

基于 POBA 快速开发平台的 ORACLE 性能优化建议

(适用于平台 V3.3RC2/ORACLE9i 以上版本)

1. 对于数据库表的常用查询字段增加索引

请在平台网站 (<http://192.168.1.220/publish>) 的“文档及资料下载”下面下载与平台版本对应的库表索引文件，并正确执行。

2. 修改可以打开的最大游标数

- (1) 查看该参数的值: `show parameter open_cursors;`
- (2) 查看当前已经打开的游标数, 例如为 500: `select count(*) from v$open_cursor;`
- (3) 修改该参数的值, 建议设置为高并情况下上面所取值的 1.5-2 倍, 如上则为 $500 \times 1.5 = 750$ 或 $500 \times 2 = 1000$: `alter system set open_cursors=1000 scope=both;`

3. 为该数据库帐号创建单独的表空间

- (1) 查看系统中的表空间列表: `select * from SYS.dba_tablespaces;`
- (2) 创建表空间: `create tablespace 表空间名称 datafile '数据目录/xxx.DBF' size 1024m autoextend on next 512m maxsize unlimited;`
- (3) 创建使用该表空间的数据库帐号: `create user Oracle 帐号 identified by 帐号口令 default tablespace 表空间名称;`
- (4) 为该帐号授权: `grant resource,connect to Oracle 帐号;`
- (5) 级联删除表空间及数据: `drop tablespace 表空间名称 including contents and datafiles;`

如果已经有了独立的表空间, 但是表空间的初始化 size 太小, 需要加到文件尺寸, 可以修改表空间:

- 增加数据文件尺寸: `alter database datafile '数据目录/xxx.DBF' resize 4000m`
- 设定数据文件自动扩展: `alter database datafile '数据目录/xxx.DBF' autoextended on next 1000m maxsize 20000m`

4. 设置合适的表存储参数

`select * from v$waitstat;`

	CLASS	COUNT	TIME
▶ 1	data block	715	548
2	sort block	0	0
3	save undo block	0	0
4	segment header	1	0
5	save undo header	0	0
6	free list	0	0
7	extent map	0	0
8	1st level bmb	0	0
9	2nd level bmb	0	0
10	3rd level bmb	0	0
11	bitmap block	0	0
12	bitmap index block	0	0
13	file header block	4	3
14	unused	0	0
15	system undo header	0	0
16	system undo block	0	0
17	undo header	11	0
18	undo block	0	0

如果发现 data block 或者 free list 类的 count 次数很大，则说明等待情况严重，需要增加 freelists

5. 设置 log_buffer 参数

根据物理内存大小，日志缓冲区大小应设置为几十兆到上百兆之间。

```
show parameter log_buffer;  
alter system set log_buffer=33987231;
```

通过下面的语句，可以查看当前的日志缓冲设置是否够用：

```
select * from v$session_wait where event='log buffer wait';
```

如果显示多条记录，则说明日志缓冲设置小了，需要增加，如果没有记录，则正常。

6. 设置 optimizer_index_caching 参数

这个参数可以设置索引的缓冲度，范围是 0 到 100，默认是 0，可以考虑设置成 90

```
show parameter optimizer_index_caching;  
alter system set optimizer_index_caching=90;
```

7. 设置 optimizer_index_cost_adj 参数

默认=100 表名索引扫描与全表扫描代价一样。将这个参数设小表名索引代价要小于全表扫描，这样就使得使用 CBO 进行成本计算时更倾向于使用索引扫描。建议把这个参数设置成 30 到 50。

```
show parameter optimizer_index_cost_adj;  
alter system set optimizer_index_cost_adj=50;
```

8. 设置合适的 SGA 相关参数

SGA 是用于存储数据库信息的内存区，该信息为数据库进程所共享。它包含 Oracle 服务器的数据和控制信息,它是在 Oracle 服务器所驻留的计算机的实际内存中得以分配，如果实际内存不够再往虚拟内存中写。

(1) 查看以下几个参数：

```
sga_max_size, 即 SGA 允许的最大值  
shared_pool_size, 共享池内存最大值  
db_cache_size, 数据缓冲区内存最大值  
show parameter sga_max_size;  
show parameter shared_pool_size;  
show parameter db_cache_size;
```

(2) 合理设置 SGA 的是三个参数

一般 sga_max_size 为物理内存的 1/3-1/2, shared_pool_size 设置为物理内存的 1/10, 一般不超过 1G, db_cache_size 一般为上述两者的差值，实际中肯定要比该差值还要小才行，如果设置的大了，会提示下面信息：



根据实际情况调整到合适的值即可。

```
alter system set sga_max_size=6G;  
alter system set shared_pool_size=300M;  
alter system set db_cache_size=3G;
```

9. 设置排序参数

(1) **sort_area_size**: 默认的用来排序的 `sort_area_size` 大小是 32K, 通常显得有点小, 一般可以考虑设置成 1M (1048576)。这个参数不能设置过大, 因为每个连接都要分配同样的排序内存。

```
show parameter sort_area_size;  
alter system set sort_area_size =1048576scope=spfile;
```

(2) **sort_multiblock_read_count**: 增大这个参数可以提高临时表空间排序性能, 该参数默认是 2, 可以改成 32 来对比一下排序查询时间变化。注意, 这个参数的最大值与平台有关系。

```
show parametersort_multiblock_read_count;  
alter system set sort_multiblock_read_count =32;
```

10. 设置最大连接数

```
show parameter processes;  
alter system set processes=1000 scope=spfile;  
select count(*) from v$process;
```