

ORACLE 数据库异常应急流 程

BY SHOUG. 林少杰
(KEVINLIN.ORA)

SH'OUG

SHANGHAI ORACLE USERS GROUP

上海ORACLE用户组

How to Find SHOUG?



The image shows a Google search interface. The search bar contains the text '上海oracle用户组'. Below the search bar, there are navigation tabs for 'Web', 'Images', 'Maps', 'Shopping', 'More', and 'Search tools'. The 'Web' tab is selected. The search results show 'About 5,960,000 results (0.36 seconds)'. The first result is titled '上海Oracle用户组| SHOUG, 走近全系Oracle技术和数据库专家' with the URL 'www.shoug.info/'. Below the title, there is a snippet of text: 'SHOUG的全称是ShangHai Oracle Users Group, 中文为上海Oracle用户组。 SHOUG的成员仅仅局限于上海地区吗? 上海是国际化大都市, 我们将以上海为中心, ...'. Below the snippet, it says 'You visited this page on 5/20/13.'. The second result is titled 'Oracle 12c新特性- ORACLE数据库数据恢复、性能优化、故障诊断来 ...' with the URL 'www.askmaclean.com/archives/.../oracle/oracle-12c'. Below the title, there is a snippet of text: 'Feb 26, 2013 - 《Oracle 12c新特性》-作者: Maclean Liu, 首发于Ask Maclean中文Oracle博客. ... 手机: 13764045638, ORA-ALLSTARS Exadata用户组QQ群:23549328 ... Database 12c进入release发布的倒计时, 可能在今年7月在上海举行 ...'. Below the snippet, it says 'You've visited this page 4 times. Last visit: 4/25/13'.

数据库异常应急流程

Steps for Database hang

本文作者：林少杰 (kevinlin.ora@gmail.com)

摘要： 无

关键词： hang

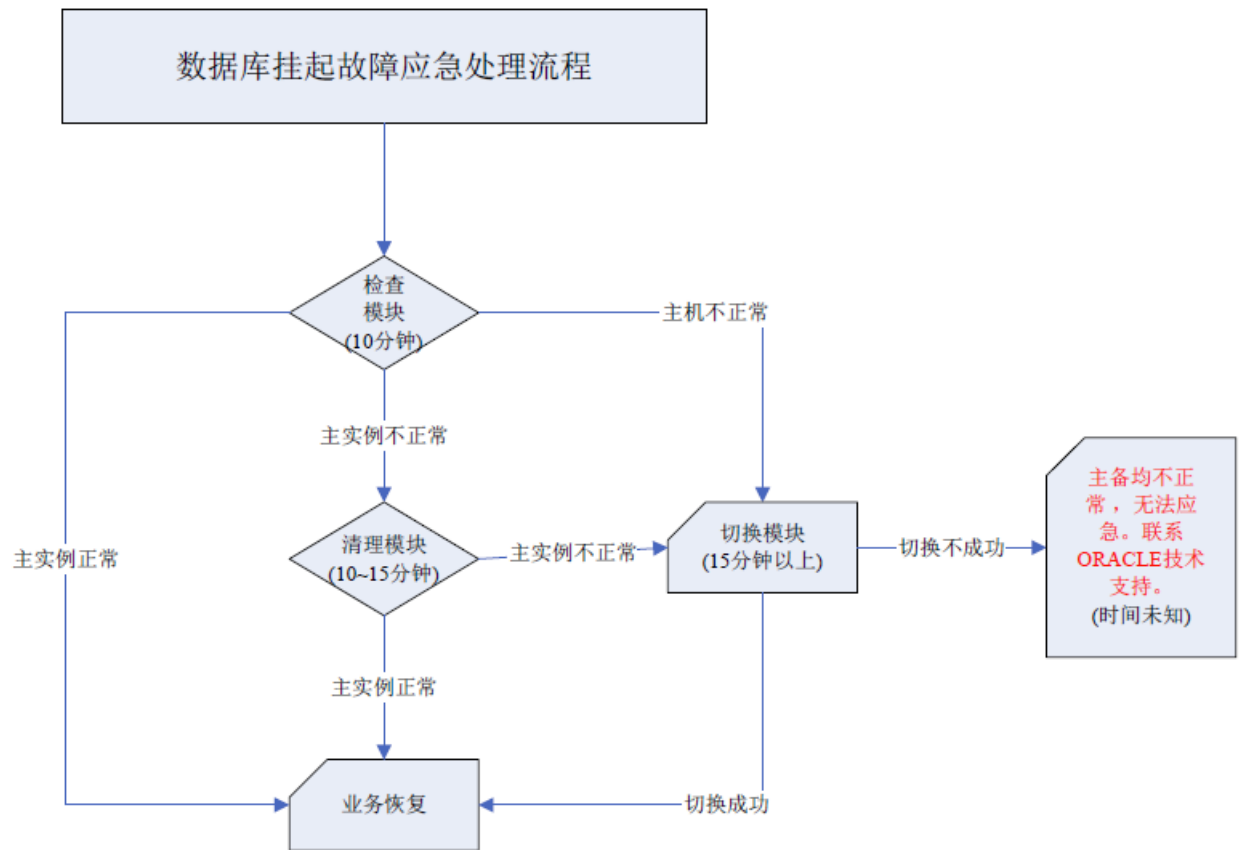
背景

为了尽快解决数据库异常问题，如挂起、阻塞、无法归档等，初步建立数据库异常应用处理机制，最终目的是减少故障对正常业务使用的影响时间。

具体内容：

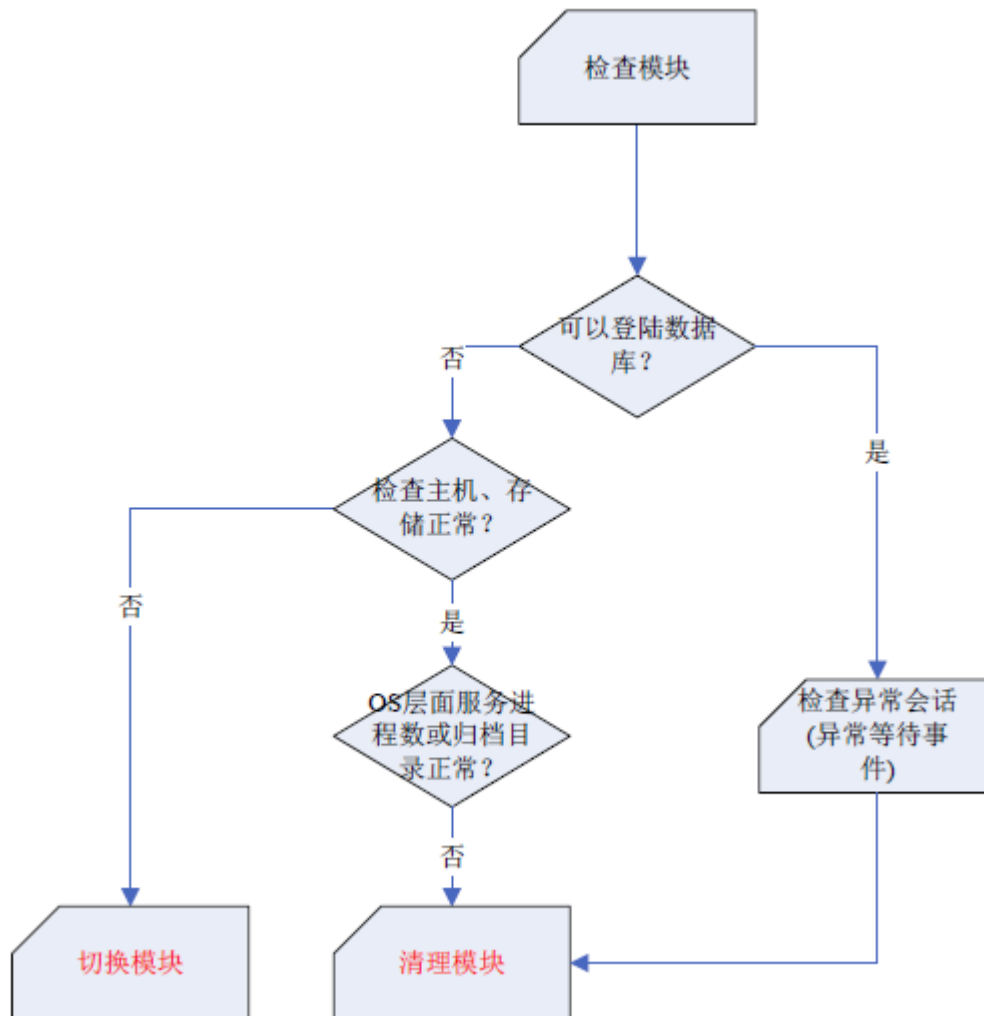
一、流程主模块

跳转：1.[主模块](#) 2.[检查模块](#) 3.[清理模块](#) 4.[切换模块](#)



二、检查模块

跳转：1.[主模块](#) 2.[检查模块](#) 3.[清理模块](#) 4.[切换模块](#)



1. 检查是否可以登录数据库

使用安装 oracle 软件的用户登录操作系统，进行以下步骤：

```
$export ORACLE_SID=sidname(指定所要登录的实例名)
```

```
$ Sqlplus /nolog
```

```
SQL> conn / as sysdba
```

```
SQL>show parameter instance
```

如返回 SQL>提示符，则可以正常登录，否则认为不可正常登录。

2. 检查主机、存储是否正常

```
$ errpt|more
```

检查详细输出

3. 检查服务进程数与归档目录

检查服务进程数：\$ ps -ef|grep LOCAL=NO|grep sidname|wc -l，检查
\$ORACLE_BASE/admin/dbname/udump 下相关时间段的 trace 文件。

检查归档目录：

```
$ df -k
```

4. 检查异常会话

登录数据库后：

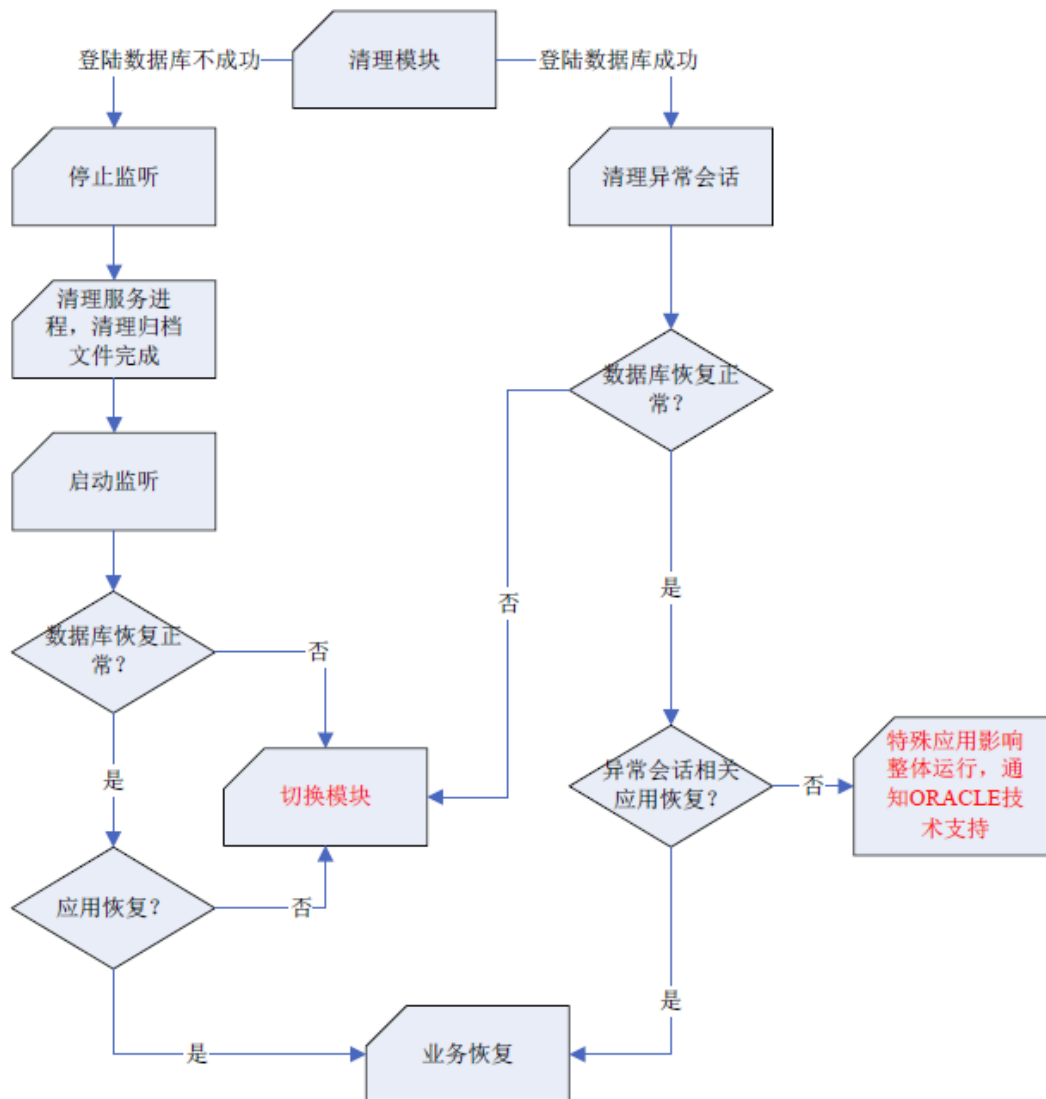
```
SQL> select sid,serial#,seq#,event,wait_time,seconds_in_wait,p1,p2,p3 from v$session  
where event not like '%SQL%client%' and event not like '%Streams%' order by event;
```

```
SQL>select
```

```
a.sid,a.program,a.sql_id,a.event,a.seq#,a.seconds_in_wait,b.sid,b.program,b.sql_id,b.event,b.seq  
#,b.wait_time,b.seconds_in_wait from V$session a,V$session b where  
b.sid=a.BLOCKING_SESSION;
```

三、清理模块

跳转：1.主模块 2.检查模块 3.清理模块 4.切换模块



1. 停止监听

\$ ps -ef|grep tns 查看监听服务名

\$ lsnrctl stop listener_name

2. 清理异常会话

检查到异常会话(此时已获取 sid)后：

SQL> select spid from V\$process where addr=(select paddr from V\$session where sid=&sid);

```
SQL>host
$ kill -9 spid
```

3. 清理服务进程

```
$ ps -ef|grep LOCAL=NO|grep sidname
将输出复制到文本编辑器，按 ospid 操作系统进程号编辑 kill -9 命令
$ 在此运行 kill -9 命令
```

4. 清理归档目录

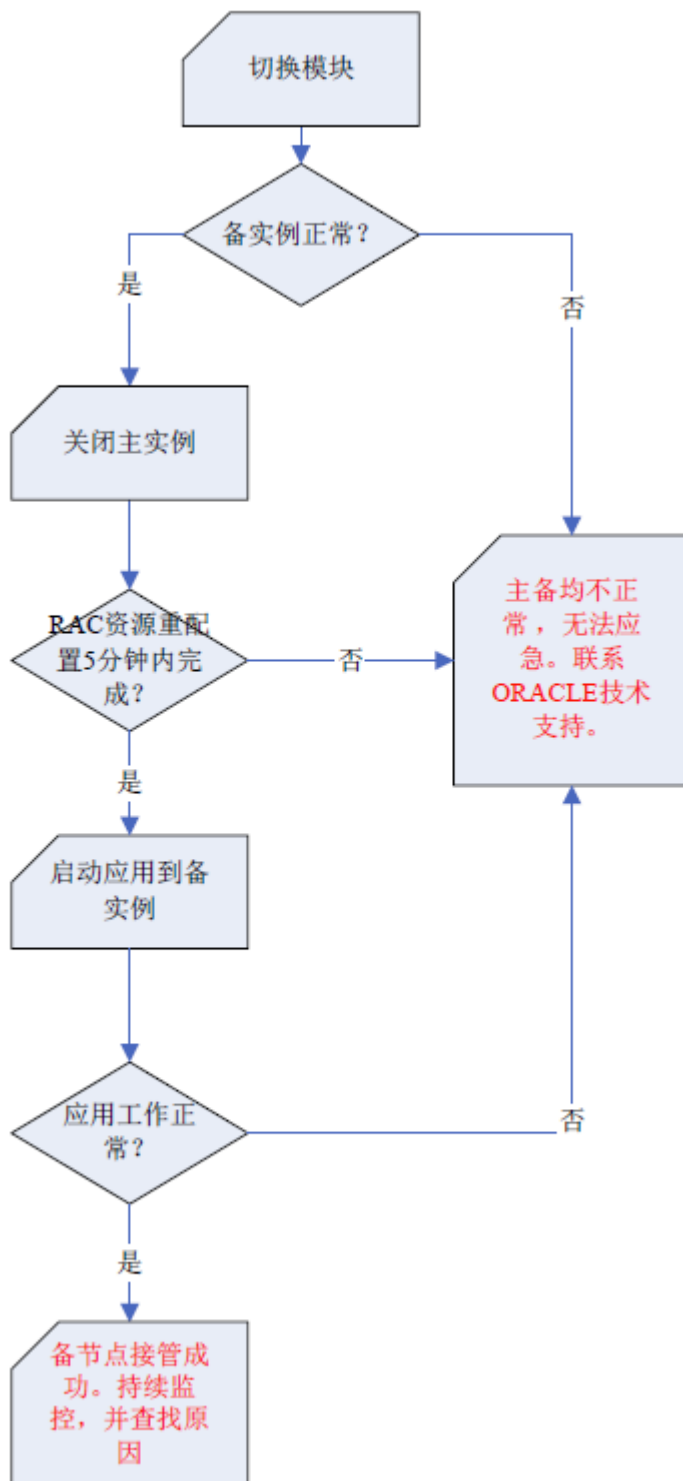
```
进入归档目录后：
$ ls -lrt
$ ls|grep arc |xargs -n 10 rm -f 或按 ls -lrt 输出在文本编辑器中编辑 rm -f 命令
如果使归档目录满的文件不是归档文件，则将相关文件删除。
```

5. 判断数据库是否正常

```
应用进行相关操作，数据库中监控：
$ export ORACLE_SID=sidname
$ sqlplus /nolog
SQL> conn / as sysdba
SQL> select * from v$lock where type<>'MR' order by type;
select sid,serial#,seq#,event,wait_time,seconds_in_wait,p1,p2,p3 from v$session where
event not like '%SQL%client%' and event not like '%Streams%' order by event;
SQL> select sid,serial#,username,program,sql_id,machine from V$session where
username is not null and status='ACTIVE' order by program,sql_id;
SQL> host
$ cd $ORACLE_BASE/admin/dbname/bdump
$ tail -300f alert_sidname.log
```


四、切换模块

跳转：1.[主模块](#) 2.[检查模块](#) 3.[清理模块](#) 4.[切换模块](#)



1. 判断备实例正常

```

$ export ORACLE_SID=sidname
$ sqlplus /nolog
SQL> conn / as sysdba
SQL> select * from v$lck where type<>'MR' order by type;
select sid,serial#,seq#,event,wait_time,seconds_in_wait,p1,p2,p3 from v$session where
event not like '%SQL%client%' and event not like '%Streams%' order by event;
  
```

```
SQL> select sid,serial#,username,program,sql_id,machine from V$session where
username is not null and status='ACTIVE' order by program,sql_id;
```

```
SQL> host
```

```
$ cd $ORACLE_BASE/admin/dbname/bdump
```

```
$ tail -300f alert_sidname.log
```

2. 关闭主实例

停止监听，服务进程清理后：

```
$ export ORACLE_SID=sidname
```

```
$ sqlplus /nolog
```

```
SQL> conn / as sysdba
```

```
SQL> shutdown immediate;
```

无法登录数据库的情况下：

```
$ ps -ef|grep pmon|grep sidname
```

```
$ kill -9 spid_of_pmon
```

```
$ cd $ORACLE_BASE/admin/dbname/bdump
```

```
$ tail -300f alert_sidname.log
```

确认数据库已关闭：

```
$ ps -ef|grep sidname
```

```
$ ipcs -m
```

3. 判断 RAC 资源重配完成

登录备实例节点：

```
$ cd $ORACLE_BASE/admin/dbname/bdump
```

```
$ tail -300f alert_sidname.log
```

当告警日志中出现以下信息，说明资源已重新配置完成：

```
“Reconfiguration complete”
```

4. 持续监控数据库

```
$ export ORACLE_SID=sidname
```

```
$ sqlplus /nolog
```

```
SQL> conn / as sysdba
```

```
SQL> select * from v$lock where type<>'MR' order by type;
```

```
select sid,serial#,seq#,event,wait_time,seconds_in_wait,p1,p2,p3 from v$session where
event not like '%SQL%client%' and event not like '%Streams%' order by event;
```

```
SQL> select sid,serial#,username,program,sql_id,machine from V$session where
username is not null and status='ACTIVE' order by program,sql_id;
```

```
$ cd $ORACLE_BASE/admin/dbname/bdump
```

```
$ tail -300f alert_sidname.log
```

五、附录- ORACLE 推荐收集的诊断信息，以及 SQL 信息获取

1. ORACLE 建议如果情况再现，收集 systemstate 或 hanganalyze 信息，用于确定原因。

```
oradebug setmypid
oradebug unlimit
oradebug setinst all
oradebug -g all hanganalyze 4
--wait several seconds
oradebug -g all hanganalyze 4
--wait several seconds
oradebug -g all hanganalyze 4

oradebug -g all dump systemstate 267
--wait several seconds
oradebug -g all dump systemstate 267
--wait several seconds
oradebug -g all dump systemstate 267

oradebug tracefile_name
```

2. 获取 SQL 历史信息及执行计划

```
spool hist_sql.lst
undefine sql_id
define sql_id='<your_SQL_ID>'

set null null
set lines 320
set pages 99
set trimspool on
col snap_beg format      a12
col iowait_delta          format 99999999.99 heading io|wait|delta|(ms)
col iowait_total          format 99999999.99 heading io|wait|total|(ms)
col ELAPSED_TIME_DELTA   format 99999999.99 heading
elapsd|time|delta|(ms)
col CPU_TIME_DELTA       format 99999999.99 heading cpu|time|delta|(ms)
col PLAN_HASH_VALUE      heading plan_hash|value
```

```

col CONCURRENCY_WAIT_delta format 99999999.99 heading conc|wait|delta|(ms)
col CLUSTER_WAIT_DELTA format 99999999.99 heading clust|wait|delta|(ms)
col PX_SERVERS_EXECS_DELTA format 99999 heading PXServ|Exec|delta
col APWAIT_DELTA format 99999 heading appl|wait|time|delta|(micro)
col PLSEXEC_TIME_DELTA format 99999 heading plsqli|exec|time|delta|(micro)
col JAVAEXEC_TIME_DELTA format 99999 heading java|exec|time|delta|(micro)
col optimizer_cost format 9999 heading opt|cost
col optimizer_mode format a10 heading optim|mode
col kept_versions format 999 heading kept|vers
col invalidations_delta format 999 heading inv|alid|dlt
col parse_calls_delta format 99999 heading parse|calls|delta
col executions_delta format 999999 heading exec|delta
col fetches_delta format 9999999 heading fetches|delta
col end_of_fetch_count_delta format 99999 heading end|of|fetch|call|delta
col buffer_gets_delta format 9999999999 heading buffer|gets|delta
col disk_reads_delta format 9999999999 heading disk|reads|delta
col DIRECT_WRITES_DELTA format 99999999 heading
direct|writes|delta
col rows_processed_delta format 9999999999 heading rows|processed|delta
col rows_ex format 999999999 heading rows|exec
col snap_id format 99999 heading snap|id
col ela_ex format 999999999.99 heading elapsed|per|execution
col cwt_ex format 999999999.99 heading cwt|per|execution
col cc_ex format 999999999.99 heading cc|per|execution
col io_ex format 999999999.99 heading io|per|execution
col instance_number format 99 heading in|ID

```

```

select dba_hist_sqlstat.instance_number, sql_id, plan_hash_value,
dba_hist_sqlstat.snap_id,
to_char(dba_hist_snapshot.BEGIN_INTERVAL_TIME,'dd-mm hh24:mi') snap_beg,
invalidations_delta,
parse_calls_delta,
executions_delta,
px_servers_execs_delta,
fetches_delta,
buffer_gets_delta,
disk_reads_delta,
direct_writes_delta,
rows_processed_delta,
elapsed_time_delta/1000 elapsed_time_delta,
cpu_time_delta/1000 cpu_time_delta,
iowait_delta/1000 iowait_delta,
clwait_delta/1000 cluster_wait_delta,
ccwait_delta/1000 concurrency_wait_delta,
substr(optimizer_mode,1,3) opt,
case when executions_delta = 0 then NULL
when cpu_time_delta = 0 then NULL
else
(cpu_time_delta/executions_delta)/1000
end cpu_ex,

```

```

case when executions_delta = 0 then NULL
when elapsed_time_delta = 0 then NULL
else
(elapsed_time_delta/executions_delta)/1000
end ela_ex
,substr(SQL_PROFILE,1,32) sql_profile
from dba_hist_sqlstat, dba_hist_snapshot
where sql_id='&&sql_id'
and dba_hist_sqlstat.snap_id=dba_hist_snapshot.snap_id
and dba_hist_sqlstat.instance_number=dba_hist_snapshot.instance_number
order by dba_hist_sqlstat.instance_number, plan_hash_value, dba_hist_sqlstat.snap_id
/

```

```

select dba_hist_sqlstat.instance_number, sql_id, plan_hash_value,
dba_hist_sqlstat.snap_id,
to_char(dba_hist_snapshot.BEGIN_INTERVAL_TIME,'dd-mm hh24:mi') snap_beg,
invalidations_delta,
parse_calls_delta,
executions_delta,
elapsed_time_delta/1000 elapsed_time_delta,
cpu_time_delta/1000 cpu_time_delta,
iowait_delta/1000 iowait_delta,
clwait_delta/1000 cluster_wait_delta,
ccwait_delta/1000 concurrency_wait_delta,
substr(optimizer_mode,1,3) opt,
case when executions_delta = 0 then NULL
when rows_processed_delta = 0 then NULL
else
(rows_processed_delta/executions_delta)
end rows_ex,
case when executions_delta = 0 then NULL
when iowait_delta = 0 then NULL
else
(iowait_delta/executions_delta)/1000
end io_ex,
case when executions_delta = 0 then NULL
when clwait_delta = 0 then NULL
else
(clwait_delta/executions_delta)/1000
end cwt_ex,
case when executions_delta = 0 then NULL
when ccwait_delta = 0 then NULL
else
(ccwait_delta/executions_delta)/1000
end cc_ex,
case when executions_delta = 0 then NULL
when cpu_time_delta = 0 then NULL
else
(cpu_time_delta/executions_delta)/1000
end cpu_ex,

```

```

case when executions_delta = 0 then NULL
when elapsed_time_delta = 0 then NULL
else
(elapsed_time_delta/executions_delta)/1000
end ela_ex
from dba_hist_sqlstat, dba_hist_snapshot
where sql_id='&&sql_id'
and dba_hist_sqlstat.snap_id=dba_hist_snapshot.snap_id
and dba_hist_sqlstat.instance_number=dba_hist_snapshot.instance_number
order by dba_hist_sqlstat.instance_number, plan_hash_value, dba_hist_sqlstat.snap_id
/

select plan_table_output from table (dbms_xplan.display_awr('&&sql_id',null, null,
'ADVANCED +PEEKED_BINDS'));

select plan_table_output from table (dbms_xplan.display_cursor('&&sql_id', null,
'ADVANCED +PEEKED_BINDS'));

spool off

```

3. sqlt Xtract 与 sql advisor

please generate and upload a sqlt Xecute for the sql when it runs quickly and sqlt Xtract for the slow sql_id='<your sql>'.

The steps below can be used to generate the sqlt_s#####.zip file:

From sqlt_instructions.txt file:

For the Xtract method:

```

# cd sqlt/run
# sqlplus apps
SQL> START sqltxtract.sql [SQL_ID][HASH_VALUE]
SQL> START sqltxtract.sql 9gqan20q8cdf

```

For the Xecute method:

```

# cd sqlt
# sqlplus apps
SQL> START [path]sqltxecute.sql [path]scriptname
SQL> START run/sqltxecute.sql input/sample/script1.sql

```

The tool will create a sqlt...zip file each time it is executed. Please upload the sqlt_s#####.zip file to the SR.

please check the SQL tuning advisor for tuning recommendations. The script listed below

```

should allow you to generate the tuning output:
Login as SYS and run the following commands
===== Begin script =====
variable stmt_task VARCHAR2(64);
variable sts_task VARCHAR2(64);

```

```
EXEC :sts_task := DBMS_SQLTUNE.CREATE_TUNING_TASK( -
sql_id => '9gqan20q8cdff', -
task_name => 'sql_tuning_task1',-
description => 'Tune my query');
EXEC DBMS_SQLTUNE.EXECUTE_TUNING_TASK('sql_tuning_task1');
SET LONG 1000000
SET LONGCHUNKSIZE 1000
SET LINESIZE 100
SELECT DBMS_SQLTUNE.REPORT_TUNING_TASK( 'sql_tuning_task1') from
DUAL;
===== End Script =====
This script may take some time to execute, so please allow it some time to complete.
Please upload the output to the SR.
```

作者个人简介

林少杰(kevinlin.ora)是多维创新有限公司(国内知名 Oracle 第三方服务)上海地区的技术总监。拥有 7 年 ORACLE 数据库系统的技术支持服务经验，工作环境主要为 AIX、Solaris 以及 HP 平台上的 Oracle 11g, Oracle 10g ,Oracle 9i, Oracle 8i 集群环境。所服务行业有电信行业、金融服务业、物流业等。

林少杰(kevinlin.ora)在数据库以及 SQL 层面的性能调优有丰富经验，熟悉备份恢复、数据库安装配置。擅长异常问题的分析诊断。

林少杰 (kevinlin.ora) 一直以来遵循业界的最佳实践，提供并实施高质量的解决方案，不断坚持与增强技术服务的质量。

作为服务团队带头人曾参与以下项目：

- 上海联通信息化的核心 Oracle 数据库维护服务项目。
- 银联数据服务有限公司的数据库技术支持服务项目。
- 延边朝鲜族自治州住房公积金管理中心数据库技术支持服务项目。
- 上海浦东国际集装箱码头有限公司数据库技术支持服务项目。
- 银联数据服务有限公司的数据库技术支持服务项目。
- 上海联通信息化的 BSS 核心 Oracle 数据库维护服务项目。
- 上海银商资讯公司的数据库技术支持服务项目。