ZFS Dedupliciation

For deduplication, it is recommended to have L2ARC cache size of 2-5GB per 1TB of disk.

- For every TB of pool data, you should expect 5 GB of dedup table data, assuming an average block size of 64K.
- This means you should plan for at least 20GB of system RAM per TB of pool data, if you want to keep the dedup table in RAM, plus any extra memory for other metadata, plus an extra GB for the OS.

ZFS dedupe (again): Is memory usage dependent on physical (deduped, compressed) data stored or on logical used?

THINK TWICE !

Never ever turn on deduplication for whole pool. It is not possible to turn it off without sending whole pool to another zfs and receiving it back. Also it is best to have plenty of RAM to fit all DDT into RAM, not SSD/NVMe.

Huge CPU usage by over 96 ZFS kernel threads noticed with open-zfs v8.0.6 (ZFS On Linux), when some big parts of data deleted (auto snapshot rotation). It is connected with deduplication enabled and causes system to almost freeze because of high CPU usage!

WARNING!

Issue when deleting large portion of data and deduplication enabled. ZFS driver creates 96 `z_fr_iss` threads. Load average of system goes immediatelly to 100 (soft watchdog can be set to reboot system when LA is too high). These threads kills CPU and IO.

- Large Deletes & Memory Consumption
- Reduce ZFS related processes/tasks
- High CPU usage by "z_fr_iss" after deleting large files
- rm / remove / delete a large file causes high load and irresponsiveness

Turn on

Enable deduplication:

```
zfs set dedup=on tank/data
```

Turn off

Once all deduped datasets are destroyed the dedup table will be removed and the performance

impact is cleared. NOTE: after removal data set with dedup enabled, it takes some time until ZFS removes all DDT entries internally.

status

List dedup flags from all volumes:

```
zfs get dedup | egrep '(on|off)'
```

```
zpool list rpool

NAME    SIZE  ALLOC   FREE  CKPOINT  EXPANDSZ   FRAG    CAP  DEDUP    HEALTH
ALTROOT
rpool   928G   888G  40,0G        -         -    25%    95%  1.19x    ONLINE
-
```

```
zpool status -D rpool
zdb -DD rpool
```

```
DDT-sha256-zap-duplicate: 550683 entries, size 474 on disk, 153 in core
DDT-sha256-zap-unique: 3204889 entries, size 505 on disk, 163 in core

DDT histogram (aggregated over all DDTs):

bucket              allocated                       referenced
______   ______________________________   ______________________________
refcnt   blocks   LSIZE   PSIZE   DSIZE   blocks   LSIZE   PSIZE   DSIZE
------   ------   -----   -----   -----   ------   -----   -----   -----
     1    3.06M    341G    324G    325G    3.06M    341G    324G    325G
     2     428K   44.2G   41.9G   42.1G     963K   98.7G   93.7G   94.0G
     4    84.9K   6.34G   5.40G   5.46G     384K   27.0G   22.7G   22.9G
     8    16.7K    245M    107M    137M     173K   2.52G   1.09G   1.40G
    16    7.61K    144M   79.0M   88.9M     146K   2.79G   1.51G   1.69G
    32      564   8.93M   3.35M   4.02M    22.8K    382M    146M    174M
    64      110   1.32M    588K    776K    9.46K    113M   51.6M   67.3M
   128       52   1.07M    670K    748K    8.78K    183M    114M    127M
   256       37    947K    576K    652K    13.7K    389M    239M    266M
   512        4   10.5K   10.5K     16K    3.17K   8.15M   8.15M   12.7M
    1K        6   43.5K     15K     28K    8.34K   60.6M   21.6M   38.8M
    2K        1   36.5K      8K      8K    2.08K   75.9M   16.6M   16.6M
 Total    3.58M    392G    372G    373G    4.75M    473G    444G    446G

dedup = 1.20, compress = 1.07, copies = 1.00, dedup * compress / copies =
1.27
```

Where DDT table memory usage can be calculated:

- `echo '(550683 * 153 + 3204889 * 163) /1024/1024' | bc` is 578 MB used memory
- `echo '(550683 * 474 + 3204889 * 505) /1024/1024' | bc` is 1792 MB used on disk

SIZES:

- DSIZE: (On Disk size) On pool there is 446GB of data stored on 373GB of disk (446 / 373 = 1,195 dedup ratio).
- LSIZE: (logical - in memory)
- PSIZE: (physical size) size required to store all data and DSIZE