1. 数据库
   1. SQL优化

充分利用索引类

* + 1. 少用in

ORACLE试图将其转换成多个表的连接，如果转换不成功则先执行IN里面的子查询，再查询外层的表记录，如果转换成功则直接采用多个表的连接方式查询。推荐方案：在业务密集的SQL当中尽量不采用IN操作符，用EXISTS 方案代替。

* + 1. 不用IS NULL 或IS NOT NULL操作（判断字段是否为空）

只要这些列中有一列含有null，该列就会从索引中排除。也就是说如果某列存在空值，即使对该列建索引也不会提高性能。数据库中空值用0或者‘ ’填入，从而：a is not null 改为 a>0 或a>’’等

* + 1. 用>=代替>

执行A>2与A>=3的效果有区别，因为A>2时ORACLE会先找出为2的记录索引再进行比较，而A>=3时ORACLE则直接找到=3的记录索引。

* + 1. Like ‘abc%’ abc前面不要带匹配符%，因为第一个字符为匹配符，从而无法使用索引。
    2. UNION all 代替union

UNION在进行表链接后会筛选掉重复的记录，所以在表链接后会对所产生的结果集进行排序运算，删除重复的记录再返回结果。UNION ALL操作只是简单的将两个结果合并后就返回。

* + 1. 不用not，<> 会全表扫描，放弃索引

其他

* + 1. Where条件限制严格的往后放，会先被解析。可以过滤掉最大数量记录
    2. SELECT子句中避免使用 ‘ \* ‘

ORACLE在解析的过程中, 会将'\*' 依次转换成所有的列名, 这个工作是通过查询数据字典完成的, 这意味着将耗费更多的时间。

分析sql语句耗费情况，v$sqlarea视图

SELECT EXECUTIONS , DISK\_READS, BUFFER\_GETS,

ROUND((BUFFER\_GETS-DISK\_READS)/BUFFER\_GETS,2) Hit\_radio,

ROUND(DISK\_READS/EXECUTIONS,2) Reads\_per\_run,ELAPSED\_TIME,

SQL\_TEXT

FROM V$SQLAREA

WHERE EXECUTIONS>0

AND BUFFER\_GETS > 0

and Parsing\_Schema\_Name='COMM\_USER' and module = 'JDBC Thin Client'

ORDER BY 6 DESC;

Ref:数据库性能优化之SQL语句优化 - 全栈开发者.mht

* 1. Join

select count(\*) from a ,b where a.id = b.sid;

红色部分等价于select count(\*) from a cross join b，产生一个笛卡尔积，a的数据条数乘以b的数据条数。形成一个临时表，再通过where条件过滤，效率比inner join低很多。Join 的on条件是在生成临时表时就起作用，所以效率高。在大数据量时能体现区别。

* 1. 分区表

创建表空间

create tablespace partition1 datafile 'C:\app\Administrator\oradata\orcl\p1.dbf' size 20m

create tablespace partition2 datafile 'C:\app\Administrator\oradata\orcl\p2.dbf' size 20m

创建表

CREATE TABLE PROBLEM\_TICKETS

(

DATE\_ENTERED DATE NOT NULL PRIMARY KEY,

STATUS VARCHAR2(20)

)

PARTITION BY LIST (STATUS)

(

PARTITION p\_plane VALUES ('plane') TABLESPACE partition1,

PARTITION p\_ship VALUES ('ship') TABLESPACE partition2

)

创建本地索引

CREATE INDEX index\_problem\_ticket ON PROBLEM\_TICKETS(DATE\_ENTERED) LOCAL;

本地索引就是每个分区一个索引，索引存放在相应表空间中。全局索引就是一个统一索引，分区改变后索引会失效。

* 1. Oracle日志分类

Alert log file(警报日志)

Trace file(跟踪日志,用户和进程信息)

undo日志（事物执行过程中的日志,用于回滚）

redo日志（重做日志恢复数据库使用,用于数据库崩溃,已经commit数据在内存缓存中,还没有同步到磁盘上,所以redo日志缓存比数据修改缓存先写入到磁盘上 Ref:理解数据库中的undo日志、redo日志、检查点）

* + 1. redo日志分为

online redo log 在线重做日志（联机重做日志）,sql语句存入本地磁盘,用于数据内存的数据未同步到硬盘上,但是机器崩溃了,重启机器时,根据本地磁盘的redo log恢复到硬盘中。

Archive redo log 归档重做日志,简称归档日志,若数据库处于归档模式下（咱们用的数据库大部分是非归档模式）,在线重做日志满了自动写入到归档日志,以本地磁盘的形式,归档模式用于整个数据库恢复

Ref:Oracle-归档日志详解(运行模式、分类)

* 1. 获取tnsnames.ora位置

如果设置环境变量TNS\_ADMIN环境变量,则所有Oracle版本均以这个为准,适用于安装多个oracle客户端的机器,通过统一一个tnsnames.ora文件连接服务器。

注册表位置：搜索ORACLE\_HOME

* 1. 表空间
     1. 表空间已用容量

select tablespace\_name, count(\*) as extends, sum(bytes) / 1024 / 1024 as MB,sum(blocks) as blocks from dba\_data\_files group by tablespace\_name;

* + 1. 表空间剩余容量

select tablespace\_name ,count(\*) as extends,round(sum(bytes)/1024/2014,2) as MB,

sum(blocks) as blocks from dba\_free\_space group by tablespace\_name

* + 1. 表空间各种属性

select \* from dba\_data\_files

* + 1. 给表空间增加数据文件

Alter tablespace dblzhf\_ts add datafile 'D:\ORACLE\PRODUCT\10.2.0\ORADATA\ORCL\DBLZHF\_1.DBF' size 100M autoextend on next 100M maxsize unlimited

* + 1. 创建表空间

create tablespace dblzhf\_ts logging datafile 'D:\oracle\product\10.2.0\oradata\orcl\dblzhf\_ts.dbf'

size 50m autoextend on next 50m extent management local;

* 1. 检索数据技巧
     1. 查询自己想要的数据库表

select \* from user\_tab\_comments where comments like '%情报%'

可以根据表的注释找，同时也可以找表名

* + 1. 检索海量数据中某一条

select \* from (select rownum rn, jssj from (select msgid, rownum, jssj from lzhf order by jssj desc) t) a where a.rn >= 100000 and a.rn < 110000

* + 1. 按照时间检索

select \* from lzhf where jssj between to\_date('2015-10-21 17:04:32', 'yyyy-mm-dd hh24:mi:ss')

and to\_date('2015-10-21 17:05:34', 'yyyy-mm-dd hh24:mi:ss')

* 1. 索引
     1. 创建索引

create index jssj\_index on lzhf(jssj)

* 1. 约束
     1. 获取所有的约束和主键约束

select owner ,constraint\_name,table\_name,column\_name from user\_cons\_columns where table\_name = 'LZHF'

* + 1. 删除主键约束

alter table lzhf drop constraint PK\_LZHF

* + 1. 删除表前先删约束

cascade constraints意识是如果其他表的约束中规定以来该表，则先删除该约束，再继续删表，保证表能被删除成功。

* 1. 启用、禁用约束

alter table 表名称 enable constraint 限制名称

启动后,出错,是因为检查约束是否可行,有的表不符合约束

alter table 表名称 disable constraint 限制名称

* 1. 给Oracle增加内存使用量以提高效率

alter system set sga\_target = 1100M scope=both;

alter system set db\_cache\_size = 500M scope=both;

* 1. Oracle sequence的作用

用来在多用户环境下产生唯一整数的数据库对象,该整数可以用来作为整数类型的主键值,在多用户并发情况下有较好的性能;

如果用程序代码生成该主键值,则需要保证互斥,效率比较低。一般在需要数据库优化的地方会使用sequence。

* 1. Win7下面oracle的odbc的位置

C:\Windows\SysWOW64\odbcad32.exe

* 1. 权限和角色

有各种各样的权限如连接数据库,建表等,将常用权限组合为角色。

* 1. 数据库自动处理脚本bat脚本解释

sqlplus /nolog @ScriptAdd.exec

登录到sqlplus,执行ScriptAdd.exec中内容

* + 1. SPOOL cat\_all.log

Spool是Oracle中输出日志使用

* + 1. @setvars.sql

设定运行的一些环境变量,这里灵活性比较强

* + 1. SET VERIFY OFF

是否显示替代变量被替代后的语句

* + 1. DEFINE root\_path=C:\oracle\product\10.2.0\oradata\orcl

定义数据文件存放目录

* + 1. DEFINE pwd\_of\_sys=orcl

-- 定义 SYS 用户启用 SYSDBA 角色时的口令

* + 1. conn dbsjgx/dbsjgx@orcl

连接某个用户

* + 1. @addScript.sql

执行sql脚本

* + 1. Disconnect 断开连接
    2. Exit直接退出
    3. Pause暂停,不退出命令行
  1. 数据库导入导出exp imp
     1. 导出

exp userid=system/manager@orcl file=c:\DBQKZH.dmp log=c:\DBQKZH.log owner=DBQKZH

如果出现exp-00003 未找到段的存储定义,则最好到数据库本机上导出

* + 1. 导入

imp userid=system/manager@orcl file=C:\DBQKZH.dmp grants=n ignore=y full=y log=C:\DBQKZH\_in.log

* + 1. PLsql导出sql文件

在export tables界面选择完表名后,右键 save object selection,记录下这次选择的表名,下次直接load这个文件就可以了。

* 1. 过程包解释

DELIMITER $$ 让$$为分隔符

DROP PROCEDURE IF EXISTS `PKG\_CSXT$SEZHDXWZLOG`$$ 如果PKG\_CSXT$SEZHDXWZLOG存在，则删除该过程包

DECLARE CONTINUE HANDLER FOR SQLSTATE '23000' SET @DUPLEX\_KEY=1; 如果主键冲突，设置DUPLEX\_KEY变量为1。

Sql语句调用过程包 CALL PKG\_CSXT$SEZHDXWZLOG ('tt','11','22','0000-00-00 00:00:00')

* 1. Blob字段

获取所有包含blob字段的表

select \* from user\_lobs

* 1. 其他杂项
     1. 无system用户密码创建用户和表空间

使用sys用户连接数据库,可在不知道system的密码情况下创建表空间和用户

先Sqlplus /nolog

Conn sys/oracle as sysdba

即可以管理员登录。

Sqlplus /nolog只是登录sqlplus,并没有连接数据库

* + 1. 大小写,字段名,关键词做字段名

Oracle的双引号之间是区分大小写的,列名自动为大写。