

## 一、SQL tuning 类

1.列举几种表连接方式

2.不借助第三方工具，怎样查看 *sql* 的执行计划

3.如何使用 *CBO*, *CBO* 与 *RULE* 的区别

4.如何定位重要(消耗资源多)的 *SQL*

5.如何跟踪某个 *session* 的 *SQL*

6.*SQL* 调整最关注的是什么

7.说说你对索引的认识(索引的结构、对 *dml* 影响、对查询影响、为什么提高查询性能)

8.使用索引查询一定能提高查询的性能吗?为什么

9.绑定变量是什么?绑定变量有什么优缺点?

10.如何稳定(固定)执行计划

11.和排序相关的内存在 *8i* 和 *9i* 分别怎样调整, 临时表空间的作用是什么

12.存在表  $T(a,b,c,d)$ , 要根据字段  $c$  排序后取第 21—30 条记录显示, 请给出 *sql*

## 二、数据库基本概念类

1.*pctused* and *pctfree* 表示什么含义有什么作用

2.简单描述 *table / segment / extent / block* 之间的关系

3.描述 *tablespace* 和 *datafile* 之间的关系

4.本地管理表空间和字典管理表空间的特点, *ASSM* 有什么特点

5.回滚段的作用是什么

6.日志的作用是什么

7.SGA 主要有那些部分，主要作用是什么

8.racle 系统进程主要有哪些，作用是什么

### 三、备份恢复类

1.备份如何分类

2.归档是什么含义

3.如果一个表在 `2004-08-04 10:30:00` 被 `drop`，在有完善的归档和备份的情况下，如何恢复

4.rman 是什么，有何特点

5. *standby* 的特点

6. 对于一个要求恢复时间比较短的系统(数据库 *50G*, 每天归档 *5G*), 你如何设计备份策略

#### 四、系统管理类

1. 对于一个存在系统性能的系统, 说出你的诊断处理思路

2. 列举几种诊断 *IO*、*CPU*、性能状况的方法

3. 对 *statspack* 有何认识

4. 如果系统现在需要在一个很大的表上创建一个索引, 你会考虑那些因素, 如何做以尽量减小对应用的影响

5. 对 *raid10* 和 *raid5* 有何认识

#### 五、综合随意类

1.你最擅长的是 *oracle* 哪部分?

2.喜欢 *oracle* 吗?喜欢上论坛吗?或者偏好 *oracle* 的哪一部分?

3.随意说说你觉得 *oracle* 最有意思的部分或者最困难的部分

4.为何要选择做 *DBA* 呢?

-----答案分割线-----

## 一、*SQL tuning* 类

1.列举几种表连接方式

*hash join/merge join/nest loop(cluster join)/index join*

2.不借助第三方工具，怎样查看 *sql* 的执行计划

*set autot on*

*explain plan set statement\_id = &item\_id for &sql;*

`select * from table(dbms_xplan.display);` [http://download-west.oracle.com/ ... /b10752/ex\\_plan.htm](http://download-west.oracle.com/.../b10752/ex_plan.htm)

### 3.如何使用 CBO,CBO 与 RULE 的区别

在 `optimizer_mode=choose` 时,如果表有统计信息(分区表外),优化器将选择 CBO,否则选 RBO。RBO 遵循简单的分级方法学,使用 15 种级别要点,当接收到查询,优化器将评估使用到的要点数目,然后选择最佳级别(最少的数量)的执行路径来运行查询。

CBO 尝试找到最低成本的访问数据的方法,为了最大的吞吐量或最快的初始响应时间,计算使用不同的执行计划的成本,并选择成本最低的一个,关于表的数据内容的统计被用于确定执行计划。

### 4.如何定位重要(消耗资源多)的 SQL

```
select sql_text
from v$sql
where disk_reads > 1000 or (executions > 0 and buffer_gets/executions > 30000);
```

### 5.如何跟踪某个 session 的 SQL

```
exec dbms_system.set_sql_trace_in_session(sid,serial#,&sql_trace);
```

```
select sid,serial# from v$session where sid = (select sid from v$mystat where rownum = 1);
```

```
exec dbms_system.set_ev(&sid,&serial#,&event_10046,&level_12,'');
```

6.SQL 调整最关注的是什么

查看该 SQL 的

response time(db block gets/consistent gets/physical reads/sorts (disk))

7.说说你对索引的认识(索引的结构、对 dml 影响、为什么提高查询性能)

b-tree index/bitmap index/function index/partitional index(local/global) 索引通常能提高 select/update/delete 的性能,会降低 insert 的速度,

8.使用索引查询一定能提高查询的性能吗?为什么

索引就是为了提高查询性能而存在的,如果在查询中索引没有提高性能,只能说是用错了索引,或者讲是场合不同

9.绑定变量是什么?绑定变量有什么优缺点?

绑定变量是相对文本变量来讲的,所谓文本变量是指在 *SQL* 直接书写查询条件,这样的 *SQL* 在不同条件下需要反复解析,绑定变量是指使用变量来代替直接书写条件,查询 *bind value* 在运行时传递,然后绑定执行。优点是减少硬解析,降低 *CPU* 的争用,节省 *shared\_pool* 缺点是不能使用 *histogram,sql* 优化比较困难

#### 10.如何稳定(固定)执行计划

*query\_rewrite\_enabled = true*

*star\_transformation\_enabled = true*

*optimizer\_features\_enable = 9.2.0*

创建并使用

stored outline [http://download-west.oracle.com/ ... /outlines.htm#26854](http://download-west.oracle.com/.../outlines.htm#26854)

#### 11.和排序相关的内存在 *8i* 和 *9i* 分别怎样调整,临时表空间的作用是什么

*8i* 中 *sort\_area\_size/sort\_area\_retained\_size* 决定了排序所需要的内存

如果排序操作不能在 *sort\_area\_size* 中完成,就会用到 *temp* 表空间

*9i* 中如果 *workarea\_size\_policy=auto* 时,

排序在 *pga* 内进行,通常 *pga\_aggregate\_target* 的 *1/20* 可以用来进行

*disk sort*;



如果 *workarea\_size\_policy=manual* 时,排序需要的内存由 *sort\_area\_size* 决定

在执行

*order by/group by/distinct/union/create index/index rebuild/minus* 等操作时,

如果在 *pga* 或 *sort\_area\_size* 中不能完成,排序将在临时表空间进行 (*disk sort*),

临时表空间主要作用就是完成系统中的 *disk sort*.

12.存在表 *T(a,b,c,d)*,要根据字段 *c* 排序后取第 21—30 条记录显示,请给出 *sql*

```
create table t(a number(,b number(,c number(,d number(;
```

```
/
```

```
begin
```

```
for i in 1 .. 300 loop
```

```
insert into t values(mod(i,2),i/2,dbms_random.value(1,300),i/4);
```

```
end loop;
```

```
end;
```

```
/
```

```
select * from (select c.*,rownum as rn from (select * from t order by c desc) c) where rn between 21 and 30;
```

/

```
select * from (select * from test order by c desc) x where rownum
```

相比之 *minus* 性能较差

## 二、数据库基本概念类

1.*pctused* and *pctfree* 表示什么含义有什么作用

*pctused* 与 *pctfree* 控制数据块是否出现在 *freelist* 中,

*pctfree* 控制数据块中保留用于 *update* 的空间,当数据块中的 *free space* 小于 *pctfree* 设置的空间时,

该数据块从 *freelist* 中去掉,当块由于 *dml* 操作 *free space* 大于 *pct\_used* 设置的空间时,该数据库块将

被添加在 *freelist* 链表中。

2.简单描述 *table* / *segment* / *extent* / *block* 之间的关系

*table* 创建时,默认创建了一个 *data segment*,

每个 *data segment* 含有 *min extents* 指定的 *extents* 数,

每个 *extent* 根据表空间的存储参数分配一定数量的 *blocks*

### 3.描述 *tablespace* 和 *datafile* 之间的关系

一个 *tablespace* 可以有一个或多个 *datafile*,每个 *datafile* 只能在一个 *tablespace* 内,

*table* 中的数据,通过 *hash* 算法分布在 *tablespace* 中的各个 *datafile* 中,

*tablespace* 是逻辑上的概念,*datafile* 则在物理上储存了数据库的种种对象。

### 4.本地管理表空间和字典管理表空间的特点, *ASSM* 有什么特点

本地管理表空间(*Locally Managed Tablespace* 简称 *LMT*)

8i 以后出现的一种新的表空间的管理模式,通过位图来管理表空间的空间使用。

字典管理表空间(*Dictionary-Managed Tablespace* 简称 *DMT*)

8i 以前包括以后都还可以使用的一种表空间管理模式,通过数据字典管理表空间的空间使用。

动段空间管理(*ASSM*),

它首次出现在 *Oracle920* 里有了 *ASSM*, 链接列表 *freelist* 被位图所取代,

它是一个二进制的数组,

能够迅速有效地管理存储扩展和剩余区块(*free block*), 因此能够改善分段存储本质,

ASSM 表空间上创建的段还有另外一个称呼叫

*Bitmap Managed Segments*(BMB 段)。

## 5.回滚段的作用是什么

事务回滚：当事务修改表中数据的时候，该数据修改前的值(即前影像)会存放在回滚段中，

当用户回滚事务(*ROLLBACK*)时，*ORACLE* 将会利用回滚段中的数据前影像来将修改的数据恢复到原来的值。

事务恢复：当事务正在处理的时候，例程失败，回滚段的信息保存在 *undo* 表空间中，

*ORACLE* 将在下次打开数据库时利用回滚来恢复未提交的数据。

读一致性：当一个会话正在修改数据时，其他的会话将看不到该会话未提交的修改。

当一个语句正在执行时，该语句将看不到从该语句开始执行后的未提交的修改(语句级读一致性)

当 *ORACLE* 执行 *SELECT* 语句时，*ORACLE* 依照当前的系统改变号(*SYSTEM CHANGE NUMBER-SCN*)

来保证任何前于当前 *SCN* 的未提交的改变不被该语句处理。可以想象：当一个长时间的查询正在执行时，

若其他会话改变了该查询要查询的某个数据块，*ORACLE* 将利用回滚段的数据前影像来构造一个读一致性视图。

## 6.日志的作用是什么

记录数据库事务,最大限度地保证数据的一致性与安全性

重做日志文件: 含对数据库所做的更改记录, 这样万一出现故障可以启用数据恢复, 一个数据库至少需要两个重做日志文件

归档日志文件: 是重做日志文件的脱机副本, 这些副本可能对于从介质失败中进行恢复很必要。

## 7.SGA 主要有那些部分, 主要作用是什么

*SGA: db\_cache/shared\_pool/large\_pool/java\_pool*

*db\_cache:*

数据库缓存(*Block Buffer*)对于 *Oracle* 数据库的运转和性能起着非常关键的作用,

它占据 *Oracle* 数据库 *SGA*(系统共享内存区)的主要部分。*Oracle* 数据库通过使用 *LRU*

算法, 将最近访问的数据块存放到缓存中, 从而优化对磁盘数据的访问.

*shared\_pool:*

共享池的大小对于 *Oracle* 性能来说都是很重要的。

共享池中保存数据字典高速缓冲和完全解析或编译的的 *PL/SQL* 块和 *SQL* 语句及控制结构

*large\_pool*:

使用 *MTS* 配置时,因为要在 *SGA* 中分配 *UGA* 来保持用户的会话,就是用

*Large\_pool* 来保持这个会话内存

使用 *RMAN* 做备份的时候,要使用 *Large\_pool* 这个内存结构来做磁盘 *I/O* 缓存器

*java\_pool*:

为 *java procedure* 预备的内存区域,如果没有使用 *java proc*,*java\_pool* 不是必须的

8.Oracle 系统进程主要有哪些,作用是什么

数据写进程(*dbwr*): 负责将更改的数据从数据库缓冲区高速缓存写入数据文件

日志写进程(*lgwr*): 将重做日志缓冲区中的更改写入在线重做日志文件

系统监控(*smon*): 检查数据库的一致性如有必要还会在数据库打开时启动数据库的恢复

进程监控(*pmon*): 负责在一个 *Oracle* 进程失败时清理资源

检查点进程(*chpt*): 负责在每当缓冲区高速缓存中的更改永久地记录在数据库中时,更新控制文件和数据文件中的数据库状态信息。

归档进程(*arcn*): 在每次日志切换时把已满的日志组进行备份或归档

作业调度器(*cjq*):负责将调度与执行系统中已定义好的 *job*,完成一些预定义的工作.

恢复进程(*reco*) :保证分布式事务的一致性,在分布式事务中,要么同时 *commit*,  
要么同时 *rollback*;

### 三、备份恢复类

#### 1.备份如何分类

逻辑备份: *exp/imp*

物理备份:

*RMAN* 备份

*full backup/incremental backup*(累积/差异)

热备份:*alter tablespace begin/end backup*;

冷备份:脱机备份(*database shutdown*)

#### 2.归档是什么含义

关于归档日志: *Oracle* 要将填满的在线日志文件组归档时,则要建立归档日志  
(*archived redo log*)。

其对数据库备份和恢复有下列用处:

数据库后备以及在线和归档日志文件,在操作系统和磁盘故障中可保证全部提交的事物可被恢复。

在数据库打开和正常系统使用下，如果归档日志是永久保存，在线后备可以进行和使用。

数据库可运行在两种不同方式下：

*NOARCHIVELOG* 方式或 *ARCHIVELOG* 方式数据库在 *NOARCHIVELOG* 方式下使用时，不能进行在线日志的归档,如果数据库在 *ARCHIVELOG* 方式下运行，可实施在线日志的归档。**3:**如果一个表在 *2004-08-04 10:30:00* 被 *drop*，在有完善的归档和备份的情况下，如何恢复？

手工拷贝回所有备份的数据文件

*startup mount;*

*sql alter database recover automatic until time '2004-08-04:10:30:00';*

*alter database open resetlogs;*

**4.rman** 是什么,有何特点？

*RMAN(Recovery Manager)*是 *DBA* 的一个重要工具，用于备份、还原和恢复 *oracle* 数据库, *RMAN* 可以用来备份和恢复数据库文件、归档日志、控制文件、系统参数文件,也可以用来执行完全或不完全的数据库恢复。*RMAN* 有三种不同的用户接口：

*COMMAND LINE* 方式、*GUI* 方式(集成在 *OEM* 中的备份管理器)、*API* 方式(用于集成到第三方的备份软件中)。



具有如下特点：

- 1)功能类似物理备份，但比物理备份强大  $N$  倍；
- 2)可以压缩空块；
- 3)可以在块水平上实现增量；
- 4)可以把备份的输出打包成备份集，也可以按固定大小分割备份集；
- 5)备份与恢复的过程可以自动管理；
- 6)可以使用脚本(存在 *Recovery catalog* 中)
- 7)可以做坏块监测

## 5. standby 的特点

备用数据库(*standby database*): *ORACLE* 推出的一种高可用性

(*HIGH AVAILABLE*)数据库方案，

在主节点与备用节点间通过日志同步来保证数据的同步，备用节点作为主节点的备份

可以实现快速切换与灾难性恢复,从 *920* 开始,还开始支持物理与逻辑备用服务器。

*9i* 中的三种数据保护模式分别是：

1)、*MAXIMIZE PROTECTION*：最大数据保护与无数据分歧，*LGWR* 将同时传送到备用节点，

在主节点事务确认之前，备用节点也必须完全收到日志数据。如果网络不好，引起 *LGWR* 不能传送数据，将引起严重的性能问题，导致主节点 *DOWN* 机。

2)、*MAXIMIZE AVAILABILITY*：无数据丢失模式，允许数据分歧，允许异步传送。

正常情况下运行在最大保护模式，在主节点与备用节点的网络断开或连接不正常时，自动切换到最大性能模式，

主节点的操作还是可以继续的。在网络不好的情况下有较大的性能影响。

3)、*MAXIMIZE PERFORMANCE*：这种模式应当可以说是从 *8i* 继承过来的备用服务器模式，异步传送，

无数据同步检查，可能丢失数据，但是能获得主节点的最大性能。*9i* 在配置

*DATA GUARD* 的时候默认就是 *MAXIMIZE PERFORMANCE*

6.对于一个要求恢复时间比较短的系统(数据库 *50G*,每天归档 *5G*)，你如何设计备份策略

*rman*/每月一号 *level 0* 每周末/周三 *level 1* 其它每天 *level 2*

#### 四、系统管理类

1.对于一个存在系统性能的系统，说出你的诊断处理思路。

(1) 做 *statspack* 收集系统相关信息

了解系统大致情况/确定是否存在参数设置不合适的地方/查看 *top 5 event* /  
查看 *top sql* 等

(2) 查 *v\$system\_event/v\$session\_event/v\$session\_wait*

从 *v\$system\_event* 开始,确定需要什么资源(*db file sequential read*)等

深入研究 *v\$session\_event*,确定等待事件涉及的会话

从 *v\$session\_wait* 确定详细的资源争用情况(*p1-p3* 的  
值:*file\_id/block\_id/blocks* 等)

(3) 通过 *v\$sql/v\$sqltext/v\$sqlarea* 表确定 *disk\_reads*、  
(*buffer\_gets/executions*)值较大的 SQL

2.列举几种诊断 IO、CPU、性能状况的方法

*top/vmstat*

*statspack*

*sql\_trace/tkprof*

查 *v\$system\_event/v\$session\_event/v\$session\_wait*

查 *v\$sqlarea*(*disk\_reads* 或 *buffer\_gets/executions* 较大的 SQL)

3.对 *statspack* 有何认识

*Statspack* 是 Oracle 公司提供的-一个收集数据库运行性能指标的软件包, 该软件包从 8i 起, 在 9i、10g 都有显著的增强。该软件包的辅助表(存储相关参

数与收集的性能指标的表)由最初的 25 个增长到 43 个。收集级别参数由原来的 3 个(0、5、10)增加到 5 个(0、5、6、7、10)通过分析收集的性能指标,数据库管理员可以详细地了解数据库目前的运行情况,对数据库实例、等待事件、SQL 等进行优化调整。利用 *statspack* 收集的 *snapshot*,可以统计制作数据库的各种性能指标的统计趋势图表。

4.如果系统现在需要在一个很大的表上创建一个索引,你会考虑那些因素,如何做以尽量减小对应用的影响

在系统比较空闲时;*nologging* 选项(如果有 *dataguard* 则不可以使用 *nologging*)大的 *sort\_ared\_size* 或 *pga\_aggregate\_target* 较大

5.对 *raid1+0* 和 *raid5* 有何认识

*RAID 10*(或称 *RAID 1+0*)与 *RAID 0+1* 不同,它是用硬盘驱动器先组成 *RAID 1* 阵列,然后在 *RAID 1* 阵列之间再组成 *RAID 0* 阵列。*RAID 10* 模式同 *RAID 0+1* 模式一样具有良好的数据传输性能,但却比 *RAID 0+1* 具有更高的可靠性。*RAID 10* 阵列的实际容量为  $M \times n / 2$ ,磁盘利用率为 50%。

*RAID 10* 也需要至少 4 个硬盘驱动器构成,因而价格昂贵。*RAID 10* 的可靠性同 *RAID 1* 一样,但由于 *RAID 10* 硬盘驱动器之间有数据分割,因而数据传输性能优良。*RAID 5* 与 *RAID 3* 很相似,不同之处在于 *RAID 5* 的奇偶校验信息也同数据一样被分割保存到所有的硬盘驱动器,而不是写入一个指定的硬盘驱动器,从而消除了单个奇偶校验硬盘驱动器的瓶颈问题。*RAID 5* 磁盘阵

列的性能比 *RAID 3* 有所提高，但仍然需要至少 3 块硬盘驱动器。其实际容量为  $M \times (n-1)$ ，磁盘利用率为  $(n-1)/n$ 。