISK 和 VOTING DISK 所应系统磁盘属性,果然是独占模式. 修改为共享模式,两节点 CRS 服务能正常开启.

```
GDHZ-SMC-MO-SV01A-IBM-RPT:/dev #lsattr -E -l hdisk2 | grep reserve_
reserve_policy single_path                Reserve Policy                True
GDHZ-SMC-MO-SV01A-IBM-RPT:/dev #lsattr -E -l hdisk3 | grep reserve_
reserve_policy single_path                Reserve Policy                True ----独占模式
GDHZ-SMC-MO-SV01A-IBM-RPT:/dev #lsattr -E -l hdisk4 | grep reserve_
reserve_policy no_reserve                 Reserve Policy                True
```

GDHZ-SMC-MO-SV01A-IBM-RPT:/dev #lsattr -E -l hdisk7 | grep reserve

reserve_policy no_reserve

Reserve Policy

True ---共享模式

解决方法:

修改为共享模式:

chdev -l hdiskXX -a reserve_lock=no

chdev -l hdiskXX -a reserve_policy=no_reserve

数据库 HANG 类

数据库 HANG 异常处理

现象描述:

数据库无法登陆

影响因素:

A、大量异常阻塞

B、资源耗尽

解决方法:

1.进行 HANG 分析，查找顶级阻塞的会话。

普通的 HANG 分析:

参考业支脚本.

全局的 HANG 分析:

oradebug setmypid (oradebug setospid 3188)

oradebug unlimit

oradebug setinst all

```
oradebug -g def hanganalyze 3
```

```
oradebug tracefile_name
```

2.紧急情况下，在操作系统使用命令: `ps -ef|grep LOCAL=NO |awk '{print $2}'|xargs kill -9`
kill 所有非本地进程,然后检查系统资源状态，检查数据库状态

话务告警库 oracle 进程 HANG 住

现象描述:

环境: SunOS+RAC+oracle 10.2.0.3.0

场景: 话务告警库打补丁后，开启各个资源后，用 `crs_stat -t` 检查个资源均为 online 状态，实际上 2 节点数据库实例未启动。

影响因素:

当天凌晨数据库打补丁，再 `startup` 数据库实例。在重启过程过程中看后台日志，发现使用未有 'instance open' 出现，实际上 ORACLE 进程 HANG 住，实例未正常启动。

解决方法:

关闭 `emagent`，`mnagent` 程序及其他所有 oracle 相关进程，重启数据库恢复正常。再打开 `emagent`。

数据库 CRASH 类

网分后台库实例重启

现象描述:

环境: Aix+RAC+oracle 10.2.0.3.0

场景: 数据库三个实例自动重启, startup 命令可以启动数据库, 但运行一个多小时会自动重启数据库。

影响因素:

最后检查到是 undo 表空间中有一个事务回滚段出错。

分析思路:

注释: 请描述出错误内容, 以及找出具体查询的方法。

解决方法:

找出次回滚的具体数据库操作, 与开发商协商解决。

实例 CRASH 应急处理

现象描述:

RAC 环境单节点 crash , RAC 环境所有节点 crash

影响因素:

暂无,CRASH 问题涉及到的问题比较复杂, 以后将逐步更新。

解决方法:

遇到实例 crash 之后, 首先看是否能重启启动:

1、能重新启动的情况下, 知会相关领导, 事后分析重启原因。

2、不能重启的情况下:

A、RAC 之单节点无法重启, 通知相关领导, 评估正常节点负载, 应用切换至正常节点。相关技术人员进行故障处理。

B、RAC 之全节点无法重启, 通知相关领导, 考虑是否切换应急库。相关技术人员进行故障处理。

传输数据库脑裂问题

现象描述:

环境: Aix+RAC+oracle 10.2.0.4

场景:传输网管数据库实例脑裂,主机和CRS无异常,但是节点二实例自动重启,并且在脑裂的过程中影响节点一数据库 hang,应用不能正常连接。

数据库日志可看到以下信息:

IPC Send timeout detected,Receiver ospid 2123460;Waiting for clusterware split-brain resolution;ORA-29740: evicted by member , group incarnation;

影响因素:

脑裂会导致数据库自动实例重启。通常出现 ORA-29740 是由以下原因引起:

- 1、 RAC 节点间时间不同步
- 2、 数据库主机资源负载过高
- 3、 心跳网络异常
- 4、 Oracle bug

解决方法:

经过检查先后发现了以下问题:

- 1、主机/var 目录空间不足,系统 dump 生成失败
- 2、Oracle 用户环境变量设置不正确
- 3、操作系统未打 APAR IV04047 补丁
- 4、RAC 两节点的操作系统版本不一致,cs_01 版本 6100-07,cs_02 版本为 6100-06
- 5、分析发现心跳交换机配置存在异常,交换机 IP 为 172.16.3.1 和 cs_01 的心跳网卡 IP 冲突。经协商后更改交换机 IP 为 172.16.3.100。并且发现在主机上无法 ping 通交换机 IP,在交换机上 cs_01 和 cs_02 连接的端口上抓包,发现抓到 172.16.3.13 和 172.16.3.14 的数据包,确认这两个 IP 为物联网数据库的心跳 IP。我们检查物联网数据库后发现,该系统也出现了 ORA-29740 引起的脑裂。抓包信息如下

356	15.550857	8e:56:c7:02:cc:04	Broadcast	ARP	60	who has 172.16.3.100? Tell 172.16.3.2
357	15.552500	172.16.3.2	172.16.3.1	IPv4	1514	Fragmented IP protocol (proto=UDP 0x11, off=4440, ID=9604)
358	15.552505	172.16.3.13	172.16.3.14	UDP	298	Source port: 48647 Destination port: 43366
359	15.552622	172.16.3.14	172.16.3.13	UDP	234	Source port: 43362 Destination port: 48647
360	15.552624	172.16.3.14	172.16.3.13	UDP	330	Source port: 43362 Destination port: 48617
361	15.552720	172.16.3.13	172.16.3.14	UDP	234	Source port: 48613 Destination port: 43362
362	15.553057	172.16.3.14	172.16.3.13	UDP	298	Source port: 43396 Destination port: 48617
363	15.553077	172.16.3.13	172.16.3.14	UDP	234	Source port: 48613 Destination port: 43396
364	15.553170	172.16.3.13	172.16.3.14	UDP	330	Source port: 48613 Destination port: 43366
365	15.553266	172.16.3.14	172.16.3.13	UDP	234	Source port: 43362 Destination port: 48613


```

[Quidway]display arp dynamic
IP ADDRESS      MAC ADDRESS    EXPIRE(M)  TYPE INTERFACE
              VLAN
-----
192.168.255.254  0819-a6d0-c045    I -  vlanif1
192.168.128.1    5cf3-fcb9-15b0    20  D-0  GE0/0/4
              1
172.16.3.100     0819-a6d0-c045    I -  vlanif100
172.16.3.13      e41f-1350-b34c    20  D-0  GE0/0/1
              100
172.16.3.14      e41f-1350-b6b8    20  D-0  GE0/0/2
              100
  
```

经过和惠讯、IBM、亿阳协商，更换一台交换机进行测试，进一步定位是否由于心跳交换机配置问题导致该故障。在几次更换心跳交换机后问题解决。

至此，前后耗时约一个月的传输脑裂问题解决，此类长时间无法定位的问题，应该扩大分析范围不仅着眼于数据库，应该和主机网络人员一道分析定位。

应用操作类

SQL 执行计划变化应急处理

现象描述：

一个 SQL 的执行计划的不稳定，

影响因素:

常见原因包含以下两种:

- 1、统计信息的变化
- 2、SQL 语句的变化

解决方法:

- 1、通过 SQL_ID 确认统计信息是否一致 (该语句会将 AWR 中所有的信息查找出来)

```
set lines 155
col execs for 999,999,999
col avg_etime for 999,999.999
col avg_lio for 999,999,999.9
col begin_interval_time for a30
col node for 99999
break on plan_hash_value on startup_time skip 1
select ss.snap_id,
       ss.instance_number node,
       begin_interval_time,
       sql_id,
       plan_hash_value,
       nvl(executions_delta, 0) execs,
       (elapsed_time_delta /
        decode(nvl(executions_delta, 0), 0, 1, executions_delta)) / 1000000 avg_etime,
       (buffer_gets_delta /
        decode(nvl(buffer_gets_delta, 0), 0, 1, executions_delta)) avg_lio,
       (disk_reads_delta /
        decode(nvl(buffer_gets_delta, 0), 0, 1, executions_delta)) avg_pio
  from DBA_HIST_SQLSTAT S, DBA_HIST_SNAPSHOT SS
 where sql_id = nvl('&sql_id', '4dqs2k5tynk61')
    and ss.snap_id = S.snap_id
    and ss.instance_number = S.instance_number
    and s.instance_number like nvl('&instance_number', s.instance_number)
    and executions_delta > 0
 order by 1, 2, 3
/
```

- 2、如果不一致, 我们需要检查统计信息是否最新, 如果统计信息更新时间比当前时间有些遥远, 收集统计信息。
-

3、如果统计信息正常，我们需要确认数据量是否有大量增加。

4、如果数据量增加的情况下，我们就要考虑清理数据或 SQL 优化。

使用绑定变量 SQL 突然运行缓慢的应急处理

现象描述:

一个 session 可能以某种方式变的非常消耗 CPU,并且没有任何有意义的 wait event. SQL 的执行计划没有改变过

影响因素:

一般包含以下几种:

- 1、SQL 代入的绑定变量值极端
- 2、等待的块有问题

解决方法:

- 1、查看 SQL 的执行计划，及等待事件。
- 2、等待事件是否正常，如果为块等待，则可以采样等待的块，通过采样的块，定位对应的表和索引，进行分析。
- 3、使用 errorstack 定位这个 SQL 的绑定变量值进行分析。

数据网管无法插入数据

现象描述:

环境: SunOS+oracle 11.2.0.1

数据网管数据库中插入数据失败。

影响因素:

主要问题来自于 I/O 的问题，由于业务增加，I/O 争用严重，导致数据库 dbwr 进程写入慢，从而影响到缓冲区不能快速释放，在用户请求时无法分配到空闲的缓冲块。产生较高的"free buffer waits"的等待事件。

解决方法:

找出正在进行的 io 读写频繁的数据库操作，与开发商协商解决。

网分前端库应用程序 timestamp 错误

现象描述:

环境: Aix+RAC+oracle 10.2.0.5.0

场景: 开发商反映网分前端数据库, 应用程序的 timestamp 一直出错

影响因素:

最后检查到是主机时区与数据库的时区设置错误导致这个错误。

分析思路:

注释: 记录错误内容。

解决方法:

调整主机时区问题解决。

来电提醒数据库物化视图刷新缓慢

现象描述:

环境: SUSE 10+ oracle 11.1.0.7.0

场景: 开发商反映数据库, 物化视图 T_OSS_USERROUTEINFO_MV 刷新十分缓慢

影响因素:

检查发现开发通过 dblink 刷新物化视图, 在数据库的另一端 undo 表空间已经占满, 导致物化视图刷新十分缓慢。

解决方法:

扩大了源端 undo 表空间, 物化视图的刷新进度正常。

分析思路:

注释: 这个问题是怎么查询到另一端的 UNDO, 是偶然还是必然, 如果是明确报错请增加错误内容。

佛山 FSCRM 插入数据时报权限错误

现象描述:

场景: 向一个表中插入数据时, 系统中报这个错误。ORA-01950: no privileges on tablespace 'RH_DATA' FILE:[RedeductRentFrame.cpp],LINE:[100]

影响因素:

很明显这是一个因权限问题引起的报错问题, 我们需要知道这里的操作与哪些权限相关。主要有以下几个方面的权限问题:

- 1、当前表的属主, 是否在相应的表空间中具有 UNLIMITED TABLESPACE 的权限。
- 2、当前表的索引属性, 是否在相应的表空间中具有 UNLIMITED TABLESPACE 的权限。
- 3、当前是否有触发器, 相应的表及索引, 其属主相应的表空间是否具有 UNLIMITED TABLESPACE 权限。
- 4、当前是否有物化视图, 相应的表及索引, 其属主相应的表空间是否具有 UNLIMITED TABLESPACE 权限。

解决方法:

给上述相关的对象增加 UNLIMITED TABLESPACE 权限。

检查程序账号权限:

```
SQL> select * from dba_sys_privs where grantee='ZSIB';
```

GRANTEE	PRIVILEGE	ADM
-----	-----	---
ZSIB	UNLIMITED TABLESPACE	NO
ZSIB	CREATE TABLE	NO

查看索引的属主:

```
sys@FSCRM> select owner,index_name,table_owner,table_name from dba_indexes where  
OWNER<>table_owner AND TABLE_NAME='IB_CBS_SERVFEEREQ';
```

其他分类

服务切换应急处理

现象描述:

场景一: 当我们在进行服务切换时, 无法从一个节点切换到另外一个节点。

影响因素:

一般包含以下几种:

- A、目标节点实例异常,服务无法切换。
- B、持有服务节点无法释放, 服务无法切换
- C、BUG

分析思路:

解决方法:

(场景一) 解决方法:

1.首先使用 `srvctl start service -d <db_name> [-s <service_name> [-i <inst_name>]]` 命令切换。

例如: `srvctl start service -d fsdb -s smm -i fsdb1`

2.当上面命令, 可以使用以下命令进行强制服务切换: `alter system set service_names='s1,s2';`

例如: `alter system set service_names='sfs,smm' scope=memory sid='FSDB1';`

然后查看是否注册成功:

SQL> show parameter service

NAME	TYPE	VALUE

service_names	string	sfs,smm

3.检查监听并测试连接测试是否生效

`$lsnrctl service`

`$sqlplus sys/oracle@node_vip:1521/service_name`

4、如果连接不成功, 按照以下命令排查.

检查实例启动时间: `select instance_number, status, startup_time from gv$instance;`

检查数据库状态: `select inst_id, open_mode from gv$database;`

检查 alert.log 日志: Completed: ALTER DATABASE OPEN (是否完成)

检查服务运行情况: `srvctl status service -d <db_name>`

检查服务配置情况: `srvctl config service -d <db_name> -a`

升级 PSU 系统报错

现象描述:

在打补丁的过程中，出现如下的报错信息:

Running prerequisite checks...

Prerequisite check "CheckActiveFilesAndExecutables" failed.

The details are:

Following executables are active :

/oracle/app/oracle/product/10.2.0/bin/tnslsnr

UtilSession failed: Prerequisite check "CheckActiveFilesAndExecutables" failed.

OPatch failed with error code 73

影响因素:

通过日志可以发现，导致当前升级 PSU 出错的原因是当前的监听器没有关闭。因此，当务之急，是将监听手工关闭。

但事实上，在出现上述错误之前，工程已经将监听关闭了。但由于该 PSU 只跟数据库相关，与集群没有关系。因此，他在升级该 PSU 时，只是通过 CRS 关闭了监听，过了一段时间以后，监群又自动将监听拉起来了，这就导致升级 PSU 时，监听仍然是 ACTIVE 状态。

因而，在升级补丁或 PSU 时，需要通过 lsnrctl stop 的方式关闭监听器。

解决方法:

通过 lsnrctl stop 的方式关闭监听器。

数据库在执行过程中使用动态采样

现象描述:

一个 SQL 在执行过程中异常缓慢，通过查看其执行计划发现，该 SQL 在执行过程中使用动态采样。

Note

- dynamic sampling used for this statement

现象分析:

动态采样本身是 Oracle 的一个特性，用于在没有统计信息的情况下，自动收集相关表的统计信息。但在某些情况下，他仍然使用动态采样，并进而导致执行效率低下。

解决方法:

1、通过 hint 禁用动态采样。/*+dynamic_sampling(t 0) */

2、这是 Oracle 的一个 bug，请打补丁 9272549。

通过 srvctl 查看数据库的状态时反应缓慢

现象描述:

数据库状态正常，但在执行"`srvctl status database -d uwnms1`"时，其节点一需要很长时间才能出结果，节点二正常。查看 alert 日志，也并未发现有告警的信息。

现象分析:

通过类似的命令查看或启动某个资源时，由于并无相应的日志，因此给我们的判断带来一些麻烦。解决此类问题，我们可以采用跟踪的办法来分析其内部的相应过程。

解决方法:

- 1、`strace -frT -o /tmp/strace2.log srvctl status database -d uwnms1`，根据其生成的 trace 文件分析内部过程，并解决。
- 2、使用 `sh -x shell.sh` 的方式去 debug 命令的执行过程，发现问题。
- 3、使用 `truss` 等命令进行时间分析。