# BTRFS – description

Btrfs is a file system designed to handle large amounts of data. It is capable of organizing volumes of up to 16 EB and supports a maximum file size of 16 EB. To ensure a consistent file system state, it uses copy-on-write, meaning it writes data to new locations and then updates the entry in the file system trees, so the changes only become visible once they have been completed. As a comparison: ext4 (which is the newest update of the ext-family of file systems traditionally used on Linux) supports 1 EB volumes (1/16) and 16 TB file sizes (1/2^20). And ext4 is a journaling file system. It records changes to be made before making them so the system state can be restored in the event of a crash.

By now this new file system is bundled in all major Linux distributions (since Debian 6.0, Ubuntu 10.10 and Fedora 13. Fedora plans to eventually use btrfs as their default file system.

## Design

Btrfs is structured as several layers of B-trees, all using the sameimplementation. The trees store generic items sorted on a 136-bit key. The first 64 bits of the key are a unique object id. The middle 8 bits are an item type field; its use is hardwired into code as an item filter in tree lookups. Objects can have multiple items of multiple types. The remaining right-hand 64 bits are used in type-specific ways. Therefore items for the same object end up adjacent to each other in the tree, ordered by type. By choosing certain right-hand key values, objects can further put items of the same type in a particular order**.**

Interior tree nodes are simply flat lists of key-pointer pairs, where the pointer is the logical block number of a child node. Leaf nodes contain item keys packed into the front of the node and item data packed into the end, with the two growing toward each other as the leaf fills up.

## New features

The original announcement listed 12 features as implemented or to do:

* Extent based file storage (2^64 max file size)
* Space efficient packing of small files
* Space efficient indexed directories
* Dynamic inode allocation
* Writable snapshots
* Subvolumes (separate internal filesystem roots)
* Object level mirroring and striping
* Checksums on data and metadata (multiple algorithms available)
* Strong integration with device mapper for multiple device support
* Online filesystem check
* Very fast offline filesystem check
* Efficient incremental backup and FS mirroring

As of Linux 3.6 (released 30 September 2012), Btrfs implements:

* Online defragmentation
* Online volume growth and shrinking
* Online block device addition and removal
* Online balancing
* Offline filesystem check
* Online data scrubbing for finding errors and automatically fixing them for files with redundant copies
* RAID0, RAID1, and RAID10
* Subvolumes
* Transparent compression
* Snapshots
* File cloning
* Checksums on data and metadata
* In-place conversion from ext3/4 to Btrfs
* File system seeding
* Block discard support
* Send/receive
* Hierarchical per-subvolume quotas

## Planned features

* Online filesystem check
* Very fast offline filesystem check
* Parity-based RAID (RAID5 and RAID6)
* Object-level RAID0, RAID1, and RAID10
* Incremental dumps
* Ability to handle swap files and swap partitions
* Data deduplication
* Encryption