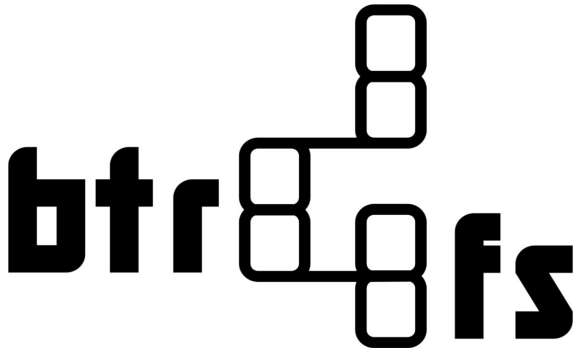


Intern: Btrfs

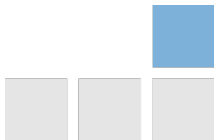
A modern file system



btrfs

Simon Fromme ■ 23.06.2017

- Abkürzung für “B-Tree Filesystem”
- 2007: Vorstellung eines COW-freundlichen B-Trees durch Ohad Rodehq (Linux Storage and File Systems Workshop)
- Konzeption von Btrfs durch Chris Mason (ehemaliger Mitarbeiter bei Reiserfs)
- “Momentan” (Kernel 4.8-rc5): 120.760 Lines of Code, 20-35 Contributors pro Release

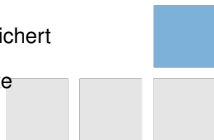


1970 entdeckt von R. Bayer und E. McCreight bzw. M. Kaufman (nicht publiziert)

1. Jeder Knoten hat maximal m Kinder
2. Jeder Knoten (außer der Wurzel) hat mindestens $m/2$ Kinder
3. Die Wurzel hat mindestens 2 Kinder, außer sie ist gleichzeitig ein Blatt
4. Alle Blattknoten haben die gleiche Tiefe und tragen keine Informationen
5. Jeder Knoten (der kein Blatt ist) mit k Kinderknoten beinhaltet $k - 1$ Schlüssel

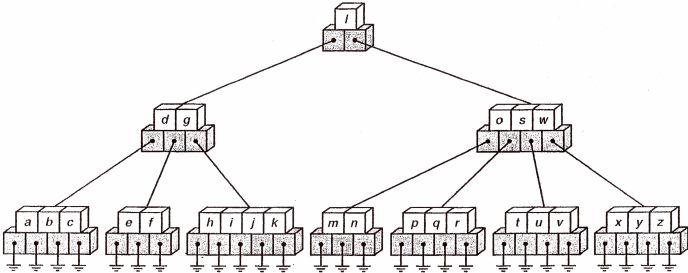
B⁺-Tree:

- eigentliche Datenelemente werden nur in den Blattknoten gespeichert
- innere Knoten enthalten lediglich Schlüssel, keine Datenelemente
- Verbindung der Blattknoten zu doppelt verketteter Liste

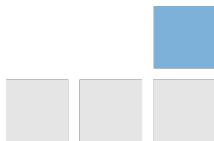


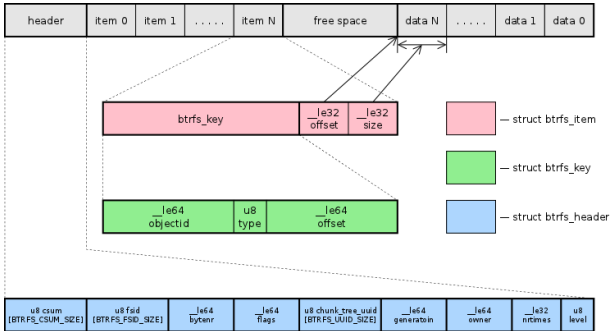


B-Trees

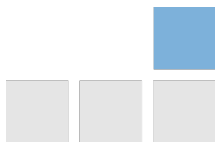


- keine Verkettung der Blattknoten
- Referenz-Zähler für Knoten, um Zahl der Baum-Traversierungen beim Löschen von Snapshots zu reduzieren
- Modifizierte Lösch-/Einfüge Algorithmen, etc.

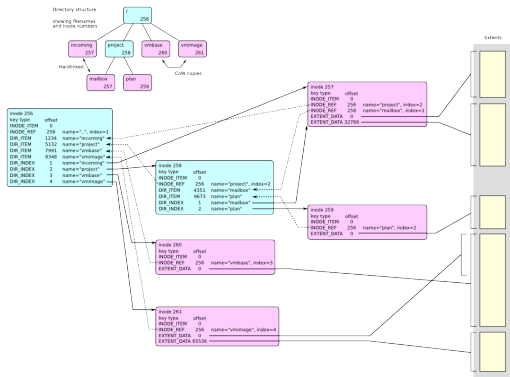




- Filesystem
- Extent-Allocation
- Checksum
- Chunk-/Device
- Reloc



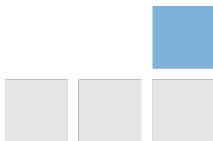
Btrfs-Trees



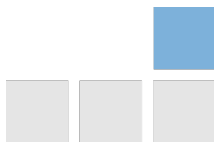
Demonstration der verschiedenen Trees und der jeweiligen Schlüssel, Datenelemente anhand des Outputs von

```
$ btrfs-debug-tree [options] <device>
```

- Speichereffizient für kleine Dateien
- Speichereffiziente Indizierung von Verzeichnissen
- Dynamische Inode Allokierung
- Writeable Shapshots, Read-Only Snapshots
- Checksummen für Dateien und Metadaten
- Kompression (zlib, LZO)
- Integrierte Unterstützung für verschiedene Devices (Striping, Mirroring, RAID, ...)
- "SSD-Aware"
- effiziente inkrementelle Backups
- Online Degragmentierung
- Offline Filesystem-Check
- "Suvolume-aware" Quota Unterstützung



- Online Filesystem Check
- Mirroring and Striping auf Objekt-Level
- Alternative Checksum Algorithmen
- In-band Deduplication
- Andere Kompressions Methoden (LZ4, ...)



- B-Tree Scheme (Seite 4):
<http://ps11.pstcc.edu/~rbarber/1020/images/Btree-complete.gif>
- Btrfs Datenstruktur (Seite 6):
<https://btrfs.wiki.kernel.org/index.php/File:Leaf-structure.png>
- Btrfs Filesystem-Trees (Seite 8):
<https://btrfs.wiki.kernel.org/index.php/File:Directory-structure.png>

