# Oracle统计、分析和优化环境配置

## 创建批处理文件Login.bat

用于初始化设置系统环境

Login.bat

@echo off

title eoda

mode con cols=140

color 85

set ORACLE\_SID=muphy

sqlplus eoda/foo

## 创建数据库脚本文件login.sql

用于初始化SQL\*PLUS运行环境

define \_editor=vim --设置编辑器为vim

set serveroutput on size 1000000 --设置打开DBMS\_OUTPUT并设置默认缓冲区

set trimspool on --设置去除命令两端的空格

set long 5000 --设置long和clob列时默认显示字节数

set linesize 1000

set pagesize 9999

column plan\_plus\_exp format a80

column table\_name format a30

column index\_name format a30

column name format a30

column value format a30

column table format a30

set sqlprompt '&\_user.@&\_connect\_identifier.> ' --设置提示符指示谁登陆到那个数据库

## 控制报告

SET AUTOTRANCE OFF

SET AUTOTRANCE ON EXPLAIN --只显示执行计划

SET AUTOTRANCE ON STATISTICS --只显示统计结果

SET AUTOTRANCE ONLY --不显示查询结果

SET AUTOTRACE TRANCEONLY EXPLAN --只显示执行计划

## 配置Statspack

$ sqlplus / as sysdba

sys/eoda > @spcreate

## 创建统计stats视图

需要有查看视图的权限

conn / as sysdba

grant select on v\_$statname to eoda;

grant select on v\_$mystat to eoda;

grant select on v\_$latch to eoda;

grant select on v\_$timer to eoda;

conn eoda/foo

drop table run\_stats;

set echo on;

create or replace view stats

as select 'STAT...' || a.name name, b.value

from v$statname a, v$mystat b

where a.statistic# = b.statistic#

union all

select 'LATCH.' || name, gets

from v$latch

union all

select 'STAT...Elapsed Time', hsecs from v$timer;

## 创建存储统计结果的run\_stats表

create global temporary table run\_stats

( runid varchar2(15),

name varchar2(80),

value int )

on commit preserve rows;

## 创建包runstats\_pkg

用于比较两个sql之间的性能，会测量3个要素：耗用时间、系统统计信息和闩定

create or replace package runstats\_pkg

as

procedure rs\_start;

procedure rs\_middle;

procedure rs\_stop(p\_difference\_threshold in number default 0 );

end;

/

create or replace package body runstats\_pkg

as

g\_start number;

g\_run1 number;

g\_run2 number;

procedure rs\_start

is

begin

delete from run\_stats;

insert into run\_stats

select 'before', stats.\* from stats;

g\_start := dbms\_utility.get\_cpu\_time;

end;

procedure rs\_middle

is

begin

g\_run1 := (dbms\_utility.get\_cpu\_time-g\_start);

insert into run\_stats

select 'after 1', stats.\* from stats;

g\_start := dbms\_utility.get\_cpu\_time;

end;

procedure rs\_stop(p\_difference\_threshold in number default 0)

is

begin

g\_run2 := (dbms\_utility.get\_cpu\_time-g\_start);

dbms\_output.put\_line( 'Run1 ran in ' || g\_run1 || ' cpu hsecs' );

dbms\_output.put\_line( 'Run2 ran in ' || g\_run2 || ' cpu hsecs' );

if ( g\_run2 <> 0 )

then

dbms\_output.put\_line

( 'run 1 ran in ' || round(g\_run1/g\_run2\*100,2) ||

'% of the time' );

end if;

dbms\_output.put\_line( chr(9) );

insert into run\_stats

select 'after 2', stats.\* from stats;

dbms\_output.put\_line

( rpad( 'Name', 30 ) || lpad( 'Run1', 16 ) ||

lpad( 'Run2', 16 ) || lpad( 'Diff', 16 ) );

for x in

( select rpad( a.name, 30 ) ||

to\_char( b.value-a.value, '999,999,999,999' ) ||

to\_char( c.value-b.value, '999,999,999,999' ) ||

to\_char( ( (c.value-b.value)-(b.value-a.value)),

'999,999,999,999' ) data

from run\_stats a, run\_stats b, run\_stats c

where a.name = b.name

and b.name = c.name

and a.runid = 'before'

and b.runid = 'after 1'

and c.runid = 'after 2'

and abs( (c.value-b.value) - (b.value-a.value) )

> p\_difference\_threshold

order by abs( (c.value-b.value)-(b.value-a.value))

) loop

dbms\_output.put\_line( x.data );

end loop;

dbms\_output.put\_line( chr(9) );

dbms\_output.put\_line

( 'Run1 latches total versus runs -- difference and pct' );

dbms\_output.put\_line

( lpad( 'Run1', 14 ) || lpad( 'Run2', 19 ) ||

lpad( 'Diff', 18 ) || lpad( 'Pct', 11 ) );

for x in

( select to\_char( run1, '9,999,999,999,999' ) ||

to\_char( run2, '9,999,999,999,999' ) ||

to\_char( diff, '9,999,999,999,999' ) ||

to\_char( round( run1/decode( run2, 0, to\_number(0), run2) \*100,2 ), '99,999.99' ) || '%' data

from ( select sum(b.value-a.value) run1, sum(c.value-b.value) run2,

sum( (c.value-b.value)-(b.value-a.value)) diff

from run\_stats a, run\_stats b, run\_stats c

where a.name = b.name

and b.name = c.name

and a.runid = 'before'

and b.runid = 'after 1'

and c.runid = 'after 2'

and a.name like 'LATCH%'

)

) loop

dbms\_output.put\_line( x.data );

end loop;

end;

end;

/

## 创建mystat.sql和mystat2.sql查看统计结果

前者统计初始情况，或者统计执行sql之后的情况和报告差异

### 创建mystat.sql脚本文件

set echo off

set verify off

column value new\_val V

define S="&1"

set autotrace off

select a.name, b.value

from v$statname a, v$mystat b

where a.statistic# = b.statistic#

and lower(a.name) = lower('&S')

/

set echo on

### 创建mystat2.sql脚本文件

set echo off

set verify off

select a.name, b.value V, to\_char(b.value-&V,'999,999,999,999') diff

from v$statname a, v$mystat b

where a.statistic# = b.statistic#

and lower(a.name) = lower('&S')

/

set echo on

## 创建SHOW\_SPACE存储过程

用于打印数据库段的空间利用率信息

create or replace procedure show\_space

( p\_segname in varchar2, -- 段名（表、索引）

p\_owner in varchar2 default user, -- 默认为当前用户

p\_type in varchar2 default 'TABLE', -- 默认为表类型

p\_partition in varchar2 default NULL ) -- 分区对象的分区名

authid current\_user

as

l\_free\_blks number;

l\_total\_blocks number;

l\_total\_bytes number;

l\_unused\_blocks number;

l\_unused\_bytes number;

l\_LastUsedExtFileId number;

l\_LastUsedExtBlockId number;

l\_LAST\_USED\_BLOCK number;

l\_segment\_space\_mgmt varchar2(255);

l\_unformatted\_blocks number;

l\_unformatted\_bytes number;

l\_fs1\_blocks number; l\_fs1\_bytes number;

l\_fs2\_blocks number; l\_fs2\_bytes number;

l\_fs3\_blocks number; l\_fs3\_bytes number;

l\_fs4\_blocks number; l\_fs4\_bytes number;

l\_full\_blocks number; l\_full\_bytes number;

-- 内部存储过程，用于格式化打印

procedure p( p\_label in varchar2, p\_num in number )

is

begin

dbms\_output.put\_line( rpad(p\_label,40,'.') ||

to\_char(p\_num,'999,999,999,999') );

end;

begin

-- 动态sql判断对象是否ASSM

begin

execute immediate

'select ts.segment\_space\_management

from dba\_segments seg, dba\_tablespaces ts

where seg.segment\_name = :p\_segname

and (:p\_partition is null or

seg.partition\_name = :p\_partition)

and seg.owner = :p\_owner

and seg.tablespace\_name = ts.tablespace\_name'

into l\_segment\_space\_mgmt

using p\_segname, p\_partition, p\_partition, p\_owner;

exception

when too\_many\_rows then

dbms\_output.put\_line

( 'This must be a partitioned table, use p\_partition => ');

return;

end;

-- 如果对象是ASSM 表空间, 调用dbms\_space.space\_usage

-- 否则调用dbms\_space.free\_blocks

if l\_segment\_space\_mgmt = 'AUTO'

then

dbms\_space.space\_usage

( p\_owner, p\_segname, p\_type, l\_unformatted\_blocks,

l\_unformatted\_bytes, l\_fs1\_blocks, l\_fs1\_bytes,

l\_fs2\_blocks, l\_fs2\_bytes, l\_fs3\_blocks, l\_fs3\_bytes,

l\_fs4\_blocks, l\_fs4\_bytes, l\_full\_blocks, l\_full\_bytes, p\_partition);

p( 'Unformatted Blocks ', l\_unformatted\_blocks );

p( 'FS1 Blocks (0-25) ', l\_fs1\_blocks );

p( 'FS2 Blocks (25-50) ', l\_fs2\_blocks );

p( 'FS3 Blocks (50-75) ', l\_fs3\_blocks );

p( 'FS4 Blocks (75-100)', l\_fs4\_blocks );

p( 'Full Blocks ', l\_full\_blocks );

else

dbms\_space.free\_blocks(

segment\_owner => p\_owner,

segment\_name => p\_segname,

segment\_type => p\_type,

freelist\_group\_id => 0,

free\_blks => l\_free\_blks);

p( 'Free Blocks', l\_free\_blks );

end if;

-- 调用dbms\_space.unused\_space获取未使用的空间信息

dbms\_space.unused\_space

( segment\_owner => p\_owner,

segment\_name => p\_segname,

segment\_type => p\_type,

partition\_name => p\_partition,

total\_blocks => l\_total\_blocks,

total\_bytes => l\_total\_bytes,

unused\_blocks => l\_unused\_blocks,

unused\_bytes => l\_unused\_bytes,

LAST\_USED\_EXTENT\_FILE\_ID => l\_LastUsedExtFileId,

LAST\_USED\_EXTENT\_BLOCK\_ID => l\_LastUsedExtBlockId,

LAST\_USED\_BLOCK => l\_LAST\_USED\_BLOCK );

p( 'Total Blocks', l\_total\_blocks );

p( 'Total Bytes', l\_total\_bytes );

p( 'Total MBytes', trunc(l\_total\_bytes/1024/1024) );

p( 'Unused Blocks', l\_unused\_blocks );

p( 'Unused Bytes', l\_unused\_bytes );

p( 'Last Used Ext FileId', l\_LastUsedExtFileId );

p( 'Last Used Ext BlockId', l\_LastUsedExtBlockId );

p( 'Last Used Block', l\_LAST\_USED\_BLOCK );

end;

/