TABLESPACE

MANAGEMENT

# System and undo are critical tablespaces we cannot offline these tablespaces.

• maximum 1022 number of datafiles’s can be added under a tablespace.

Physical :

=======

• Tablespace contains one or more datafiles it is a physical file.

Logicaldefination of tablespace:

========================

• logical definition is collection of contents segments

• Segments is a collection of extents.

• Extents is collection of contiguous oracle blocks

———————————————————————

# Main view of tablespace:

====================

• desc dba\_tablespaces

• select contents from dba\_tablespaces;

• Select distinct contents from dba\_tablespaces;

# To check tablespace permanent /Temp/:

===============================

• select tablespace\_name,contents from dba\_tablespaces;

# Tablespaces and uses:

===================

1. system :

========

• metadata tables of the database is stored in system tablespace.

2. sysaux:

=======

• it stores the snap id of the database it was interduces from 10g.

• Performance related like awr report will be stored.

• Auditing.

3. users:

=======

• when ever user is creating objects it stores that objects.

4. undo:

======

• when the operations like dml are happening and the transaction has be rolled back the previous information will be stored under undo tablespace.

5. Temp:

=======

• when the user retrieving data from multiple tables that data need some storage point.Temp will hold the data for temporarily.

———————————————————————

# To check users default\_tablespace:

============================

• select username,default\_tablespace from dba\_users where username=‘Palikila’;

———————————————————————

# To change users default tablespace:

=============================

• alter user username default tablespace mouli;

———————————————————————

TO check how many users linked with tablespace:

=============================

• select username from dba\_users where default\_tablespace=‘palikila’;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# TO ADD TABLESPACE TO USER:

=========================

• alter user u1 default tablespace palikila;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# SMALL FILE & BIG FILE TABLESPACE

==============================

1. Smallfile:

========

• Small file tablespace :It’s size in GB

• It contain one or more datafiles .and by default all tablespace are created with small file .

• We can add datafile when this tablespace filled.

(8k block size 32gb)

(4K block size 16gb)

(2k block size 8gb)

2. Big file:

=======

• Big file tablespace :it’s size in TB

• It contain only one datafile.

• We cannot add datafile when this tablespace filled.

• we can also created big file tablespace

• Create bigfile tablespace palikila datafile ‘/prod/hyd/oradata/palikila01.dbf’ size 30 m;

(8k block size 32T)

(4K block size 16T)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# TO CHECK TABLESPACES:

======================

• select name from v$tablespace;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# CREATE TABLESPACE:

==================

• create tablespace tablespace name datafile ‘/path/ tablespace name01.dbf’. Size 100m;

•

To know datafile location:

~~~~~~~~~~~~~~~~~~~

• select name from v$ datafile;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# To check datafiles under tablespace:

=============================

• select file\_name from dba\_data\_files where tablespace\_name=‘palikila’;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# TO DROP TABLESPACE:

===================

• drop tablespace palikila including contents and datafiles;

Tablespace dropped along with datafiles and data .

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Rename tablespace: Classic

=================

• alter tablespace (old name) rename to (new name);

• alter tablespace classic rename to palikila;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# TO CREATE USER ALONG WITH TABLESPACE :

===========================

• create user u1 identified by u1 default tablespace palikila;

We can also add profile :

• create user u1 identified by u1 default tablespace palikila profile p1;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Autoextend:

==========

• By default autoextend is enable for tablespace.

• When it is enable & and when the tablespace reached threshold level it automatically extends the size up to 32 gb.

# To check datafile is autoextend enable or not:

====================================

•

• select FILE\_NAME,BYTES/1024/1024,MAXBYTES/1024/1024,AUTOEXTEND ON from dba\_data\_files;

•

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# IF TABLESPACE FILLED UP :

======================

Check OS level size :

==============

df -h

1. resize the existing datafile

2. Add new datafile

1. resize datafile : palikila

~~~~~~~~~~~

• alter database datafile ‘/prod/hyd/oradata/palikila01.dbf’ resize 60m;

2. Adding datafile to’ filled tablespace:plikila

~~~~~~~~~~~~~~~~~~~~~~~~~~~

• alter tablespace palikila add datafile ‘/prod/hyd/oradata/palikila02.dbf’ size 30m;

We can also increase max size :palikila

~~~~~~~~~~~~~~~~~~~~~~~

• alter database datafile ‘/prod/hyd/oradata/palikila01.dbf’autoextend on maxsize 300 m;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# TO KNOW Block size:

===============

• desc dba\_tablespaces;

• select TABLESPACE\_NAME,BLOCK\_SIZE from dba\_tablespaces;

• Shows tablespaces block size

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# TO CHANGE BLOCK SIZE :

=====================

• we have to change parameter

• Show parameter cache

2k,4K,8k,16k,32k

Shows different parameters

• Alter system set db\_32k\_Cache\_Size=100m scope=both;

# create tablespace with different blocksize:

=================================

• create tablespace kota datafile ‘/prod/hyd/oradata/kota01.dbf’ size 30m blocksize 32768;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# TO CHECK TABLESPACE SIZE & AUTOENTENDABLE:

===============

• desc dba\_data\_files;

Shows some ……..

• select FILE\_NAME,BYTES/1024/1024,MAXBYTES/1024/1024,AUTOEXTENSIBLE from dba\_data\_files;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# TABLESPACE AUTO EXTEND ON/OFF:

=============================

• alter database datafile ‘ path of tablespace datafile.dbf’ autoextend on;

Alter database datafile ‘/prod/hyd/oradata/ palikila01.dbf’ autoextend on;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# TABLESAPCE LOGGING & NO LOGGING:

================================

• desc dba\_tablespaces

• if tablespace is in nologging the data in tablespace will not convert as archives.

We can create nologging tablespace and change to logging :

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

• create tablespace mouli datafile ‘ /prod/hyd/oradata/mouli01.dbf’ size 30m nologging;

To change from no logging to loggin:

~~~~~~~~~~~~~~~~~~~~~~~~~~~

• alter tablespace mouli loggin;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Tablespace status online / offline :

==========================

• online - read write.

• read only - we cannot load any data.we. Can perform only select operations

• select tablespace \_name,status from dba\_tablespaces;

To offline

~~~~~~~

• alter tablespace mouli offline;

To online

~~~~~~~

• alter tablespace mouli online;

[ we can’t offline system and undo tablespace and datafile. ]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Datafile online / offline & file # number

==============================

• desc dba\_data\_files

• desc v$datafile;

• select file\_name, status from dba\_data\_files;

• select file#,name,status from v$datafile;

• to offline the datafile db be must be enable with archive log mode.

To offline :

~~~~~~~

• alter database datafile ‘/prod/hyd/oradata/mouli01.dbf’ offline;

IT goes to recover state :

• recover datafile ‘/prod/hyd/oradata/mouli01.dbf’;

(OR)

• recover datafile 1;

Now it goes to offline state :

To online:

~~~~~~~

• alter database datafile ‘/prod/hyd/oradata/mouli01.dbf’ online;

[we can’t offline system and undo tablespace and datafile ]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Datafile rename :

==============

1. by offline tablespace (method)

2. Datafile offline(method)

11g:

~~~

# datafile offline method:

===================

• select name, status from v$datafile;

• Offline the datafile which we want to rename

• alter database datafile ‘/path/.dbf’ offline;

• datafile goes to recover state .

• recover it .

• recover datafile ‘ /path/.dbf’;

• now it goes to offline state .

shut down the database.

===================

• Shut immediate;

move data files from source to required destination.

===============================

Change in os level:

==============

• mv /prod/hyd/oradata/palikila01.dbf /prod/hyd/oradata/palikila01.dbf

start database in mount state

=======================

Startup mount;

Update in (CF) :

============

• alter database rename file ‘/prod/hyd/oradata/palikila01.dbf’ to ‘/prod/hyd/oradata/palikila\_01.dbf’;

• online the datafile.

In 12c & 19c:

~~~~~~~

# rename the datafile :

================

• select name,status from v$ datafile;

• alter database move datafile ‘/prod/hyd/oradata/palikila01.dbf’ to ‘ /prod/hyd/oradata/palikila\_01.dbf’;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Extent management:

=================

• desc dba\_tablespace

• Select Tablespace\_name,Extent\_managemet from dba\_tablespaces;

• in 11g system tablespace is in DISCRIONARY state.

• From 12c all the tablespaces are local.

# Based on extent management tablespace are of 2 types:

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

1. Dictionary managed tablespace

2. Locally managed tablespace

• by default all the tablespaces are build locally.

• every tablespace has a header and some body.

• If it is locally managed tablespace the all extents allocation information is stored in the header.

• When the extents information is stored in system tablespace it is discriminate managed tablespace.

• Disctionary tablespace requires more I/O OPERATIONS

# Disctionary:

===========

• based on the extents allocation information.(logical definition)

• logical definition is collection of contents segments

• Segments is a collection of extents.

• Extents is collection of contiguous oracle blocks

# Dictionary managed Tablespace:

~~~~~~~~~~~~~~~~~~~~~~~~~~

• create tablespace mouli datafile ‘/path/mouli01.dbf’ size 30m extent management dictionary;

# convert Disctionary to locally:

========================

• exec dbms\_space\_admin.Tablespace\_migrate\_To\_local(‘MOULI’);

• From

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# To check tablespace size:

====================

◦