<https://blogs.oracle.com/database4cn/%E5%A6%82%E4%BD%95%E5%9C%A8asm%E4%B8%8A%E5%AE%9A%E4%BD%8D%E6%95%B0%E6%8D%AE%E5%9D%97>

| Sunday, August 27, 2017

**Share**

**如何在asm上定位数据块**

By: [Bin Mao](https://blogs.oracle.com/author/bin-mao)

我们都知道当db使用传统的文件系统时，定位block是就是通过dba\_extents 中的blockid即可，然后

通过bbed或dd 直接操作对应数据文件的相应block即可，从10g开始，oracle使用ASM来管理磁盘，

因为所有数据文件都在ASM上，所以传统的bbed，dd都不能直接访问新型"文件系统"上的某个数据文件了

当我们需要在特殊场景下修改某个block的内存的时候该怎么办呢？下面给您演示一下：

首先我在这里先普及一下ASM的一点基础知识，需要知道的是ASM在磁盘上存储数据，分配的时候都是AU（allocation unit）为单位的，默认情况AU是1M，所以一个2G的数据文件

需要至少分配2×1024=2048个AU，当然存储这些AU的相关信息（如编号）也是需要存储空间的，这种元数据也是需要占用AU的，所以

实际上从ASM层面来看分配给某个数据文件的au数量要高于它的需求数量。对于ASM来讲，数据库的所有文件，如控制文件，spfile，

数据文件都是作为一种叫文件目录(file directory)类型的数据进行管理的。每一条这种类型的数据就是一个ASM管理的文件，其中

ASM为了管理方便会给每个文件目录分配一个唯一的编号，并且会在第一个文件目录的AU里面为其分配一个4k的block来

存放它分配的AU情况。而这个block编号就是分配给文件目录的编号。因为au默认是1M，所以一个au能分配出来256个block，也就是文件目录

编号前255(oracle 里面编号都是从0开始的)都在第一个AU里面，大于255小于512的都在第一个文件目录第二个au里面某个4kblock里面，

那么如何知道某个数据文件的编号呢，实际上oracle在创建文件的时候就已经告诉我们了。例如我们在往users表空间添加数据文件名为

user2.dbf的文件后，oracle在ASM上创建数据文件的名字类似如下：

  user2.dbf => +DATA/R11204/DATAFILE/USERS.328.944473935

  其中328就是表明这个数据文件的ASM 文件目录的编号。所以我们就通过它来入手。

介绍完如上的知识，我们看看如何定位一个数据文件的某个block：

首先创建一个表，为了好验证，我插入的数据都是a，

create table test\_t ( a number, b varchar2(100));

insert into test\_t values(1,'abcdefg');

insert into test\_t select \* from test\_t;

insert into test\_t select \* from test\_t;

insert into test\_t select \* from test\_t;

commit;

查询这个表的block情况

select \* from dba\_extents where segment\_name='TEST\_T';

MAOB    TEST\_T        TABLE    USERS    0    4    168    65536    8    4

select file\_id,file\_name from dba\_data\_files;

4 +DATA/r11204/datafile/users.300.908780493 <<file\_number=300

 我们可以看到这个表在file\_id=4的数据文件里只分配了一个8个block的extent，第一个blockid是168，

根据oracle ASSM表空间的存储相关的知识，extent0 的前三个block分别是L1,L2,L3的metadata，所以第一个data block是171，

那么我们就找blockid=171这个block的ASM上存储位置：

首先我们根据file\_id=4 blockid=171信息计算出这个数据文件在文件目录里面AU信息和block信息

AU信息：因为oracle数据块默认是8k的，所以171个block会占用多少个1M的au呢？

select  171\*8/1024 from dual

1.3359375

是1.3个Au，所以这个171的block一定放在第二个au里面的某个block上

au里的block信息：

select (171\*8-1\*1024)/8 from dual

43

这个blockid=171的block放在第二个au里面的第43个block位置，

然后我们就需要找到这个数据文件的第二个au相对于磁盘的位置，根据之前介绍的知识，

下我们用kfed来读取磁盘头信息：我们要得到第一个文件目录的au信息

[grid@rac1 disks]$ kfed read /dev/oracleasm/disks/VOL1 |grep f1b1

kfdhdb.f1b1locn:                     10 ; 0x0d4: 0x0000000a  file1的第一个au

磁盘头里面的f1b1locn存放 就是目录文件1的第一个au，之前介绍了，第一个au里面

只能存放文件编号小于256的文件目录，我们的数据文件4的名字是 users.300.908780493

所以这个文件编号是300-256=44，也就是说会放在au=2的第44个block里面

      查看文件目录1里面的au分配信息

    [oracle@rac1 ~]$ kfed read /dev/oracleasm/disks/VOL1 aun=10 blkn=1 |grep au

kfffde[0].xptr.au:                   10 ; 0x4a0: 0x0000000a

kfffde[1].xptr.au:                   61 ; 0x4a8: 0x0000003d   <<第二个au的编号

kfffde[2].xptr.au:           4294967295 ; 0x4b0: 0xffffffff

查看文件目录44的au分配信息

[grid@rac1 ~]$ kfed read /dev/oracleasm/disks/VOL1 aun=61 blkn=44 | grep au

kfffde[0].xptr.au:                 9931 ; 0x4a0: 0x000026cb

kfffde[1].xptr.au:                 9932 ; 0x4a8: 0x000026cc <<file 300的第二个au编号

kfffde[2].xptr.au:                 9933 ; 0x4b0: 0x000026cd

kfffde[3].xptr.au:                 9934 ; 0x4b8: 0x000026ce

 。。。。。。。。。。。。。。

得到file 300的第二个au编号之后，我们就可以直接copy他的信息进行验证

dd if=/dev/oracleasm/disks/VOL1 skip=9932 of=/tmp/users.300 bs=1024k count=1

BBED> set blocksize 8192

BBED> set block 0

BBED-00309: out of range block number (0)  说明blockid是从1开始的，那么block43对应44

BBED> set block 44

BBED> p kcbh

struct kcbh, 20 bytes       @0

   ub1 type\_kcbh            @0        0x06

   ub1 frmt\_kcbh            @1        0xa2

   ub1 spare1\_kcbh          @2        0x00

   ub1 spare2\_kcbh          @3        0x00

   ub4 rdba\_kcbh            @4        0x010000ab  <<file4 block 171

   ub4 bas\_kcbh             @8        0x0080737b

   ub2 wrp\_kcbh             @12       0x0000

   ub1 seq\_kcbh             @14       0x01

   ub1 flg\_kcbh             @15       0x04 (KCBHFCKV)

   ub2 chkval\_kcbh          @16       0xffc6

   ub2 spare3\_kcbh          @18       0x0000

BBED> set offset 8100

    OFFSET             8100

BBED> dump /v

 File: users.300 (1)

 Block: 44      Offsets: 8100 to 8191  Dba:0x0040002C

-------------------------------------------------------

 64656667 2c010202 c1020761 62636465 l defg,......abcde

 66672c01 0202c102 07616263 64656667 l fg,......abcdefg

 2c010202 c1020761 62636465 66672c01 l ,......abcdefg,.

 0202c102 07616263 64656667 2c010202 l .....abcdefg,...

 c1020761 62636465 66672c01 0202c102 l ...abcdefg,.....

其中在定位文件目录300的au编号的时候也可以通过如下简单办法：

select disk\_kffxp, AU\_kffxp, xnum\_kffxp from x$kffxp

where group\_kffxp=1

and number\_kffxp=300 ;

DISK\_KFFXP   AU\_KFFXP XNUM\_KFFXP

---------- ---------- ----------

     0     9931           0

     0     9932           1