**数据库巡检手册**



青岛慧海联创信息技术有限公司

文档信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 文档名称编号 | 数据库巡检手册（HHLC-ITSS-SJKXJSC） | | | |
| 编制单位 | 青岛慧海联创信息技术有限公司 | | | |
| 文档版本 | 版本日期 | 版本说明 | 作者 | 审核 |
| V1.0 | 2025-5-5 | 发布版本 | 郑永伟 | 张仲全 |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **数据库检测** | | |
| **硬件机型** | | **HP** **DL580** **G7** |
| **是否集群** | | **是** |
| **系统实际用户数** | | **10** |
| **数据库进程** | | |
| **进入操作系统，登陆** **Oracle** **用户，命令：su** **-** **oracle** | | |
| **Process** | **进程情况** | **进入操作系统，登陆** **Oracle** **用户**  **命令：ps** **-ef|grep** **ora\_** |
| **说明** | Oracle 10g 后台进程  SMON(System Monitor)用于执行历程恢复、合并空间碎片并释放临时段。  PMON(Process Monitor)用于监视服务器进程的执行，摒弃在服务器进程失败时清除该服务器进程。  DBWR(Database Writer)用于将数据库缓存的脏缓冲区数据写入到数据文件中。  LGWR(Log Writer)用于将重做日志缓冲区所记载的全部内容写入到充作日志文件中。  CKPT(Checkpoint Process)用于发出检查点(Checkpoint)，检查点会同步数据库的数据文件、控制文件和重做日志，当发出检查点时，后台进程 CKPT将检查点时刻的 SCN(System Change Number)写入到控制文件和数据文件头部，同时促使后台进程 DBWR 将所有脏缓冲区写入到数据文件中。  ARCH(Archive Process)用于将重做日志的事物变化复制到归档日志中，该后台进程只有在 archivelog 模式下才有意义。  Oracle 11g 新增的后台进程  DBRM  说明: 数据库资源管理进程(The database resource manager process),负责设置资源计划和其他的资源管理的工作.  DIAG  说明 : 数据库诊断进程(The diagnosibility process),负责维护管理各种用于诊断的转储文件,并执行 oradebug命令。  DIA0  说明 : 另一个数据库诊断进程，负责检测 Oracle 数据库中的挂起(hang)和死锁的处理。  PSP0  说明 : process spawner，用于产生 oracle 进程  SMCO  说明: space management coordinator，该进程负责空间管理协调管理工作，负责执行空间的分配和回收。  Wnnn  说明 : 命名为 W000，W001，W002.....，由 smcO 动态产生执行上述相关任 |

|  |  |  |
| --- | --- | --- |
|  |  | 务。  VKTM  说明 : virtual keeper of time，用于提供 wall-clock time，（每秒钟更新一次） 。提供每二十毫秒更新一次的  reference-time counter，看起来有点类似计时器的功能。  GMON  说明 : 用于维护 asm 磁盘组的磁盘之间的关系。  KATE  说明 : 当 ASM 的磁盘离线的时候，该进程负责 asm 的元文件的 io 读写。  MARK  说明 : 如果有向 asm 离线磁盘的missed 写请求，该进程将 ASM 分配的单元的状态标记为 stale  FBDA  说明 : 涉及到 flashback-data-archive 新特性的一个进程，The flashback data archiver proces。用于将"轨表" (tracked  tables)的历史数据进行归档。当"轨表"上的事务提交以后，fbda 进程负责将数据的前镜像保存到 flashback archive 区域。  该进程还负责 flashback 的数据归档的空间管理、分配、保留，跟踪 tracked transactions。  什么是"轨表" (tracked tables) : 是指启用了 flashback archive 特性的表。  RMSn  说明 : The Oracle RAC management processes,负责执行 Oracle RAC 的管  理任务，比如 RAC 相关资源的创建和集群中新实例的添加。  DSKM  说明: The slave diskmon process ， 负责 oracle 实例、asm 实例和磁盘的管理进程之间的 io fencing 信息的交换。如果  使用 SAGE 的存储，该进程还负责 SAGE 存储的一些信息的管理。 RAC 进程  GSD global services daemon 全局服务守护进程lock process(LCK) 锁管理进程  DIAG:DIAGNOSABILITY DAEMON 失败进程的诊断信息捕获进程Operating System-Dependent(OSD) 操作系统资源访问进程LMS - Gobal Cache Service Process 全局缓存服务进程  LMD - Global Enqueue Service Daemon 全局查询服务守护进程LMON-Global Enqueue Service Monitor 全局查询服务监视进程LCK0 - Instance Enqueue Process 实例查询进程 |
| **进程状态** |  |
| **Space** | **使用情况** | df -h |
| **Listener** | **监听状态** | 命令：lsnrctl status  配置正常 |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **运行情况** | 命令：sqlplus “/as sy sdba ” |
| **查看日志** | 命令：  More /u01/app/oracle/product/11.2.0/db\_1/network/log/sqlnet.log有错误才有日志 |
| **Tnsname** | **配置情况** | 标准配置  位置：/u01/app/oracle/product/11.2.0/db\_1/network/admin |
| **运行情况** | 正常 |
| **查看状态** | Tnsping tnsname(数据库实例名) |
| **Alert** | **配置情况** | 标准配置  /u01/app/oracle/diag/rdbms/<SID>/<INSTANCE>/trace/alert\_sid.log |
| **运行情况** | 正常  More alert\_sid.log |
| **CRS** | **服务运行情况** | crsctl check crs  检查 crs 的健康情况 |
| **资源运行情况** | crs\_stat -t  用来查看 RAC 中各节点上 resources 的运行状况,Resources 的属性等 |
| **OCR** | **运行情况** | ocrcheck  验证 OCR 的状态以及空间使用情况 |
| **Voting** **Disk** | **运行情况** | crsctl query css votedisk  Votingdisk 状态查询 |
| **ASM** | **运行情况** | crs\_stat -t | grep asm |
| **使用情况** | # su - grid  $ asmcmd  ASMCMD> ls  DATA/  ORA\_DATA/  ASMCMD> lsdg ora\_data |
| 数据库  su - oracle  sqlplus “/as sy sdba ” | | |
|  | | 诊断结果及建议 |
| **DBstatus** | **数据库状态** | select status from v$instance ; |
| **配置情况** | Select \* from v$parameter ; |
| **运行情况** | 正常 |
| **使** **用资** **源情况** | select \* from v$resource\_limit； |

|  |  |  |
| --- | --- | --- |
| **ASM** | **使用情况** | select group\_number,name,total\_mb,free\_mb from v$asm\_diskgroup ; |
| **DBfile** | **数据文件状态** | select name,status from v$datafile ; |
| **控制文件状态** | select status,name from v$controlfile ; |
| **日志文件状态** | select group#,members,archived,status from v$log ; |
| **表空间使用率** | set pagesize 50  column "Tablespace" format a13  column "UsedMB" format 99,999,999  column "FreeMB" format 99,999,999  column "Total MB" format 99,999,999 select  fs.tablespace\_name "Tablespace",  (df.totalspace - fs.freespace) "Used MB",  fs.freespace "FreeMB",  df.totalspace "Total MB",  round(100 \* (fs.freespace / df.totalspace)) "Pct Free" from  (select  tablespace\_name,  round(sum(bytes)/1048576) TotalSpace  from  dba\_data\_files  group by  tablespace\_name ) df,  (select  tablespace\_name,  round(sum(bytes)/1048576) FreeSpace  from  dba\_free\_space  group by  tablespace\_name  ) fs  where df.tablespace\_name=fs.tablespace\_name  order by "Pct Free" / |
| **运行情况** | 正常 |
| **Sessions** | **并发数** | select count(\*) from v$session where status='ACTIVE' ; |

|  |  |  |
| --- | --- | --- |
| **Redo** | **使用情况** | select  b.THREAD#,a.GROUP#,a.STATUS,a.MEMBER,b.BYTES,b.ARCHIVED,b.STATUS from v$logfile a,v$log b where a.GROUP#=b.GROUP# ; |
| **Performan** **ce** | **配置情况** |  |
| **察看数据库锁表** | 锁表有时候是瞬间的，长时间锁定的表才可能是死锁。  select l.\*, s.OSUSER, s.ACTION, o.OBJECT\_NAME  from gv$locked\_object l, gv\_$session s, all\_objects o where l.SESSION\_ID=s.SID  and o.OBJECT\_ID=l.OBJECT\_ID |
| **死锁** | set linesize 200  column oracle\_username for a16  column os\_user\_name for a12  column object\_name for a30  SELECT l.xidusn,  l.object\_id,l.oracle\_username,l.os\_user\_name,l.process, l.session\_id,s.serial#, l.locked\_mode,o.object\_name  FROM v$locked\_object l,dba\_objects o,v$session s  where l.object\_id = o.object\_id and s.sid = l.session\_id ; select  t2.username||' ' ||t2.sid||' ' ||t2.serial#||' ' ||t2.l ogon\_time||' ' ||t3.sql\_text  from v$locked\_object t1,v$session t2,v$sqltext t3 where t1.session\_id=t2.sid  and t2.sql\_address=t3.address  order by t2.logon\_time ; |
| **enqueue等待** | Selecteq\_type"lock",total\_req# "gets",total\_wait#  "waits",cum\_wait\_time from v$enqueue\_stat where total\_wait#>0 ; |

|  |  |  |
| --- | --- | --- |
|  | **等待事件** | set linesize 200  column username for a12  column program for a30  column event for a28  column p1text for a15  column p1 for 999,999,999,999,999  select s.username,s.program,sw.event,sw.p1text,sw.p1 from v$session s,v$session\_wait sw  where s.sid=sw.sid and s.status='ACTIVE' order by sw.p1 ;  select event,p1 "File #",p2 "Block #",p3 "Reason Code" from  v$session\_wait order by event ;  where event = 'buffer busy waits' ;  select owner,segment\_name,segment\_type,file\_id,block\_id from dba\_extents  where file\_id = &P1 and &P2 between block\_id and block\_id + blocks -1 ; column event for a35 ;  column p1text for a40 ;  select sid,event,p1,p1text from v$session\_wait order by event； |
| **RAC** **全** **局等** **待** **事件** | select \* from v$event\_name where NAME like 'gc%' and WAIT\_CLASS='Cluster' ; |
| **数据文件IO** | select fs.name name,f.phyblkrd pbr,f.phyblkwrt pbw,f.phyblkrd pyr,f.readtim,f.writetim  from v$filestat f, v$dbfile fs  where f.file# = fs.file# order by 2,3,4 desc ; |
| **表空间** **IO** | select tablespace\_name,sum(f.phyblkrd) pbr,sum(f.phyblkwrt) pbw,sum(f.phyblkrd) pyr,sum(f.readtim),sum(f.writetim)  from v$filestat f, dba\_data\_files fs  where f.file# = fs.file\_id group by tablespace\_name  order by 2,3,4 desc ; |

|  |  |  |
| --- | --- | --- |
|  | **长事务** | set linesize 200  column name for a16  column username for a10  select a.name,b.xacts,c.sid,c.serial#,c.username,d.sql\_text  from v$rollname a,v$rollstat b,v$session c,v$sqltext  d,v$transaction e where a.usn=b.usn and b.usn=e.XIDUSN and c.taddr=e.addr  and c.sql\_address=d.ADDRESS  and c.sql\_hashvalue=d.hash\_value order by a.name,c.sid,d.piece ; |
| **大事务** | select sid,serial#,to\_char(start\_time,'yyyy-mm-dd hh24:mi :ss')  start\_time,sofar,totalwork, (sofar/decode(totalwork,0,1,totalwork))  \*100 ratio,message from v$session\_longops where message like '%RMAN%' ;  select sid,serial#,to\_char(start\_time,'yyyy-mm-dd hh24:mi :ss')  start\_time,sofar,totalwork, (sofar/decode(totalwork,0,1,totalwork))  \*100 ratio,message from v$session\_longops where sofar <> totalwork ;  where (sofar/totalwork)\*100 < 100 ; |
| **察看用户session数量** | Select count(\*) from v$session ;  如果 session 过多，察看 select \* from v$session，察看是什么程序 |
| **归档的生成频率** | set linesize 120  column begin\_time for a26  column end\_time for a26  select a.recid,to\_char(a.first\_time,'yyyy-mm-dd hh24:mi :ss') begin\_time,  b.recid,to\_char(b.first\_time,'yyyy-mm-dd hh24:mi :ss') end\_time, round((b.first\_time - a.first\_time)\*24\*60,2) minutes  from v$log\_history a,v$log\_history b where b.recid = a.recid+1 ; |
| **缓冲区命中率** | SELECT (1 - (SUM(DECODE(NAME, 'physical reads', VALUE, 0)) / (SUM(DECODE(NAME, 'db block gets', VALUE, 0)) +  SUM(DECODE(NAME, 'consistent gets', VALUE, 0))))) \* 100 "缓冲区命中率"  FROM V$SYSSTAT ; |

|  |  |  |
| --- | --- | --- |
|  | **重做日志命中率** | SELECT name, gets, misses, immediate\_gets, immediate\_misses,  100 - Decode(gets,0,0,misses/(gets+misses))\*100 ratio1,  100 - Decode(immediate\_gets+immediate\_misses,0,0,immediate\_misses/(imm ediate\_gets+immediate\_misses))\*100 ratio2  FROM v$latch WHERE name IN ('redo allocation', 'redo copy') ; |
| **低效** **SQL** | SELECT EXECUTIONS , DISK\_READS, BUFFER\_GETS,  ROUND((BUFFER\_GETS-DISK\_READS)/BUFFER\_GETS,2) Hit\_radio, ROUND(DISK\_READS/EXECUTIONS,2) Reads\_per\_run,  SQL\_TEXT  FROM V$SQLAREA  WHERE EXECUTIONS>0 AND BUFFER\_GETS > 0  AND (BUFFER\_GETS-DISK\_READS)/BUFFER\_GETS < 0.8 ORDER BY 4 DESC ; |
| **TOP** **SQL** | select sql\_text,buffer\_gets,executions,buffer\_gets/executions from v$sqlarea where buffer\_gets>1000000 order by 4 desc； |
| **临时表空间情况** | SELECT se.username 用户名, se.sid,  se.serial#,  se.sql\_address, se.machine,  se.program,  su.tablespace,  su.blocks\*8192/1024/1024 "Used Space(M)", su.segtype,  su.contents  FROM v$session se, v$sort\_usage su  WHERE se.saddr=su.session\_addr ; |
| **在线日志情况** | SELECT  B.RECID,B.FIRST\_TIME,A.FIRST\_TIME,ROUND((A.FIRST\_TIME-B.FIRST\_TI ME)\*24\*60,2) MINATES  FROM V$LOG\_HISTORY A,V$LOG\_HISTORY B  WHERE A.RECID=B.RECID +1 AND A.FIRST\_TIME>SYSDATE - 20 AND ROUND((A.FIRST\_TIME-B.FIRST\_TIME)\*24\*60,2)<30  ORDER BY A.FIRST\_TIME DESC ; |
| **无效索引** | SELECT OWNER,INDEX\_NAME,INDEX\_TYPE,TABLE\_NAME,STATUS FROM DBA\_INDEXES  WHERE STATUS = 'UNUSABLE' ; |