

ASM Metadata and Internals

A collection of facts on configuration and and diagnostic of Oracle ASM. More on RAC and ASM configuration and performance of CERN Physics DBs in Inside_Oracle_ASM_LC_CERN_UKOU07.ppt and in Presentation at E42014.

ASM metadata, V\$ and X\$:

View Name	X\$ Table name	Description
V\$ASM_DISK	X\$KFDSK, X\$KFKID	performs disk discovery, lists disks and their usage metrics
V\$ASM_DISKGROUP	X\$KFGRP	performs disk discovery and lists diskgroups
V\$ASM_DISKGROUP_STAT	X\$KFGRP_STAT	diskgroup stats without disk discovery
V\$ASM_DISK_STAT	X\$KFDSK_STAT, X\$KFKID	lists disks and their usage metrics
V\$ASM_FILE	X\$KFFIL	lists ASM files, including metadata/asmdisk files
V\$ASM_ALIAS	X\$KFALS	lists ASM aliases, files and directories
V\$ASM_TEMPLATE	X\$KFTMTA	lists the available templates and their properties
V\$ASM_CLIENT	X\$KFNCL	lists DB instances connected to ASM
V\$ASM_OPERATION	X\$KFGMG,X\$KFGBRB(11g)	lists rebalancing operations
N.A.	X\$KFKLIB	available libraries, includes asmlib path
N.A.	X\$KFDPARTNER	lists disk-to-partner relationships
N.A.	X\$KFFXP	extent map table for all ASM files
N.A.	X\$KF DAT	extent list for all ASM disks
N.A.	X\$KFBH	describes the ASM cache (buffer cache of ASM in blocks of 4K (_asm_blksize))
N.A.	X\$KFCCE	a linked list of ASM blocks. to be further investigated

This list is obtained querying v\$fixed_view_definition and v\$fixed_table: `select * from v$fixed_view_definition where view_name like '%ASM%';` and `select * from v$fixed_table where name like 'X$KF%';` (ASM fixed tables use the X\$KF prefix).

New in 11g (11.2.0.2 is taken as reference):

View Name	X\$ Table name	Description
V\$ASM_ACFSSNAPSHOTS	X\$KRVACFSS	snapshots of ACFS filesystems
V\$ASM_ACFSVOLUMES	X\$KRVACFSV	info on monted ACFS volumes
V\$ASM_ACFSENCRYPTION_INFO	X\$KRVACFSENCR	info on ACFS encryption config
V\$ASM_ACFSSSECURITY_INFO	X\$KRVACFSREALM	info on ACFS security (realm) config
V\$ASM_ATTRIBUTE	X\$KRVENV	ASM DG attributes. Data stored in file #9 of each DG Notes: the X\$ table shows also 'hidden' attributes, Example to turn off variable extents <code>alter diskgroup set attribute '_extent_counts'='214748367 0 0';</code>
V\$ASM_DISK_IOSTAT	X\$KRVNSDSKIOST	I/O usage statistics
V\$ASM_FILESYSTEM	X\$KRVACFS	ACFS filesystems
V\$ASM_USER	X\$KRVZUDR	os users info
V\$ASM_USERGROUP	X\$KRVZGDR	creators of ASM file access control

ASM_Internals < PDBService < TWiki

		group
V\$ASM_USERGROUP_MEMBER	X\$KFZUAGR	members of ASM file access control groups
V\$ASM_VOLUME	X\$KFVOL, X\$KFFIL	info on ADVN volumes created on ASM SGs
V\$ASM_VOLUME_STAT	X\$KFVOL,X\$KFVOLSTAT	stats on ADVN volumes created on ASM SGs
N.A.	X\$X\$KFCBH	
N.A.	X\$KFCLLE	
N.A.	X\$KFDDD	
N.A.	X\$KFDFS	
N.A.	X\$KFFOF	reports the list of open files. it is the source for lsof in asmcmd
V\$ASM_OPERATION in 11g	X\$KFGBRB	
N.A.	X\$KFGBRW	
N.A.	X\$KFKLSOD	reports the list of open devices. it is the source for lsod in asmcmd
N.A.	X\$KFMGRP	
N.A.	X\$KFRC	
N.A.	X\$KFVOFS	no more there in 11.2.0.3
N.A.	X\$KFVOFSV	no more there in 11.2.0.3

New in 12c (12.1.0.1 is taken as reference):

N.A.	X\$KFDAP	
N.A.	X\$KFDSO	
N.A.	X\$KFDSR	
N.A.	X\$KFDXEXT	
N.A.	X\$KFGBRC	
N.A.	X\$KFGBRS	
N.A.	X\$KFIAS_FILE	
N.A.	X\$KFIAS_PROC	
N.A.	X\$KFNRC	
N.A.	X\$KFCSTAT	I.O. statistics
GV\$ASM_ESTIMATE	X\$KFGXP	
GV\$IOS_CLIENT	X\$KFIAS_CLNT	currently undocumented GV\$
GV\$ASM_ACFS_SEC_ADMIN	X\$KVVACFSADMIN	
GV\$ASM_ACFS_SEC_CMDRULE	X\$KVVACFSCMDRULE	
GV\$ASM_ACFS_SEC_REALM_FILTER	X\$KVVACFSREALMFILTER	
GV\$ASM_ACFS_SEC_REALM_GROUP	X\$KVVACFSREALMGROUP	
GV\$ASM_ACFS_SEC_REALM	X\$KVVACFSREALMS	
GV\$ASM_ACFS_SEC_REALM_USER	X\$KVVACFSREALMUSER	
GV\$ASM_ACFSREPL	X\$KVVACFSREPL	
GV\$ASM_ACFSREPLTAG	X\$KVVACFSREPLTAG	
GV\$ASM_ACFS_SEC_RULE	X\$KVVACFSRULE	
GV\$ASM_ACFS_SEC_RULESET		X\$KVVACFSRULESET
GV\$ASM_ACFS_SEC_RULESET_RULE	X\$KVVACFSRULESETRULE	
GV\$ASM_ACFSTAG	X\$KVVACFSTAG	
N.A.	X\$KFZPBLK	info on the password files in asm

X\$KFFXP (metadata, file extent pointers)

This X\$ table contains the mapping between files, extents and allocation units. It allows to track the position of all the extents of a given file striped and mirrored across storage. Note: RDBMS read operations access only the primary extent of a mirrored couple (unless there is an IO error) . Write operations instead write all mirrored extents to disk.

X\$KFFXP Column Name	Description
ADDR	x\$ table address/identifier
INDX	row unique identifier
INST_ID	instance number (RAC)
GROUP_KFFXP	ASM disk group number. Join with v\$asm_disk and v\$asm_diskgroup
NUMBER_KFFXP	ASM file number. Join with v\$asm_file and v\$asm_alias
COMPOUND_KFFXP	File identifier. Join with compound_index in v\$asm_file
INCARN_KFFXP	File incarnation id. Join with incarnation in v\$asm_file
PXN_KFFXP	Progressive file extent number
XNUM_KFFXP	ASM file extent number (mirrored extent pairs have the same extent value) a value of 2147483648 is for the triple-mirrored file metadata
DISK_KFFXP	Disk number where the extent is allocated. Join with v\$asm_disk can have the value 65534 when AU not present on physical storage (applies to normal or high redundancy DG)
AU_KFFXP	Relative position of the allocation unit from the beginning of the disk. The allocation unit size (1 MB) in v\$asm_diskgroup can have the value 4294967294 when AU not present on physical storage because of failure for example (applies to normal or high redundancy DG)
LXN_KFFXP	0->primary extent, ->mirror extent, 2->2nd mirror copy (high redundancy and metadata)
FLAGS_KFFXP	N.K.
CHK_KFFXP	N.K.
SIZE_KFFXP	11g , to support variable size AU, integer value which marks the size of the extent in AU size units. extent sizes are determined by the diskgroup parameter _extent_sizes, the default value in 11gR2 and 12c this is: '1 4 16' and the extent sizes by _extent_counts, default= 20000 20000 214748367, that is the first 20000 extents have size 1 AU, then the next 20000 extents have size 4 AUs, all the subsequent extents have size 16 AUs.

Example - find location of ASM files extents using x\$kfxfp

- Find the 2 mirrored extents of an ASM file (the spfile in this example)

```
sys@+ASM1> select GROUP_KFFXP,DISK_KFFXP,AU_KFFXP from x$kfxfp where
number_kfxfp=(select file_number from v$asm_alias where name='spfiletest1.ora');
```

```
GROUP_KFFXP DISK_KFFXP AU_KFFXP
-----
1           20         379
1           3          101
```

- find the diskname

```
sys@+ASM1> select disk_number,path from v$asm_disk where
group_number=1 and disk_number in (3,20);
```

DISK_NUMBER PATH

```
-----
      3  /dev/mapper/itstor417_2p1
     20  /dev/mapper/itstor419_2p1
```

- access the data directly from disk with dd

```
dd if=/dev/mapper/itstor417_2p1 bs=1024k count=1 skip=101|strings|more
```

- Example: extract extent map for a given datafile (487 in group 1 in the example):

```
select xnum_kffxp,lxn_kffxp,pxn_kffxp,(select path from v$asm_disk where disk_number=disk_kffxp),
```

X\$KFDAT (metadata, disk-to-AU mapping table)

This X\$ table contains details of **all allocation units** (free and used).

X\$KFDAT Column Name	Description
ADDR	x\$ table address/identifier
INDX	row unique identifier
INST_ID	instance number (RAC)
GROUP_KFDAT	diskgroup number, join with v\$asm_diskgroup
NUMBER_KFDAT	disk number, join with v\$asm_disk
COMPOUND_KFDAT	disk compund_index, join with v\$asm_disk
AUNUM_KFDAT	Disk allocation unit (relative position from the beginning of the disk), join with x\$kffxp.au_kffxp
V_KFDAT	V=this Allocation Unit is used; F=AU is free
FNUM_KFDAT	file number, join with v\$asm_file
I_KFDAT	N.K.
H_KFDAT	11g , N.K.
XNUM_KFDAT	Progressive file extent number join with x\$kffxp.pxn_kffxp
RAW_KFDAT	raw format encoding of the disk,and file extent information
SIZE_KFDAT	11g , N.K.
FMT_KFDAT	11g , N.K.

Example2 - list allocation units of a given file from x\$kfdat

- similarly to example 1 above, another way to retrieve ASM file allocation maps:

```
sys@+ASM1> select GROUP_KFDAT,NUMBER_KFDAT,AUNUM_KFDAT from x$kfdat where
      fnum_kfdat=(select file_number from v$asm_alias where name='spfiletest1.ora');
```

```
GROUP_KFDAT NUMBER_KFDAT AUNUM_KFDAT
-----
          1             3          101
          1            20          379
```

Example3 - list extents belonging to voting disk in ASM (11gR2)

```
select * from x$kfdat where group_kfdat=1 and fnum_kfdat=1048572 order by
number_kfdat,AUNUM_KFDAT;
```

Example4 - from strace data of an oracle user process

- from the strace file of a user (shadow) process identify IO operations:
 - ◆ ex: `strace -p 30094 2>&1|grep pread`
 - ◆ `pread(257, "#\242\0\0\33\0@\2\343\332\177\303s\5\1\4\211\330\0\0"..., 8192, 473128960) = 8192`
 - ◆ it is a read operation of 8KB (oracle block) at the offset 473128960 (=451 MB + 27*8KB) from file descriptor FD=257
- using `/proc/30094/fd -> find FD=257 is /dev/mapper/itstor420_1p1`
- I find the group and disk number of the file:

```
sys@+ASM1> select GROUP_NUMBER,DISK_NUMBER from v$asm_disk
where path='/dev/mapper/itstor420_1p1';
```

GROUP_NUMBER	DISK_NUMBER
1	30

- using the disk number, group number and offset (from strace above) I find the file number and extent number:
 - ◆ note in this example we cover an extent with `size_kffxp=1`, the case of an extent spanning more AUs requires additional calculations.

```
sys@+ASM1> select number_kffxp, XNUM_KFFXP,size_kffxp from x$kffxp where group_kffxp=1 and disk_k
```

NUMBER_KFFXP	XNUM_KFFXP	SIZE_KFFXP
268	17	1

- from `v$asm_file fnum=268` is file of the users' tablespace:

```
sys@+ASM1> select name from v$asm_alias where FILE_NUMBER=268
```

NAME
USERS.268.612033477

```
sys@DB> select file#,name from v$datafile where upper(name) like '%USERS.268.612033477';
```

FILE#	NAME
9	+TEST1_DATADG1/test1/datafile/users.268.612033477

- from `dba extents` finally find the owner and segment name relative to the original IO operation:

```
sys@TEST1> select owner,segment_name,segment_type from dba_extents
where FILE_ID=9 and 27+17*1024*1024/8192 between block_id and block_id+blocks;
```

OWNER	SEGMENT_NAME	SEGMENT_TYPE
SCOTT	EMP	TABLE

Extent and AU allocations in asmcmd 12c

In 12c `asmcmd` has 2 new commands to help navigating ASM extent pointers and disk allocations: `mapextent` and `mapau`. Example, how to find the allocation units for the first extents of a given files:

```
ASMCMD> mapextent '+ORCL_MYTEST/ORCL/DATAFILE/mytest.256.844901607' 1
Disk_Num      AU      Extent_Size
```

```
1          107          1
0          107          1
```

Example, given a disk number and allocation unit number, how to find the file number and extent number:

```
ASMCMD> mapau
usage: mapau [--suppressheader] <dg number> <disk number> <au>
help: help mapau
ASMCMD> mapau 1 1 107
File_Num      Extent      Extent_Set
261           1273         636
```

X\$KFDPARTNER

This X\$ table contains the disk-to-partner (1-N) relationship. Two disks of a given ASM diskgroup are partners if they each contain a mirror copy of the same extent. Therefore partners must belong to different failgroups of the same diskgroup. This mechanism is in place to reduce the chance of losing both sides of the mirror in case of double disk failure. The limit to the number of partners per disk is:

_asm_partner_target_disk_part (default 8 in recent versions). We also have _asm_partner_target_fg_rel (target maximum number of failure group relationships for repartnering, default 4):

X\$KFDPARTNER Column Name	Description
ADDR	x\$ table address/identifier
INDX	row unique identifier
INST_ID	instance number (RAC)
GRP	diskgroup number, join with v\$asm_diskgroup
DISK	disk number, join with v\$asm_disk
COMPOUND	disk identifier. Join with compound_index in v\$asm_disk
NUMBER_KFDPARTNER	partner disk number, i.e. disk-to-partner (1-N) relationship
MIRROR_KFDPARTNER	=1 in a healthy normal redundancy config
PARITY_KFDPARTNER	=1 in a healthy normal redundancy config
ACTIVE_KFDPARTNER	=1 in a healthy normal redundancy config
11g, DISKFGNUM	failgroup number of the disk
11g, PARTNERFGNUM_KFDPARTNER	failgroup number of the partner disk

X\$KFFIL and metadata files

Three types of metadata:

- diskgroup metadata: files with NUMBER_KFFIL <256 ASM metadata and ASMlog files. These files have high redundancy (3 copies) and block size =4KB.
 - ◆ ASM log files are used for ASM instance and crash recovery when a crash happens with metadata operations (see below COD and ACD)
 - ◆ at diskgroup creation 6 files with metadata are visible from x\$kffil
- disk metadata: disk headers (typically the first 2 AU of each disk) are not listed in x\$kffil (they appear as file number 0 in x\$kfdat). Contain disk membership information. This part of the disk has to be 'zeroed out' before the disk can be added to ASM diskgroup as a new disk.
- file metadata: 3 mirrored extents with file metadata, visible from x\$kfpx and x\$kfdat * note: metadata i triple mirrored if at least 3 failgroups are available

Example: list all files, system and users' with their sizes:

ASM_Internals < PDBService < TWiki

- `select group_kffil group#, number_kffil file#, filsiz_kffil
filesize_after_mirr, filspc_kffil raw_file_size from x$kffil;`

Example: List all files including metadata allocated in the ASM diskgroups

- `select group_kfdat group#,FNUM_KFDAT file#, sum(1) AU_used from
x$kfdat where v_kfdat='V' group by group_kfdat,FNUM_KFDAT,v_kfdat;`

Description of metadata files

References: Oracle Automatic Storage Management, Oracle Press Nov 2007, N. Vengurlekar, M. Vallath, R.Long and [<http://asmsupportguy.blogspot.com>] Bane Radulovic

- on each disk, AU=0: disk header (disk name, etc), first stride of the Allocation Table (AT) and Free Space Table (FST)
- on each disk, AU=1: space allocate for the Partner Status Table (PST) (not all disks have PST data)
- on each disk, AU=11 block 1: **12c** additional copy of the disk header

- File#1: File Directory (files and their extent pointers)
- File#2: Disk Directory
- File#3: Active Change Directory (ACD) The ACD is analogous to a redo log, where changes to the metadata are logged. Size=42MB * number of instances
- File#4: Continuing Operation Directory (COD). The COD is analogous to an undo tablespace. It maintains the state of active ASM operations such as disk or datafile drop/add. The COD log record is either committed or rolled back based on the success of the operation.
- File#5: Template directory
- File#6: Alias directory
- File#8: 11g ?? content N.K.
- 11g, File#9: Attribute Directory
- 11g, File#12: Staleness directory, allocated when needed to track offline disks
- 12c, File #13 ASM password directory
- 11g, File#253: ASM spfile in ASM (11gR2 feature)
- 11g, File#254: Staleness registry, allocated when needed to track offline disks
- 11g, File#255: OCR FILE in ASM (11gR2 feature)

- 11g, File#1048572 (Hex=FFFFC), special file, does not appear in x\$kffxp: it contains the mirrored copies of the voting disk in ASM (11gR2 and 12c), 3 copies for normal redundancy
- 11g, File#1048575 (Hex=FFFFFF), not a real file#, does not appear in x\$kffxp, content N.K., it appears to allocate a relatively small size at the end of each ASM disk.

Tnsnames entries and ASM (relevant for 10g)

TIP: An example of tnsnames entry to be used to connect to ASM instances via Oracle*NET (note the extra keyword (UR=A)). More generally UR=A allows to connect to 'blocked services'. Example connect sys/pass@ASM1 as sysdba (an asm password file is also needed on the server). The extra keyword (UR=A) applies to 10g, it is not needed in 11g.

```
ASM1 =  
(DESCRIPTION =  
  (ADDRESS = (PROTOCOL = TCP) (HOST = [hostname]) (PORT = [portN]))  
  (CONNECT_DATA =  
    (SERVER = DEDICATED) (SERVICE_NAME = +ASM) (INSTANCE_NAME = +ASM1)  
    (UR=A)  
  ) )
```

DBMS_DISKGROUP, an internal ASM package

dbms_diskgroup is an Oracle 'internal package' (C implementation, as opposed to PL/SQL), it provides and API to access ASM data. It is used by external programs, for example asmcmd 12c. A list of available procedures obtained from strings \$ORACLE_HOME/bin/oracle|grep -i dbms_diskgroup: Note on how to further research this: asmcmd in 11g and 12c is a collection of PERL scripts who use dbms_diskgroup for asm manipulation. Use find \$ORA_CRS_HOME -name asmcmd*|xargs grep -i dbms_diskgroup.

```

dbms_diskgroup.abortfile(:handle)
dbms_diskgroup.addcreds(:osuname,:clusid,:uname,:passwd);
dbms_diskgroup.asmcopy (:src_path, :dst_name, :spfile_number, :fileType, :blkSz, :spfile_number2, :
dbms_diskgroup.checkfile (v_AsmFileName,v_FileType,v_lbks,v_offstart,v_FileSize)
dbms_diskgroup.close (:handle);
dbms_diskgroup.commitfile (:handle);
dbms_diskgroup.copy ('', '', '', :src_path, :src_ftyp, :src_blksize, :src_fsiz, '', '', '', :dst_path,
dbms_diskgroup.createclientcluster (:cname, :direct_access)
dbms_diskgroup.createdir (:NAME);
dbms_diskgroup.createfile (:NAME, :type, :lblksize, :fsz, :handle, :pblksize, :genfname);
dbms_diskgroup.dropdir (:DIRNAME)
dbms_diskgroup.dropfile (:NAME, :type);
dbms_diskgroup.getfileattr (:src_path, :fileType, :fileSz, :blkSz)
dbms_diskgroup.getfileattr (:NAME, :type, :fsz, :lblksize, 1, :hideerr);
dbms_diskgroup.getfilephyblksize (:fileName, :flag, :pblksize)
dbms_diskgroup.gethdlattr (:handle, :attr, :nval, :sval);
dbms_diskgroup.gpnpsesp (:spfile_path)
dbms_diskgroup.mapau (:gnum, :disk, :file, :extent, :xsn)
dbms_diskgroup.mapextent (:NAME, :xsn, :mapcount, :extsize, :disk1, :au1, :disk2, :au2, :disk3, :au3);
dbms_diskgroup.mkdir (:DIRNAME)
dbms_diskgroup.open (:NAME, :fmode, :type, :lblksize, :handle, :pblksize, :fsz);
dbms_diskgroup.openpfile (:NAME, :lblksize, :fsz, :handle, :pblksize, :fmode, :genfname, :dbname);
dbms_diskgroup.patchfile (v_AsmFilename, v_filetype, v_lbks, v_offstart, 0, v_numblks, v_FsFilename, v_f
dbms_diskgroup.read (:handle, :offset, :length, :buffer, :reason, :mirr);
dbms_diskgroup.remap (:gnum, :fnum, :vxn)
dbms_diskgroup.renamefile (:NAME, :tname, :type, :genfname);
dbms_diskgroup.resizefile (:handle, :fsz);
dbms_diskgroup.write (:handle, :offset, :length, :buffer, :reason);

```

ASM Oracle kernel components and prefixes

SQL> oradebug doc component asm

ASM	Automatic Storage Management (kf)
KFK	KFK (kfk)
KFKIO	KFK IO (kfkio)
KFKSB	KFK subs (kfksubs)
KFN	ASM Networking subsystem (kfn)
KFNU	ASM Umbillicus (kfnm, kfns, kfnb)
KFNS	ASM Server networking (kfns)
KFNC	ASM Client networking (kfnc)
KFIS	ASM Intelligent Storage interfaces (kfis)
KFM	ASM Node Monitor Interface Implementation (kfm)
KFMD	ASM Node Monitor Layer for Diskgroup Registration (kfmd)
KFMS	ASM Node Monitor Layers Support Function Interface (kfms)
KFFB	ASM Metadata Block (kffb)
KFFD	ASM Metadata Directory (kffd)
KFZ	ASM Zecurity subsystem (kfz)
KFC	ASM Cache (kfc)
KFR	ASM Recovery (kfr)
KFE	ASM attributes (kfe)
KFDP	ASM PST (kfdp)
KFG	ASM diskgroups (kfg)
KFDS	ASM staleness registry and resync (kfds)

KFIA	ASM Remote (kfia)
KFIAS	ASM IOserver (kfias)
KFAC	ASM IOserver client (kfiac)
KFFSCRUB	ASM Scrubbing (kffscrub)
KFIO	ASM translation I/O layer (kfio)
KFIOER	ASM translation I/O layer (kfioer)
KFV	ASM Volume subsystem (kfv)
KFVSU	ASM Volume Umbillicus (kfvsu)
KFVSD	ASM Volume Background (kfvsd)
KFDX	ASM Exadata interface (kfdx)
KFZP	ASM Password File Layer (kfzp)
KFA	ASM Alias Operations (kfa)

ASM parameters and underscore parameters

Query from X\$ tables that expose underscore parameters. 102 parameters in 12.1.0.1!

```
select a.kspinm "Parameter", a.kspdesc "Description", c.kspstvl "Instance Value"
  from x$kspci a, x$kspcv b, x$kspsv c
 where a.indx = b.indx and a.indx = c.indx
       and kspinm like '%asm%'
 order by a.kspinm;
```

ASM-related acronyms

- **PST** - Partner Status Table. Maintains info on disk-to-diskgroup membership.
- **COD** - Continuing Operation Directory. The COD structure maintains the state of active ASM operations or changes, such as disk or datafile drop/add. The COD log record is either committed or rolled back based on the success of the operation. (source Oracle whitepaper)
- **ACD** - Active Change Directory. The ACD is analogous to a redo log, where changes to the metadata are logged. The ACD log record is used to determine point of recovery in the case of ASM operation failures or instance failures. (source Oracle whitepaper)
- **OSM** Oracle Storage Manager, legacy name, synonymous of ASM
- **CSS** Cluster Synchronization Services. Part of Oracle clusterware, mandatory with ASM even in single instance. CSS is used to heartbeat the health of the ASM instances.
- **RBAL** - Oracle background process. In an ASM instance coordinated rebalancing operations. In a DB instance, opens and mount diskgroups from the local ASM instance.
- **ARBx** - Oracle background processes. In an ASM instance, a slave for rebalancing operations
- **PSPx** - Oracle background processes. In an ASM instance, Process Spawners
- **GMON** - Oracle background processes. In an ASM instance, diskgroup monitor.
- **ASMB** - Oracle background process. In an DB instance, keeps a (bequeath) persistent DB connection to the local ASM instance. Provides hearthbeat and ASM statistics. During a diskgroup rebalancing operation ASM communicates to the DB AU changes via this connection.
- **O00x** - Oracle background processes. Slaves used to connected from the DB to the ASM instance for 'short operations'.

Additional links

ASM utilities: kfed and amdu: [ASM_utilities](#)

Revisions:

First version, Jan 2006, Luca.Canali@cernSPAMNOT.ch

Major additions, Jan 2007, L.C.

Additions and corrections, Nov 2007, L.C.

11gR1 updates, Jun 2008, L.C.

11gR2 updates, Jan 2010, L.C.

12c updates, May 2014, L.C.

This topic: PDBService > ASM_Internals

Topic revision: r48 - 20 Jun 2014 - LucaCanali



Copyright 1999 - 2013 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding TWiki? Send feedback