

# BTRFS vs ZFS

My target is to:

- protect private files (Photos, documents) (of course I have scheduled backups to external device). On ZFS I can create subvolume with copies=2. On BTRFS with single disc it is not possible.
- reduce e-mail disc space usage (compression, deduplication). On both I can enable compression with similar results. Deduplication works better on BTRFS, because I can specify small block size 4kB. Also is performed by external tool and it is not consuming RAM all the time, like dedup tables in ZFS.
- deduplicate mp3 files grouped in folders

With single disc setup, there is no big difference between features provided by both filesystems.

Things to consider:

- BTRFS is integrated by default, ZFS-fuse is old, ZFS native is provided as DKMS but can cause problem if kernel will be upgraded
- BTRFS lacks of data redundancy (ZFS copies=2), but I can live without it because I have external backup.
- BTRFS deduplication looks better (no constant RAM consumption)
- ZFS offers better tooling
- BTRFS support attributes (like +c = compress)
- user quota ???

## Features

Feature	ZFS	BTRFS	remarks
License	CDDL	GPL	
Copy-on-write	Yes	Yes	
Checksums	Yes	Yes	ZFS: 256bit fletcher2 or SHA256; BTRFS: CRC32c
Encryption	No	No	workaround with encryption of underlying devices
Online defragmentation		Yes	
Online growing / shrinking		Yes	
Online block dev add/rem	Yes(-)	Yes	ZFS: only mirrored devices can be removed
Online deduplication	block	No	btrfs: planned, external tool available
Online compression	Yes	LZO / ZLIB	similar compression level
Subvolumes	Yes	Yes	
Subvolumes inside subvol.	Yes	Yes	
Limit subvolume size	Yes	Yes	ZFS: quota, BTRFS: quatagroup
Linux root filesystem	?	Yes	
Striping			
Mirroring			
Redundancy on single disc	Yes	metadata	ZFS: copies=n param
Online scrubbing	Yes	Yes	scrub - scan and check data integrity

Feature	ZFS	BTRFS	remarks
In place ext2/3/4 conversion	No	Yes	BTRFS: with rollback possible! DO NOT USE, it is not well tested and causes errors!
maturity	Yes	Linux 3.10	
GRUB loader support		Grub2	
Snapshots	Yes	Yes	
Performance	fuse	kernel	fuse is anymore bottleneck with moder kernels
<b>FS Overhead</b>			
<b>EXT4</b>	<b>BTRFS</b>	<b>ZFS</b>	
270 371	269 540		

<http://richardhartmann.de/blog/posts/2012/02/RAID-sucks/>

[http://docs.oracle.com/cd/E37670\\_01/E37355/html/ol\\_use\\_case1\\_btrfs.html](http://docs.oracle.com/cd/E37670_01/E37355/html/ol_use_case1_btrfs.html)

Check and compare:

- dedup: <https://btrfs.wiki.kernel.org/index.php/Deduplication>
- single file compression (btrfs: chattr +c ...)
- online convert from RAID0 to RAID1
- direct mounting subvolumes (btrfs: by subvol path or subvol ID)
- file snapshot (btrfs: cp -reflink /mnt/sv1/test1 /mnt/sv1/test3)
- ZFS: online defrag (BTRFS: mount -o autodefrag)
- btrfs: convert from ext4 (btrfs-convert /dev/sdb1) and rollback
- btrfs as root
- Locate modified files from last snapshot (btrfs subvolume find-new)
- SSD upport (trim/discard)

## ZFS

- (-) compression on/off per subvolume - no support for +c attribute
- (+) All features are controlled by subvolumes (datasets) attributes (compression, deduplication)
- (-) CDDL license is not compatible with Linux kernel. Cannot be distributed as derived work.
- (-) Only mirrored block devices can be removed. Non-redundant and raidz devices cannot be removed from a pool.
- (-) Deduplication on block level, cannot find shifted blocks.

## BTRFS

- (+) compression on during mount, but also using attribs
- (?) deduplication not implemented. External tool available: [\[https://github.com/markfasheh/duperemove\]](https://github.com/markfasheh/duperemove) External tool can be also a benefit. To reduce CPU and memory you can start deduplication process when you want, and only on selected files/dirs.
- (-) All features controlled by mount options
- (+) Every block device can be removed, data are automatically copied (balanced) to free space on other devices
- (+) Metadata are automatically duplicated between block devices, data are striped

From:  
<https://niziak.spox.org/wiki/> - **niziak.spox.org**

Permanent link:  
[https://niziak.spox.org/wiki/linux:fs:btrfs\\_vs\\_zfs](https://niziak.spox.org/wiki/linux:fs:btrfs_vs_zfs)

Last update: **2016/04/21 13:49**

