I have set up a new FreeNAS for media-files (large musicfiles of ~10MB or ~40MB). Poolperformance of 8disk-Z2 is very limited (w400/r225 MB/s). Perhaps you have suggestions?

Set-up:

- SuperMicro X8DTL-iF
- Dual Intel Xeon X5670 CPU's (3Ghz - 6c/12t)
- 4x 8GB ECC-RAM
- NIC: 10Gb Emulex dual-port SFP+
- IT-flashed Dell H310 controller
- Pool:
  - Pool of 8x 8TB WD-Red hard-disks (Z2-VDev)
  - Disks: 8TB WD-Red 'EMAZ'; 5400rpm; 256MB Cache (512e)
- .
- Latest version of FreeNAS (11.2 U5)
- (connected Workstation includes dual Xeons e5-2670, runs Windows10)(Samba/SMB-share via that 10Gb-interface).

Test-results:
(new/empty pool/disks; ran writes/reads between Pool and RAM-disk on Workstation)
(20GB-testruns of avg 10MB-files and runs of avg 40MB-files)(showing avg speeds below)

r/w speeds of the Pool (MB/s):

	r/w to/from RAM-disk (copy-paste)	
	Write	Read
(Single-disk max seq speed)	(200)	(200)
Calculated 8disk-Z2 poolspeed	1100	1500
Vs.		
Tested 8disk-Z2 poolspeed	400	225
More tested poolspeeds:		
1disk ('stripe')	200	160
4disk-Z1	325	225
2x3disk-Z1	375	225
8disk-Z2	400	225
2x4disk-Z1	400	225
8disk-stripe	400	225

	write/read within Pool (dataset to dataset)
8disk-Z2	80

How the speed is calculated:

- single-disk seq speed = 200MB/s

- avg seektime = 8,4 millisec
- avg file-size = 25MB
- => effective r/w speed per disk = 1/(25/200 + 0,0084)x25=187 MB/s
- => writing: 8-2=6 disks => ~1100 MB/s
- => reading: 8 disks => ~1500 MB/s.

Checking possible IOPS-limitation:

- Single-disk IOPS (data from online tests):
  - write: 500-600 (random 4k/qd1-qd32)
  - read: 200-600 (random 4k/qd1-qd32)
- So Pool/Vdev-IOPS:
  - same as for 1 single disk:
    - so a limiting factor when ZFS does purely random r/w (seek/write block by block) (being when pool gets too full)
    - but normally, with enough free pool-space, ZFS will do seek/write for series of files (e.g 10 files in 1 IO-action)
    - So IOPS not a limitation during these (100% free pool) testings.

<u>Settings used:</u>

- ashift on 12 (4k); just to be sure
- TLER is not on 'on' with these EMAZ-types (can via script be set on 'on' for reducing seektime in case of errors)
- max Recordsize = 1MB
- no ZIL/sync disabled (not to give extra write-burden on the disks)(and there will be no sync write anyhow)
- prefetch off (no extra random reads)
- Atime off (no need for writing/logging last time file was read)
- compression on (dual Xeon CPU's have no sweat doing this)(+ relevant for the large blocks)
- deduplication off (has heavy burden on CPU's, might slow stuff down, but not when off)
- transactiongroup size: kept on default (being 4GB for my 32GB RAM)
- ARC: trying not to have more then marginal reads from file-cache/ARC
  - I set the minimum for Metadata-cache on 28GB (RAM being 32, OS needing say 1-2 GB there would be say 2 GB for file-cache)
  - I put approx 500GB of data on the Pool, and I used 20GB-testbatches, so only small parts of that testbatch would be in cache (if any); which is the idea: we want to see the performance of the disk-pool in terms of reading/writing
- L2ARC; none mounted
- .
- Created Datasets with Windows-type share; Samba/SMB-share.
- Set the adapter (linked to the 10Gb-NIC on the NAS) to 9000Bytes/ Jumbo-frames.
- Created network-connection/folder in Explorer on Windows10-Workstation.

<u>Tested varying settings:</u>
(destroyed the datasets, created new all the time)

- Prefetch on and off: had no effect at all
- Tested various record/blocksizes (64k, 128k, 1MB, 4MB); made no difference at all (only 16k and 4k slowed down speed)
- Sync on 'standard': made no difference at all
- Atime on; no difference
- Setting 'max_pending' Tunable for max queue-depth to 8x32=256 (WD-Reds giving max read-IOPS=600 at qd32; IOPS=200 at qd1): no effect
- Used 'cut-paste' in stead of 'copy-paste': gave ~30% slower speeds
- Switched disk-controller to different PCIE-slot (using x8PCIEv2 in stead of x4PCIEv2)(pulled out other cards to have all lanes): no effect
- Did various settings/tests to align the record/block-size with the 4k-blocksize of the disks (avoiding possible heavy r/w-effect due to 'read-modify-write activities')(file-blocks of 512 or other size being adapted to 4k-size on disk):
    - Note:
        - Controller is Dell H310 having chip SAS2008; no 4kN-function, only working with 512B-blocks;
        - disks are type '512e' (as usual now)(physical 4k-blocks; emulating 512-blocks)
        - Normally 8x 512B file-blocks are read, bundled and written to disk as 1x 4kB-block
        - When the file-blocks are not 512B, the disk goes into 'read-modify-write' mode (reading 4k, modifying to 512B, writing 4k-block)
        - I got the impression this type of workload could cause that '75% extra workload/processtime', so did several tests.
    - Tests done:
        - set blocksize to 1MB (later 4MB)
        - set up Pool with 6disk-Z2 Vdev (4 disks doing the data-writes)(4x 4k ligning with the 1MB (and 4MB) size)
        - => no effect at all
        - also switched off the compression (this making all file-blocks un-aligned, right?)
        - => no effect at all.

<u>Observations:</u>

- Getting 400/225, with seq speed being 200 etc, so calculated speed being 4-5 times higher, we miss say 75% of speed;
    - CPU's and RAM are not really active (CPU's 10-15%, RAM still free space)
    - So the process is disks-bound
    - So disks are doing something else that consumes 75% of the time? What?
- When using 1 disk as pool, we get exactly the specified speed for writes (and only little less for reads)
    - Clearly having the number of 1 or just having 1 disk there is no 'slow-down'/extra –process?
    - The more disks that are used, the more intense the 'slow-down-effect' is
- When having 2 Vdevs and even when having 8 disks in stripe, we get same speed;
    - so clearly there is no shortage of IOPS? That is not cause of slow speeds?

    - so the disks are doing (quite) some more then just seek and write/read?
- Using much smaller or larger record/blocksizes does not effect the r/w-speeds at all (only when using ridiculously low 16k and 4k blocks speeds is slower)
    - So the process is running on same (small?) blocksize all the time, setting does not matter?
- Larger/40MB-files run at 50% faster r/w-speeds then 10MB-files:
    - So the system is not sequencing series of files (1 seek; many files-write/read); works file per file? Although pool is 100% free?
    - There is a certain processing that has to be done per file? (not per block)
- Reads are about 50% slower then writes. But when reading it uses all 8 disks; when writing 8 – 2 = 6 disks.
    - So reads should be faster, right? But are clearly slower (r 225 vs w 400).
    - Reads suffer the most from 1 cause that writes also suffer from; or reads suffer from 2 causes?

<u>Questions:</u>

- In my view the 400/225 w/r-speeds for the 8disk-Z2 pool (running these large 10/40MB-files) are very low; correct?
- What can I adjust to get a better speed?
- How can I see what the disks are doing, during these r/w tests?

Thanks in advance for any suggestions.