Pool layouts

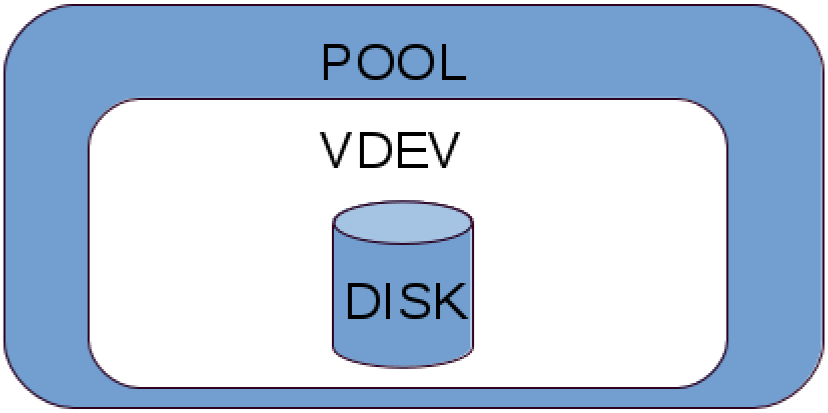
Previous: Hardware and software requirements		Next: ZFS naming conventions

ZFS pool is a collection of disk devices that are grouped together to allow crafting filesystems from it. In that regards it's similar to Linux LVM (Logical Volume Manager). Disks are grouped together in logical devices called vdevs (virtual devices). VDEV determine redundancy level of the pool. A pool can consist of a single vdev or multiple vdevs. vdevs of different layout can be mixed together in one pool, but that goes against best practices and should not be considered. Below are examples of pools layout along with explanation of redundancy level.

For various RAIDZ configurations few things are worth of note. First and most important is that, due to ZFS being able to manage both filesystem and volume management, it offers solution to so called RAID write hole. A write hole can occur when, during the write of data block is interrupted by a system crash (most notably a power failure). It can result in parity data being insufficient for data recovery. Battery powered RAID controllers can mitigate this problem. ZFS takes another approach. It write variable length data stripes and makes this operation atomic. Due to Copy on Write and transactional nature, filesystem is always in consistent state - block will be committed or rolled back.

Single disk

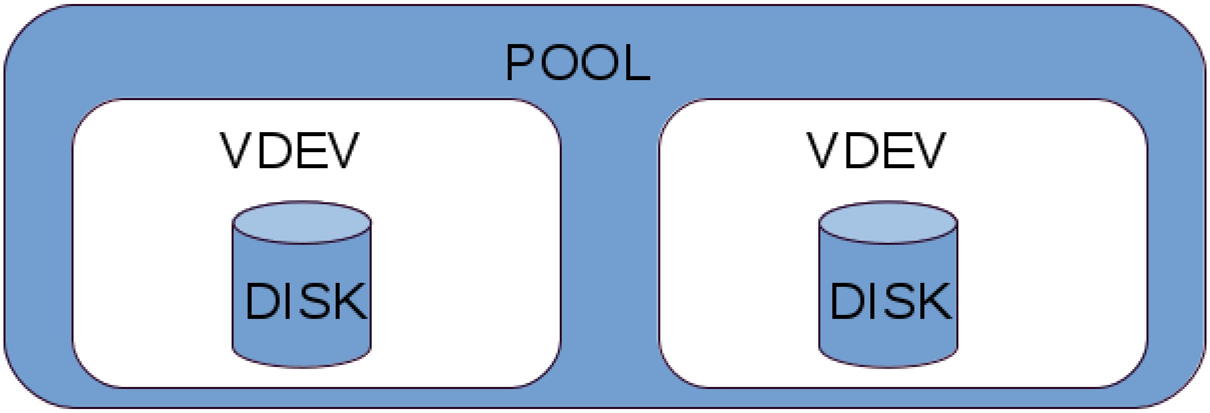
Pool always consists of vdevs. A single disk pool contains a single vdev that contains a single disk.



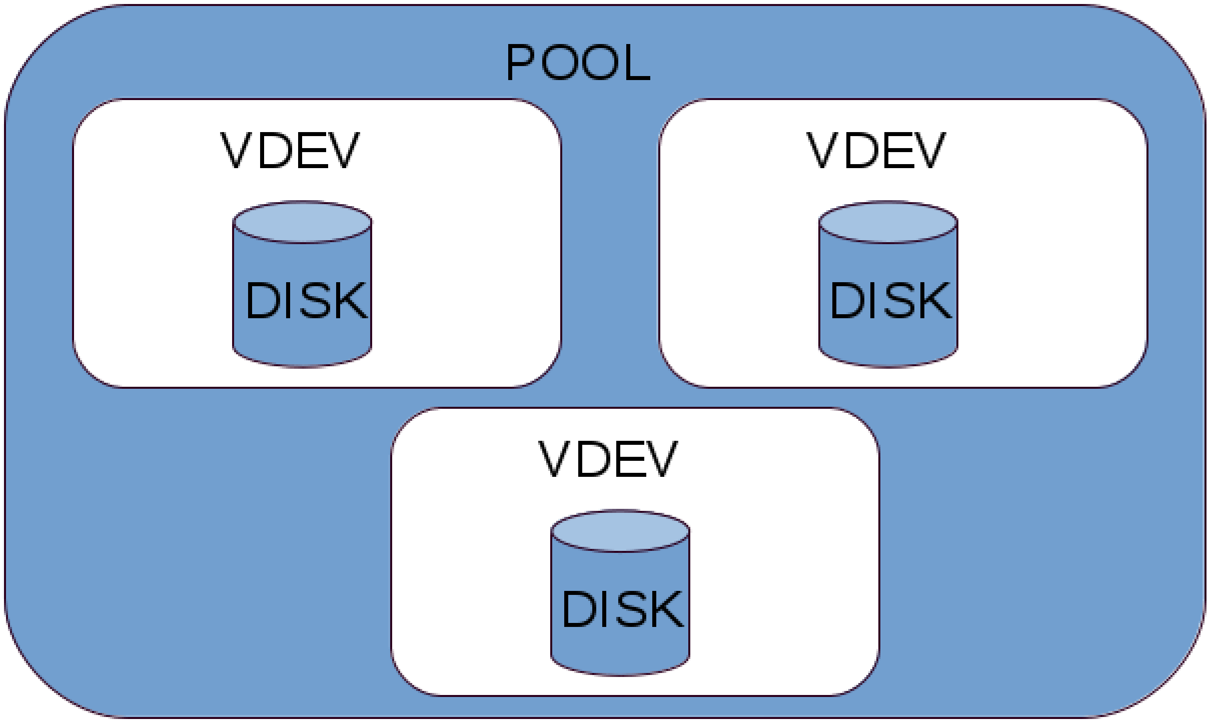
This layout does not allow for any level of redundancy.

Two, three, four disks - stripe,

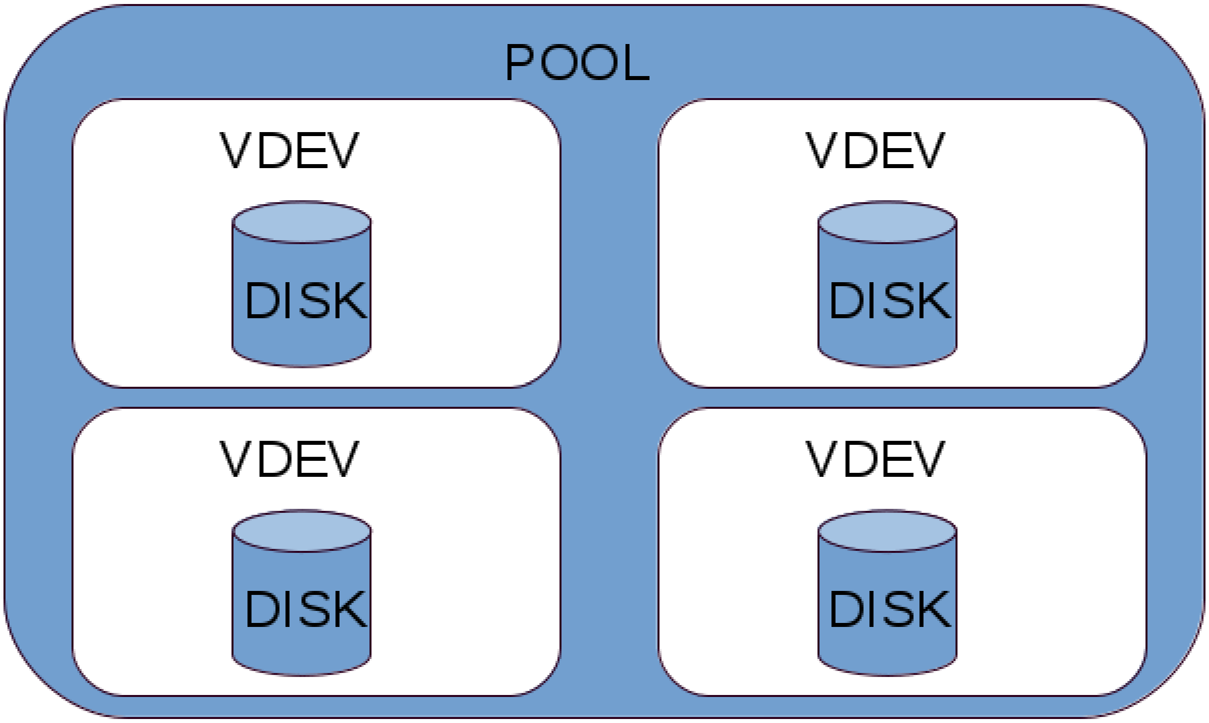
A two disks pool with two vdevs. No redundancy. Data is written to vdevs in round robin fasion. Loosing one disk means loosing whole pool.



Three disks stripe. No redundancy.



Four disks stripe. No redundancy.

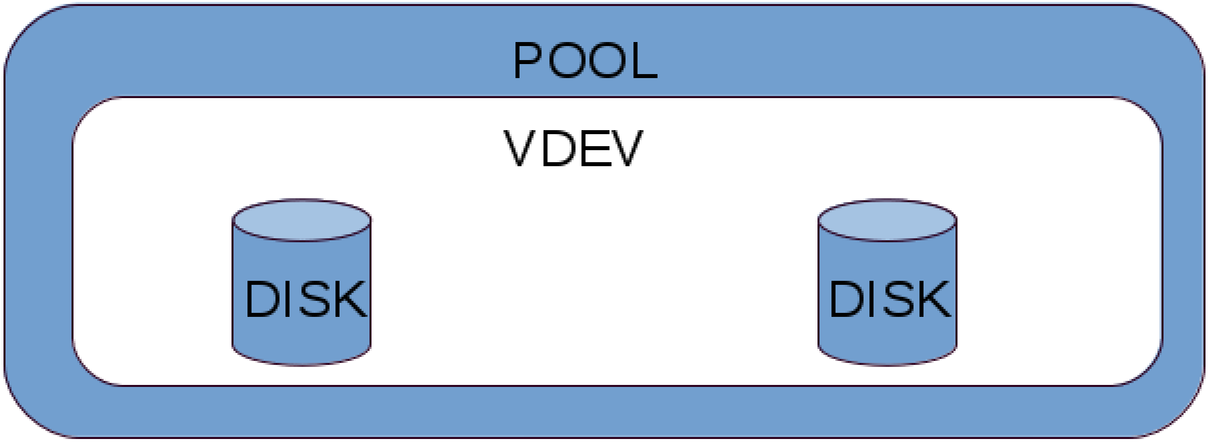


This can be continued basically without the end. As noted earlier, this layout gives no redundancy at all.

Please note, that a stripe pool can be easily turned into a mirrored pool. This is exampled in chapter explaining management of ZFS pools.

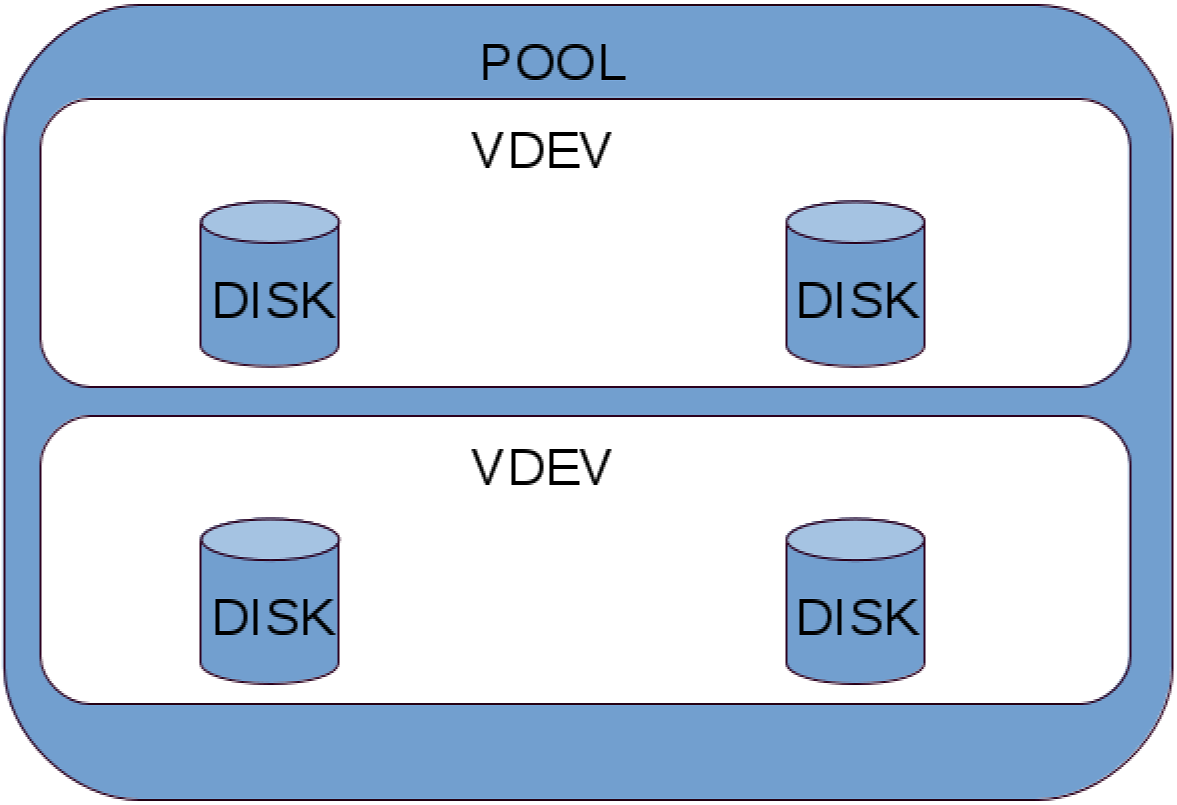
Two disks, single vdev - mirror

This is the simplest redundancy level, using only two disks. Disks are grouped together in a single vdev. Data are mirrored on both drives. This pool can loose one drive and still be operational. A vdev capacity is a single disk capacity, as all data is mirrored.



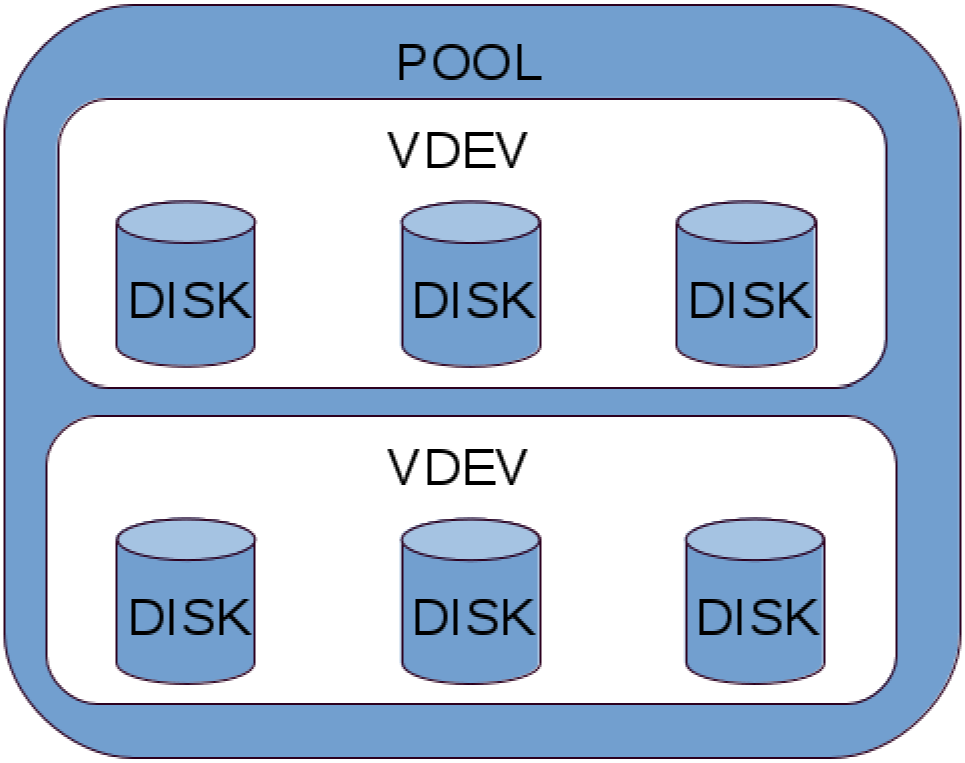
Four disks, two vdevs - mirrors

This is a stripe of two mirrors - two vdevs containing two disks each. Mirroring of data is done between disks in the same vdev. Pool can contain many mirror vdevs, pools consisting of hundreds of mirror vdevs have been observed in real life installations. A pool capacity can be grown by adding a new vdev to it. A vdev capacity is a single disk capacity, as all data is mirrored. This is a rough equivalent of RAID1+0, which is a stripe of mirrors.



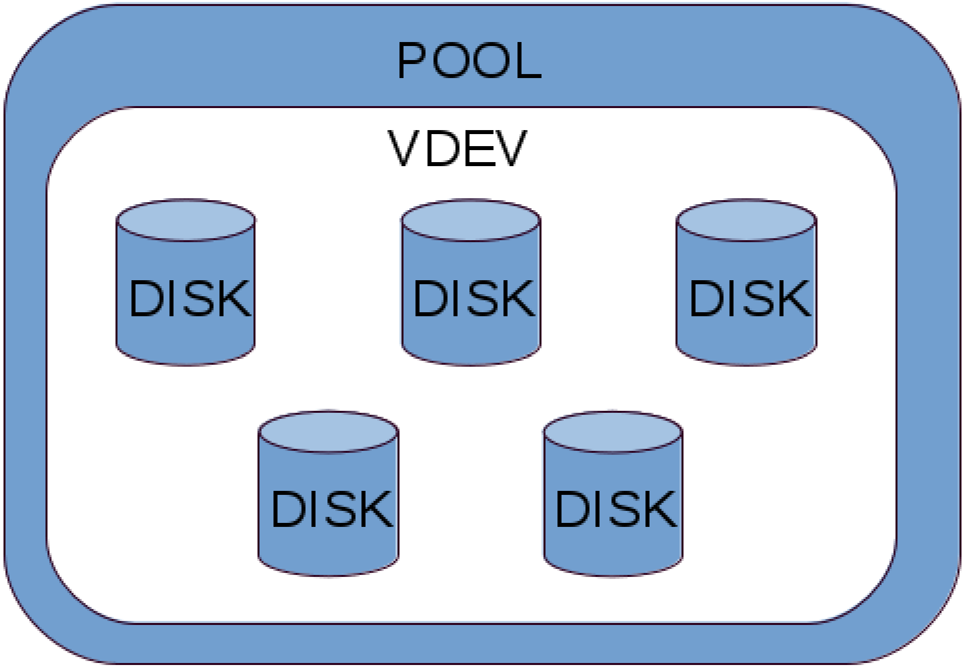
Six disks, two vdevs - triple mirror

The mirror redundancy can be hightened by creating a mirror that consists of three disks. As previously a pool with triple mirror vdevs can be as big as to contain hundreds of vdevs. A regular mirro vdev can be turned into a triple mirror and vice versa by use of zpool command. A vdev capacity is a single disk capacity, as all data is mirrored.



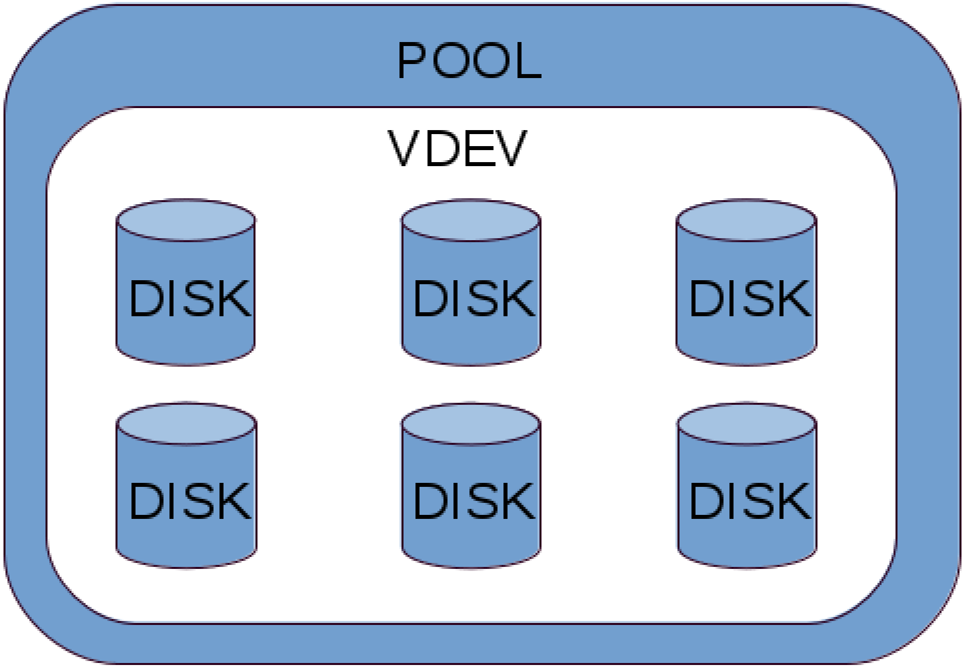
Five disks, single vdev - raidz

A raidz vdev is a rough equivalent of RAID5 in traditional RAIDs. In raidz one disk is a parity disk. A vdev can loose as many as one disk with still being operational. In this layout the overall vdev capacity equals all disks summed capacity minus one disk (parity disk). A pool capacity is a sum of all its vdev capacities.



Six disks, single vdev - raidz2

A raidz2 is a rough equivalent of RAID6 in traditional RAIDs. It contains two pariti disks, so a single vdev capacity is a sum of all disks capacities minus two disks.



In addition to layouts above other types of media can be used in so called hybrid pools. Additionally to having platter disks in a pool storage administrator can add SSD and DRAM based disks as Cache and ZFS Intent Log devices.

Adding SSD disk as a cache device creates a Layer 2 Cache (thus the name l2arc). It is a layer living between in RAM cache (which is most costly but fastest) and platter disks storage, which is least costly but slowest medium in the pool. In ideal world hottest data live in RAM, warm data live on SSD, cold data sit on disks.

Adding SSD (or better battery powered DRAM devices) as ZFS Intent Log allows for grouping of data blocks to be committed to disks in large groups and flushing them in one large I/O. That not only takes off load from disk writes but also helps preventing pool fragmentation.

Previous: Hardware and software requirements		**Next: ZFS naming conventions**