



Dbvisit
THE SMART ALTERNATIVE

Evolution of Oracle Database Redo Logs Through Versions

UGF2264

Vit Spinka

Agenda

- What is redo, after all?
- Basic structure of redo
- Change record in detail
- Across versions: changes to data change records
- Across versions: new change record types
- Q&A

Vit Spinka

- Working with Oracle Database since *8i*
- Oracle Certified Master
- Principal developer of Dbvisit Replicate
- ... which gets its data by parsing Oracle redo logs
- @vitspinka
- vit.spinka@dbvisit.com
- This presentation for download at <http://vitspinka.cz/download.html>



Dbvisit



- HQ in New Zealand, US subsidiary, partners throughout the world
- Used in 80+ Countries
- Database Replication is our playground
- Worldwide leader in DR solutions for Oracle Standard Edition
- Product Engineers with “real world” DBA Experience
- Regular presenters at Oracle events such as OOW, Collaborate and NZOUG
- Passionate about Oracle Technology



Trusted in 80+ countries. . .

CDLQ ALDO ARUP Dbvisit THE SMART ALTERNATIVE Mercedes-Benz UK FUELS at&t Pythian love your data AGFA HealthCare IBM Boehringer Ingelheim First Citizens Bank Amazon web services Alcatel-Lucent Verizon NHS imation ORION HEALTH

. . . By 800+ companies.

What is redo, after all

- Basic goal: recover database
- Online redo: instance/crash recovery
- Archive redo: media recovery
- Every change is written to redo before data files
- Allows to replay changes

What is redo, after all

- Optimized for recovery and for performance
- Binary format
- Even when dumped to text, still pretty cryptic
- Saves space by using object ids; wastes space by aligning to blocks
- Block size “fixed” (actually: default 512 bytes, HP-UX 1024 bytes, 11.2+ and Advanced Disk Format: 4096 bytes)

Redo dump

- MOS Note 103181.6
- ALTER SYSTEM DUMP LOGFILE 'filename';
- The resulting trace file is huge and contains most (but not all!) interesting information from the redo in a text format
- We will see these dumps further in this session
- Database can dump redo from a different database, as long as the endian (big/little) match and version is the same or higher than the source db*

Basic structure of redo

- Header

FILE HEADER:

```
Compatibility Vsn = 202375168=0xc100000
Db ID=1365223133=0x515fa6dd, Db Name='ORCL'
Activation ID=1365239005=0x515fe4dd
Control Seq=3327=0xcff, File size=102400=0x19000
File Number=2, Blksiz=512, File Type=2 LOG
```

```
descrip:"Thread 0001, Seq# 0000000104, SCN 0x0000002d22aa-0xffffffffffff"
thread: 1 nab: 0xffffffff seq: 0x00000068 hws: 0x1 eot: 1 dis: 0
resetlogs count: 0x31ea5b5f scn: 0x0000.001a3f12 (1720082)
prev resetlogs count: 0x30a7312f scn: 0x0000.00000001 (1)
Low scn: 0x0000.002d22aa (2957994) 07/08/2014 15:58:30
Next scn: 0xffff.ffffffff 01/01/1988 00:00:00
```

FILE HEADER:

Compatibility Vsn = 202375168=0xc100000

Db ID=1365223133=0x515fa6dd, Db Name='ORCL'

Activation ID=1365239005=0x515fe4dd

Control Seq=3327=0xcff, File size=102400=0x19000

File Number=2, Blksiz=512, File Type=2 LOG

descrip:"Thread 0001, Seq# 0000000104, SCN 0x0000002d22aa-0xffffffffffffffff"

thread: 1 nab: 0xffffffff seq: 0x00000068 hws: 0x1 eot: 1 dis: 0

resetlogs count: 0x31ea5b5f scn: 0x0000.001a3f12 (1720082)

prev resetlogs count: 0x30a7312f scn: 0x0000.00000001 (1)

Low scn: 0x0000.002d22aa (2957994) 07/08/2014 15:58:30

Next scn: 0xffff.ffffffff 01/01/1988 00:00:00

Basic structure of redo

- Blocks
- Only a physical structure
- They are used in the RBA – redo block address

Basic structure of redo

- Redo record
- Spans one or more redo blocks
- One redo block can have multiple redo records (if created by the same server process)
- Each redo record has an RBA, pointing to start of the redo record
- Header – SCN, time, size; trace will lie and show these even if they are not in the header
- Change records – actual data

```
REDO RECORD - Thread:1 RBA: 0x000068.00000015.0010 LEN: 0x007c VLD: 0x05 CON_UID: 3345156736  
SCN: 0x0000.002d22c3 SUBSCN: 1 07/08/2014 15:58:58  
(LWN RBA: 0x000068.00000015.0010 LEN: 0002 NST: 0001 SCN: 0x0000.002d22c0)
```

REDO RECORD - Thread:1 RBA: 0x000068.00000015.0010 LEN: 0x007c VLD: 0x05

CON_UID: 3345156736

SCN: 0x0000.002d22c3 SUBSCN: 1 07/08/2014 15:58:58

(LWN RBA: 0x000068.00000015.0010 LEN: 0002 NST: 0001 SCN: 0x0000.002d22c0)

Basic structure of redo

- Change record
- Describes actual change
- Data changes: change to undo, change to data (tables, indexes)
- Transactions: changes to undo (allocate undo entry), commit/rollback
- Maintenance: space allocation, ASSM bitmaps...
- All is referenced by ids valid on the source: object id, data object id, datafile Absolute File Number, DBA, undo segment id...

Basic structure of redo

- Since 9i, redo is used by LogMiner and Streams
- More information needed:
- How do you assemble row pieces together
- How do you identify row if you cannot use rowid
- Answer: supplemental logging
- Adds row assembly information
- Adds primary key information (or whatever you specify)
- Oracle also added actual DDL text to redo
- Sadly, none of this is visible in the text dump

Change record in detail

- Type of change is in OP, consisting of “Layer” and “Opcode”
- Interesting layers:
 - 5: Undo (5.2 allocate undo header, 5.1 undo change, 5.4 commit)
 - 11: Table data (insert, update, delete, lock, ...)
 - 10: Index data
 - And some more (19.1 direct load, 24.1 DDL)
- One change record (usually) modifies one database block

Change record in detail

- Example: one insert, then commit
- Start transaction:

```
CHANGE #2 CON_ID:1 TYP:0 CLS:19 AFN:4 DBA:0x01000090 OBJ:4294967295 SCN:0x0000.002d228d SEQ:1 OP:5.2 ENC:0  
RBL:0  
ktudh redo: slt: 0x0013 sqn: 0x000008d3 flg: 0x0052 siz: 112 fbi: 0  
          uba: 0x01003327.0187.04      pxid: 0x0000.000.00000000      pdbid:3345156736
```

- (strictly speaking: this is undo block allocation and can happen multiple times in a transaction)
- SCN in change header is not interesting – SCN of change is in redo header

CHANGE #2 CON_ID:1 TYP:0 CLS:19 AFN:4 DBA:0x01000090 OBJ:4294967295
SCN:0x0000.002d228d SEQ:1 OP:5.2 ENC:0 RBL:0
ktudh redo: slt: 0x0013 sqn: 0x000008d3 flg: 0x0052 siz: 112 fbi: 0
uba: 0x01003327.0187.04 pxid: 0x0000.000.00000000
pdbid:3345156736

Change record in detail

- Example: insert into hr.jobs values ('TE_TEST', 'Test job', 1, 100)
- Redo

```
CHANGE #1 CON_ID:3 TYP:0 CLS:1 AFN:13 DBA:0x034000bb OBJ:95424 SCN:0x0000.00285536 SEQ:2 OP:11.2 ENC:0 RBL:0
KTB Redo
op: 0x01 ver: 0x01
compat bit: 4 (post-11) padding: 1
op: F xid: 0x0002.013.000008d3 uba: 0x01003327.0187.04
KDO Op code: IRP row dependencies Disabled
  xtype: XA flags: 0x00000000 bdba: 0x034000bb hdba: 0x034000ba
itli: 1 ispac: 0 maxfr: 4858
tabn: 0 slot: 0(0x0) size/delt: 26
fb: --H-FL-- lb: 0x1 cc: 4
null: ----
col 0: [ 7] 54 45 5f 54 45 53 54
col 1: [ 8] 54 65 73 74 20 6a 6f 62
col 2: [ 2] c1 02
col 3: [ 2] c2 02
```

CHANGE #1 CON_ID:3 TYP:0 CLS:1 AFN:13 DBA:0x034000bb OBJ:95424

SCN:0x0000.00285536 SEQ:2 OP:11.2 ENC:0 RBL:0

KTB Redo

op: 0x01 ver: 0x01

compat bit: 4 (post-11) padding: 1

op: F xid: 0x0002.013.000008d3 uba: 0x01003327.0187.04

KDO Op code: IRP row dependencies Disabled

xtype: XA flags: 0x00000000 bdba: 0x034000bb hdba: 0x034000ba

itli: 1 ispac: 0 maxfr: 4858

tabn: 0 slot: 0(0x0) size/delt: 26

fb: --H-FL-- lb: 0x1 cc: 4

null: ----

col 0: [7] 54 45 5f 54 45 53 54

col 1: [8] 54 65 73 74 20 6a 6f 62

col 2: [2] c1 02

col 3: [2] c2 02

Change record in detail



```
CHANGE #5 CON_ID:1 TYP:0 CLS:20 AFN:4 DBA:0x01003327 OBJ:4294967295 SCN:0x0000.002d228c SEQ:1 OP:5.1 ENC:0
RBL:0
ktudb redo: siz: 112 spc: 7776 flg: 0x0012 seq: 0x0187 rec: 0x04
           xid: 0x0002.013.000008d3
ktubl redo: slt: 19 rci: 0 opc: 11.1 [objn: 95424 objd: 95424 tsn: 3]
Undo type:  Regular undo           Begin trans      Last buffer split:  No
Temp Object:  No
Tablespace Undo:  No
              0x00000000 prev ctl uba: 0x01003327.0187.03
prev ctl max cmt scn: 0x0000.002d1e27 prev tx cmt scn: 0x0000.002d1e39
txn start scn: 0x0000.002d22c3 logon user: 114 prev brb: 16790299 prev bcl: 0 BuExt idx: 0 flg2: 0
KDO undo record:
KTB Redo
op: 0x03 ver: 0x01
compat bit: 4 (post-11) padding: 1
op: Z
KDO Op code: DRP row dependencies Disabled
      xtype: XA flags: 0x00000000 bdba: 0x034000bb hdba: 0x034000ba
itli: 1 ispac: 0 maxfr: 4858
tabn: 0 slot: 0(0x0)
```

CHANGE #5 CON_ID:1 TYP:0 CLS:20 AFN:4 DBA:0x01003327 OBJ:4294967295 SCN:0x0000.002d228c

SEQ:1 OP:5.1 ENC:0 RBL:0

ktudb redo: siz: 112 spc: 7776 flg: 0x0012 seq: 0x0187 rec: 0x04

 xid: 0x0002.013.000008d3

ktubl redo: slt: 19 rci: 0 opc: 11.1 [objn: 95424 objd: 95424 tsn: 3]

Undo type: Regular undo Begin trans Last buffer split: No

Temp Object: No

Tablespace Undo: No

 0x00000000 prev ctl uba: 0x01003327.0187.03

prev ctl max cmt scn: 0x0000.002d1e27 prev tx cmt scn: 0x0000.002d1e39

txn start scn: 0x0000.002d22c3 logon user: 114 prev brb: 16790299 prev bcl: 0 BuExt

idx: 0 flg2: 0

KDO undo record:

KTB Redo

op: 0x03 ver: 0x01

compat bit: 4 (post-11) padding: 1

op: Z

KDO Op code: DRP row dependencies Disabled

 xtype: XA flags: 0x00000000 bdba: 0x034000bb hdba: 0x034000ba

itli: 1 ispac: 0 maxfr: 4858

tabn: 0 slot: 0(0x0)

Across versions: changes in time

- How the redo record header evolved:

- **9i**
REDO RECORD - Thread:1 RBA: 0x0006e8.00000007.0018 LEN: 0x00e8 VLD: 0x01
SCN: 0x0000.006b5967 SUBSCN: 1 07/13/2014 14:44:35

- **11g**
REDO RECORD - Thread:1 RBA: 0x00000c.00000005.0050 LEN: 0x00a4 VLD: 0x01
SCN: 0x0000.000cc465 SUBSCN: 1 07/13/2014 14:38:52
(LWN RBA: 0x00000c.00000005.0050 LEN: 0003 NST: 0001 SCN: 0x0000.000cc465)

- **12c (non-CDB)**

REDO RECORD - Thread:1 RBA: 0x0008e6.00000002.0010 LEN: 0x006c VLD: 0x05 CON_UID: 0
SCN: 0x0000.00f51e47 SUBSCN: 1 07/13/2014 19:17:13
(LWN RBA: 0x0008e6.00000002.0010 LEN: 0003 NST: 0001 SCN: 0x0000.00f51e47)

- **12c (multitenant)**

REDO RECORD - Thread:1 RBA: 0x000068.00000017.0010 LEN: 0x006c VLD: 0x05 CON_UID: 3918633952
SCN: 0x0000.002d22ca SUBSCN: 1 07/08/2014 15:59:07
(LWN RBA: 0x000068.00000017.0010 LEN: 0003 NST: 0001 SCN: 0x0000.002d22c7)

REDO RECORD - Thread:1 RBA: 0x0006e8.00000007.0018 LEN: 0x00e8 VLD: 0x01
SCN: 0x0000.006b5967 SUBSCN: 1 07/13/2014 14:44:35

9i

REDO RECORD - Thread:1 RBA: 0x00000c.00000005.0050 LEN: 0x00a4 VLD: 0x01
SCN: 0x0000.000cc465 SUBSCN: 1 07/13/2014 14:38:52

11g

(LWN RBA: 0x00000c.00000005.0050 LEN: 0003 NST: 0001 SCN: 0x0000.000cc465)

REDO RECORD - Thread:1 RBA: 0x0008e6.00000002.0010 LEN: 0x006c VLD: 0x05
CON_UID: 0

12c

SCN: 0x0000.00f51e47 SUBSCN: 1 07/13/2014 19:17:13

(LWN RBA: 0x0008e6.00000002.0010 LEN: 0003 NST: 0001 SCN: 0x0000.00f51e47)

REDO RECORD - Thread:1 RBA: 0x000068.00000017.0010 LEN: 0x006c VLD: 0x05
CON_UID: 3918633952

12c pdb

SCN: 0x0000.002d22ca SUBSCN: 1 07/08/2014 15:59:07

(LWN RBA: 0x000068.00000017.0010 LEN: 0003 NST: 0001 SCN: 0x0000.002d22c7)

Across versions: ordering

- Thanks to optimizations, redo is less and less sequential
- Changes in one redo record can be in different order (some 11g, much 12c)

```
CHANGE #1 CON_ID:3 TYP:0 CLS:1 AFN:13 DBA:0x034000bb OBJ:95424 SCN:0x0000.00285536 SEQ:2 OP:11.2 ENC:0 RBL:0
CHANGE #2 CON_ID:1 TYP:0 CLS:19 AFN:4 DBA:0x01000090 OBJ:4294967295 SCN:0x0000.002d228d SEQ:1 OP:5.2 ENC:0 RBL:0
CHANGE #3 CON_ID:3 TYP:0 CLS:1 AFN:13 DBA:0x034000c3 OBJ:95425 SCN:0x0000.002d22c3 SEQ:1 OP:10.2 ENC:0 RBL:0
CHANGE #4 CON_ID:1 TYP:0 CLS:19 AFN:4 DBA:0x01000090 OBJ:4294967295 SCN:0x0000.002d22c4 SEQ:1 OP:5.4 ENC:0 RBL:0
CHANGE #5 CON_ID:1 TYP:0 CLS:20 AFN:4 DBA:0x01003327 OBJ:4294967295 SCN:0x0000.002d228c SEQ:1 OP:5.1 ENC:0 RBL:0
CHANGE #6 CON_ID:1 TYP:0 CLS:20 AFN:4 DBA:0x01003327 OBJ:4294967295 SCN:0x0000.002d22c4 SEQ:1 OP:5.1 ENC:0 RBL:0
```

- And redo records are in different order, too (private strands?) – LWN (some 10g, much more 11g)

```
REDO RECORD - Thread:1 RBA: 0x000068.00000060.0010 LEN: 0x0078 VLD: 0x06 CON_UID: 1
SCN: 0x0000.002d236c SUBSCN: 1 07/08/2014 15:59:26
```

```
REDO RECORD - Thread:1 RBA: 0x000068.0000005f.0010 LEN: 0x0078 VLD: 0x06 CON_UID: 1
SCN: 0x0000.002d236c SUBSCN: 2 07/08/2014 15:59:26
```

```
(LWN RBA: 0x000068.00000060.0010 LEN: 0002 NST: 0002 SCN: 0x0000.002d236c)
```

CHANGE #1 CON_ID:3 TYP:0 CLS:1 AFN:13 DBA:0x034000bb OBJ:95424 SCN:0x0000.00285536
SEQ:2 OP:11.2 ENC:0 RBL:0

CHANGE #2 CON_ID:1 TYP:0 CLS:19 AFN:4 DBA:0x01000090 OBJ:4294967295
SCN:0x0000.002d228d SEQ:1 OP:5.2 ENC:0 RBL:0

CHANGE #3 CON_ID:3 TYP:0 CLS:1 AFN:13 DBA:0x034000c3 OBJ:95425 SCN:0x0000.002d22c3
SEQ:1 OP:10.2 ENC:0 RBL:0

CHANGE #4 CON_ID:1 TYP:0 CLS:19 AFN:4 DBA:0x01000090 OBJ:4294967295
SCN:0x0000.002d22c4 SEQ:1 OP:5.4 ENC:0 RBL:0

CHANGE #5 CON_ID:1 TYP:0 CLS:20 AFN:4 DBA:0x01003327 OBJ:4294967295
SCN:0x0000.002d228c SEQ:1 OP:5.1 ENC:0 RBL:0

CHANGE #6 CON_ID:1 TYP:0 CLS:20 AFN:4 DBA:0x01003327 OBJ:4294967295
SCN:0x0000.002d22c4 SEQ:1 OP:5.1 ENC:0 RBL:0

REDO RECORD - Thread:1 RBA: 0x000068.00000060.0010 LEN: 0x0078 VLD: 0x06 CON_UID: 1
SCN: 0x0000.002d236c SUBSCN: 1 07/08/2014 15:59:26

REDO RECORD - Thread:1 RBA: 0x000068.0000005f.0010 LEN: 0x0078 VLD: 0x06 CON_UID: 1
SCN: 0x0000.002d236c SUBSCN: 2 07/08/2014 15:59:26

(LWN RBA: 0x000068.00000060.0010 LEN: 0002 NST: 0002 SCN: 0x0000.002d236c)

Across versions: changes in time



- How the change record header evolved:

- **9i** CHANGE #3 TYP:0 CLS: 1 AFN:4 DBA:0x0100001a SCN:0x0000.006b5965 SEQ: 2 OP:11.2

- **10g** CHANGE #3 TYP:0 CLS: 1 AFN:5 DBA:0x01400046 OBJ:51677 SCN:0x0000.000fff16 SEQ: 1 OP:11.2

- **11.1** CHANGE #3 TYP:0 CLS: 1 AFN:4 DBA:0x01005fa7 OBJ:157491 SCN:0x0000.01380043 SEQ: 2 OP:11.2 ENC:0

- **11.2** CHANGE #3 TYP:0 CLS:1 AFN:5 DBA:0x014000bf OBJ:73299 SCN:0x0000.000cc464 SEQ:1 OP:11.2 ENC:0 RBL:0

- **12c (non-CDB)**

CHANGE #3 CON_ID:0 TYP:0 CLS:1 AFN:2 DBA:0x028000bf OBJ:91537 SCN:0x0000.00f51e47 SEQ:1 OP:11.2 ENC:0 RBL:0

- **12c (multitenant)**

CHANGE #3 CON_ID:1 TYP:2 CLS:1 AFN:3 DBA:0x00c17afb OBJ:91833 SCN:0x0000.002d1f4c SEQ:1 OP:11.2 ENC:0 RBL:0

- **9i**

CHANGE #3 TYP:0 CLS: 1 AFN:4 DBA:0x0100001a SCN:0x0000.006b5965 SEQ: 2 OP:11.2

- **10g**

CHANGE #3 TYP:0 CLS: 1 AFN:5 DBA:0x01400046 OBJ:51677 SCN:0x0000.000fff16 SEQ: 1
OP:11.2

- **11.1**

CHANGE #3 TYP:0 CLS: 1 AFN:4 DBA:0x01005fa7 OBJ:157491 SCN:0x0000.01380043 SEQ:
2 OP:11.2 ENC:0

- **11.2**

CHANGE #3 TYP:0 CLS:1 AFN:5 DBA:0x014000bf OBJ:73299 SCN:0x0000.000cc464 SEQ:1
OP:11.2 ENC:0 RBL:0

- **12c (non-CDB)**

CHANGE #3 CON_ID:0 TYP:0 CLS:1 AFN:2 DBA:0x028000bf OBJ:91537 SCN:0x0000.00f51e47
SEQ:1 OP:11.2 ENC:0 RBL:0

- **12c (multitenant)**

CHANGE #3 CON_ID:1 TYP:2 CLS:1 AFN:3 DBA:0x00c17afb OBJ:91833 SCN:0x0000.002d1f4c
SEQ:1 OP:11.2 ENC:0 RBL:0

Across versions: changes in time

- How the change records evolved – 9i undo

```
CHANGE #2 TYP:0 CLS:24 AFN:2 DBA:0x008020c4 SCN:0x0000.006b5953 SEQ: 1 OP:5.1
ktudb redo: siz: 100 spc: 680 flg: 0x0012 seq: 0x04a1 rec: 0x3e
          xid: 0x0004.020.000015bd
ktubl redo: slt: 32 rci: 0 opc: 11.1 objn: 7655 objd: 7655 tsn: 5
Undo type: Regular undo          Begin trans    Last buffer split: No
Temp Object: No
Tablespace Undo: No
          0x00000000 prev ctl uba: 0x008020c4.04a1.3d
prev ctl max cmt scn: 0x0000.006b013c prev tx cmt scn: 0x0000.006b0144
KDO undo record:
KTB Redo
op: 0x03 ver: 0x01
op: Z
KDO Op code: DRP row dependencies Disabled
  xtype: XA bdba: 0x0100001a hdba: 0x01000019
itli: 1 ispac: 0 maxfr: 4863
tabn: 0 slot: 0(0x0)
```

Across versions: changes in time



- How the change records evolved – 12c multitenant, same undo operation

```
CHANGE #2 CON_ID:1 TYP:0 CLS:18 AFN:4 DBA:0x010008eb OBJ:4294967295 SCN:0x0000.002d227a SEQ:1 OP:5.1 ENC:0 RBL:0
```

```
ktudb redo: siz: 112 spc: 4090 flg: 0x0012 seq: 0x022a rec: 0x23
```

```
      xid: 0x0001.010.0000075b
```

```
ktubl redo: slt: 16 rci: 0 opc: 11.1 [objn: 91833 objd: 91833 tsn: 1]
```

```
Undo type: Regular undo      Begin trans      Last buffer split: No
```

```
Temp Object: No
```

```
Tablespace Undo: No
```

```
      0x00000000 prev ctl uba: 0x010008eb.022a.22
```

```
prev ctl max cmt scn: 0x0000.002d1df7 prev tx cmt scn: 0x0000.002d1e0b
```

```
txn start scn: 0xffff.ffffffff logon user: 0 prev brb: 16779495 prev bcl: 0 BuExt idx: 0 flg2: 0
```

```
KDO undo record:
```

```
KTB Redo
```

```
op: 0x03 ver: 0x01
```

```
compat bit: 4 (post-11) padding: 1
```

```
op: Z
```

```
KDO Op code: DRP row dependencies Disabled
```

```
      xtype: XA flags: 0x00000000 bdba: 0x00c17afb hdba: 0x00c01e12
```

```
itli: 2 ispac: 0 maxfr: 4858
```

```
tabn: 0 slot: 1 (0x1)
```

CHANGE #2 TYP:0 CLS:24 AFN:2 DBA:0x008020c4 SCN:0x0000.006b5953 SEQ: 1 OP:5.1

ktudb redo: siz: 100 spc: 680 flg: 0x0012 seq: 0x04a1 rec: 0x3e

xid: 0x0004.020.000015bd

9i

ktubl redo: slt: 32 rci: 0 opc: 11.1 objn: 7655 objd: 7655 tsn: 5

Undo type: Regular undo Begin trans Last buffer split: No

Temp Object: No

Tablespace Undo: No

0x00000000 prev ctl uba: 0x008020c4.04a1.3d

prev ctl max cmt scn: 0x0000.006b013c prev tx cmt scn: 0x0000.006b0144

KDO undo record:

KTB Redo

op: 0x03 ver: 0x01

op: Z

KDO Op code: DRP row dependencies Disabled

xtype: XA bdba: 0x0100001a hdba: 0x01000019

itli: 1 ispac: 0 maxfr: 4863

tabn: 0 slot: 0(0x0)

CHANGE #2 CON_ID:1 TYP:0 CLS:18 AFN:4 DBA:0x010008eb OBJ:4294967295 SCN:0x0000.002d227a
SEQ:1 OP:5.1 ENC:0 RBL:0

12c

ktudb redo: siz: 112 spc: 4090 flg: 0x0012 seq: 0x022a rec: 0x23
xid: 0x0001.010.0000075b

ktubl redo: slt: 16 rci: 0 opc: 11.1 [objn: 91833 objd: 91833 tsn: 1]

Undo type: Regular undo Begin trans Last buffer split: No

Temp Object: No

Tablespace Undo: No

0x00000000 prev ctl uba: 0x010008eb.022a.22

prev ctl max cmt scn: 0x0000.002d1df7 prev tx cmt scn: 0x0000.002d1e0b

txn start scn: 0xffff.ffffffff logon user: 0 prev brb: 16779495 prev bcl: 0 BuExt idx:
0 flg2: 0

KDO undo record:

KTB Redo

op: 0x03 ver: 0x01

compat bit: 4 (post-11) padding: 1

op: Z

KDO Op code: DRP row dependencies Disabled

xtype: XA flags: 0x00000000 bdba: 0x00c17afb hdba: 0x00c01e12

itli: 2 ispac: 0 maxfr: 4858

tabn: 0 slot: 1(0x1)

Across versions: since 9i

- DDL in redo
- Useless for recovery, useful for LogMiner, Streams etc.
- No detail in dump

```
REDO RECORD - Thread:1 RBA: 0x00099a.0000002d.00e8 LEN: 0x0228 VLD: 0x01  
SCN: 0x0000.00fefec0 SUBSCN: 1 07/19/2014 14:16:25  
CHANGE #1 MEDIA RECOVERY MARKER SCN:0x0000.00000000 SEQ:0 OP:24.1 ENC:0
```

- Even recursive DDL is there
- Later versions added other, non-interesting 24.1s

- **But it *is* there**

```
REDO RECORD - Thread:1 RBA: 0x00099a.0000002d.00e8 LEN: 0x0228 VLD: 0x01
SCN: 0x0000.00fefec0 SUBSCN: 1 07/19/2014 14:16:25
CHANGE #1 MEDIA RECOVERY MARKER SCN:0x0000.00000000 SEQ:0 OP:24.1 ENC:0
#XID: 0x0005.003.00002236
#part 1 of 1
#logon schema = SYS
#user schema = SYS
#object_id = 117610
#SQL text = CREATE TABLE hr.non_comptab
AS
SELECT * FROM all_objects
WHERE SUBSTR(object_name,1,1) BETWEEN 'A' AND 'WZZZZZ' and rownum<=1000
#object_owner = HR
#object_name = NON_COMPTAB
#base_object_id = 117610
```

Compression

- First, a normal heap table CTAS, processed as direct insert:

```
CHANGE #2 CON_ID:0 TYP:1 CLS:1 AFN:6 DBA:0x01845d4b OBJ:117610 SCN:0x0000.00fefec7 SEQ:1 OP:19.1 ENC:0 RBL:0
Direct Loader block redo entry
seg/obj: 0x1cb6a csc: 0x00.fefec7 itc: 3 flg: E typ: 1 - DATA
 data_block_dump,data header at 0x7ff9a25aa090
=====
flag=-----
0xe:pti[0] nrow=76 offs=0
0x12:pri[0] offs=0x1f29
...
0xa8:pri[75] offs=0x404
block_row_dump:
tab 0, row 0, @0x1f29
tl: 87 fb: --H-FL-- lb: 0x0 cc: 18
col 0: [ 3] 53 59 53
col 1: [ 6] 49 5f 43 4f 4c 32
...
col 17: [ 1] 59
```

CHANGE #2 CON_ID:0 TYP:1 CLS:1 AFN:6 DBA:0x01845d4b OBJ:117610 SCN:0x0000.00fefecc
SEQ:1 OP:19.1 ENC:0 RBL:0

Direct Loader block redo entry

seg/obj: 0x1cb6a csc: 0x00.fefec7 itc: 3 flg: E typ: 1 - DATA

data_block_dump,data header at 0x7ff9a25aa090

=====

flag=-----

0xe:pti[0] nrow=76 offs=0

0x12:pri[0] offs=0x1f29

...

0xa8:pri[75] offs=0x404

block_row_dump:

tab 0, row 0, @0x1f29

t1: 87 fb: --H-FL-- lb: 0x0 cc: 18

col 0: [3] 53 59 53

col 1: [6] 49 5f 43 4f 4c 32

...

col 17: [1] 59

Compression

- OLTP compression:

```
CHANGE #2 CON_ID:0 TYP:1 CLS:1 AFN:6 DBA:0x01845d63 OBJ:117611 SCN:0x0000.00fefef2 SEQ:1 OP:19.1 ENC:0  
RBL:0
```

```
Direct Loader block redo entry
```

```
seg/obj: 0x1cb6b csc: 0x00.fefef2 itc: 3 flg: E typ: 1 - DATA
```

```
bdba: 0x610f0000
```

```
data_block_dump,data header at 0x7ff9a25aa090
```

```
=====
```

```
76543210
```

```
flag--0-----
```

```
ntab=2
```

```
nrow=356
```

```
    r0_9ir2=0x0
```

```
    mec_kdbh9ir2=0x20
```

```
                76543210
```

```
    shcf_kdbh9ir2=-----
```

```
                76543210
```

(see following slides)

CHANGE #2 CON_ID:0 TYP:1 CLS:1 AFN:6 DBA:0x01845d63 OBJ:117611 SCN:0x0000.00fefef2
SEQ:1 OP:19.1 ENC:0 RBL:0

Direct Loader block redo entry

seg/obj: 0x1cb6b csc: 0x00.fefef2 itc: 3 flg: E typ: 1 - DATA

bdba: 0x610f0000

data_block_dump,data header at 0x7ff9a25aa090

=====

76543210

flag=-0-----

ntab=2

nrow=356

r0_9ir2=0x0

mec_kdbh9ir2=0x20

76543210

shcf_kdbh9ir2=-----

76543210

flag_9ir2=--R--OC Archive compression: N

fcls_9ir2[0]={ }

perm_9ir2[18]={ 7 15 0 16 17 13 11 14 12 1 4 5 2 10 3 8 9 6 }

```
0x28:pti[0]      nrow=70   ofs=0
0x2c:pti[1]      nrow=286  ofs=70
0x30:pri[0]      ofs=0x1d6d
```

```
...
block_row_dump:
```

```
tab 0, row 0, @0x1d6d
```

```
tl: 8 fb: --H-FL-- lb: 0x0  cc: 15
```

```
col 0: *NULL*
```

```
...
```

```
col 14: [ 7] 78 71 05 18 0c 34 1d
```

```
bindmp: 00 23 0f 0c 28 3b 29 28
```

```
...
```

```
tab 0, row 39, @0x1e23
```

```
tl: 10 fb: --H-FL-- lb: 0x0  cc: 1
```

```
col 0: [ 7] 78 71 05 18 0c 34 1e
```

```
bindmp: 00 0a cf 78 71 05 18 0c 34 1e
```

```
...
```

```
tab 1, row 0, @0x1ce1
```

```
tl: 17 fb: --H-FL-- lb: 0x0  cc: 18
```

```
col 0: *NULL*
```

```
...
```

```
col 17: [ 2] c1 32
```

```
bindmp: 2c 00 04 01 ce 49 5f 43 4f 4c 32 ca c1 32 ca c1 32
```


Compression

- HCC compression*:

```
CHANGE #2 CON_ID:0 TYP:1 CLS:1 AFN:6 DBA:0x01802743 OBJ:117643 SCN:0x0000.00ff5bff SEQ:1 OP:19.1 ENC:0  
RBL:0
```

```
Direct Loader block redo entry
```

```
seg/obj: 0x1cb8b csc: 0x00.ff5bf7 itc: 3 flg: E typ: 1 - DATA
```

```
bdba: 0x00000000
```

```
data_block_dump,data header at 0x7f90bbd59090
```

```
=====
```

```
76543210
```

```
flag--0-----
```

```
ntab=1
```

```
nrow=1
```

```
    r0_9ir2=0x0
```

```
    mec_kdbh9ir2=0x0
```

```
                76543210
```

```
    shcf_kdbh9ir2=-----
```

```
                76543210
```

(see following slides)

CHANGE #2 CON_ID:0 TYP:1 CLS:1 AFN:6 DBA:0x01802743 OBJ:117643 SCN:0x0000.00ff5bff
SEQ:1 OP:19.1 ENC:0 RBL:0

Direct Loader block redo entry

seg/obj: 0x1cb8b csc: 0x00.ff5bf7 itc: 3 flg: E typ: 1 - DATA

bdba: 0x00000000

data_block_dump,data header at 0x7f90bbd59090

=====

76543210

flag=-0-----

ntab=1

nrow=1

r0_9ir2=0x0

mec_kdbh9ir2=0x0

76543210

shcf_kdbh9ir2=-----

76543210

flag_9ir2=---R----- Archive compression: Y

fcls_9ir2[0]={ }

0x16:pti[0] nrow=1 offs=0

0x1a:pri[0] offs=0x16e6

0x16:pti[0] nrow=1 offs=0

0x1a:pri[0] offs=0x16e6

block_row_dump:

tab 0, row 0, @0x16e6

tl: 2202 fb: --H-FL-- lb: 0x0 cc: 1

col 0: [2196]

Compression level: 04 (Archive High)

CU header:

CU total length: 2184

CU flags: NC-U-CRD-OP

ncols: 18

nrows: 11900

CU decomp length: 602 len/value length: 999600

row pieces per row: 1

num deleted rows: 0

START_CU:

00 00 08 94 24 00 00 00 4b 44 5a 30 84 61 09 95 00 00 08 88 eb 06 00 12 2e ...

END_CU

bindmp: 2c 00 01 fe 94 08 00 00 08 94 24 00 00 00 4b 44 5a 30 84 61 09 95 00 ...

end_of_block_dump

Across versions: LOBs

- Basic LOB example
- Data in row (LOB locator and/or inline LOBs)

```
CHANGE #2 CON_ID: 0 TYP:0 CLS:1 AFN:6 DBA:0x01823dee OBJ:117598 SCN:0x0000.00fef392 SEQ:3 OP:11.2 ENC:0 RBL:0
KTB Redo
op: 0x02 ver: 0x01
compat bit: 4 (post-11) padding: 1
op: C uba: 0x01000a3f.09b1.04
KDO Op code: IRP row dependencies Disabled
itli: 1 ispac: 0 maxfr: 4858
tabn: 0 slot: 1(0x1) size/delt: 51
fb: --H-FL-- lb: 0x1 cc: 3
null: ---
col 0: [ 2] c1 03
col 1: [ 7] 41 41 41 41 41 41 41
col 2: [36]
00 54 00 01 02 0c 80 00 00 02 00 00 00 01 00 00 00 44 82 e0 00 10 09 00 00
00 00 00 00 00 00 00 00 00 00 00
```

CHANGE #2 CON_ID: 0 TYP:0 CLS:1 AFN:6 DBA:0x01823dee OBJ:117598 SCN:0x0000.00fef392
SEQ:3 OP:11.2 ENC:0 RBL:0

KTB Redo

op: 0x02 ver: 0x01

compat bit: 4 (post-11) padding: 1

op: C uba: 0x01000a3f.09b1.04

KDO Op code: IRP row dependencies Disabled

itli: 1 ispac: 0 maxfr: 4858

tabn: 0 slot: 1(0x1) size/delt: 51

fb: --H-FL-- lb: 0x1 cc: 3

null: ---

col 0: [2] c1 03

col 1: [7] 41 41 41 41 41 41 41

col 2: [36]

00 54 00 01 02 0c 80 00 00 02 00 00 00 01 00 00 00 44 82 e0 00 14 05 00 00
00 00 00 1f 40 00 00 00 00 00 02 01 82 3d f6

Or inline:

col 2: [42]

00 54 00 01 02 0c 80 00 00 02 00 00 00 01 00 00 00 44 82 df 00 16 09 00 00
00 00 00 00 06 00 00 00 00 00 01 00 58 00 58 00 58

Across versions: LOBs

- Basic LOB example
- Undocumented auxiliary info

```
CHANGE #1 MEDIA RECOVERY MARKER SCN:0x0000.00000000 SEQ:0 OP:11.17 ENC:0
#LOB:pre-11g basicfile LOB
#KDO Op code: LLB row dependencies Disabled
#LOGMINER DATA:
# LOB RCI MARKER:
# xid: 0x0003.00b.00002468
# (OBJN, OBJV, FLAGS, FLAGS2)=(117598, 1, -1, -1)
# LOB SEGCOL NUMBERS:
# col 0: segcol#: 3
# Non-lob Column Info:
#SUPLOG@col 1: [ 2] c1 03
#SUPLOG@col 2: [ 7] 41 41 41 41 41 41 41
```

CHANGE #1 MEDIA RECOVERY MARKER SCN:0x0000.00000000 SEQ:0 OP:11.17 ENC:0

#LOB:pre-11g basicfile LOB

#KDO Op code: LLB row dependencies Disabled

#LOGMINER DATA:

LOB RCI MARKER:

xid: 0x0003.00b.00002468

(OBJN, OBJV, FLAGS, FLAGS2)=(117598, 1, -1, -1)

LOB SEGCOL NUMBERS:

col 0: segcol#: 3

Non-lob Column Info:

#SUPLOG@col 1: [2] c1 03

#SUPLOG@col 2: [7] 41 41 41 41 41 41 41

Across versions: LOBs

- Basic LOB example
- Undocumented auxiliary info

```
CHANGE #1 MEDIA RECOVERY MARKER SCN:0x0000.00000000 SEQ:0 OP:11.17 ENC:0
#LOB:11g LOB, op = 102
#KDO Op code: LLB row dependencies Disabled
# op: 102
# xid: 0x0003.00b.00002468
# lid: 000000010000004482e0
# column: 3
# lobsize?: 4000
# psize?: 8132
# offset?: 1
# objn: 117598
# objv: 1
```


CHANGE #1 MEDIA RECOVERY MARKER SCN:0x0000.00000000 SEQ:0 OP:11.17 ENC:0

#LOB:11g LOB, op = 102

#KDO Op code: LLB row dependencies Disabled

op: 102

xid: 0x0003.00b.00002468

lid: 000000010000004482e0

column: 3

lobsize?: 4000

psize?: 8132

offset?: 1

objn: 117598

objv: 1

Across versions: LOBs

- Basic LOB example
- Actual data

```
CHANGE #1 CON_ID: 0 TYP:1 CLS:1 AFN:6 DBA:0x01823df6 OBJ:117599 SCN:0x0000.00fef394 SEQ:1 OP:19.1 ENC:0  
RBL:0
```

```
Direct Loader block redo entry
```

```
Long field block dump:
```

```
Object Id 117599
```

```
LobId: 0000000100000004482E0 PageNo 0
```

```
Version: 0x0000.00000001 pdba: 25312752
```

```
00 2a 00 2a 00 2a 00 2a 00 2a 00 2a 00 2a 00 2a 00 2a 00 2a 00 2a
```

```
...
```

```
00 20 00 20 00 20 00 20 00 20 00 20 00 20 00 20 00 20 00 20 00 20
```

- (note padding by space (0x20), that's not in the table data)

Across versions: LOBs

- SecureFiles LOB example (11g)
- Very similar data in row, similar 11.7 (but different)
- A lot of chunk management
- Different operation for actual data

```
CHANGE #1 CON_ID: 0 TYP:10 CLS:1 AFN:6 DBA:0x01845b8b OBJ:117602 SCN:0x0000.00fef421 SEQ:1 OP:26.6 ENC:0 RBL:0
KDLI common [12]
type 0x20 [data]
  psiz 8060
  dba 0x01845b8b
KDLI fpload [11.32]
  scn 0x0000.00fef421
  xid 0x000c.01b.000012a1
  objd 117602
KDLI load data [4.56]
bdba [0x01845b8b]
...
```

CHANGE #1 CON_ID: 0 TYP:10 CLS:1 AFN:6 DBA:0x01845b8b OBJ:117602 SCN:0x0000.00fef421

SEQ:1 OP:26.6 ENC:0 RBL:0

KDLI common [12]

type 0x20 [data]

psiz 8060

dba 0x01845b8b

KDLI fpload [11.32]

scn 0x0000.00fef421

xid 0x000c.01b.000012a1

objd 117602

KDLI load data [4.56]

bdba [0x01845b8b]

KDLI data load [0x0.8000]

00 2a 00 2a 00 2a 00 2a 00 2a 00 2a 00 2a 00 2a 00 2a 00 2a 00 2a 00 2a

...

00 2a 00 2a 00 2a 00 2a 00 2a 00 2a 00 2a 00 2a

KDLI suplog [9.24]

xid 0x0012.01b.000012a1

objn 117601

col# 3

New in 12c: extended datatype

- It's really just LOB
- Inline or out-of-line, as length requests
- New OCI calls (OCIBindByName2, OCIBindByPos2)
- And you can index them (if length is small enough to fit into an index block)
- But internally, it's just a LOB

New in 12c: column data



```
CHANGE #2 CON_ID: 0 TYP:0 CLS:1 AFN:6 DBA:0x0189eadd OBJ:119473 SCN:0x0000.01080d76 SEQ:1 OP:11.2 ENC:0 RBL:0
KTb Redo
op: C uba: 0x010033a5.09d0.10
KDO Op code: IRP row dependencies Disabled
itli: 1 ispac: 0 maxfr: 4858
tabn: 0 slot: 2(0x2) size/delt: 219
fb: --H-FL-- lb: 0x1 cc: 6
null: -----
col 0: [ 2] c1 04
col 1: [56]
 00 54 00 01 01 0c 00 80 00 01 00 00 00 01 00 00 00 49 1f 25 00 24 48 90 00
 1e 00 00 1a 01 00 6e 00 6d 00 6d 00 6d 00 6d 00 6d 00 6d 00 6d 00 6d 00 6d
 00 6d 00 6d 00 6d
col 2: [44]
 00 54 00 01 01 0c 00 80 00 01 00 00 00 01 00 00 00 49 1f 7e 00 18 40 90 00
 12 21 00 7f fe 01 01 01 01 89 ed ee 02 01 01 89 ee b7 03
col 3: [44]
 00 54 00 01 01 0c 00 80 00 01 00 00 00 01 00 00 00 49 1f 7f 00 18 40 90 00
 12 21 00 7f ff 01 01 01 01 89 ed f6 02 01 01 89 ef 37 03
```

CHANGE #2 CON_ID: 0 TYP:0 CLS:1 AFN:6 DBA:0x0189eadd OBJ:119473 SCN:0x0000.01080d76
SEQ:1 OP:11.2 ENC:0 RBL:0

KTB Redo

op: C uba: 0x010033a5.09d0.10

KDO Op code: IRP row dependencies Disabled

itli: 1 ispac: 0 maxfr: 4858

tabn: 0 slot: 2(0x2) size/delt: 219

fb: --H-FL-- lb: 0x1 cc: 6

null: -----

col 0: [2] c1 04

col 1: [56]

00 54 00 01 01 0c 00 80 00 01 00 00 00 01 00 00 00 49 1f 25 00 24 48 90 00
1e 00 00 1a 01 00 6e 00 6d 00 6d 00 6d 00 6d 00 6d 00 6d 00 6d 00 6d
00 6d 00 6d 00 6d

col 2: [44]

00 54 00 01 01 0c 00 80 00 01 00 00 00 01 00 00 00 49 1f 7e 00 18 40 90 00
12 21 00 7f fe 01 01 01 01 89 ed ee 02 01 01 89 ee b7 03

col 3: [44]

00 54 00 01 01 0c 00 80 00 01 00 00 00 01 00 00 00 49 1f 7f 00 18 40 90 00
12 21 00 7f ff 01 01 01 01 89 ed f6 02 01 01 89 ef 37 03

New in 12c: index on extended

- Really nothing special

```
CHANGE #3 CON_ID: 0 TYP:0 CLS:1 AFN:6 DBA:0x0189eae7 OBJ:119484 SCN:0x0000.01080d78 SEQ:1 OP:10.8 ENC:0  
RBL:0
```

```
index redo (kdxlne): (count=12) init header of newly allocated leaf block
```

```
...
```

```
CHANGE #2 CON_ID: 0 TYP:0 CLS:1 AFN:6 DBA:0x0189eae3 OBJ:119484 SCN:0x0000.01080d92 SEQ:1 OP:10.15 ENC:0  
RBL:0
```

```
index redo (kdxbin) : insert branch block row, count=3
```

```
KTB Redo
```

```
...
```

```
CHANGE #2 CON_ID: 0 TYP:0 CLS:1 AFN:6 DBA:0x0189eae6 OBJ:119484 SCN:0x0000.01080d91 SEQ:2 OP:10.8 ENC:0  
RBL:0
```

```
index redo (kdxlne): (count=12) init leaf block being split
```

```
...
```

```
CHANGE #3 CON_ID: 0 TYP:0 CLS:1 AFN:6 DBA:0x0189eae7 OBJ:119484 SCN:0x0000.01080d92 SEQ:1 OP:10.5 ENC:0  
RBL:0
```

```
index redo (kdxlre): restore leaf row
```


CHANGE #3 CON_ID: 0 TYP:0 CLS:1 AFN:6 DBA:0x0189eae7 OBJ:119484 SCN:0x0000.01080d78
SEQ:1 OP:10.8 ENC:0 RBL:0

index redo (kdxlne): (count=12) init header of newly allocated leaf block

...

CHANGE #2 CON_ID: 0 TYP:0 CLS:1 AFN:6 DBA:0x0189eae3 OBJ:119484 SCN:0x0000.01080d92
SEQ:1 OP:10.15 ENC:0 RBL:0

index redo (kdxbin) : insert branch block row, count=3

KTB Redo

...

CHANGE #2 CON_ID: 0 TYP:0 CLS:1 AFN:6 DBA:0x0189eae6 OBJ:119484 SCN:0x0000.01080d91
SEQ:2 OP:10.8 ENC:0 RBL:0

index redo (kdxlne): (count=12) init leaf block being split

...

CHANGE #3 CON_ID: 0 TYP:0 CLS:1 AFN:6 DBA:0x0189eae7 OBJ:119484 SCN:0x0000.01080d92
SEQ:1 OP:10.5 ENC:0 RBL:0

index redo (kdxlre): restore leaf row

Across versions: new features

- Oracle added locally managed tablespaces and ASSM:
- Instead of DML on SYS tables (uet\$, fet\$), we have layer 13 operations:

```
CHANGE #1 CON_ID:3 TYP:0 CLS:8 AFN:12 DBA:0x01014e10 OBJ:91563 SCN:0x0000.002d1daf SEQ:1 OP:13.22 ENC:0 RBL:0  
Redo on Level1 Bitmap Block  
Redo for state change  
Len: 1 Offset: 12 newstate: 1
```

- Can be ignored if you are interested only in actual data

CHANGE #1 CON_ID:3 TYP:0 CLS:8 AFN:12 DBA:0x01014e10 OBJ:91563 SCN:0x0000.002d1daf
SEQ:1 OP:13.22 ENC:0 RBL:0

Redo on Level1 Bitmap Block

Redo for state change

Len: 1 Offset: 12 newstate: 1

Across versions: new features

- And sometimes you can only guess
- (MOS: something with DBFS / SecureFile Lobs?)

```
CHANGE #1 CON_ID:1 TYP:0 CLS:10 AFN:3 DBA:0x00c17e87 OBJ:99142 SCN:0x0000.002d0ca6  
SEQ:1 OP:13.53 ENC:0 RBL:0
```

```
KTSL - PUA redo record:
```

```
1: Sync of 1 chunks
```

```
Chunk dba: c16f9d : Chunk length : 3 : Allocator xid: 0x0004.015.000008a5 : Allocation  
scn: 0.2958179 : Mark flag: 2
```

CHANGE #1 CON_ID:1 TYP:0 CLS:10 AFN:3 DBA:0x00c17e87 OBJ:99142 SCN:0x0000.002d0ca6
SEQ:1 OP:13.53 ENC:0 RBL:0

KTSL - PUA redo record:

1: Sync of 1 chunks

Chunk dba: c16f9d : Chunk length : 3 : Allocator xid: 0x0004.015.000008a5 :

Allocation scn: 0.2958179 : Mark flag: 2

QA



THE SMART ALTERNATIVE

Twitter: @dbvisit

Blog: blog.dbvisit.com

Forum: www.dbvisit.com/forums

Example script used in this session



```
delete hr.jobs where job_id='TE_TEST';
alter system switch logfile;

insert into hr.jobs values ('TE_TEST', 'Test job', 1, 100);
commit;

column member new_value fname
select member from v$logfile join v$log using (group#) where
v$log.status='CURRENT' and rownum =1;
alter system dump logfile '&fname';
oradebug setmypid
oradebug tracefile_name
```