
CERN

Lost Writes, a DBA's Nightmare?

Luca Canali – CERN

Marcin Blaszczyk - CERN

Outline

- CERN & Oracle Databases
- Lost write overview
- Lost write in real life
- Testing Lost writes
- Lessons Learned

Outline

- CERN & Oracle Databases
- Lost write overview
- Lost write in real life
- Testing Lost writes
- Lessons Learned

CERN & LHC Experiments

- European Organization for Nuclear Research
- 20 Member States, 7 Observer States + UNESCO and UE
- 60 Non-member States collaborate with CERN
- 2400 staff members work at CERN as personnel
- 10 000 more researchers from institutes world-wide

Large Hadron Collider (LHC) – particle accelerator used to collide beams at very high energy

- 27 km long circular tunnel
- Located ~100m underground
- Protons currently travel at 99.9999972% of the speed of light

Collisions are analyzed with usage of special detectors and software in the experiments dedicated to LHC

Higgs particle discovered in 2012!

CERN congratulates François Englert and Peter W. Higgs on 2013 Nobel prize in physics

CERN's Databases

- ~100 Oracle databases, most of them RAC
  - Mostly NAS storage plus some SAN with ASM
  - ~300 TB of data files for production DBs in total
- Examples of critical production DBs:
  - LHC logging database ~170 TB, expected growth up to ~70 TB / year
  - 13 Production experiments' databases ~120 TB in total
- Relational DBs play a key role in the LHC production chains
  - Accelerator logging and monitoring systems
  - Online acquisition, offline: data (re)processing, data distribution, analysis
  - Grid infrastructure and operation services
  - Data management services
  - Metadata and transaction processing for tape storage system

Outline

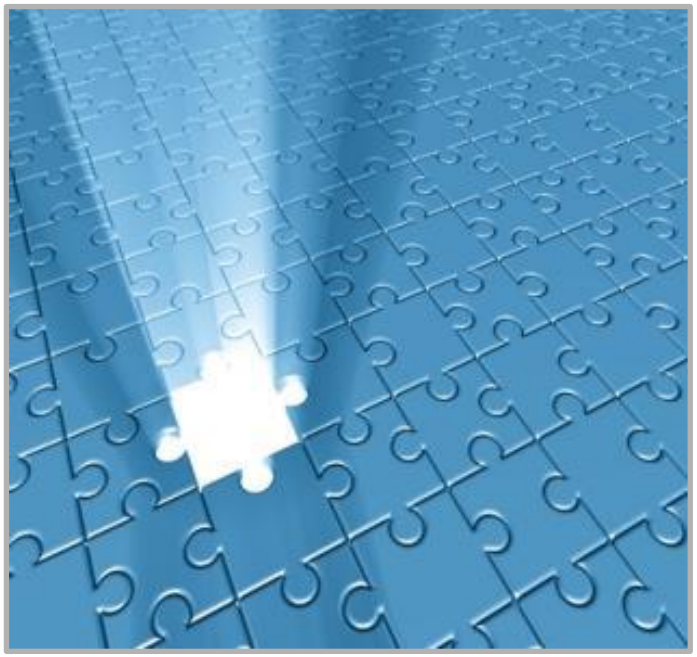
- CERN & Oracle Databases
- Lost write overview
- Lost write in real life
- Testing Lost writes
- Lessons Learned

Lost Write Overview

- What is a lost write?

"A data block lost write occurs when an I/O subsystem acknowledges the completion of the block write, while in fact the write did not occur in the persistent storage"

(support note 1302539.1)



Lost Write Causes

- What can cause lost writes?
  - Just about anything:
    - Faulty disks and disk controllers
    - Faulty memory
    - Faulty network components
    - Firmware, operating system, volume manager, NFS or third party software defects
    - Oracle Database software defects

Lost Write Symptoms

- First thought - Corrupt block
  - Logical corruption
  - Physical corruption

- But also…
  - Data Block can be physically and/or logically correct
  - Old version of the block means staleness

Outline

- CERN and Oracle
- Lost write overview
- Lost write in real life
- Testing Lost writes
- Lessons Learned

Just Another ORA 600?

- Critical production with two Active Data Guards:
  - One for disaster recovery
  - One for load balance read only load

- Both standbys stop at the same time with:
  - ORA-00600 [3020]
  - Recovery interrupted!

```
ORA-00600: internal error code, arguments: [3020], [283], [14756924], [14756924], [], [], [], [], [], [], [], []
ORA-10567: Redo is inconsistent with data block (file# 283, block# 14756924, file offset is 629637120 bytes)
ORA-10564: tablespace XXX
ORA-01110: data file 283: 'XXX'
ORA-10561: block type 'TRANSACTION MANAGED INDEX BLOCK', data object# 9538525
(…)
MRP0: Background Media Recovery terminated with error 448
(…)
Recovery interrupted!
```

“Houston, we have a problem”

- We know that there is corruption somewhere!
  - Is it the prod DB?
  - Is it the standby DB?
  - Are the redo logs?

- How to restore ADG service?
  - ADG services are stale
  - Failover to standby would mean data loss!

- Investigating the problem
  - How to attack the issue?

DBAs at Work

- Support note 1265884.1 *"Resolving ORA-752 or ORA-600 [3020] During Standby Recovery"*
- Service request severity one
- Our own investigations (block dumps, log mining, etc.)
  - Trying to find differences between primary and standby
  - When did the problem enter the system?
  - Are there more undiscovered lost writes?
  - Is this an Oracle bug, a storage issue?
- While we work on it… time is ticking
  - ADG services are stale (MRP recovery is stuck)

Restoring the services

- We have discovered that the corrupt block was on an index block on Primary

- On Primary
  - Easy then, rebuild online

```
SQL> ALTER INDEX <index_name> REBUILD ONLINE;
```

- On the ADG
  - allow corruption to go through:

```
SQL> RECOVER AUTOMATIC STANDBY DATABASE ALLOW 1 CORRUPTION;
```

  - Affected blocks are overwritten by MRP redo apply, i.e. the index structure is now “rebuilt online”

Debriefing

- Do we understand the root cause?
  - Oracle bug or storage issue?
- Suspect: storage configuration change
  - Change was reverted
  - Lost write didn’t reappear
- What can we do to protect the DB
  - What about Oracle recommendations?

From Oracle Best Practices

- Set at Primary Database:
  - DB_BLOCK_CHECKSUM=FULL
  - DB_BLOCK_CHECKING=FULL or MEDIUM
  - DB_LOST_WRITE_PROTECT=TYPICAL

- Use Data Guard and set:
  - DB_BLOCK_CHECKSUM=FULL
  - DB_BLOCK_CHECKING=FULL or MEDIUM
  - DB_LOST_WRITE_PROTECT=TYPICAL

- Note: There is overhead when using these parameters

DB_LOST_WRITE_PROTECT

- 11g new feature
  - Recommended value: ‘TYPICAL’, default is ‘NONE’
- On primary:
  - Physical read to buffer cache causes the generation of a redo entry called block read redo (BRR)
- On Standby:
  - Recovery checks blocks referred in BRR for lost writes
- This is actually a reactive detection mechanism

Additional Effects

- Setting db_lost_write_protect='typical' also means

- Lost write error:
  - instead of ORA-600 [3020]
  - ORA-00752: recovery detected a lost write of a data block
  - In general: additional useful information in systems' logs

- Yet another source of redo
  - Redo log entries for commit-time block cleanout
  - "_log_committime_block_cleanout" = TRUE (needs instance restart)
  - This is desirable in a Data Guard environment

Outline

- CERN & Oracle Databases
- Lost write overview
- Lost write in real life
- Testing Lost writes
- Lessons Learned

Why Testing Lost Writes?

- Validate as a failure scenario, training
  - Lost write can cause quite complex recovery
  - Investigating is difficult and time consuming

- What to test?
  - Artificially introduce a lost write
  - Observe how the system reacts
  - Practice analysis of the issue
  - Practice solutions and workarounds

Techniques on Reproducing LW

- Read/Write a single block from Oracle data files:
  - Filesystem: example for block number 132
    - Read

```
dd if=testlostwritetbs.dbf bs=8192 count=1
skip=132 of=blk132.dmp
```

    - Write

```
dd of=testlostwritetbs.dbf bs=8192 count=1
seek=132 if=blk132.dmp conv=notrunc
```

Techniques on Reproducing LW

- Read/Write a single block from Oracle data files:
  - ASM: example for block number 132
    - Read

```
./asmblk_edit -r -s 132
-a +TEST_DATADG1/test/datafile/testlostwritetbs.4138.831900273
-f blk132.dmp
```

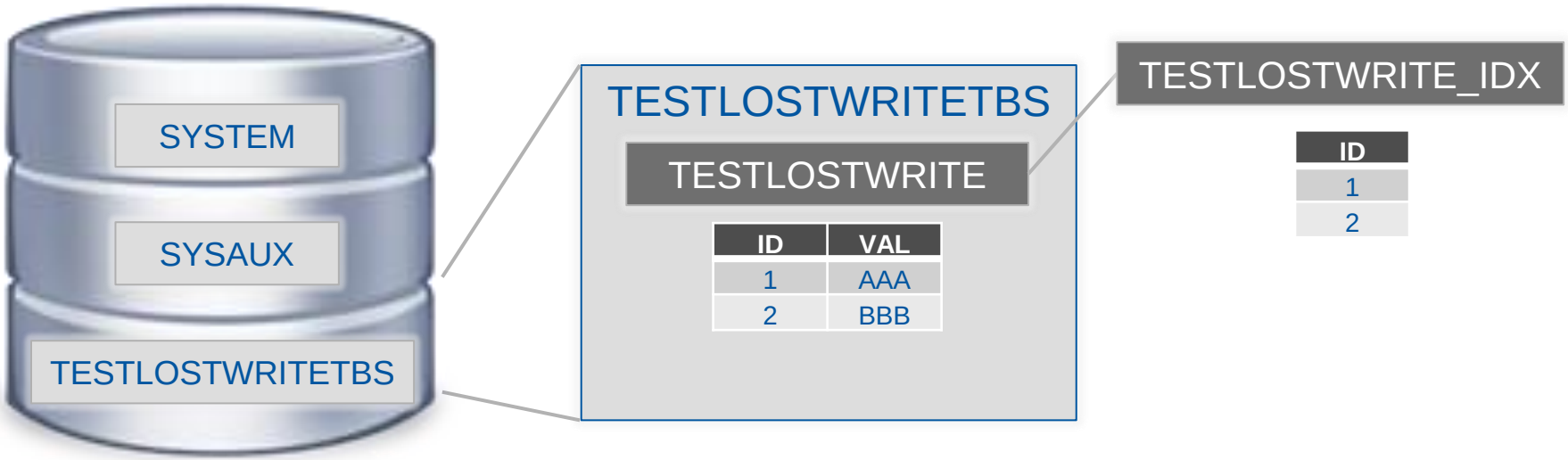
    - Write

```
./asmblk_edit -w -s 132
-a +TEST_DATADG1/test/datafile/testlostwritetbs.4138.831900273
-f blk132.dmp
```

    - Note: asmblk_edit is based on DBMS_DISKGROUP
    - Download: http://cern.ch/canali/resources.htm

How to Generate a Lost Write

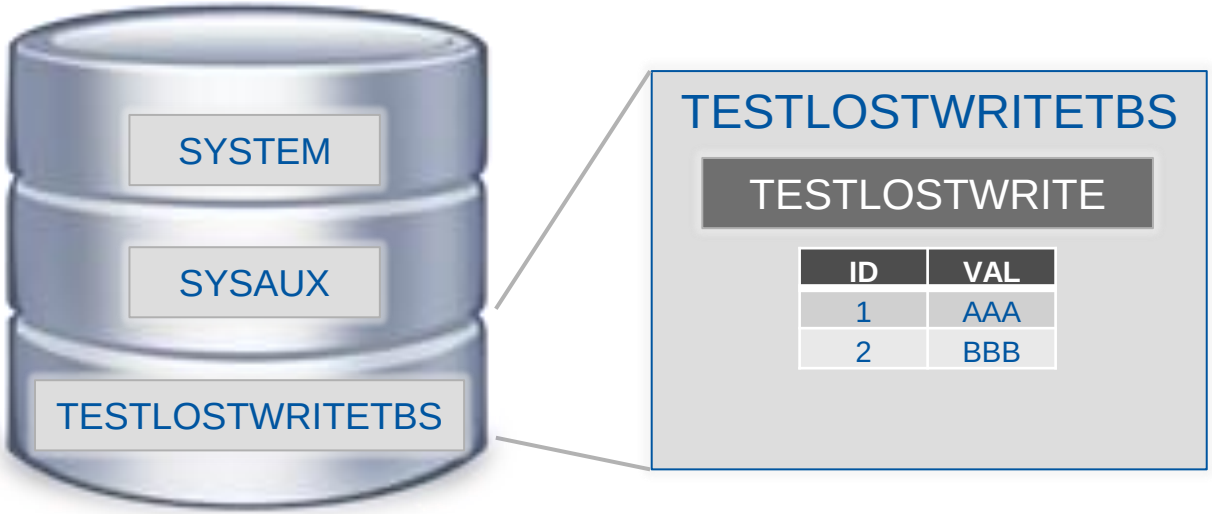
1. Create tablespace, table with few rows & index




```
SQL> CREATE TABLESPACE testlostwritetbs DATAFILE SIZE 100M;

SQL> CREATE TABLE testlostwrite (ID NUMBER, TEXTVAL VARCHAR2(100)) TABLESPACE
testlostwritetbs;

SQL> INSERT INTO testlostwrite VALUES(1,'AAA');
SQL> INSERT INTO testlostwrite VALUES(2,'BBB');
SQL> COMMIT;

SQL> CREATE INDEX testlostwrite_IDX on testlostwrite (id);
```

How to Generate a Lost Write

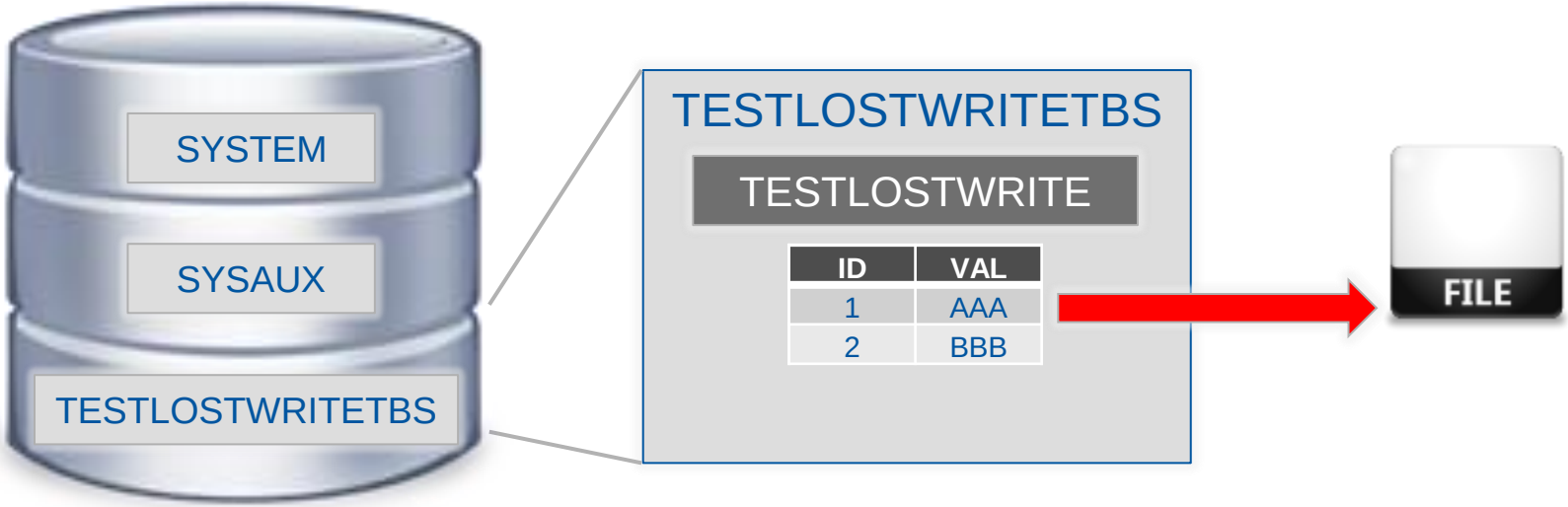
2. Determine the file & block number




```
SQL> SELECT DBMS_ROWID.ROWID_BLOCK_NUMBER(ROWID) BLK,
DBMS_ROWID.ROWID_TO_ABSOLUTE_FNO(ROWID,'TEST','TESTLOSTWRITE') FNO, a.*
FROM testlostwrite a;

           BLK              FNO               ID TEXTVAL
-------------- -------------- -------------- ----------
           132             21              1 AAA
           132             21              2 BBB
```

How to Generate a Lost Write

3. Make a copy of the block (dd or asmblk_edit)




```
SQL> ALTER TABLESPACE testlostwritetbs OFFLINE;
```

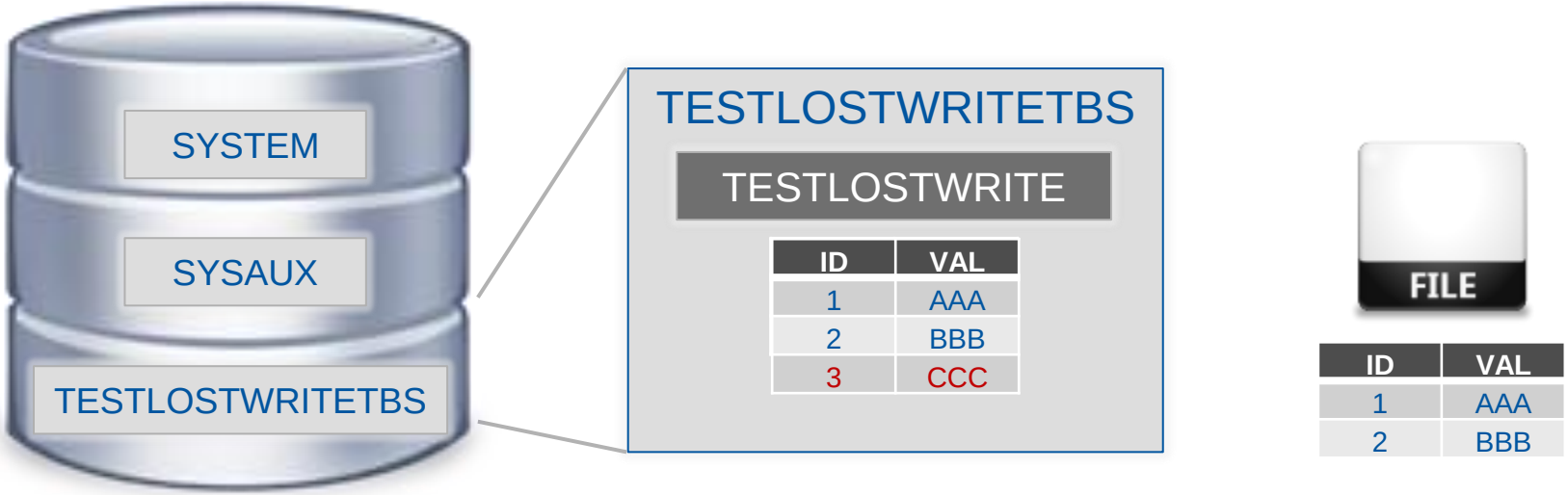
```
./asmblk_edit -r -s 132 -a
+TEST_DATADG1/test/datafile/testlostwritetbs.4138.831900273 -f blk132.dmp
```

```
dd if=testlostwritetbs.dbf bs=8192 count=1 skip=132 of=blk132.dmp
```

```
SQL> ALTER TABLESPACE testlostwritetbs ONLINE;
```

How to Generate a Lost Write

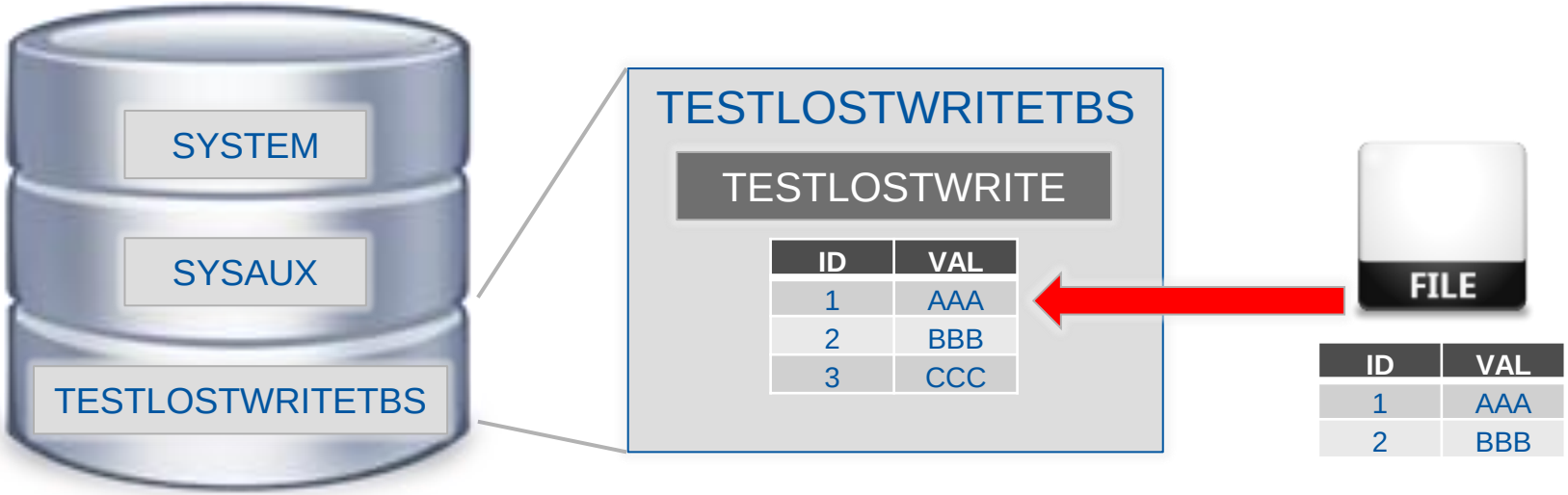
4. Insert additional row into a table – commit the change




```
SQL> INSERT INTO testlostwrite VALUES(3,'CCC');
SQL> COMMIT;
```

How to Generate a Lost Write

5. Overwrite a block to with a copy done before insert




```
SQL> ALTER TABLESPACE testlostwritetbs OFFLINE;
```

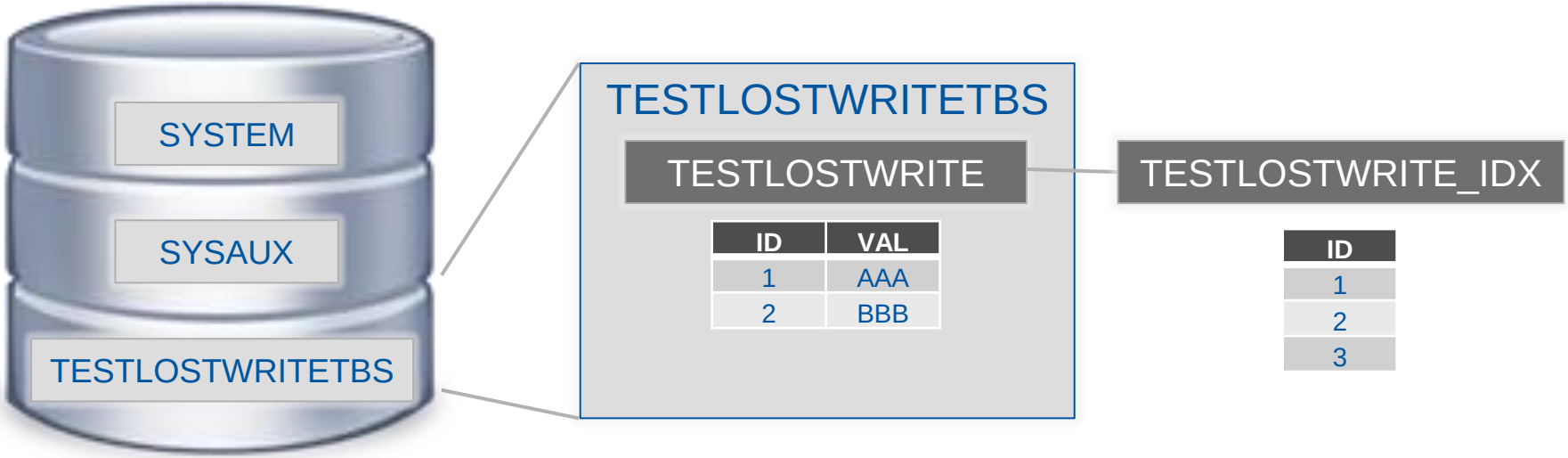
```
./asmblk_edit -w -s 132 -a
+TEST_DATADG1/test/datafile/testlostwritetbs.4138.831900273 -f blk132.dmp
```

```
dd of=testlostwritetbs.dbf bs=8192 count=1 seek=132 if=blk132.dmp conv=notrunc
```

```
SQL> ALTER TABLESPACE testlostwritetbs ONLINE;
```

How to Generate a Lost Write

6. Now we have a Lost Write!




```
SQL> SELECT  /*+ FULL(a)*/ id FROM testlostwrite a;

     COUNT(ID)
--------------
           2
SQL> SELECT /*+ INDEX_FFS(a)*/ id FROM testlostwrite a WHERE id is not null;

     COUNT(ID)
--------------
             3
```

Some Scenarios To Start Testing

- Lost write with and without data guard
  - Experiment with db_lost_write_protect parameter
- Lost write on index
- Lost write on table
  - On a user table
  - On a system object

Techniques to Practice

- Block dumps
  - Datafile block dumps
  - Flashback query
- Investigating redo
  - Logminer
  - Redo log dumps
- How to fix tables with lost writes
  - Generate transaction that compensate for lost DML

The Details and Homework

- Resolving ORA-752 or ORA-600 [3020] During Standby Recovery (Doc ID 1265884.1)
- Best Practices for Corruption Detection, Prevention, and Automatic Repair - in a Data Guard Configuration (Doc ID 1302539.1)
- Testing Lost Writes with Oracle and Data Guard
  - http://externaltable.blogspot.ch/2013_03_01_archive.html
- Data Guard Protection From Lost-Write Corruption demo at
  - http://www.oracle.com/technetwork/database/features/availability/demonstrations-092317.html

Outline

- CERN & Oracle Databases
- Lost write overview
- Lost write in real life
- Testing Lost writes
- Lessons Learned

Lessons Learned

- It’s hard to find the root causes of lost writes
  - How can we tell if the problem is with the storage?
  - Or with the Oracle software (bug)?
  - Or something in between?

- Pain points:
  - Multiple lost write incidents in production while we work on finding the root causes
  - We are caught in between a finger point battle between vendors

Lessons Learned

- Without Data Guard
  - We may not discover lost writes affecting your data
- With (Active) Data Guard
  - MRP recovery will stop and data in the standby become stale

- Pain points:
  - Reduced quality of service for ADG
    - ADG is stale
  - Impact on the disaster recovery strategy
    - Data loss in case of failover

Lessons Learned

- Primary keeps working after lost write
- Redo generated on primary cannot be applied on Data Guard and/or a backup restore

- Pain points:
  - After lost write detection should we failover immediately to standby?
  - Primary is logically corrupted, what is the business impact?

Lessons Learned

- How to be proactive on lost write detection?
  - Can we check if a DB is silently suffering from lost writes?
- We are not aware of a tool that can check this
  - Manual checks are possible
  - Difficult in practice for busy transactional systems
- Pain point:
  - Lost writes can be time bombs

Conclusions

- High Availability systems
  - Test lost write failure and recovery scenarios
- Oracle protection, actions
  - Set db_lost_write_protect=typical
  - Use Data Guard
  - Do test recoveries
- Storage needs to be solid too
- Lost writes
  - Small failures, but a lot of trouble!

Acknowledgements

- CERN Database Group
  - In particular
    - Ruben Gaspar
    - Szymon Skorupinski
    - Emil Pilecki
    - Eric Grancher

Thank you!

Luca.Canali@cern.ch

Marcin.Blaszczyk@cern.ch

www.cern.ch