

1.多通道并行备份

RMAN> report schema;

=====

File Size(MB)	Tablespace	RB segs	Datafile Name
---------------	------------	---------	---------------

1	1024	SYSTEM	***
/u01/app/oracle/oradata/sztech1/system01.dbf			
2	650	SYSAUX	***
/u01/app/oracle/oradata/sztech1/sysaux01.dbf			
3	105	UNDOTBS1	***
/u01/app/oracle/oradata/sztech1/undotbs01.dbf			
4	1024	USERS	***
/u01/app/oracle/oradata/sztech1/users01.dbf			
5	346	EXAMPLE	***
/u01/app/oracle/oradata/sztech1/example01.dbf			
6	10	TS_INVENTORY	***
/u01/app/oracle/oradata/sztech1/ts_inventory01.dbf			

RMAN> **run {**

allocate channel c1 device type disk;

allocate channel c2 device type disk;

allocate channel c3 device type disk;

backup incremental level 0

(datafile 1,6 channel c1)

(datafile 2,3 channel c2)

(datafile 4 channel c3);

sql "alter system archive log current";

}

--上面的脚本中手动指定每个通道备份哪些数据文件，但漏写了数据文件5，因此5号文件将不会被备份。

--备份时也可能又新增了数据文件，可以使用backup database not backed up备份新增的或未备份的数据文件。

RMAN> **run {**

allocate channel c1 device type disk;

allocate channel c2 device type disk;

allocate channel c3 device type disk;

backup incremental level 0

(datafile 1,6 channel c1)

(datafile 2,3 channel c2)

(datafile 4 channel c3);

backup database not backed up;

```
sql "alter system archive log current";
}
```

--备份"在限定时间周期内"没有被备份的数据文件
backup database not backed up since time='sysdate-2';

--我们可以只分配多通道，而不指定每个通道具体备份哪几个数据文件，系统自动根据文件大小进行通道分配。

```
RMAN> run {
  allocate channel c1 device type disk;
  allocate channel c2 device type disk;
  allocate channel c3 device type disk;
  backup incremental level 0 database ;    --未指定每个通道对应的数据文件
  sql "alter system archive log current";
}
```

2.监控Rman进程

①查看rman的进程信息

```
SQL> select s.sid,p.spid,s.client_info
       from v$process p,v$session s
       where p.addr=s.paddr
       and  client_info like 'rman%';
```

SID	SPID	CLIENT_INFO
151	4041	rman channel=c1
17	4042	rman channel=c2
146	4043	rman channel=c3

--查看操作系统资源使用情况,wa表示IO资源等待。

```
[root@dbserver ~]# top
```

```
top - 10:46:35 up 1:52, 3 users, load average: 0.35, 0.48, 0.49
```

```
Tasks: 177 total, 1 running, 176 sleeping, 0 stopped, 0 zombie
```

```
Cpu(s): 0.2%us, 0.3%sy, 0.0%ni, 99.3%id, 58.9%wa, 0.0%hi, 0.2%si, 0.0%st
```

```
Mem: 4151252k total, 3865768k used, 285484k free, 64932k buffers
```

```
Swap: 2064380k total, 4k used, 2064376k free, 3423972k cached
```

.....

--可以操作系统查看这些进程，如果对系统影响很大，可以kill掉这些进程。

```
[root@dbserver ~]# ps -ef |grep ora
```

....

```
oracle 4041 4029 4 10:42 ? 00:00:04 oraclesztech1 (DESCRIPTION=
(LOCAL=YES)(ADDRESS=(PROTOCOL=beq)))
```

```
oracle 4042 4029 3 10:42 ? 00:00:03 oraclesztech1 (DESCRIPTION=
(LOCAL=YES)(ADDRESS=(PROTOCOL=beq)))
```

```
oracle 4043 4029 0 10:42 ? 00:00:00 oraclesztech1 (DESCRIPTION=
(LOCAL=YES)(ADDRESS=(PROTOCOL=beq)))
```

②为每个备份进程加上标志 在2个RMAN窗口各自执行备份:

```
RUN
{
SET COMMAND ID TO 'session1';
BACKUP DATABASE;
}
```

```
RUN
{
SET COMMAND ID TO 'session2';
BACKUP DATABASE;
}
```

```
SQL> SELECT SID, SPID, CLIENT_INFO
FROM V$PROCESS p, V$SESSION s
WHERE p.ADDR = s.PADDR
AND CLIENT_INFO LIKE '%id=sess%';
```

SID	SPID	CLIENT_INFO
152	4034	id=session1
149	4156	id=session1

3.监控RMAN进行执行情况

使用视图v\$session_longops

该视图存放总运行时间超过6秒的数据库相关进程，比如备份、全表扫描等。

```
SQL> SELECT OPNAME, CONTEXT, SOFAR,
TOTALWORK, ELAPSED_SECONDS/60, TIME_REMAINING/60,
ROUND (SOFAR/TOTALWORK*100,2) "%_COMPLETE"
FROM V$SESSION_LONGOPS
WHERE OPNAME LIKE 'RMAN%'
AND OPNAME NOT LIKE '%aggregate%'
AND TOTALWORK != 0
AND SOFAR <> TOTALWORK;
```

```
SELECT OPNAME, CONTEXT, SOFAR, TOTALWORK, ELAPSED_SECONDS/60, TIME_REMAINING/60,
ROUND (SOFAR/TOTALWORK*100,2) "%_COMPLETE"
FROM V$SESSION_LONGOPS
WHERE OPNAME LIKE 'RMAN%'
AND OPNAME NOT LIKE '%aggregate%'
AND TOTALWORK != 0
AND SOFAR < TOTALWORK;
```

	OPNAME	CONTEXT	SOFAR	TOTALWORK	ELAPSED_SECONDS/60	TIME_REMAINING/60	%_COMPLETE
1	RMAN: full datafile backup ...	1	285340	404384	0.2	0.0833333333333333	70.56

4.RMAN输出错误信息

- ①命令行输出的错误信息
- ②rman的跟踪文件
- ③报警文件:Alert.log
- ④服务器跟踪文件
- ⑤sbtio.log文件（使用磁带备份才产生的日志文件）

5.rman调试debug信息

- ①rman target / debug --开启调试，并将调试信息显示在屏幕上
- ②rman target / debug trace rmantrace.log --开启调试，并将调试信息记录在指定的文件中
- ③rman>run{
 debug on; --开启调试
 backup datafile 6;
 debug off; --关闭调试
 backup datafile 4;
 }

6.Rman报错信息

阅读错误信息从后往前看。

7.Rman调优和多路复用

- ①寻找Rman备份恢复的瓶颈，比如IO资源
- ②备份时不是越快越好，不能影响正常业务读写。
- ③多路复用

maxopenfiles：针对一个通道可同时处理的文件数

filesperset：针对每个备份集包含的文件数（可在该备份集中同时读写的文件数）

```
RMAN> run {
  allocate channel c1 device type disk maxopenfiles 3;
```

```
        backup datafile 4 filesperset 2;  
    }
```

为每个通道指定可打开的最大文件数为3，并且指定每个备份集包含2个数据文件。
那么最终以小的为准，步此该示例的多路复用级别就是2。
基于性能考虑，不要设置filesperset的值大于8。

磁盘备份和恢复时的缓冲区分配：

当多路复用级别小于等于4时，每个磁盘缓冲区为1M，所有缓冲区总大小为16M；
当多路复用级别大于4并且小于8时，每个磁盘缓冲区为512K，总大小最多不超过16M；

当多路复用级别大于8时，每个磁盘缓冲区为128K，并为每个通道固定分配总512K的大小；

磁带备份和恢复时的缓冲区分配：

磁带占用的缓冲区固定为4个256K，总大小为1M。

①当backup_tape_io_slaves=true时，那么缓冲区分配在SGA的large pool.

②当backup_tape_io_slaves=true时，那么缓冲区分配在PGA中。

④同步和异步IO

同步IO:每写一个缓冲区后就立即写入存储中

异步IO:将分配的多个缓冲区全部写完，持续写入存储中

⑤异步IO瓶颈监控

V\$BACKUP_ASYNC_IO:

LONG_WAITS/IO_COUNTS越大证明IO瓶颈越严重；

SHORT_WAIT_TIME_TOTAL和LONG_WAIT_TIME_TOTAL越接近0表示没有IO瓶颈。

⑥同步IO瓶颈监控

V\$BACKUP_SYNC_IO:

DISCRETE_BYTES_PER_SECOND表示每秒离散读的速率，
可对照设备的最大速率进行对照来分析IO性能。

BUFFER_SIZE表示缓冲区大小

BUFRER_COUNT表示每个文件使用占用的缓冲区个数

8.优化通道

①限定通道的单个备份片大小

```
run{  
    allocate channel c1 device type disk maxpiecesize 50m;  
    backup datafile 1 ;  
}
```

②限制通道的备份速率，使备份最小化影响数据库的正常使用

```
run{
    allocate channel c1 device type disk rat 5m; --每秒5M的速度
    backup datafile 1 ;
}
```

③设置多路复用级别

④设置多个磁盘同时备份，类似于多磁盘的条带化写入

⑤为磁带设备多个通道，允许每个通道写入不同的数据文件

9.调优备份命令

①maxpiecesize 设置最大的备份片大小

②backupspec 设置每个备份集最大的文件数，默认为64个

③maxopenfiles 设置最大可同时打开的文件数，默认为8个

④backup duration 设置备份持续的时间，在指定时间内备份完成。

```
run {
    backup duration 00:10 minimize load database ; --在10分钟之内以最小负载的方式完成备份
}
```

说明：

【minimize load】表示最小负载，备份速度最慢，对系统响应最小

【minimize time】表示最少时间，备份速度最快，对系统响应最大。

10.Rman 提升备份性能调优步骤

①去掉通道的rate设置

②在使用同步IO的情况下，设置dbwr_io_slaves

【dbwr_io_slaves】:模拟异步i/o，如果操作系统不支持异步i/o可以将这个参数设为非0值来模拟。dbwr 直接把数据块丢给slaves 就当是写完了。

且系统如果配置了large pool时就使用大池，否则使用shared pool.

如果你的系统支持异步I/O，请将该参数设置为初始值0。

③设置larget_pool_size

④调优磁带的瓶颈

⑤查询V\$相关视图解决备份瓶颈

11.设置large_pool_size

用于备份buffers的内存分配

①如果不设置large_pool_size的值，oracle将尝试从shared pool中进行分配用于buffers

②如果设置large_pool_size的值但不够用时，oracle不会再从从shared pool中分配内存用于buffers

③服务器没有足够的sga内存时，那么将从PGA中分配buffers，同时写入报警信息到报警文件中

12.模拟（测试）备份

backup validate datafile 4; --测试备份，不真实备份数据文件,无输出文件

backup validate database;

如果以**validate**方式测试备份的时间很快，但真正备份又很慢时，表示我们的瓶颈就出在备份设备的IO上。