

Understanding Oracle Locking Internals

Arup Nanda

Longtime Oracle DBA

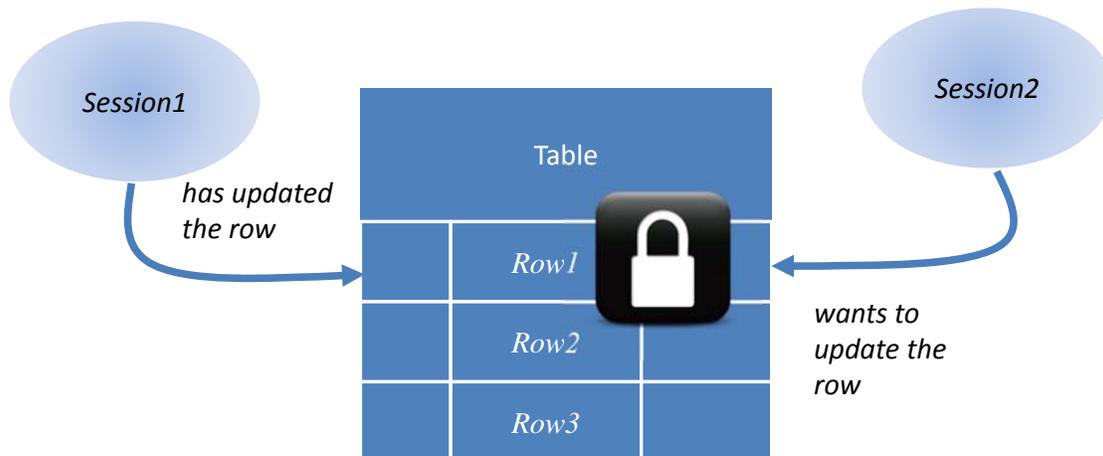
@ArupNanda

Copyright

IMPORTANT

Please feel free to use the material presented here for your benefit. All I ask is please attribute the source, which is this material, in your usage.

Locking



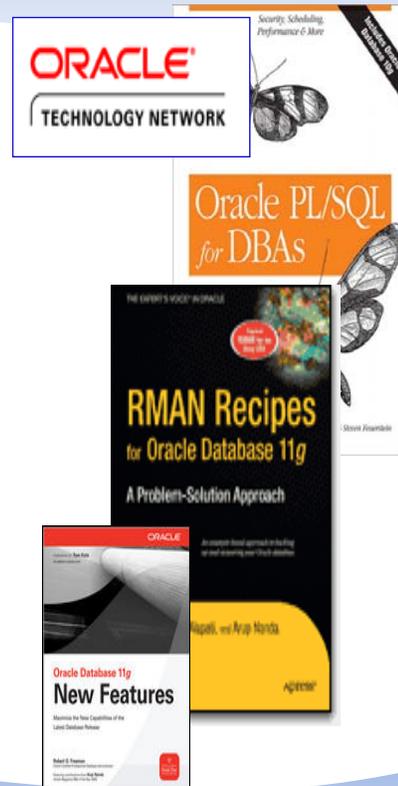
Is there a *central lock store* in Oracle?

Agenda

- What this is about?
 - How Oracle Locking Works
 - Understanding locking behavior
 - Tuning locking operations
- Tools
 - SQL*Plus

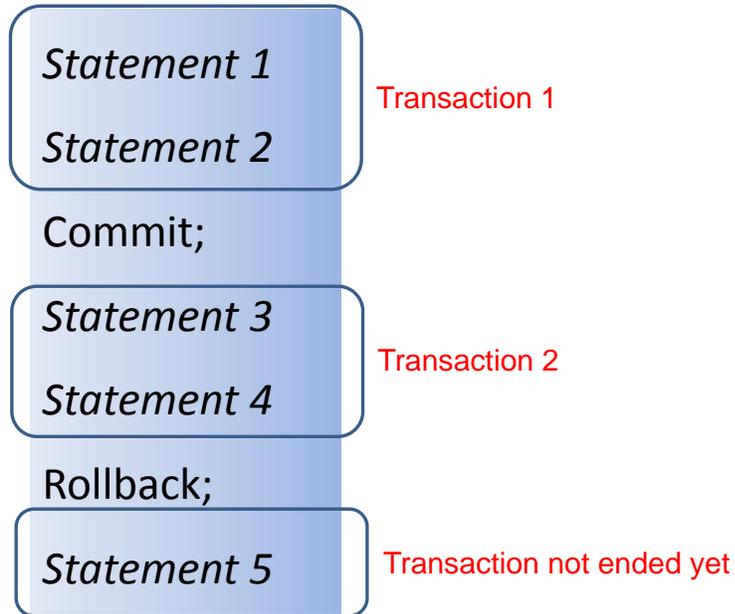
About Me

- Oracle DBA for 20 years and counting
- Speak at conferences, write articles, 4 books, provide trainings, security audits
- Blog: arup.blogspot.com
- Tweeter: @arupnanda
- Facebook.com/ArupKNanda

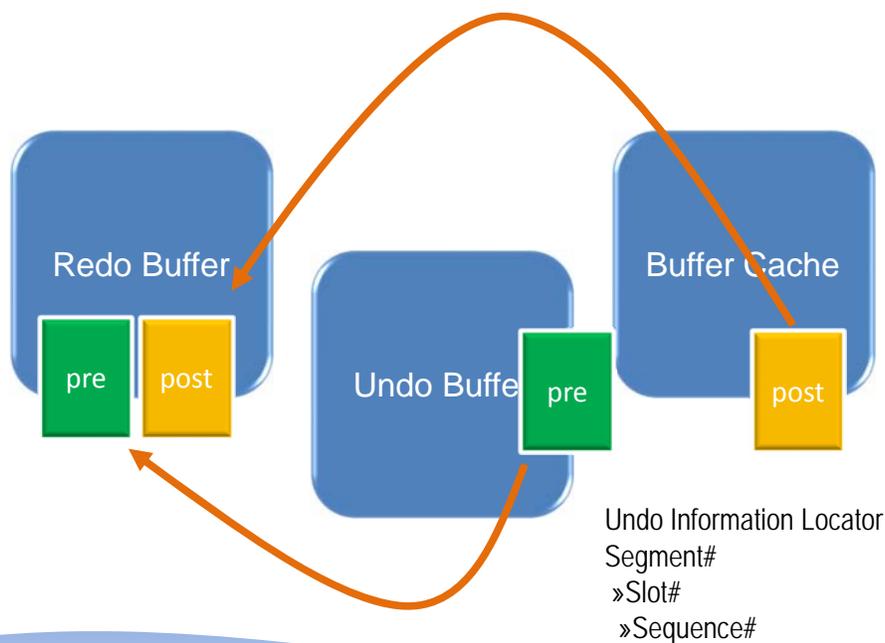


Transaction

- A transaction is a block which is ended by a commit or a rollback



Transaction Data



Transaction ID

- Checking for Transaction ID in own session
 - `dbms_transaction.local_transaction_id`

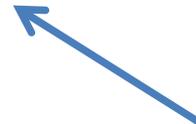
10.25.31749



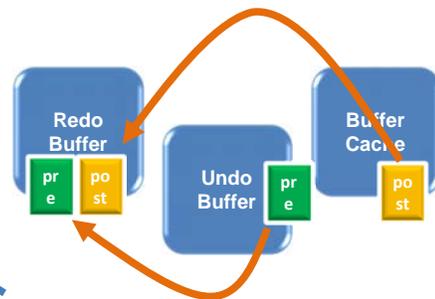
Undo
Segment#



Slot# in that
segment



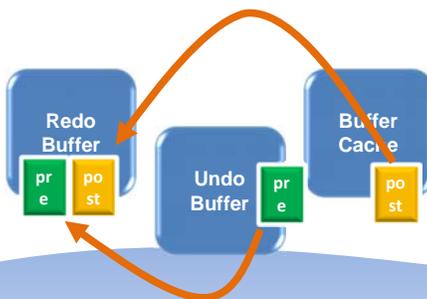
Sequence# in
that slot



Transaction Table

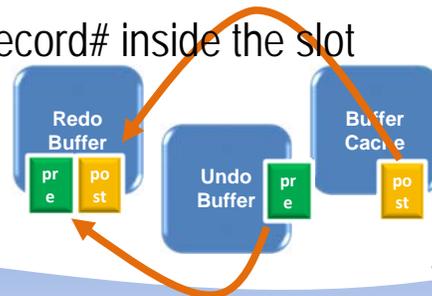
- A memory structure
- In SGA
- Exposed as `X$KTCXB`
- Visible as `V$TRANSACTION`

Transaction ID	Other Relevant Information
10.25.31749	Active/Inactive, Undo Blocks, etc.
10.25.10234	
10.25.32345	



Checking for Txns

- All the transactions in the instance
 - select addr, xidusn, xidslot, xidsqn
from v\$transaction;
 - ADDR: the address of the transaction – a raw value
 - XIDUSN: the undo segment number
 - XIDSLOT: the slot#
 - XIDSQN: the sequence# or record# inside the slot



Txn and Session

- To Know Active Txns of a Session, join with V\$SESSION

```
select sid
from v$session s,
v$transaction t
where t.ses_addr =
s.saddr
```

Txn1.sql

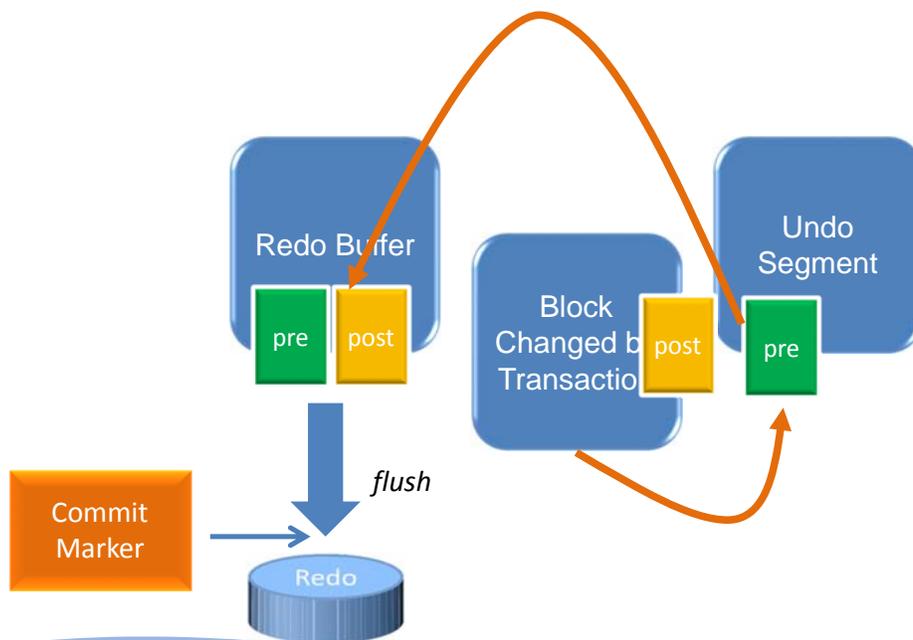
OR

V\$TRANSACTION	V\$SESSION
ADDR	TADDR
SES_ADDR	SADDR

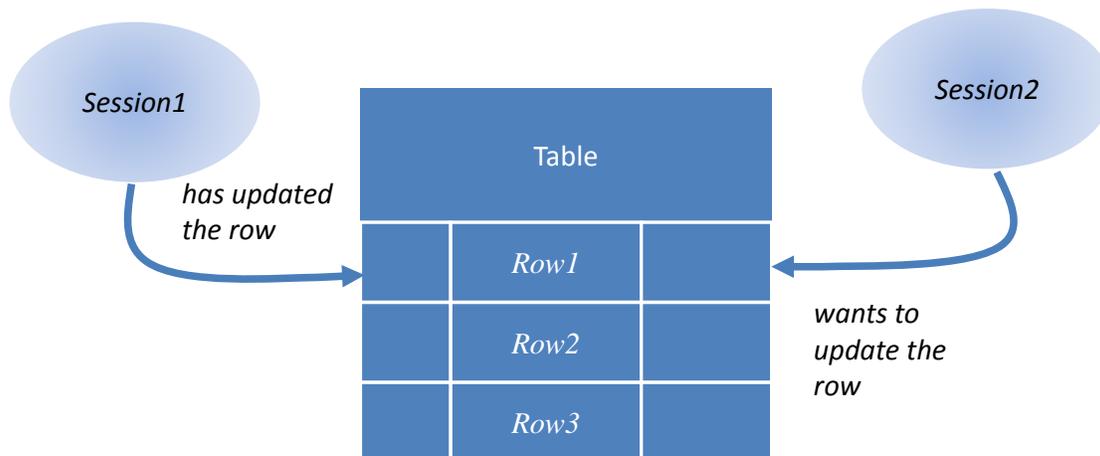
```
select sid
from v$session s,
v$transaction t
where t.addr = s.taddr
```

Txn2.sql

Commit



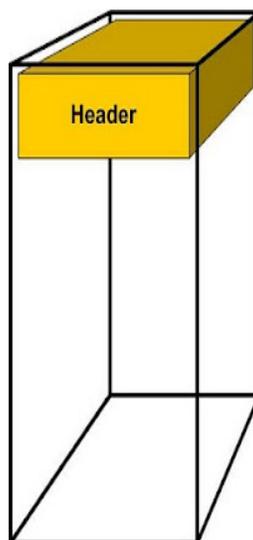
Locking



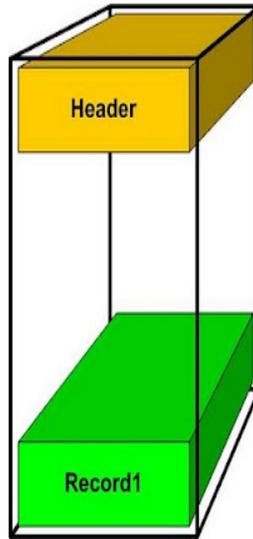
Lock Location

- There is no central locking facility
- Oracle puts the lock for a row in the block itself
- In slots called ITL Entry

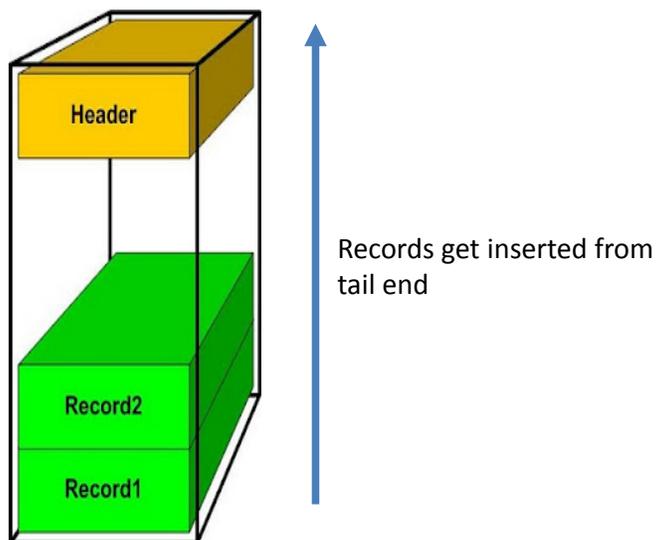
Empty Block



Records Getting Inserted

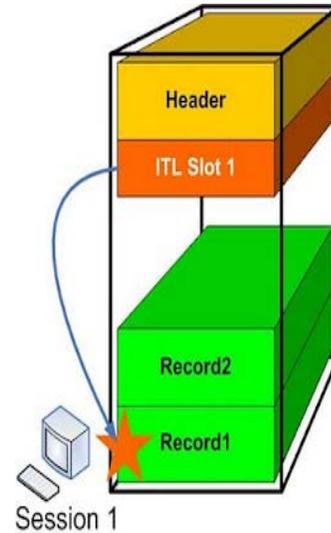


2nd Record



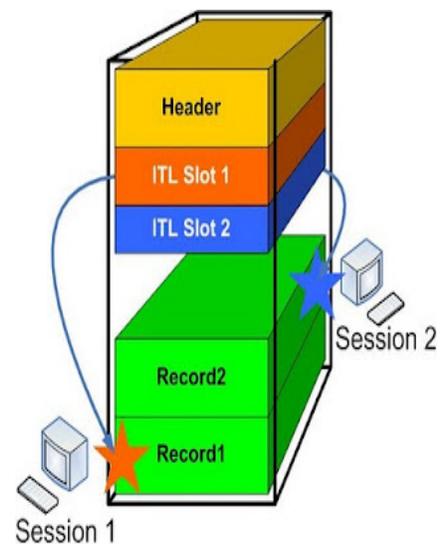
Locking

- Session 1 locks row 1
- It puts that information in the header of that block
- The transaction is “interested” in that row, hence “Interested Transaction”



Locking #2

- Session 2 locks row 2
- It also puts an interested transaction entry
- Now there is a “list” of interested transactions known as ITL

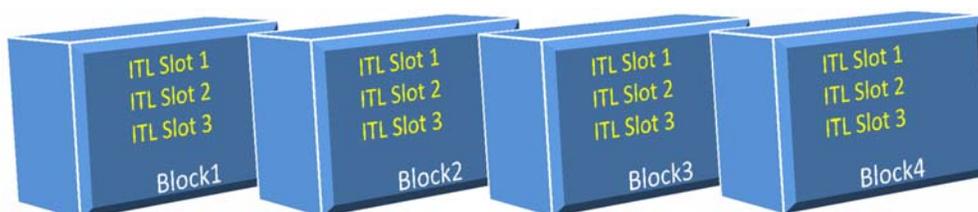


ITLs Continued

- Each ITL slot takes 24 bytes
- The total number of ITL slots can grow, as long as there is room in the block
- Can't exceed 50% of the block
- Max ITL is 255

Checking ITL

- Getting a block dump
`alter system dump datafile <DF#> block min <block#> block max <block#>;`
- Creates a tracefile with the dump of the block



ITL Record

Itl	Xid	Uba	Flag	Lck	Scn/Fsc
0x01	0x000a.019.00007c05	0x00c00288.1607.0e	----	1	fsc 0x0000.00000000
0x02	0x0003.017.00009e24	0x00c00862.190a.0f	C---	0	scn 0x0000.02234e2b

Transaction ID, corresponding to V\$TRANSACTION
Undo Segment Information
Flag – locked, etc.
Number of rows locked

Commit and ITL



- ITL is not updated when commit happens
 - Commits are superfast
- When a new transaction encounters a lock in ITL
 - it must check the transaction table
 - If ACTIVE, then it's actually locked
- ITLs are cleared during cleanout

V\$LOCKED_OBJECT

- Shows you the object locked



- OBJECT_ID
- SESSION_ID
- ORACLE_USERNAME
- OS_USER_NAME
- PROCESS
- **LOCKED_MODE**

0 - None
1 - Null
2 - Row Share
3 - Row Exclusive
4 - Share
5 - Sub Share Exclusive
6 - Exclusive

```
select
  owner                object_owner,
  object_name          object_name,
  session_id           oracle_sid,
  oracle_username      db_user,
  decode(locked_mode,
    0, 'None',
    1, 'Null',
    2, 'Row Share',
    3, 'Row Exclusive',
    4, 'Share',
    5, 'Sub Share Exclusive',
    6, 'Exclusive',
    locked_mode) locked_mode
from v$locked_object lo, dba_objects do
where (xidusn||'.'||xidslot||'.'||xidsqn) = ('&transid')
and do.object_id = lo.object_id
```

Blocking Session

- To find out the session that holds the lock this session is asking for

```
select
  blocking_session,
  blocking_instance,
  seconds_in_wait
from v$session
where sid = <sid>;
```

The SID of the session holding the lock

The Instance of the other session holding the lock

How long it has been waiting

Locked Row

- Checking for the row information

```
select row_wait_obj#,
       row_wait_file#,
       row_wait_block#,
       row_wait_row#
from v$session
where sid = <SID>;
```

To get the object information:

```
select owner, object_type, object_name,
       data_object_id
from dba_objects
where object_id = 241876;
```

OWNER	OBJECT_TYPE	OBJECT_NAME	DATA_OBJECT_ID
ARUP	TABLE	T1	241877

```
ROW_WAIT_OBJ# ROW_WAIT_FILE# ROW_WAIT_BLOCK# ROW_WAIT_ROW#
-----
241876          1024          2307623          0
```

Row from RowID

```
select * from arup.t1
where rowid = dbms_rowid.rowid_create (
    rowid_type      => 1,
    object_number   => 241877,
    relative_fno    => 1024,
    block_number    => 2307623,
    row_number      => 0
);
```

Note:
DATA_OBJECT_ID;
not OBJECT_ID

```
COL1 C
```

```
-----
```

```
1 X
```

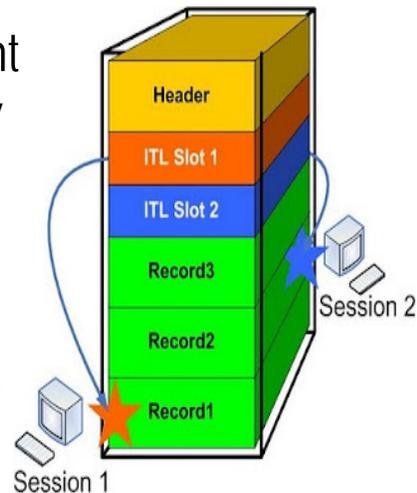
← This is the row on which the lock is being requested

A Useful Script

```
select
    object_type,
    owner||'.'||object_name||':'||nvl(subobject_name, '-')
obj_name,
    dbms_rowid.rowid_create (
        1,
        o.data_object_id,
        row_wait_file#,
        row_wait_block#,
        row_wait_row#
    ) row_id
from v$session s, dba_objects o
where sid = &sid
and o.object_id = s.row_wait_obj#
/
```

ITL Shortage

- ITL can't grow when the block is full
- The session will wait with an event `enq: TX - allocate ITL entry`
- To avoid it
 - Have plenty of room in the block
 - Increased PCTFREE, etc.
 - MINIMIZE_RECORDS_PER_BLOCK
 - Have a larger INITRANS



Finding ITL Shortage

- Query

```
select statistic_name, value
from v$segment_statistics
where object_name = '<Object Name>;
```

- Output

STATISTIC_NAME	VALUE
-----	-----
logical reads	7216
ITL waits	2
...	

STATISTIC_NAME

ITL waits
row lock waits

Historical

- AWR Repository

```
select snap_id, itl_waits_total, itl_waits_delta
from dba_hist_seg_stat
where obj# = <ObjID>
order by snap_id;
```

- Stats of Interest

- ITL_WAITS_TOTAL
- ITL_WAITS_DELTA
- ROW_LOCK_WAITS_TOTAL
- ROW_LOCK_WAITS_DELTA

Summary

- There is no central locking in Oracle
- A txn marks the rows locked in the block itself
- This is called Interested Transaction List (ITL)
- If no ITL slot is available, one is created if there is space; otherwise txn waits with ITL waits
- ITL entry shows undo information
- ITL is not updated as a part of commit
- Blog: <http://arup.blogspot.com/2011/01/how-oracle-locking-works.html>



Thank You!

Blog: arup.blogspot.com

Tweeter: [@arupnanda](https://twitter.com/arupnanda)

Facebook.com/[ArupKNanda](https://www.facebook.com/ArupKNanda)