

This Wiki entry has been copied and reformatted from <http://blog.monsted.dk/posts/ashift/>

That blog seems to be no longer updated and has some serious formatting issues, so hopefully the author doesn't mind that we're mirroring a copy with better formatting.

As you're probably aware (since you're reading this), some of the new hard drives with "Advanced Format" lie about their sector size and thus the partitioning and/or file systems end up unaligned with the disk and performance suffers. Boo, Western Digital. You guys are worse than Hitler. But, since the WD20EARS was cheap and i didn't think anyone would design something so stupid, i bought it.

Ofcourse, ZFS is smart enough to use the larger sector size through the "ashift" setting, but only if it actually knows the sector size of the drive. This option is set at pool (or rather VDEV) creation and can't be changed.

The easy way around this is to use gtop to prevent the drive from lying.

```
# zpool create tank raidz ada0 ada1 ada2
# zdb | grep ashift
    ashift: 9
```

Oe noes! 2^9 is 512, the wrong sector size. Let's try that again.

```
# zpool destroy tank
# gtop create -S 4096 ada0
# zpool create tank raidz ada0.nop ada1 ada2
# zdb | grep ashift
    ashift: 12
```

Woohoo! 2^{12} is 4096, as it should be. Note that you only need to tag one of the drives. zpool is clever enough to use the largest size of them all. Also note that you need to do this every time you add a new VDEV, otherwise you'll have varying ashift values within your pool. Only problem now is that the drive list is a bit ugly, but that's easily fixed.

```
# zpool status
pool: tank
  state: ONLINE
  scan: none requested
config:
    NAME                STATE          READ  WRITE CKSUM
    tank                 ONLINE         0     0     0
      raidz1-0          ONLINE         0     0     0
        ada0.nop        ONLINE         0     0     0
        ada1            ONLINE         0     0     0
        ada2            ONLINE         0     0     0
# zpool export tank
# gtop destroy ada0.nop
# zpool import tank
# zpool status
pool: tank
```

```
state: ONLINE
scan: none requested
config:
  NAME          STATE      READ WRITE CKSUM
  tank          ONLINE    0     0     0
    raidz1-0    ONLINE    0     0     0
      ada0      ONLINE    0     0     0
      ada1      ONLINE    0     0     0
      ada2      ONLINE    0     0     0
```

There we go. Destroying the gnop node just removes the sector size override, the data is safe and sound. You've now got ashift set right without fiddling with patches and such.

Simple Mirror Creation Example

```
nas4free: ~ # bash
[root@nas4free ~]# gnop create -S 4096 ada0
[root@nas4free ~]# gnop create -S 4096 ada1
[root@nas4free ~]# zpool create dancer mirror ada0.nop ada1.nop
[root@nas4free ~]# zpool status -v
  pool: dancer
  state: ONLINE
  scan: none requested
config:
  NAME          STATE      READ WRITE CKSUM
  dancer        ONLINE    0     0     0
    mirror-0    ONLINE    0     0     0
      ada0.nop  ONLINE    0     0     0
      ada1.nop  ONLINE    0     0     0
errors: No known data errors
[root@nas4free ~]# zpool export dancer
[root@nas4free ~]# gnop destroy ada0.nop
[root@nas4free ~]# gnop destroy ada1.nop
[root@nas4free ~]# zpool import dancer
[root@nas4free ~]# zpool status -v
  pool: dancer
  state: ONLINE
  scan: none requested
config:
  NAME          STATE      READ WRITE CKSUM
  dancer        ONLINE    0     0     0
    mirror-0    ONLINE    0     0     0
      ada0      ONLINE    0     0     0
      ada1      ONLINE    0     0     0
errors: No known data errors
```

From:
<https://www.xigmanas.com/wiki/> - **XigmaNAS**

Permanent link:
<https://www.xigmanas.com/wiki/doku.php?id=zfs:4k>

Last update: **2018/07/08 16:29**

