

Adatbázis korrupció védelme – egy életből vett eset bemutatása

KÉSZÍTETTE : FEJES TIBOR

BUDAPEST, 2018.11.12

Tartalom

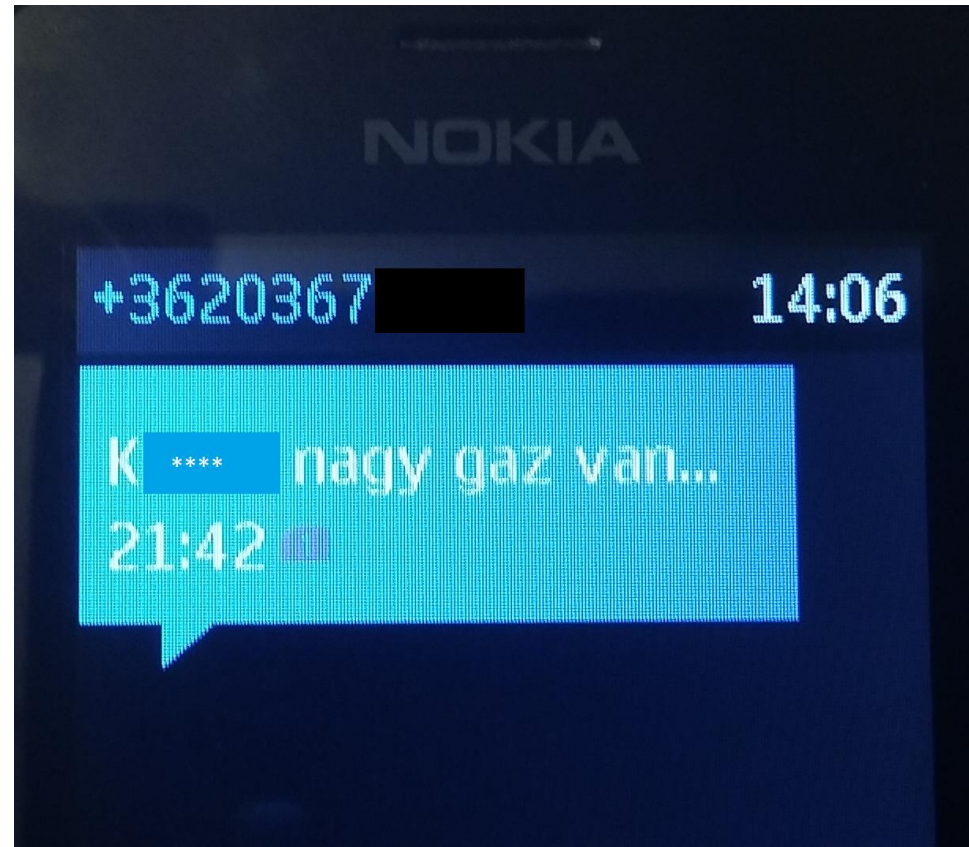
- Mi történt? - esettanulmány
- Mi okozta a hibát és az eseményeket? - nyomozás
- Mi az adatkorruptió
- Milyen lehetőségek vannak a megelőzésre?
- Adatkorruptió felderítése
- Adatkorruptió javítási lehetőségek

Esettanulmány - mi és hogyan történt?

Rendszer architektura:

- 2 site (különböző fizikai lokáció)
- 2 Node RAC site-onként
- DataGuard szinkronizáció

Esettanulmány - mi és hogyan történt?



Esettanulmány - mi és hogyan történt?

- Primary oldalon ASM Disk hozzáadás után: Cluster wait event
- A v\$asm lekérdezések „ASM File Metadata” wait eventeket mutattak
- ASMCMD LSDG ill. LSDSK adminisztratív műveletek nem futottak le
- Adatbázis instance konzisztens leállítási kísérlete nem sikerült
- Naplóváltás nem volt lehetséges – „log file switch archiving needed” wait event
- Az adatbázis kezelő nem tudta kiarchiválni a tranzakciós logokat
- DataGuard Broker konfiguráció elérhetetlenné vált
- Az adatbázis instanceok enq: CF – contenton várakozással vártak a controlfile elérésre
- Az ASM logban korrupció mutatkozott az egyik DG DISK-jére
- CRSCTL STOP CRS -F parancsra nem reagált a clusterware

Esettanulmány - mi és hogyan történt?

Hibakezelés:

- Lábankénti újraindítás nem volt sikeres
- DataGuard Broker kikapcsolása és kézi vezérlésre állítása
- Archiválási és Online logok átirányítása egy ACFS útvonalra
- Az adatbázis instance ezután el tudott indulni és szinkronizált
- Tranzakciós logok kiíródtak az új lokációra, így a DG szinkron helyreállt
- Switchovereztünk a Standby oldalra

Esettanulmány - mi okozta a hibát?

- LOG - DISKGROUP adatkorruptió

```
"WARNING: cache read a corrupt block: group=4 (DATADG) dsk=10  
blk=269 disk=10 (DATADISK23) incarn=4042308738 au=0 blk=269  
count=1,,
```

- 1 héttel a hiba előtt a mostani PRI adatbázis még SBY volt
- RMAN incremental 0 mentés során adatfile korruptió detektálás
- A PRI oldalról javítás történt (adatfileok szinkronizálásával)

Hogy lehet az, hogy ha javítva volt, most mégis előjött az adatkorruptió?

Esettanulmány - mi okozta a hibát?

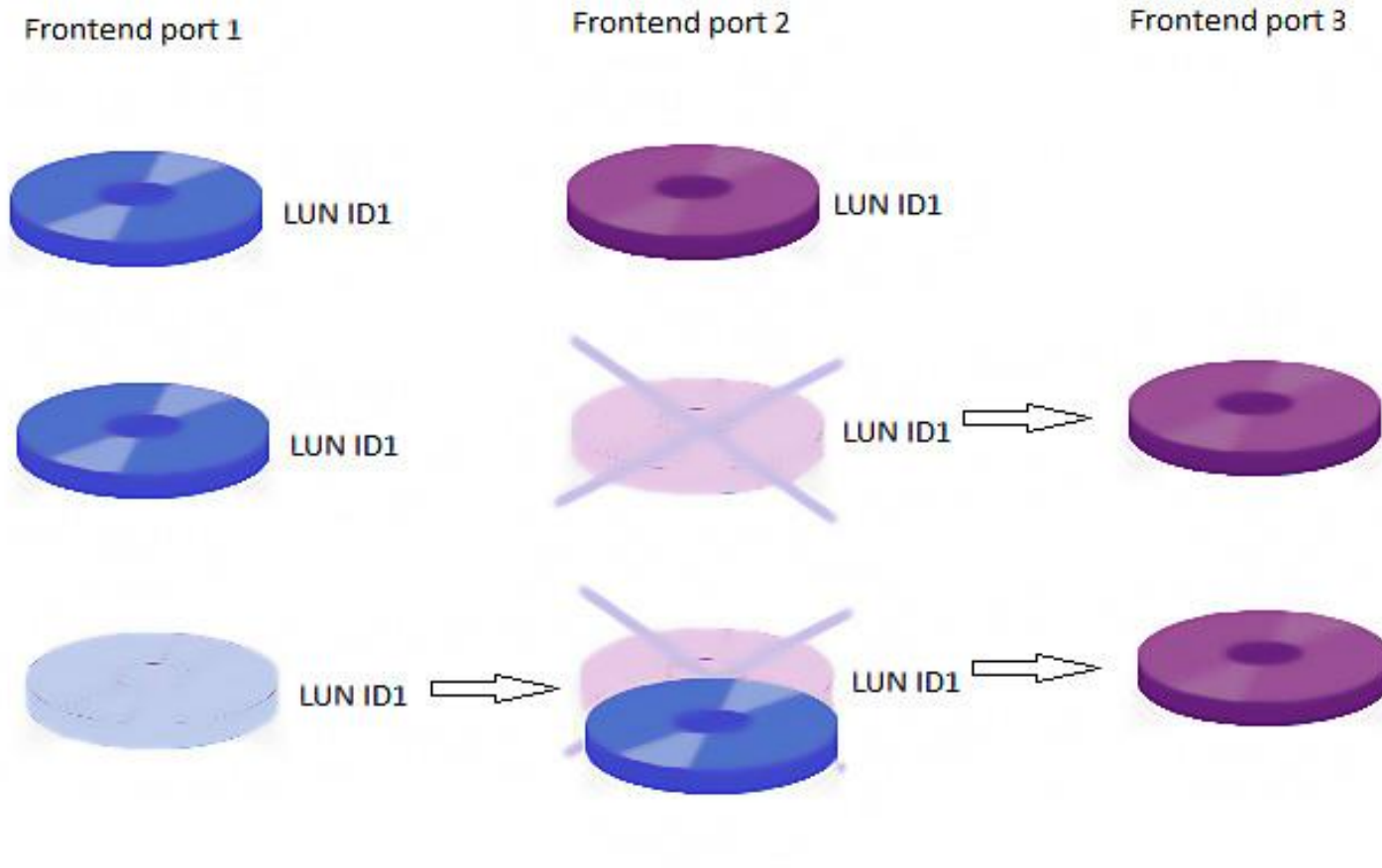
További vizsgálatok:

- A korrupció nem csak az adatfileokat hanem az ASM metaadatokat is érintette
- Az új ASM DISK hozzáadásakor a REBALANCE művelet végrehajtásakor okozott problémát

Hogyan lehetett adatfile és ASM metaadat korrupció egyszerre?

- Online storage oldali karbantartás történt (storage frontend port szeparáció)

Esettanulmány - mi okozta a hibát?



Mi az adatkorruptió?

- Block != Oracle DBMS elvár
- Adatbázis blokk korrupt ha tartalma vagy formátuma megváltozik ahhoz képest amit az Oracle Adatbázis kezelő várna.
- Adatvesztés → Adatbázis működés → Cluster működés

Logikai blokk korruptió:

- Oracle Bug
- Oracle szoftver működése során történő hiba
- Direct Load (SQLLOADER betöltés) No Loggingal

Fizikai blokk korruptió:

- Média korruptió
- Blokk Checksum eltérés
- Filesystem korruptió

Megelőzési lehetőségek

ASM Filter Driver (ASMFD) 12C

- Oracle RAC shared storage kizárva a nodeoknak
- DISK eszközök neve ugyanaz az ASM ben
- Ownership, permission beállítása → ASMLib technológia
- Más processz beleírhat a DISK eszközbe (pl. ROOT)
- ASMFD filter beállítás → blokkol nem Oracle process általi írást
- ASMFD vs ASMLib konfiguráció hasonló
- Lehetséges migráció ASMLib → ASMFD

Megelőzési lehetőségek

```
ASMCMD> afd_state
```

```
ASMCMD-9526: The AFD state is 'LOADED' and filtering is 'ENABLED' on host 'CORRUPTDB'
```

```
ASMCMD> afd_lsdk
```

```
-----  
Label           Filtering Path  
=====
```

DATADISK1	ENABLED	/dev/sdb1
-----------	---------	-----------

```
ASMCMD> lsdk -G DATADG
```

```
Path
```

```
AFD:DATADISK1
```

```
create tablespace TESZTCORR datafile '+DATADG';  
create table T_TESZTCORR (text varchar2(30)) tablespace TESZTCORR;  
insert into T_TESZTCORR values ('NINCS KORRUPCIO');  
commit;
```

Megelőzési lehetőségek

```
[root@CORRUPTDB ~]# grep --byte-offset --only-matching --text "NINCS KORRUPCIO" /dev/sdb1  
372454621:NINCS KORRUPCIO
```

```
[root@CORRUPTDB ~]# od -t c --skip-bytes=$(( 372454621 - 10 )) --read-bytes=40 --width=80 /dev/sdb1  
124625635 \0 \0 \0 \0 \0 \0 , 001 001 \r N I N C S   K O R R U P C I O 001 005 )405 005 133 \0 \0 207 \0
```

```
[root@CORRUPTDB ~]# dd of=/dev/sdb1 if=/dev/zero seek=$((372454621 + 5 )) bs=1 count=5  
conv=notrunc  
5+0 records in  
5+0 records out  
5 bytes (5 B) copied, 0.000164038 s, 30.5 kB/s
```

Nincs error, elvileg sikeres volt az 5 karakter beleírás.

Megelőzési lehetőségek

Alert log :

```
[root@CORRUPTDB]#
```

```
Nov 11 22:13:13 CORRUPTDB kernel: Buffer I/O error on device sdb1, logical block 23821
```

```
Nov 11 22:13:13 CORRUPTDB kernel: lost page write due to I/O error on sdb1
```

```
Nov 11 22:13:13 CORRUPTDB kernel: sd 4:0:2:0: [sdb1] Incomplete mode parameter data
```

```
Nov 11 22:13:13 CORRUPTDB kernel: sd 4:0:2:0: [sdb1] Assuming drive cache: write through
```

```
Nov 11 22:13:13 CORRUPTDB kernel: sdb1: unknown partition table
```

Az alert log szerint az írás elveszett. Az ASMFD blokkolta a diszk írását.

Lekérdezve ismét nincs változás:

```
[root@CORRUPTDB ~]# od -t c --skip-bytes=$(( 372454621 - 10 )) --read-bytes=40 --width=80 /dev/sdb1  
124625635 \0 \0 \0 \0 \0 \0 , 001 001 \r N I N C S   K O R R U P C I O 001 005 )405 005 133 \0 \0 207 \0
```

Megelőzési lehetőségek

Kapcsoljuk ki az FILTERING-et:

```
ASMCMD> afd_filter -d --all
```

```
ASMCMD> afd_state
```

```
ASMCMD-9526: The AFD state is 'LOADED' and filtering is 'DISABLED' on host 'CORRUPTDB'
```

```
ASMCMD> afd_lsdk
```

```
-----  
Label           Filtering Path  
=====
```

DATADISK1	DISABLED	/dev/sdb1
-----------	----------	-----------

```
[root@CORRUPTDB ~]# dd of=/dev/sdb1 if=/dev/zero seek=$(( 97574895 + 5 )) bs=1 count=5
```

```
conv=notrunc
```

```
5+0 records in
```

```
5+0 records out
```

```
5 bytes (5 B) copied, 0.000134231 s, 30.4 kB/s
```

Megelőzési lehetőségek

Alert logban semmi információ nincs sem a hibáról, sem az írásról:

```
[root@CORRUPTDB]#
```

```
Nov 11 22:41:23 CORRUPTDB kernel: sd 4:0:2:0: [sdb1] Incomplete mode parameter data
```

```
Nov 11 22:41:23 CORRUPTDB kernel: sd 4:0:2:0: [sdb1] Assuming drive cache: write through
```

```
Nov 11 22:41:23 CORRUPTDB kernel: sdb1: unknown partition table
```

```
[root@CORRUPTDB ~]# od -t c --skip-bytes=$(( 372454621 - 10 )) --read-bytes=40 --width=80 /dev/sdb1  
124625635 \0 \0 \0 \0 \0 \0 , 001 001 \r N I N C S \0 \0 \0 \0 \0 P C I O 001 005 )405 005 133 \0 \0 207 \0
```

```
SQL> SELECT * FROM T_TESZTCORR;
```

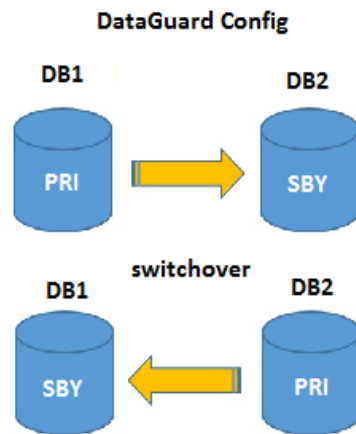
```
ERROR at line 1:
```

```
ORA-01578: ORACLE data block corrupted (file # 5, block # 133)
```

```
ORA-01110: data file 5: '+DATADG/DATAFILE/tesztcorr.123.35532231'
```

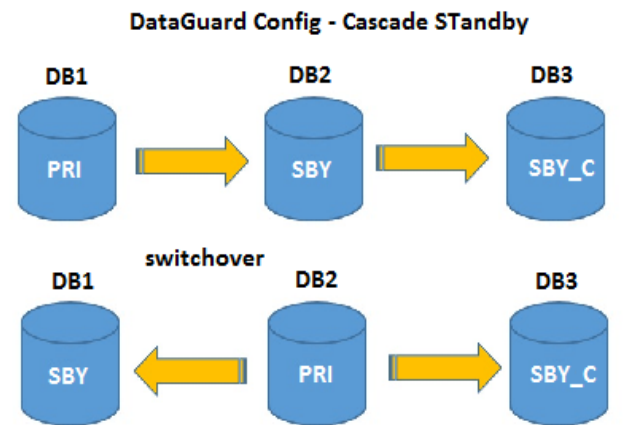

Megelőzési lehetőségek

DataGuard és DataGuard Cascade Standby



Protection Mode: MaxPerformance
Members:
 DB1 - Primary database
 DB2 - Physical standby database

Protection Mode: MaxPerformance
Members:
 DB2 - Primary database
 DB1 - Physical standby database



Protection Mode: MaxPerformance
Members:
 DB1 - Primary database
 DB2 - Physical standby database
 DB3 - Physical standby database (receiving archived redo)

Protection Mode: MaxPerformance
Members:
 DB2 - Primary database
 DB1 - Physical standby database
 DB3 - Physical standby database

Megelőzési lehetőségek

DB_ULTRA_SAFE parameter 11g

- Mielőbbi detektálás annál hamarabbi javítás
- Meta parameter – hatására más paraméterek állítódnak be
- DATA_ONLY, DATA_AND_INDEX, OFF
- DATA_ONLY
 - DB_BLOCK_CHECKING will be set to MEDIUM.
 - DB_LOST_WRITE_PROTECT will be set to TYPICAL.
 - DB_BLOCK_CHECKSUM will be set to FULL.
- DATA_AND_INDEX
 - DB_BLOCK_CHECKING will be set to FULL. (in memory korrupciót is figyel)
 - DB_LOST_WRITE_PROTECT will be set to TYPICAL.
 - DB_BLOCK_CHECKSUM will be set to FULL.

Megelőzési lehetőségek

db_block_checking

- Adat blokk integritást (sérthetetlenség) vizsgál mielőtt az adatblokk kiíródik a diszkre.
- Performancia overhead tapasztalható minden block change-nél

db_block_checksum

- DBWR processz és az sqlloader (direct write) is elvégző checksum műveletet és eltárolja az adatblokk cache headerjében mielőtt kiírja a diszkre.
- Redo és Datablokk korrupciót detektál a PRI oldalon és védi a SBY oldalt.

db_lost_write_protect

- PRI oldalról érkező elveszett írást detektál a SBY oldalon.
- A storage rendszer úgy érzékeli, hogy megtörtént a blokk írás, de valójában nem íródott ki az adat

Megelőzési lehetőségek

Standby adatbázis alert logja:

ORA-00752: recovery detected a lost write of a data block

ORA-10567: Redo is inconsistent with data block (file# 8, block# 351475)

ORA-10564: tablespace TESZTCORR

ORA-01110: data file 8: '+DATADG/datafile/corrteszt01.dbf'

ORA-10561: block type 'TRANSACTION MANAGED DATA BLOCK', data object# 105684

ORA-10877: error signaled in parallel recovery slave PR00

ORA-00752: recovery detected a lost write of a data block

MRP0: Background Media Recovery process shutdown (CORRUPTDB)

11R2 verziótól már javítja is automatikusan Media Recoveryvel ha talál lost write-ot.

Megelőzési lehetőségek

- ASM Filter Driver 12c
- DataGuard
- DataGuard Cascade Standby
- Active DataGuard (Block Media Recovery)
- DB_ULTRA_SAFE 11R1 ill. 11R2

Adatkorruptió felderítése

BACKUP VALIDATE

- RMAN beépített tool, korrupt blokk detektálás

```
RMAN>
spool log to /tmp/backup_validate.log
run
{
allocate channel ch01 device type disk;
backup validate check logical database section size 5000M;
}

...
channel ch01: backup set complete, elapsed time: 00:00:27
List of Datafiles
=====
File Status Marked Corrupt Empty Blocks Blocks Examined High SCN
-----
78 OK 0 7 4194296 8919344567625
File Name: +DATADG/DATAFILE/CORRTESZT.232.937575771
Block Type Blocks Failing Blocks Processed
-----
Data 0 4184811
Index 0 0
Other 0 9478
...
```

Adatkorruptió felderítése

RMAN incremental level 0 mentés:

RMAN>

...

RMAN-00571: =====

RMAN-00569: ===== ERROR MESSAGE STACK FOLLOWS =====

RMAN-00571: =====

RMAN-03009: failure of backup command on ch04 channel at 06/11/2018 21:50:16

ORA-19566: exceeded limit of 0 corrupt blocks for file +DATADG/DATAFILE/CORRTESZT.124.23433535

RMAN-03009: failure of backup command on ch01 channel at 06/11/2018 21:21:56

ORA-19566: exceeded limit of 0 corrupt blocks for file +DATADG/DATAFILE/TESZTREN.321.22555333

RMAN-03009: failure of backup command on ch02 channel at 06/11/2018 21:10:05

ORA-19566: exceeded limit of 0 corrupt blocks for file +DATADG/DATAFILE/TESZT2.112.34335356

RMAN-03009: failure of backup command on ch02 channel at 06/11/2018 21:07:50

...

Adatkorruptió felderítése

DBVERIFY utility:

- Egy konkrét adatfile ellenőrzése

```
oracle@CORRUPTDB:~> dbv file=+DATADG/DATAFILE/TESZT2.112.34335356
DBVERIFY: Release 12.1.0.2.0 - Production on Fri Nov 9 08:30:50 2018
Copyright (c) 1982, 2014, Oracle and/or its affiliates. All rights reserved.
DBVERIFY - Verification starting : FILE = +DATADG/DATAFILE/TESZT2.112.34335356
Page 2296082 is marked corrupt
Corrupt block relative dba: 0x23230912 (file 140, block 2296082)
Bad check value found during dbv:
Data in bad block:
last change scn: 0x0806.60b000d8 seq: 0x1 flg: 0x04
consistency value in tail: 0x00d80601
check value in block header: 0x1bc1
computed block checksum: 0xa1aa
block checksum disabled
Page 2296092 is marked corrupt
```


Adatkorruptió felderítése

DBVERIFY - Verification complete

Total Pages Examined : 4194176
 Total Pages Processed (Data) : 3870692
 Total Pages Failing (Data) : 0
 Total Pages Processed (Index): 499
 Total Pages Failing (Index): 0
 Total Pages Processed (Other): 322962
 Total Pages Processed (Seg) : 0
 Total Pages Failing (Seg) : 0
 Total Pages Empty : 20
 Total Pages Marked Corrupt : 3
 Total Pages Influx : 0
 Total Pages Encrypted : 0
 Highest block SCN : 0 (0.0)

SQL> select * from V\$DATABASE_BLOCK_CORRUPTION ;

FILE#	BLOCK#	BLOCKS	CORRUPTION_CHANGE#	CORRUPTIO	CON_ID
-----	-----	-----	-----	-----	-----
112	2296082	1	0	CHECKSUM	0

Adatkorruptió felderítése

- DB_ULTRA_SAFE
- BACKUP VALIDATE
- RMAN incremental mentés
- DBVERIFY utility

Adatkorruptió javítása

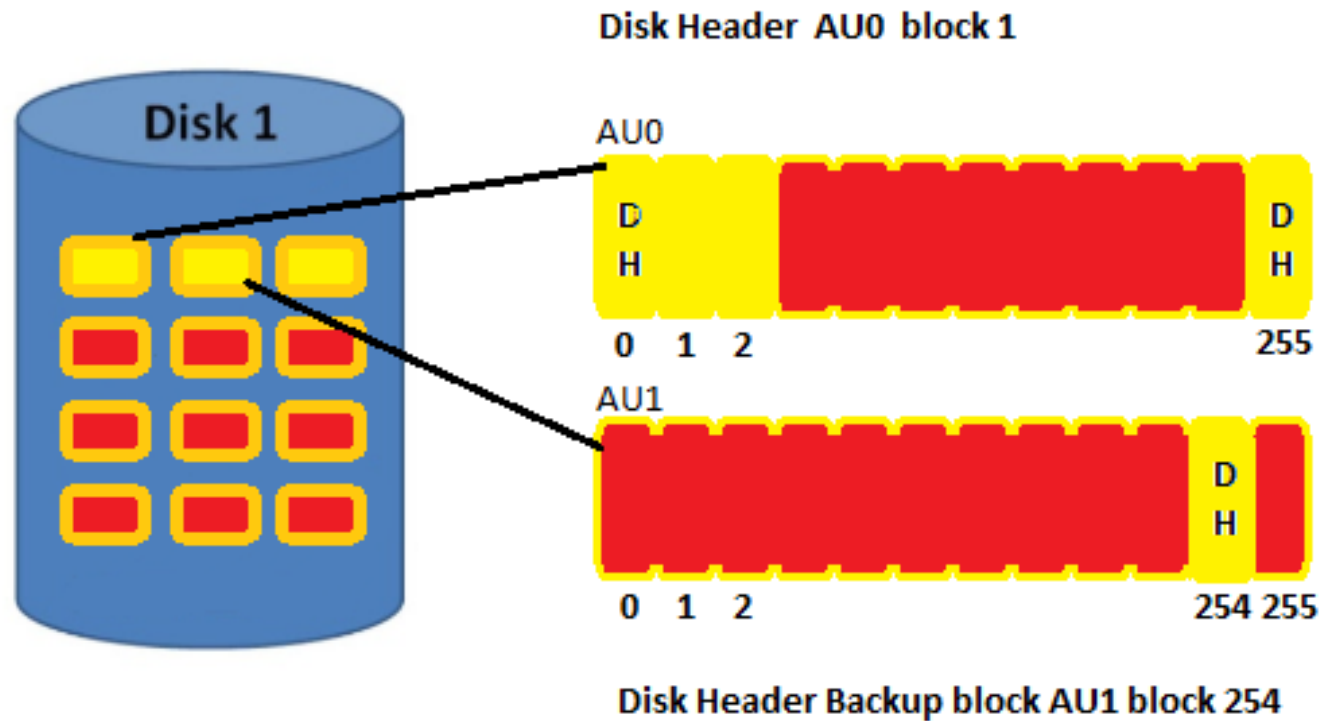
ASM DISK HEADER CORRUPTION & REPAIR (KFED)

- ASM DISK allocation unitokból (AU) épül fel
- AU-k ASM metaadatot és adatbázis adatot tartalmaznak
- Azok az AU-k amik metaadatot tartalmaznak ASM metadata blokként vannak formázva
- AU0 mindig tartalmaz metaadatot és az AU első blokkja a diszk headerje
- AU1 254. blokkja mindig tartalmaz egy backupot a diszk headerről

ASM disk header:

- Információk diszkről, diszkgroupról vagy clusterwareről

Adatkorruptió javítása



Adatkorruptió javítása

Kernel file editor (KFED) 11g ASM tool:

```
$ kfed read /dev/oracleasm/disks/DATA1 aun=0 blk=0
```

```
kfbh.endian:          0 ; 0x000: 0x00
kfbh.hard:            0 ; 0x001: 0x00
kfbh.type:            0 ; 0x002: KFBTYP_DISKHEAD
kfbh.datfmt:         0 ; 0x003: 0x00
kfbh.block.blk:      0 ; 0x004: T=0 NUMB=0x0
kfbh.block.obj:      0 ; 0x008: TYPE=0x0 NUMB=0x0
kfbh.check:          0 ; 0x00c: 0x00000000
kfbh.fcn.base:       0 ; 0x010: 0x00000000
kfbh.fcn.wrap:       0 ; 0x014: 0x00000000
kfbh.spare1:         0 ; 0x018: 0x00000000
kfbh.spare2:         0 ; 0x01c: 0x00000000
```

Adatkorruptió javítása

```
$ dd if=/dev/zero of=/dev/oracleasm/disks/DATA1 bs=4096 count=1  
1+0 records in  
1+0 records out  
4096 bytes (4.1 kB) copied, 3.8e-05 seconds, 108 MB/s
```

```
$ kfed read /dev/oracleasm/disks/DATA1 aun=0 blkn=0
```

```
kfbh.endian:          0 ; 0x000: 0x00  
kfbh.hard:            0 ; 0x001: 0x00  
kfbh.type:           0 ; 0x002: KFBTYP_INVALID
```

Az ASM header corrupt: KFBTYP_INVALID

Adatkorruptió javítása

```
SQL> select DISK_NUMBER,HEADER_STATUS,substr(PATH,1,20),label from v$asm_disk;
```

DISK_NUMBER	HEADER_STATUS	PATH	LABEL
0	CANDIDATE	ORCL:DATA1	DATA1
1	MEMBER	ORCL:DATA2	DATA2
0	MEMBER	ORCL:DATA3	DATA3

Megnézzük a backupot:

```
$ kfed read /dev/oracleasm/disks/DATA1 aun=1 blkn=254 |grep KFBTYP  
kfbh.type: 1 ; 0x002: KFBTYP_DISKHEAD
```

Adatkorruptió javítása

Használjuk kfed repairt a javításhoz:

```
$ kfed repair /dev/oracleasm/disks/VOL1
```

```
$ kfed read /dev/oracleasm/disks/VOL1 aun=0 blk=0 | grep KFBTYP  
kfbh.type:          1 ; 0x002: KFBTYP_DISKHEAD
```


Adatkorruptió javítása

REPAIR & CHECK 11g

- Hibás blokkokat tud detektálni és javítani menetközben
- Tükör diszket használ
- Normál és High redundancy diszkekre használható

```
SQL> ALTER DISKGROUP DATADG CHECK REPAIR;
```

Adatkorruptió javítása

SCRUBBING 12c

- Logical korruptiót tud detektálni és javítani
- Tükör diszket használ
- Normál és High redundancy diszkekre használható
- Rebalance power beállítás alkalmazásával javítható a performancia

```
SQL> select NAME, TYPE, STATE from v$asm_diskgroup_stat;
```

NAME	TYPE	STATE
-----	-----	-----
DATADG	NORMAL	MOUNTED

```
SQL> select DISK_NUMBER, PATH from v$asm_disk_stat;
```

DISK_NUMBER	PATH
-----	-----
0	AFD:DATADISK1
2	AFD:DATADISK2

Adatkorruptió javítása

Tegyük korrupttá a mirror oldalt.

```
[root@CORRUPTDB]# dd if=block_corrupt.txt of=/dev/sdb1 bs=8192 count=1 seek=1462088  
1+0 records in  
1+0 records out  
8192 bytes (8.2 kB) copied, 0.00160027 s, 5.1 MB/s
```

```
SQL> alter diskgroup DATA scrub file '+DATADG/DATAFILE/teszt2.307.897911659' wait;
```

```
...
```

Adatkorruptió javítása

Alert logban látszik is a találat:

Nov 08 13:25:48 2018

```
SQL> alter diskgroup DATA scrub file '+DATADG/DATAFILE/teszt2.307.897911659' wait
```

```
Starting background process SCRIB
```

Nov 08 13:25:48 2018

```
SCRIB started with pid=25, OS id=4585
```

Nov 08 13:25:48 2018

```
NOTE: Waiting for scrubbing to finish
```

Nov 08 13:25:49 2018

```
NOTE: Corrupted block 200 found in file +DATADG/DATAFILE/teszt2.307.897911659
```

```
SUCCESS: alter diskgroup DATA scrub file '+DATADG/DATAFILE/teszt2.307.897911659' wait
```

Adatkorruptió javítása

Javítsuk ki:

```
SQL> alter diskgroup DATA scrub file '+DATADG/DATAFILE/teszt2.307.897911659' repair wait;  
...
```

Adatkorruptió javítása

Alert logban látszik a javítás („scrubbed”).

Nov 08 13:35:02 2018

```
SQL> alter diskgroup DATA scrub file '+DATADG/DATAFILE/teszt2.307.897911659' repair wait
```

Nov 08 13:35:02 2018

NOTE: Waiting for scrubbing to finish

Nov 08 13:35:03 2018

NOTE: Corrupted block 200 found in file '+DATADG/DATAFILE/teszt2.307.897911659'

Nov 08 13:35:04 2018

NOTE: Scrubbing block 200 in file 307.897911659 in slave

NOTE: Successfully scrubbed block 200 in file 307.897911659

Nov 08 13:35:53 2018

NOTE: Scrubbing finished

Nov 08 13:35:53 2018

```
SUCCESS: alter diskgroup DATA scrub file '+DATADG/DATAFILE/teszt2.307.897911659' repair wait
```

Adatkorrupció javítása

- ASM DISK HEADER CORRUPTION & REPAIR (KFED) 11g
- REPAIR & CHECK 11g
- SCRUBBING 12c

Köszönöm a figyelmet!