

Erledigt

Hackintosh für Videobearbeitung mit externer SSD

Beitrag von „Brumbaer“ vom 19. Januar 2018, 13:35

USB 3.0 ist von der Übertragungsrate ausreichend für eine SSD.

Ein Gerät, das einen Raid 0 Controller hat (im Gegensatz zu 2 USB Geräten, die als Raid 0 betrieben werden), benötigt USB 3.1 für volle Geschwindigkeit

Allerdings haben Geräte, die über USB angeschlossen werden, geringere IOPS als Geräte, die über SATA angeschlossen werden.

SSDs unterscheiden benutzte und unbenutzte Blöcke und unbenutzte können gelöscht oder nicht gelöscht sein.

Ein Block in einer SSD kann nicht überschrieben werden - ohne ihn vorher zu löschen. Das dauert relative lange, deshalb wird der Inhalt in einen unbenutzten und schon gelöschten Block geschrieben und der alte Block als unbenutzt und nicht gelöscht gekennzeichnet. Das funktioniert, weil es eine Tabelle gibt in der steht wo der Inhalt eines Blockes im Speicher untergebracht ist. Der Inhalt muss nicht immer an der selben Stelle stehen, solange man dafür sorgt, dass die Tabelle auf die richtige Stelle zeigt.

Wenn die SSD gerade nichts macht, wird die Garbage Collection gestartet und die löscht nun die unbenutzten, aber nicht gelöschten Blöcke, so dass sie für die nächste Schreiboperation zur Verfügung stehen.

Unbenutzt bedeutet hier unbenutzt aus Sicht der SSD. Ein Block der aus Benutzer/Betriebssystem Sicht gelöscht wird, ist aus Sicht der SSD immer noch benutzt, denn die SSD weiß nicht, dass das OS den Block als gelöscht ansieht.

Dafür gibt es das Trim Kommando. Das sagt der SSD, dass ein Block vom OS nicht mehr benötigt wird und er deshalb gelöscht und zu den anderen fürs Schreiben verfügbaren Blöcken hinzugefügt werden kann.

In der Praxis, wird der Block vermutlich nicht sofort gelöscht, sondern er wird nur als unbenutzt gekennzeichnet und bei der nächsten Garbage Collection gelöscht.

Trim ist ein SATA Kommando.

NVMe ist nicht SATA und kennt deshalb keinen Trim-Befehl, aber es kennt Deallocate. Deallocate markiert einen Block als unbenutzt und dann kann er bei der

Wir schauen uns mal 5 Speicherbereiche(SSD-Blöcke) in der SSD an.

Block 77 Daten für den OS-Block 48
Block 78 unbenutzt, nicht gelöscht
Block 105 Daten für den OS-Block 44
Block 111 Daten für den OS-Block 45
Block 115 unbenutzt, gelöscht

Nun schreibt das System Block 45 neu, es sucht sich einen unbenutzten gelöschten Block und verwendet den

Block 77 Daten für den OS-Block 48
Block 78 unbenutzt, nicht gelöscht
Block 105 Daten für den OS-Block 44
Block 111 unbenutzt, nicht gelöscht
Block 115 Daten für den OS-Block 45

Morgens um halb zehn in Deutschland, der Anwender(in) isst sein K..., und die SSD hat Zeit die Garbage Collection laufen zu lassen

Block 77 Daten für den OS-Block 48
Block 78 unbenutzt, gelöscht
Block 105 Daten für den OS-Block 44
Block 111 unbenutzt, gelöscht
Block 115 Daten für den OS-Block 45

Jetzt löscht das System die Datei, die Block 44 enthält.

System ohne Trim/Deallocte

Block 77 Daten für den OS-Block 48
Block 78 unbenutzt, gelöscht
Block 105 Daten für den OS-Block 44
Block 111 unbenutzt, gelöscht
Block 115 Daten für den OS-Block 45

Wie sie sehen, sehen sie nichts, denn die SSD weiß nicht, dass der Block nicht mehr gebraucht wird.

System mit Trim/Deallocte

Block 77 Daten für den OS-Block 48
Block 78 unbenutzt, gelöscht

Block 105 unbenutzt, nicht gelöscht
Block 111 unbenutzt, gelöscht
Block 115 Daten für den OS-Block 45
Jetzt weiß es die SSD und gibt den Block frei

Garbage Collection und Trim/Deallocate haben unterschiedliche Aufgaben und ersetzen sich nicht gegenseitig.

NVMe benötigt keinen Trim Befehl, weil deallocate schon Teil des "normalen" NVMe Verhalten ist, während Trim "ein neues (relativ zu Ur-SATA) Feature" ist.