

Storage Alignment and VMFS Block Sizes

Introduction

There are two common questions about storage and VMware Infrastructure that appear on a weekly basis:

1. If my VMFS has a block size of 1MB, does that mean each block read from the guest read 1MB from the SAN LUN?
2. Should I align my VMFS; should I align my Guest partitions?

VMware have written a white paper on this (see Resources), but the key information from that paper has been lifted into here - as well as referring to a great doc from EMC (see Resources) on ESX optimization.

Intended Audience

VMware Certified Professionals (VCPs) and storage experts when understanding and building VMware Infrastructure on block storage (FC).

Outline

1. Aligning Guest -> VMFS -> LUN
2. Understanding VMFS block size

1. Aligning Guest + VMFS + LUN

Understanding alignment

In a SAN environment, the smallest hardware unit used by a SAN storage array to build a LUN out of multiple physical disks is called a chunk or a stripe. To optimize I/O, chunks are usually much larger than sectors. Thus a SCSI I/O request that intends to read a sector in reality reads one chunk.

On top of this, in a Windows environment NTFS is formatted in blocks ranging from 1MB to 8MB.

The file system used by the guest operating system optimizes I/O by grouping sectors into so-called clusters (allocation units).

Figure 1 shows that an unaligned structure may cause many additional I/O operations when only one cluster is ready by the guest operating system.

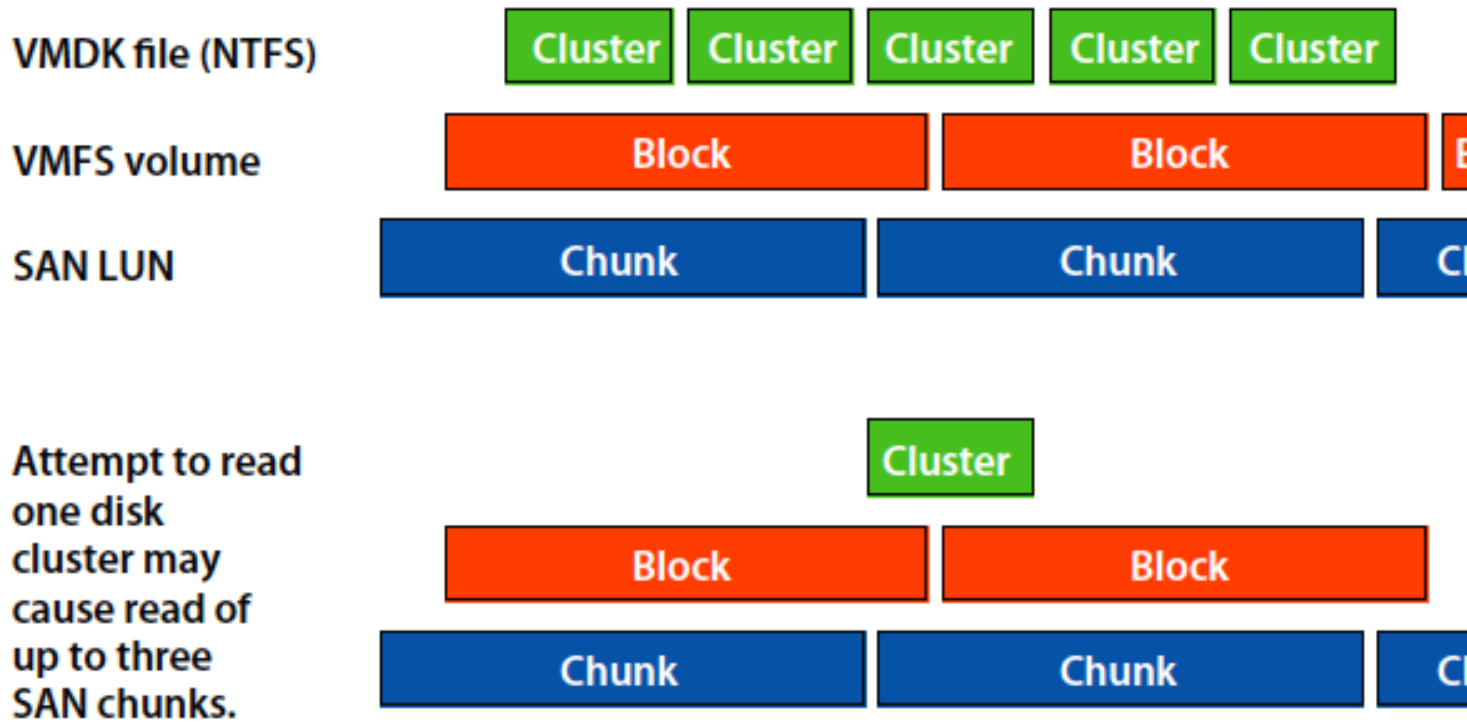


Figure 1: Unaligned partitions result in additional I/O

Figure 2 shows I/O improvements on a properly aligned Windows NTFS volume in a VMDK on a SAN LUN.

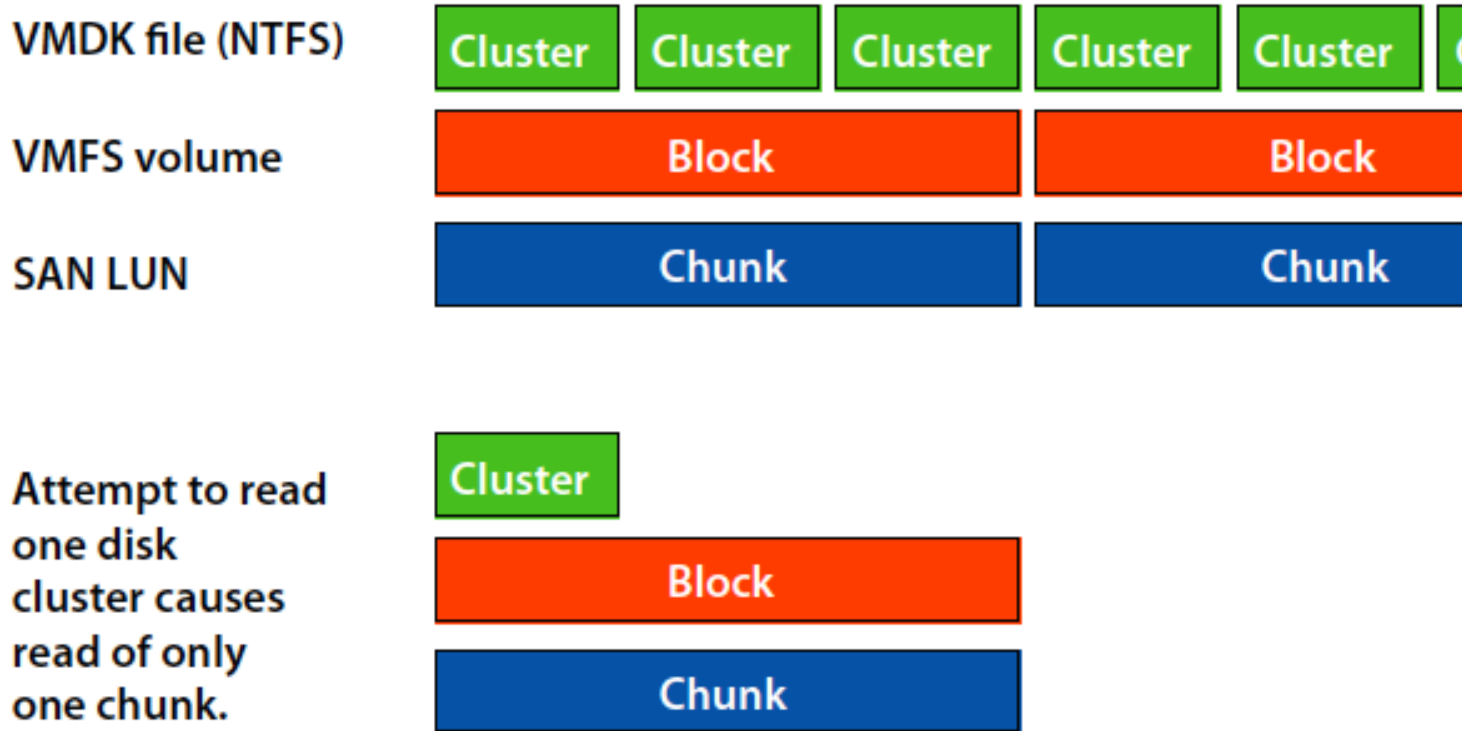


Figure 2 Aligned partitions reduce I/O

Also, operating systems on x86 architectures create partitions with a master boot record (MBR) that consumes 63 sectors. This is due to legacy BIOS code from the PC that used cylinder, head, and sector addressing instead of logical block addressing (LBA). Without LBA, the first track is reserved for the boot code, and the first partition starts at cylinder 0, head 1, and sector 1. This is LBA 63 and is therefore unaligned.

An unaligned partition results in a track crossing and an additional I/O, incurring a penalty on latency and throughput. The additional I/O (especially if small) can impact system resources significantly on some host types. An aligned partitions ensures that the single I/O is serviced by a single device, eliminating the additional I/O and resulting in overall performance improvement.

During ESX Server 3.0 installation, however, the installation procedure creates a default VMware VMFS partition that is unaligned. Administrators should consider manually aligning the default VMware VMFS partition with fdisk before use.

Should I bother aligning Guest -> VMFS -> LUN partitions?

Weigh up the cost and the benefit -

- Cost - administration time in setup, which is running a few more commands during the one-off build - ie. minor

- Benefit - reduced latency and increased throughput, esp. for sequential., but for random workloads the improvement was negligible.

The VMware white paper (see Resources) concluded:

Track alignment for both physical machines and VMware VMFS partitions yields I/O performance improvements such as reduced latency and increased throughput. Creating VMware VMFS partitions using the VI Client results in a 64KB-aligned partition table and provides the foundation for a best practices storage layout.

And the white paper also provides their test results which confirms that for sequential there is a good improvement, but for random workloads it is negligible. Neither of this benchmark-like tests are like a "normal file server" workload, so the best advice is to:

1. Test your own workloads on unaligned and aligned partitions - what improvement can you measure? That should be your guide.
2. Work out how costly it is to align partitions at your site - if it is minimal, with high success rates, then the burden of justifying alignment should be less (ie. JFDI).

If you decide to do the alignment, then you should align the VMFS to the LUN when you create the VMFS, then align the Guest to the VMFS when you create the VMDKs.

Align VMFS to LUN

To check that your existing partitions are aligned, issue the command:

```
fdisk -lu
```

The output is similar to:

```
Device boot Start End Blocks Id System
/dev/sdj1 128 167766794 83883333+ fb Unknown
```

Aligned partitions start at 128. If the Start value is 63 (the default), the partition is not aligned.

If you choose not to use the VI Client and create partitions with vmkfstools, or if you want to align the default installation partition before use, take the following steps to use fdisk to align a partition manually from the ESX Server service console:

1. Enter `fdisk /dev/sd<x>` where `<x>` is the device suffix.
2. Determine if any VMware VMFS partitions already exist. VMware VMFS partitions are identified by a partition system ID of `fb`. Type `d` to delete to delete these partitions.

Note: This destroys all data currently residing on the VMware VMFS partitions you delete. Ensure you back up this data first if you need it.

3. Type n to create a new partition.
4. Type p to create a primary partition.
5. Type 1 to create partition No. 1.
6. Select the defaults to use the complete disk.
7. Type t to set the partitions system ID.
8. Type fb to set the partition system ID to fb (VMware VMFS volume).
9. Type x to go into expert mode.
10. Type b to adjust the starting block number.
11. Type 1 to choose partition 1.
12. Type 128 to set it to 128 (the arrays stripe element size).
13. Type w to write label and partition information to disk.

To script this for a kickstart or a post installation script, use the following method:

Create an answer file using nano, vi or a text editor that can produce unix formatted text. Enter the following in the text file:

```
n
p
1

t
fb
x
b
1
128
w
```

Then you can put the following into a script:

```
fdisk /dev/sdb < fdisk.txt
```

NOTE: This script example assumes that the disk to be partitioned and formatted is located at /dev/sdb. Caution should be used when automating any type of disk partitioning and formatting because you could accidentally erase partitions and all of the data (VMs) contained on it.

Align Guest to VMFS

Once you have aligned your VMware VMFS partitions, you also need to align the data file system partitions within your virtual machines.

Note: Aligning the boot disk in the virtual machine is neither recommended nor required. Align only the data disks in the virtual machine.

The following sections discuss how to align guest operating system partitions in Linux and Windows environments.

Linux

A best practice for Linux physical as well as virtual machines is to align file system partitions using fdisk. Use the fdisk procedure in the previous section of this paper, and instead of setting the partition system id to fb, set it to 83 (Linux) or other appropriate partition system ID.

Also, using a larger allocation unit (also called cluster size) improves performance of NTFS volumes.

Note: The previous version of diskpart.exe was diskpar.exe. The main difference is that diskpart.exe creates partitions in sectors (512 bytes) while diskpart uses KB.

This example assumes the new disk is disk 0. To create an aligned partition, take the following steps:

Windows

For Windows virtual machines there is an additional layer of NTFS partitioning that requires alignment using the diskpart.exe tool from the Microsoft Download Center or MSDN.

1. Open a command prompt and enter diskpart
2. Enter select disk 1
3. Enter create partition primary align=64
4. Exit

5. Format the partition with a 32K allocation unit size

Just like using a script to run fdisk in the ESX console, you can use a script to automate the creation of partitions in Windows using diskpart.exe. You will need to create and answer file (diskanswer.txt) containing the following lines:

```
list disk
select disk 1
create partition primary align=64
select partition 1
remove noerr
assign letter=E noerr
```

Then add the following command to a script:

```
diskpart.exe /s diskanswer.txt
```

Finally, add the format command to the script:

```
format.exe E: /FS:NTFS /A:32K
```

2. Understanding VMFS block size

The default block size for a VMFS, set when you create the VMFS on a new LUN, determines the maximum possible size of a VMDK (1MB = max 256GB vmdk) and is the unit used when allocating/deallocating files such as when extending VMDKs like REDO logs.

The VMFS block size is not used for guest read/writes.

When your Linux/Windows/Solaris guest reads/writes data, it does so in blocks and these blocks are usually around 64KB. For example, when your guest issues a SCSI read, then **nothing happens to the VMFS at all** - the read is "passed through" by the vmkernel straight to the SAN LUN.

VMFS does not get in the way of your guest read/writes, it does not do any kind of caching, the VMFS block size is irrelevant for guest I/O.

Where the VMFS block size is important is when fixing the maximum VMDK size:

VMFS Block Size	Maximum VMDK Size
1 Mb	256Gb
2Mb	512Gb
4Mb	1024Gb
8Mb	2048Gb

Resources

- This doc on the web @ [Storage block size and alignment](#)
- VMware White paper: [Recommendations for Aligning VMFS Partitions \(pdf\)](#)
- EMC White Paper: [VMware ESX Server Using EMC CLARiiON Storage Systems Solutions Guide](#)
- EMC White paper: [VMware ESX Server Using EMC Symmetrix Storage Systems Solutions Guide](#)
- EMC Proven Practice on VIOPS: [VMware ESX Server Optimization with EMC Celerra](#)

Author

[Steve Chambers](#)

Contributor

[Dave Convery, VCDX, VMware vExpert](#)

Disclaimer

You use this proven practice at your discretion. VMware, VIRTERRA and the author do not guarantee any results from the use of this proven practice. This proven practice is provided on an as-is basis and is for demonstration purposes only.