

SSDs als Archiv-Laufwerk

Für ein Daten-Archiv sollte man nur SSDs mit Flash-Chips kaufen, die den Firmennamen eines Flash-Herstellers wie Samsung, SK Hynix, Toshiba/Kioxia oder WD/SanDisk tragen, um nicht Gefahr zu laufen, Flash-Komponenten geringer Qualität zu erhalten. Genauso wichtig ist es, ein frisches Medium zu verwenden und keine ausrangierte SSD. Die Abnutzung sollte immer deutlich unter 10 Prozent bleiben. QLC-Speicher eignet sich noch nicht zu Archivierung, 2D-MLC wäre sehr gut geeignet, ist aber kaum noch zu bekommen. Folglich bleibt derzeit nur TLC-Speicher.

Ist die SSD nicht fabrikneu, sollte zur Vorbereitung als Archivmedium ein Secure Erase erfolgen, um alle internen Zuweisungstabellen zu löschen. Die SSD ist dann leer, der logische Inhalt aller Speicheradressen ist undefiniert. Das beschleunigt das Schreiben und vermeidet unnötige Abnutzung durch die Garbage Collection und das Wear Leveling.

Der Einsatz spezieller Flash-Dateisysteme wie exFAT bringt gegenüber komplexen Dateisystemen wie NTFS oder ext4 keinen Vorteil für die Zuverlässigkeit eines Archivs. Wichtiger ist die Frage, welche Dateisysteme in 15 Jahren noch direkt von den Betriebssystemen unterstützt werden. Sieht man sich die Langlebigkeit von FAT16 und FAT32 und die noch vorhandene Unterstützung von ext2 im Linux-Kernel an, sind für die Zukunft auch mit NTFS und ext4 jedoch keine Probleme zu erwarten.

Offene Blöcke

Von einer Consumer-SSD kann man 5 Jahre Data Retention bei 10 Prozent Abnutzung erwarten. Bleibt man deutlich unter 100 Zyklen, sind auch 10 Jahre bei Raumtemperatur kein Problem.

Eine Schwachstelle gibt es aber: Beim Befüllen des Archivs werden die Pages in jedem Block sequenziell geschrieben. Ist ein Block gefüllt, kommt der nächste Block an die Reihe. Es gibt zu fast jeder Zeit mindestens einen Block, der nicht mehr gelöscht, aber auch noch nicht ganz gefüllt ist. Genauer sind es mindestens zwei Blöcke, da einer im TLC-Modus mit Nutzerdaten gefüllt wird und ein zweiter im pSLC-Modus mit internen Metadaten wie Verwaltungstabellen.

Diese Blöcke nennt man offene Blöcke. Durch die Parallelschaltung mehrerer Planes und Dies entstehen große logische TLC-Blöcke, sogenannte Superblöcke, die aus 8, 16 oder 32 physischen Blöcken bestehen. Mit der Parallelität erhöht sich der Durchsatz und auch die Anzahl der offenen physischen Blöcke.

Offene Blöcke zeigen beim Auslesen eine hohe Bitfehlerrate, da die noch leeren Pages die resultierenden Spannungen verschieben. Dies lässt sich beim zeitnahen Lesen zwar von der Firmware durch ein Verschieben der Schwellwertspannungen so weit kompensieren, dass die Daten lesbar sind; jedoch haben offene TLC-Blöcke eine schlechte Data Retention Time. Offene pSLC-Blöcke sind nicht betroffen, da diese durch ihre zwei Zustände unempfindlich sind.

Bei langer Lagerung werden offene TLC-Blöcke als erste Lesefehler aufweisen. Falls die zuletzt geschriebenen Daten Verwaltungsdaten des Dateisystems sind, kann es auch zu größeren Datenverlusten kommen. Zur Vermeidung dieses Problems sollten die offenen Blöcke von der Firmware der SSD selbstständig mit Zufallsdaten gefüllt und somit geschlossen werden, wenn das Laufwerk vom Betriebssystem abgemeldet wird. Dem Autor sind aber keine Consumer-SSDs bekannt, die über eine solche Funktion verfügen. Daher bleibt nur die Option, die Zeit bis zum nächsten Auffrischen deutlich kürzer zu wählen, als der Flash theoretisch ermöglicht.

Der Anwender hat keine Möglichkeit, die Existenz offener Blöcke zu vermeiden. Spätestens die im Hintergrund unbemerkt laufende Garbage Collection kann jederzeit neue offene Blöcke erzeugen, sofern die SSD mit Strom versorgt wird.

Auffrischung

Wegen der möglichen offenen Blöcke sollten die Daten alle 3 Jahre aufgefrischt werden, bei deutlich unter 100 Zyklen Abnutzung spätestens alle 5 Jahre. Nach 15 Jahren ist die Produktlebensdauer überschritten, denn elektronische Bauteile altern auch im abgeschalteten Zustand. Dann sollte man die Daten auf ein größeres Medium der zu diesem Zeitpunkt aktuellen Technologie umkopieren – bevor man irgendwann keine Hardware mehr besitzt, die den

Anschluss von SATA-, NVMe- oder USB-SSDs erlaubt.

Beim Auffrischen gibt es zwei Möglichkeiten: Entweder schreibt man selbst alle Daten durch Umkopieren neu oder überlässt es der Firmware der SSD, den Zustand zu bewerten und nur die Blöcke aufzufrischen, bei denen es die Firmware aufgrund der Bitfehlerrate für nötig erachtet.

Will man die Entscheidung der Firmware überlassen, so muss man lediglich einmal alle logischen Adressen lesen. Unter Windows nutzt man `chkdsk /r`, unter Linux bietet sich das Kommando `dd` an:

```
dd if=/dev/LAUFWERK of=/dev/null
bs=8M status=progress
```

Trifft die SSD-Firmware dabei auf eine hohe Bitfehlerrate oder muss die Firmware sogar die Schwellwertspannungen zur Unterscheidung der einzelnen Zustände anpassen, wird sie den Block auffrischen. Je nach Hersteller wird die Firmware einen kritischen Block möglicherweise nicht sofort auffrischen, sondern zunächst die nächsten Leseanforderungen des Hosts weiter bearbeiten. In solchen Fällen merkt sich das Data-Care-Management den betroffenen Block, um ihn später aufzufrischen, wenn die SSD nicht voll ausgelastet ist.

Da nun viele, wenn nicht sogar alle Blöcke eine Auffrischung benötigen, wird sich die Firmware nicht alle Blocknummern merken können. Dann greift ein zweiter Mechanismus: Die Firmware aktiviert einen Auto-Scan.

Nach der Überprüfung wird die Firmware im Hintergrund alle Blöcke erneut lesen und bei Bedarf sofort auffrischen. Daher sollte die SSD wegen der im Vergleich zum Lesen geringeren Schreibgeschwindigkeit noch mindestens die achtfache Zeit angesteckt bleiben, die für die vorherige Überprüfung benötigt wurde.

Bei diesem Verfahren muss man darauf vertrauen, dass der SSD-Hersteller sein Data-Care-Management entsprechend dem aktuellen Stand der Technik implementiert hat. Wem das zu unsicher ist, dem bleibt nur das manuelle Auffrischen, also das Kopieren der Daten auf eine leere SSD mit erneut deutlich weniger als 10 Prozent Abnutzung.