Understanding the SCN

By **sandeep makol**
Oct 29, 2007

Note: This article was written for educational purpose only. Please refer to the related vendor documentation for detail.

Understanding the SCN

In order to understand how Oracle performs recovery, it's first necessary to understand Oracle's SCN in terms of the various places where it can be stored and how it's used for instance and media recovery.

The SCN is an internal number maintained by the database management system (DBMS) to log changes made to a database. The SCN increases over time as changes are made to the database by Structured Query Language (SQL). By understanding how the SCN is used, you can understand how Oracle recovery works. Oracle9i enables you to examine the current SCN using the following SQL:

```
SQL> select dbms_flashback.get_system_change_number from dual;
```

Whenever an application commits a transaction, the log writer process (LGWR) writes records from the redo log buffers in the System Global Area (SGA) to the online redo logs on disk. LGWR also writes the transaction's SCN to the online redo log file. The success of this atomic write event determines whether your transaction succeeds, and it requires a synchronous (wait-until-completed) write to disk.

Note: The need for a synchronous write upon commit is one of the reasons why the online redo log can become a bottleneck for applications and why you should commit as infrequently as is practical. In general, Oracle writes asynchronously to the database datafiles for performance reasons, but commits require a synchronous write because they must be guaranteed at the time they occur.

SCN and Checkpoints:

A checkpoint occurs when all modified database buffers in the Oracle SGA are written out to datafiles by the database writer (DBWn) process. The checkpoint process (CKPT) updates all datafiles and control files with the SCN at the time of the checkpoint and signals DBWn to write out the blocks. A successful checkpoint guarantees that all database changes up to the checkpoint SCN have been recorded in the datafiles. As a result, only those changes made after the checkpoint need to be applied during recovery. Checkpoints occur automatically as follows:

- Whenever a redo log switch takes place
- Whenever the time set by the LOG_CHECKPOINT_TIMEOUT initialization parameter is reached
- Whenever the amount of redo written reaches the number of bytes associated with the LOG_CHECKPOINT_INTERVAL

Typically, LOG_CHECKPOINT_INTERVAL is chosen so that checkpoints only occur on log switches. Oracle stores the SCN associated with the checkpoint in four places: three of them in the control file and one in the datafile header for each datafile.

The System Checkpoint SCN:

After a checkpoint completes, Oracle stores the system checkpoint SCN in the control file. You can access the checkpoint SCN using the following SQL:

```
SQL> select checkpoint_change# from v$database;


  CHECKPOINT_CHANGE#
--------------------
```

```
          292767
```

The Datafile Checkpoint SCN:

After a checkpoint completes, Oracle stores the SCN individually in the control file for each datafile. The following SQL shows the datafile checkpoint SCN for a single datafile in the control file:

```
SQL> select name,checkpoint_change# from v$datafile where name like '%users01%';

 NAME                              CHECKPOINT_CHANGE#
----------------------------------- --------------------
/u02/oradata/OMFD1/users01.dbf                292767
```

The Start SCN:

Oracle stores the checkpoint SCN value in the header of each datafile. This is referred to as the start SCN because it is used at instance startup time to check if recovery is required. The following SQL shows the checkpoint SCN in the datafile header for a single datafile:

```
SQL> select name,checkpoint_change# from v$datafile_header where name like '%users01%';

 NAME                              CHECKPOINT_CHANGE#
----------------------------------- --------------------
/u02/oradata/OMFD1/users01.dbf                292767
```

The Stop SCN:

The stop SCN is held in the control file for each datafile. The following SQL shows the stop SCN for a single datafile when the database is open for normal use:

```
SQL> select name,last_change# from v$datafile where name like '%users01%';

NAME                              LAST_CHANGE#
----------------------------------- ------------
/u02/oradata/OMFD1/users01.dbf
```

During normal database operation, the stop SCN is NULL for all datafiles that are online in read-write mode. SCN Values while the Database Is Up Following a checkpoint while the database is up and open for use, the system checkpoint in the control file, the datafile checkpoint SCN in the control file, and the start SCN in each datafile header all match. The stop SCN for each datafile in the control file is NULL. SCN after a Clean Shutdown After a clean database shutdown resulting from a SHUTDOWN IMMEDIATE or SHUTDOWN NORMAL of the database, followed by STARTUP MOUNT, the previous queries on v$database and v$datafile return the following:

```
SQL> select checkpoint_change# from v$database;

  CHECKPOINT_CHANGE#
--------------------
          293184

SQL> select name,checkpoint_change#,last_change# from v$datafile where name like '%user%';

NAME                              CHECKPOINT_CHANGE#   LAST_CHANGE#
----------------------------------- -------------------- --------------
/u02/oradata/OMFD1/users01.dbf                293184         293184


SQL> select name,checkpoint_change# from v$datafile_header where name like '%users01%';

 NAME                              CHECKPOINT_CHANGE#
----------------------------------- --------------------
/u02/oradata/OMFD1/users01.dbf                293184
```

During a clean shutdown, a checkpoint is performed and the stop SCN for each datafile is set to the start SCN from the datafile header. Upon startup, Oracle checks the start SCN in the file header with the datafile checkpoint SCN. If they match, Oracle checks the start SCN in the datafile header with the datafile stop SCN in the control file. If they match, the database can be opened because all block changes have been applied, no changes were lost on shutdown, and therefore no recovery is required on startup. After the database is opened, the datafile stop SCN in the control file once again changes to NULL to indicate that the datafile is open for normal use.

SCN after an Instance Crash

The previous example showed the behavior of the SCN after a clean shutdown. To demonstrate the behavior of the checkpoints after an instance crash, the following SQL creates a table (which performs an implicit commit) and inserts a row of data into it without a commit:

```
create table x(x number) tablespace users;

insert into x values(100);
```

If the instance is crashed by using SHUTDOWN ABORT, the previous queries on v$database and v$datafile return the following after the database is started up in mount mode:

```
SQL> select checkpoint_change# from v$database;

  CHECKPOINT_CHANGE#
--------------------
           293185

 SQL> select name,checkpoint_change#,last_change# from v$datafile where name like '%users01%';

NAME                              CHECKPOINT_CHANGE#   LAST_CHANGE#
---------------------------------- -------------------- --------------
/u02/oradata/OMFD1/users01.dbf                293185


SQL> select name,checkpoint_change# from v$datafile_header where name like '%users01%';

NAME                              CHECKPOINT_CHANGE#
---------------------------------- --------------------
/u02/oradata/OMFD1/users01.dbf                293185
```

In this case, the stop SCN is not set, which is indicated by the NULL value in the LAST_CHANGE# column. This information enables Oracle, at the time of the next startup, to determine that the instance crashed because the checkpoint on shutdown was not performed. If it had been performed, the LAST_CHANGE# and CHECKPOINT_CHANGE# values would match for each datafile as they did during a clean shutdown. If an instance crashes at shutdown, then instance crash recovery is required the next time the instance starts up.

Recovery from an Instance Crash

Upon the next instance startup that takes place after SHUTDOWN ABORT or a DBMS crash, the Oracle DBMS detects that the stop SCN for datafiles is not set in the control file during startup. Oracle then performs crash recovery. During crash recovery, Oracle applies redo log records from the online redo logs in a process referred to as roll forward to ensure that all transactions committed before the crash are applied to the datafiles. Following roll forward, active transactions that did not commit are identified from the rollback segments and are undone before the blocks involved in the active transactions can be accessed. This process is referred to as roll back. In our example, the following transaction was active but not committed at the time of the SHUTDOWN ABORT, so it needs to be rolled back:

```
SQL> insert into x values(100);
```

After instance startup, the X table exists, but remains empty. Instance recovery happens automatically at database startup without database administrator (DBA) intervention. It may take a while because of the need to apply large amounts of outstanding redo changes to data blocks for transactions that completed and those that didn't complete and require roll back.

Recovery from a Media Failure

Up until this point, the checkpoint start SCN in the datafile header has always matched the datafile checkpoint SCN number held in the control file. This is reasonable because during a checkpoint, the datafile checkpoint SCN in the control file and the start SCN in the datafile header are both updated, along with the system checkpoint SCN. The following SQL shows the start SCN from the datafile header and datafile checkpoint SCN from the control file for the same file:

```
SQL> select 'controlfile' "SCN location",name,checkpoint_change# from v$datafile where name like
'%users01%'
union
select 'file header',name,checkpoint_change# from v$datafile_header where name like '%users01%';

SCN location   NAME                              CHECKPOINT_CHANGE#
-------------- ---------------------------------- --------------------
controlfile    /u02/oradata/OMFD1/users01.dbf                293188
file header    /u02/oradata/OMFD1/users01.dbf                293188
```

Unlike the v$datafile view, there is no stop SCN column in the v$datafile_header view because v$datafile_header is not used at instance startup time to indicate that an instance crash occurred. However, the v$datafile_header does provide the Oracle DBMS with the information it requires to perform media recovery. At instance startup, the datafile checkpoint SCN in the control file and the start SCN in the datafile header are checked for equality. If they don't match, it is a signal that media recovery is required.

For example, media recovery is required if a media failure has occurred and the original datafile has been replaced with a backup copy. In this case, the start SCN in the backup copy is less than the checkpoint SCN value in the control file, and Oracle requests archived redo logs—generated at the time of previous log switches—in order to reapply the changes required to bring the datafile up to the current point in time.

In order to recover the database from a media failure, you must run the database in ARCHIVELOG mode to ensure that all database changes from the online redo logs are stored permanently in archived redo log files. In order to enable ARCHIVELOG mode, you must run the command ALTERDATABASE ARCHIVELOG when the database is in a mounted state.

You can identify files that need recovery after you have replaced a datafile with an older version by starting the instance in mount mode and running the following SQL:

```
SQL> select file#,change# from v$recover_file;

     FILE#    CHANGE#
---------- ----------
```

```
        4     313401
```

In this example, file 4 is the datafile in the USERS tablespace. By reexecuting the previous SQL to display the datafile checkpoint SCN in the control file and the start SCN in the datafile header, you can see that the start SCN is older due to the restore of the backup datafile that has taken place:

```
SQL> select 'controlfile' "SCN location",name,checkpoint_change#
from v$datafile where name like '%users01%'
union
select 'file header',name,checkpoint_change#
from v$datafile_header where name like '%users01%';

SCN location   NAME                                CHECKPOINT_CHANGE#
-------------- ---------------------------------- --------------------
controlfile    /u02/oradata/OMFD1/users01.dbf              313551
file header    /u02/oradata/OMFD1/users01.dbf              313401
```

If you were to attempt to open the database, you would receive errors like the following:

```
ORA-01113: file 4 needs media recovery
ORA-01110: datafile 4: '/u02/oradata/OMFD1/users01.dbf'
```

You can recover the database by issuing RECOVER DATABASE from SQL*Plus while the database is in a mounted state. If the changes needed to recover the database to the point in time before the crash are in an archived redo log, then you will be prompted to accept the suggested name:

```
ORA-00279: change 313401 generated at 11/10/2001 18:50:23 needed for thread
ORA-00289: suggestion : /u02/oradata/OMFD1/arch/T0001S0000000072.ARC
ORA-00280: change 313401 for thread 1 is in sequence #72

Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
```

If you respond to the prompt using AUTO, Oracle applies any archived redo logs it needs, followed by any necessary changes in the online redo logs, to bring the database right up to the last committed transaction before the media failure that caused the requirement for the restore.

So far, we've considered recovery scenarios where the goal is to recover the database to the most recent transaction. This is known as complete recovery. The RECOVER DATABASE command has several other options that enable you to recover from a backup to a point in time before the most recent transaction by rolling forward and then stopping the application of the redo log changes at a specified point. This is known as incomplete recovery. You can specify a time or an SCN as the recovery point. For example,

```
recover database until time '2001-11-10:18:52:00';
recover database until change 313459;
```

Before you perform incomplete recovery, it's recommended that you restore a complete database backup first. After incomplete recovery, you must open the mounted database with ALTER DATABASE OPEN RESETLOGS. This creates a new incarnation of the database and clears the contents of the existing redo logs to make sure they can't be applied.

Recovery from a Media Failure Using a Backup Control File

In the previous example, we had access to a current control file at the time of the media failure. This means that none of the start SCN values in the datafile headers exceeded the system checkpoint SCN number in the control file. To recap, the system checkpoint number is given by the following:

```
SQL> select checkpoint_change# from v$database;
```

You might be wondering why Oracle needs to maintain the last system checkpoint value in the control file as well as checkpoint SCNs in the control file for each datafile (as used in the previous example). There are two reasons for this. The first is that you might have read-only tablespaces in your database. In this case, the database checkpoint SCN increases, and the checkpoint SCN for the datafiles in the read-only tablespace remains frozen in the control file.

The following SQL report output shows a database with a read-write tablespace (USERS) and read-only tablespace (TEST). The start SCN in the file header and the checkpoint SCN in the control file for TEST are less than the system checkpoint value. Once a tablespace is read only, checkpoints have no effect on the files in it. The other read-write tablespace has checkpoint values that match the system checkpoint:

```
SCN location         NAME                             CHECKPOINT_CHANGE#
-------------------- -------------------------------- ----------------
controlfile          SYSTEM checkpoint                       355390
file header          /u02/oradata/OD2/users01.dbf            355390
file in controlfile  /u02/oradata/OD2/users01.dbf            355390
file header          /u02/oradata/OD2/test01.dbf             355383
file in controlfile  /u02/oradata/OD2/test01.dbf             355383
```

The second reason for the maintenance of multiple checkpoint SCNs in the control file is that you might not have a current control file available at recovery time. In this case, you need to restore an earlier control file before you can perform a recovery. The system checkpoint in the control file may indicate an earlier change than the start SCN in the datafile headers.

The following SQL shows an example where the system checkpoint SCN and datafile checkpoint SCN indicate an earlier change than the start SCN in the datafile header:

```
SQL> select 'controlfile' "SCN location",'SYSTEM checkpoint' name,checkpoint_change#
from v$database
```

```
union
select 'file in controlfile',name,checkpoint_change#
from v$datafile where name like 'users01%'
union
select 'file header',name,checkpoint_change#
from v$datafile_header where name like '%users01%';
```

```
SCN location        NAME                          CHECKPOINT_CHANGE#
------------------- ------------------------------   ------------------
controlfile         SYSTEM checkpoint                          333765
file header          /u02/oradata/OD2/users01.dbf                355253
file in controlfile /u02/oradata/OD2/users01.dbf                333765
```

If try you to recover a database in the usual way in this situation, Oracle detects that the control file is older than some of the datafiles, as indicated by the checkpoint SCN values in the datafile headers, and reports the following message:

```
SQL> recover database
ORA-00283: recovery session canceled due to errors
ORA-01610: recovery using the BACKUP CONTROLFILE option must be done
```

If you want to proceed with recovery in this situation, you need to indicate to Oracle that a noncurrent control file—possibly containing mismatches in the SCN values identified by the previous error messages—is about to be specified for recovery by using the following command:

```
recover database using BACKUP CONTROLFILE;
```

Comments/Reviews on this article:

Abi Abi
Aug 14, 2011

One of the Best Articles I have gone through on Backup and Recovery

mohammedsalim memon
Aug 19, 2010

great.............
Thank you,sandeep
i like your explanation and again thank you for sharing knowledge.

subramaniyan balraj
Feb 15, 2008

it is very clear. i understood easily...

update furter articals.

anna farah
Feb 28, 2008

I appreciate your work .i couldn't fully get abt SCN before,but now i easily understood.
Thanx very much.

Niel Sahastrabuddhe
Jun 03, 2011

Thanks Sandep for nice explanation.
but i want to know 1) how long uncommited transaction remain in datafile.and which process remove this uncommited transaction if database is open.please explain

Praveen Malgi
Mar 30, 2010

Superb explanation , gr8 job !!
I am very clear with instance recovery done considering SCN

Ruslan Amalatov
May 07, 2009

Thank you, Sandeep.
Explanation is very clear and helpful. Good job.

Ruslan Amalatov
May 07, 2009

Thank you, Sandeep.
Explanation is very clear and helpful. Good job.

dharmendra singh
Oct 19, 2010

simply great
keep it up.....

About author:

Please login to post your comments