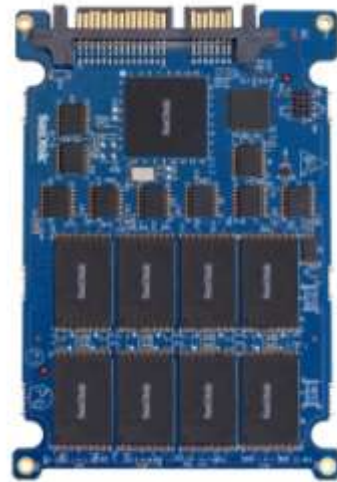


[Guide] – SSD Optimierungen



Sie bestimmen das IT-Jahr 2009: **SSD, Solid State Drives.**

In diesem Guide werden wir behandeln, wie man seine Betriebssysteme für SSDs optimiert und welche Anpassungen / Einstellungen nötig sind. Manche können sogar auf die klassische Festplatte angewandt werden.

1. **Einführung:**

Wie mittlerweile in jedem Gebiet von aktueller Hard- und Software gibt es nicht „die“ SSD. Einerseits ist es zwar schön, aus einer großen Produktvielfalt aussuchen zu können, andererseits können unbedarfte Anwender auch sehr schnell den Überblick verlieren.

Bei SSDs hält sich das ganze jedoch noch in Grenzen – man unterscheidet grundsätzlich zwischen 2 Arten von SSDs, die sich vor allem durch deutliche Differenzen bei den Preisen unterscheiden.

MLC-basierte:

→ Bei MLC-basierten SSDs handelt es sich um SSDs mit MLC (multi level cell) Speicherchips. Sie sind in der Regel deutlich billiger. Nachteil ist jedoch die Lebensdauer einer Zelle, die mit ca. 10.000 Schreibzyklen beziffert wird. Laut Herstellerangaben von Intel oder OCZ sollen die Laufwerke dennoch Jahre halten, echte Erfahrungswerte wird man in erst in paar Jahren erhalten.

SLC-basierte:

→ SLC-Chips (single level cell) speichern nur ein Bit pro Zelle. Sie sind deutlich teurer, zumal für gleiche Speicherkapazitäten auch mehr Chips verbaut werden müssen. Vorteil von ihnen sind die 100.000 Schreibzyklen. Sie halten also ca. 10x so lange, wenn nicht ein Chip einfach kaputt geht.

Während sich die Endkunden-orientierten Unternehmen wie OCZ, A-Data oder G.Skill hauptsächlich auf MLC-Laufwerke spezialisieren und in kurzen Abständen neue Produkte auf den Markt werfen, konzentrieren sich Firmen wie Mtron oder Memoright auf den Unternehmensbereich, aus dem auch diese Technologie entstammt und schon länger bekannt ist.

Dennoch möchte Mtron auch etwas vom Kuchen abhaben und führte die Mobi-Serie ein. Sie bieten zwar nicht die absolute Leistungskrone, für den Endkunden sind sie aber völlig ausreichend und besser als die meisten Festplatten.

2. **Optimierung Windows:**

Wir beschränken uns hier auf Windows Vista, einige Tipps sind aber bestimmt auch unter XP anwendbar. Zu gegebener Zeit werden wir den Artikel bzgl. Windows 7 aktualisieren. Dort sind allerdings schon viele Sachen voreingestellt und generell der Umgang mit SSDs optimiert.

→ Indizierung abschalten:

Wie auch schon im Tuning-Guide beschrieben empfiehlt es sich auch hier, die Vista-Suche abzuschalten. Sollte man wirklich etwas suchen müssen, geschieht dies bei einer SSD derart schnell, dass die Vorteile des Vista-Index verloren gehen.

Dazu klicken wir uns in die Systemsteuerung – Verwaltung – Dienste und suchen in der langen Liste die Windows-Suche. Dort einen Rechtsklick auf diesen Eintrag und auf Eigenschaften. Im erscheinenden Fenster wählen wir neben „Starttyp“ den Punkt „Deaktiviert“.

→ Schreibcache aktivieren:

Ähnlich wie bei USB-Sticks empfiehlt es sich, den Schreibcache für Laufwerke zu aktivieren. Dazu klicken wir auf Computer, dort einen Rechtsklick auf die SSD und auf Eigenschaften. Im Reiter „Hardware“ machen wir einen Doppelklick auf das SSD-Laufwerk. Im öffnenden Fenster klicken wir auf den Reiter „Richtlinien“ und wählen den Punkt „Für Leistung optimieren“. Nun sollten sich darunter 2 Unterpunkte befinden, einmal „Aktiviere Schreibcache auf dem Laufwerk“ und einmal „Aktiviere erhöhte Schreibleistung“. Vor beiden Menüpunkten einen Haken setzen und mit OK bestätigen.

→ Auslagerungsdatei abschalten:

!Achtung!: Dieser Schritt kann zu Inkompatibilitäten mit alten Programmen führen. Gerade bei MLC-basierten SSDs empfiehlt er sich jedoch, da so erheblich weniger Schreibzugriffe seitens Windows stattfinden. Auch sollte man genügend Arbeitsspeicher haben. Alternativ zum komplettem Abschalten der Auslagerungsdatei kann man diese auch auf eine andere Festplatte, ExpressCard-SSD oder schnelle Speicherkarte im CardReader verlegen. Dazu sollten entsprechende Alternativen auch leistungsfähig genug sein. Den Pfad der Auslagerungsdatei passt man im gleichen Menü an, siehe folgend:

Zunächst doppelklickt man auf „Computer“ und dort auf „Systemeigenschaften“ im grün-blau hinterlegten Menü oben. Im erscheinenden Fenster klickt man links auf „Erweiterte Systemeinstellungen“ und bestätigt die UAC-Meldung mit Fortsetzen. Im Reiter „Erweitert“ klickt man auf den Punkt „Einstellungen“ im Bereich „Leistung“. Im nun erscheinenden Fenster klickt man auf den Register „Erweitert“ und auf „Ändern“ im Bereich „Virtueller Arbeitsspeicher“. Im sich öffnenden Fenster klickt man auf den Punkt vor „Keine Auslagerungsdatei“ und rechts daneben auf „Festlegen“. Vista wird nun auffordern, den Computer neu zu starten – das kann aber erst einmal warten.

→ Timestamp abschalten:

Der Timestamp hatte den eigentlichen Sinn, dass sich das Betriebssystem merkt, wann eine Datei zum letzten Mal geöffnet oder geändert wurde. Achtung: Diesen Timestamp bitte nicht verwechseln mit der Eigenschaft „Zuletzt geändert am“ im Explorer – dieser ist davon nicht betroffen.

Unter Windows öffnet man dazu einfach die cmd.exe mit Administratorrechten (rechtsklick auf die cmd.exe und „Ausführen als Administrator“ anklicken) und gibt folgenden Befehl ein:

```
FSUTIL behavior set disablelastaccess 1
```

Dies minimiert die Schreibzugriffe auf die SSD enorm und ist vor allem für MLC-User zu empfehlen, aber auch die SLC'ler verlängern das Leben ihrer Laufwerke.

→ Defragmentierung abschalten:

Auch diesen Punkt hatten wir schon im Vista-Tuning-Guide angesprochen. Defragmentierung allgemein wird bei SSDs keine Rolle mehr spielen. Zu schnell sind die Zugriffe, zu marginal die Unterschiede zwischen Fragmentiert und nicht.

Deshalb gehen wir im Startmenü auf „Alle Programme“, „Zubehör“, „Systemprogramme“ und „Defragmentierung“ und entfernen den Haken vor der automatischen Defragmentierung.

→ Windows-Logging-Dienst abschalten/umleiten:

Achtung: Diesen Dienst benötigen einige wenige Programme (z.B. 3DMark06). Es empfiehlt sich also, das Arbeitsverzeichnis dafür auf eine Festplatte umzuleiten.

Dazu geht man im Registry-Editor (regedit.exe unter Ausführen eingeben) in den Registry-Schlüssel

HKEY_LOCAL_MACHINE\Software\Microsoft\WBEM

Im Unterpunkt **CIMOM** wählen wir rechts die Unterpunkte Working Directory, Repository Directory und Logging Directory. Dort sind bereits Pfade definiert die in etwa so aussehen:

%systemroot%\system32\wbem\Logs\

Aus dem Teil %systemroot% machen wir z.B. ein F: oder einen anderen Laufwerksbuchstaben, je nachdem wie die eigene Situation aussieht. Bei Notebooks, in denen man nur ein Laufwerk einbauen kann, beschränkt sich die Möglichkeit sowieso auf eine ExpressCard-SSD oder eine SD-Karte im Card-Reader. Alternativ kann man auch einfach den WBEM-Dienst auf der SSD lassen.

→ Sonstige Maßnahmen:

Im Forum von OCZ (ocztechnologyforum.com), dem ersten Hersteller, der SSDs dem Massenmarkt zur Verfügung stellte (und dabei ziemlich floppte) wurden einige Tweaks für Vista 32bit/64bit veröffentlicht. Den Thread findet man hier: [Link](#)

[Das folgende bezieht sich auf diesen Beitrag!]

Einige Tweaks haben wir bereits angesprochen. PreFetch und SuperFetch zu deaktivieren macht die wenigsten Probleme, wenn nur ein Laufwerk verbaut ist (z.B. im Notebook). Da Programme dort automatisch auf der SSD installiert sind und sehr schnell starten, ist es nicht nötig, diese direkt nach dem Starten in den RAM zu cachern.

In Desktopsystemen, in denen mehrere Laufwerke verbaut sind und Programme die häufig genutzt werden auf einer anderen Festplatte liegen wird das ganze schon kritischer. Sicherlich, die Dienste können problemlos deaktiviert werden. Jedoch ist ein deutliches Leistungsminus zu spüren, wenn keine Programme mehr gecached werden und komplett von der Festplatte geladen werden müssen. Hier muss also jeder User für sich selbst entscheiden.

Die Tweaks im Schlüssel

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management

bzgl. 'clearPageFilesAtShutdown', 'LargeSystemCache' un 'SecondLevelDataCache' sind eher allgemeiner Natur und können problemlos angewandt werden. Für den Punkt SecondLevelDataCache noch ein paar Werte:

AMD Phenom 9x00 Reihe: 2048

Core 2 Duo T-Reihe: 4096

AMD Phenom II 9x0 Reihe: 2048

AMD AthlonX2: 1024 oder 2048

Auch die Tweaks im benachbarten Registry-Schlüssel

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\FileSystem

können ohne Bedenken angewandt werden.

Der Wert für NtfsDisable8dot3NameCreation deaktiviert die Möglichkeit im NTFS-Dateisystem, dass Namen in verkürzter Form abgespeichert werden – erhöht die Dateisystemperformance und hat wenig mit einer SSD zu tun.

Der Wert in NtfsMemoryUsage = 2 erhöht den Dateicache – auch eher ein Tuning für das Dateisystem und weniger für die SSD.

Den Teil bzgl. der Windows-Dienste haben wir ja schon weiter oben abgehakt.

Die restlichen „Tweaks“ liegen im Ermessen des Benutzers. Die Indexierung für die SSD kann man getrost deaktivieren. Die Aktivierung des „Performance Mode“ deaktiviert die meisten grafischen Schmankerl von Vista. Beides hat natürlich keine Auswirkungen auf die SSD.

Der letzte (sinnvolle) Punkt bzgl. dem Hibernation-File liegt ebenfalls im Ermessen des Benutzers. Wer häufig den Hibernate-Modus benötigt (beim Hibernate wird eine Kopie des Systemzustandes in der Größe des verbauten Arbeitsspeichers auf der Festplatte abgelegt und der PC komplett ausgeschaltet. Im Gegensatz dazu wird im Standby der aktuelle Systemstatus in den Arbeitsspeicher geladen und diese konstant mit einer gewissen Spannung versorgt, so dass dieser die Informationen behält). Wer also auf Hibernate verzichtet, kann so noch einmal ein paar Gb auf der SSD freischaufeln.

→ Alignment:

Für optimale Performance muss die SSD richtig „aligned“ werden. Während XP damit überhaupt nix anfangen kann, wird bei Vista und „7“ zwar aligned, aber nicht optimal für SSDs. Diese Werte sind aber vom Controller in der SSD abhängig. SSDs mit JM602- und Indilinx Barefoot-Controller sollten 64k Alignment-Werte bzw. 128 Offset-Sektoren erhalten, SSDs mit internen RAIDs (G.Skill Titan oder OCZ Apex) das doppelte. Ebenso auf klassischen RAIDs aus 2 solcher Laufwerke jeweils den doppelten Wert.

Um das Alignment zu verändern, muss die SSD blank und unpartitioniert sein. Weiterhin benötigt man das Tool [diskpar](#) (XP). Vista User haben das alternative Tool diskpart bereits vorinstalliert. Dieses kleine Kommandozeilen-Tool ermöglicht also das editieren der Laufwerke. Dazu geht man wie folgt vor:

Zunächst muss man herausfinden, welche Datenträger-Nummer die SSD in dem System hat. Dazu klickt man sich in die Datenträgerverwaltung von XP. (Systemsteuerung – Verwaltung). Dort notiert man sich die Datenträger Nummer seiner SSD.

Unter XP öffnet man nun via cmd.exe diskpar mit dem Befehl „diskpar –s x“, wobei x die Nummer des Datenträgers sein muss, sonst gibt es Chaos. Zunächst wird man nach den Offset-Sektoren gefragt, meistens (JM602 / Indilinx Barefoot) muss man hier 128 eingeben. Für andere SSDs findet man diese Informationen auch meist bei google in User-Foren. Danach muss man noch die Partitionsgröße angeben (in Mb), bei 30Gb wären dies 30553. Diese Information findet man aber auch im diskpar-Fenster ein Stückchen weiter oben. Danach sollte man diese Partition noch formatieren, meist mit NTFS und 4096kb große Zuordnungseinheiten (standard).

Unter Vista/7 hat man bereits das Tool diskpart vorinstalliert. Dazu klickt man auf das „Start“-Menü und tippt im Ausführen-Feld „diskpart“ ein. Nach Bestätigen der UAC-Meldung befindet man sich direkt im Tool (ebenfalls Kommandozeile). Dort kann man sich mittels „list disk“ seine Laufwerke anzeigen lassen und danach direkt *select disk x*, wobei x für die Nummer der SSD steht. Danach gibt man den Befehl „create partition primary size=30553 align=64“ ein und erhält somit eine Partition mit dem richtigen Alignment. Danach noch ein active um die Partition aktiv zu schalten und dann kann das Tool auch wieder geschlossen werden. Danach noch eine Schnellformatierung auf NTFS und alles ist fertig.

Wichtig: Im Tool diskpar unter XP muss man die Offset-Sektoren angeben, unter diskpart (Vista/7) die Größe des Alignments. Macht man dieses Alignment nicht, *kann* es zu extremen Leistungseinbrüchen kommen.

→ Der letzte allgemeine Hinweis:

SSDs sind mit sogenannten „wear-leveling“-Algorithmen ausgestattet, die dafür sorgen, dass die Flash-Zellen einer SSD gleichmäßig abgenutzt werden. Erkennt der Controller also eine defekte Zelle wird diese als defekt markiert und eine andere Zelle verwendet. Dies geht solange gut, bis der Controller keine Ausweichmöglichkeit mehr findet. Dieser Prozess kann entscheidend verlängert werden, wenn eine SSD nicht bis zum randvoll gefüllt ist. Als allgemeine Richtlinie kann man 80% sehen, grade bei MLC-Laufwerken.

→ SSDs und Linux:

Auch unter Linux sind einige Anpassungen notwendig um die SSD möglichst lange am Leben zu halten, auch wenn hier deutlich weniger Maßnahmen möglich/nötig sind.

Einen ähnlichen „Tweak“ wie unter XP/Vista/7 gibt es jedoch auch für Linux. Die Auslagerungsdatei bzw. das „Swap“-Verzeichnis, was unter Linux weniger genutzt wird als unter Windows, kann auch hier auf ein nicht-SSD-Laufwerk gelegt werden. Dazu mountet man das /swap-Verzeichnis einfach auf einer SD-Karte oder einer ExpressCard-SSD. Wobei hier noch gesagt werden sollte: Sollte in einem PC 2Gb oder mehr verbaut sein, kann man die swap-Thematik fast vergessen. Bei 4Gb wird das swap-Verzeichnis überhaupt nicht mehr benötigt, deshalb mounten wir es auch nach wie vor auf die SSD.

→ Linux: Disk Cache:

Sollte dein Mainboard einen „Disk Cache“ unterstützen (bei Notebooks wird man diese Option eher selten finden, da dort das BIOS auf das Minimum beschnitten ist) empfiehlt es sich, das Laufwerk anders ansteuern zu lassen.

Normalerweise müsste im BIOS der Wert für den Disk Cache auf „Write Through“ stehen. Bei SSDs sollte man diesen auf „Write Back“ stellen und den Parameter

```
hdparm -W1 /dev/sda (sda=SSD)
```

um den Rückschreibecache zu aktivieren. Damit man dies nicht nach jedem Booten erneut eingeben muss, fügt man diese Zeile in die Datei /etc/rc.local ein.

→ Linux: IO-Scheduler ändern:

Linux unterstützt von Haus aus eine Technik, die sehr dem NCQ ähnelt. Mit einem speziellen I/O-Scheduler wird dabei das Lesen und Schreiben von Festplatten-Plattern (die Scheiben einer Festplatte) in einer bestimmten Reihe und Sequenz durchgeführt.

Da dies bei SSDs aber keinen Sinn mehr macht, booten wir mit einem alternativen Scheduler.

Dazu fügen wir in unsere menu.lst folgendes hinzu:

```
title Fedora 10 x64
root (hd0,1)
kernel /boot/vmlinuz-2.6.27.18-x64... ..quiet elevator=noop
initrd /boot/initrd.img-2.6.17.18-x64
```

Das fett markierte aktiviert den noop-elevator und erhöht die allgemeine Performance der SSD unter Linux.

→ Linux: Letzter Dateizugriff nicht schreiben:

Auch dieses kennen wir bereits von XP/Vista. Auch unter Linux wird der letzte Dateizugriff für jede Datei gespeichert und erzeugt unnötige Schreibzugriffe auf die SSD. Da dieser Tweak direkt von Linus Torvalds (dem Entwickler des Linux-Kernels) kommt, kann man dies getrost anwenden.

Dazu editieren wir als root die /etc/fstab (z.B. su – password / sudo gedit /etc/fstab) und setzen hinter die Partitionen, die auf der SSD liegen ein noatime.

Eine Zeile würde dann entsprechend so aussehen:

```
/dev/sdb1 / ext3 noatime,errors=remount-ro 0 1
```

→ Linux: Dateisystem:

Neuere Dateisysteme setzen allesamt auf ein sogenanntes "journaling", um schneller auf Dateien zugreifen zu können. Dieses Journal macht bei SSDs wiederum keinen wirklichen Sinn mehr und erzeugt eher unnütze Schreibzugriffe, da bei jeder Änderung das Journal neu geschrieben werden muss.

Dazu empfiehlt es sich, dass Dateisystem für Linuxinstallationen auf ein älteres Dateisystem wie z.B. ext2 zu setzen. Sollte man bereits auf ext3 installiert haben und möchte nun auf ext2 wechseln, so kann man in der /etc/fstab die Partitionen als ext2 mounten, damit wird ebenfalls auf das Journal verzichtet.

Dies waren nun die ersten und allgemeinen Tipps und Tricks rund um SSDs.

Bei aller Panikmache um die Haltbarkeit von MLCs und SLCs:

OCZ brachte mit der **Core Series** die ersten SSDs in den Massenmarkt und sorgte schnell für Enttäuschung. Grund hierfür ist der verkorkste JMicron JMF602-Controller welcher alles andere als ausgereift ist. Schnell wurde eine B-Revision (JMF602B) herausgebracht, die die größten Mängel beseitigten. SSDs der ersten Generation litten in häufigen Fällen an sogenannten „Systemlags“. Da die Leistung des Controllers sehr beschränkt ist und besonders im 4k-Write die IOPS extrem niedrig sind, konnte eine Installation eines Spiels oder Betriebssystems länger dauern als auf einer klassischen Festplatte. Diese Laufwerke verschwinden nun jedoch langsam aus den Händler-Portfolios. OCZ legte nach mit der Core V2 mit leicht erhöhten Transferraten und beseitigten Systemlags. Diese können zwar nach wie vor auftreten, aber es bedarf nun mehr Anforderungen gleichzeitig.

Schnell folgten neue Serien wie die Solid Series, Apex Series oder Vertex Series. Ein Blick in Händlerlisten wie geizhals.at offenbart jedoch: Es gibt sehr viele Laufwerke mit exakt den gleichen Leistungsdaten. Diese benutzen dann auch allesamt den besagten JMicron-Controller. Lobenswerte Ausnahmen stellen dort zur Zeit die OCZ Vertex Serie und die Samsung SSDs dar. Während Samsung einen eigenentwickelten Controller verwendet und über ein paar OEM-Partner vertreibt (Corsair, PNY, Hama) setzt die OCZ Vertex und ein paar andere Hersteller auf den Indilinx Barefoot Controller. Zwar hat auch dieser seine Schwächen, aber dank eigenem Cache und neuer Entwicklung ist er deutlich leistungsfähiger als andere Controller.

Allen MLC-Laufwerken haftet jedoch noch der fade Beigeschmack der Haltbarkeit an:
MLC-Zellen halten ca. 10.000 Schreibzyklen aus, SLC-Chips ca. 10x mehr.

Mal abgesehen davon, dass Festplatten wesentlich früher kaputt gehen können (allein schon wegen den mechanischen Teilen) so können auch MLC-Laufwerke sehr lange halten. Selbst wenn man jeden Tag 50Gb auf die SSD schreibt, können sie jahrelang halten und funktionieren. Und wie realistisch ist das?

Gar nicht, selbst als Systemlaufwerk schreibt man nicht 50Gb am Tag – und wie lange behält man schon eine Festplatte/Laufwerk in seinem System?

Für ein XP-System mit ein paar Programmen kann eine 16Gb SSD ausreichen, besser ist eine mit 32Gb. Für Vista kann eine 32Gb SSD ebenfalls ausreichen (nachdem man Vista etwas bereinigt hat), will man jedoch noch ein paar Programme nutzen empfehlen sich 64Gb. In diesen Größen sind durchaus SLC-SSDs bezahlbar. Für den gleichen Preis einer 64Gb SSD bekommt man jedoch eine 128Gb SSD mit MLC-Speicher.

Es muss also jeder für sich selbst entscheiden, was er sich zutraut und was er benutzen möchte.
Abschließend noch ein paar Links zum Ein- und Weiterlesen:

- [Allgemeiner SSD-Thread](#) (forumdeluxx.de)
- [Was ist eine SSD?](#) (wikipedia.de)
- [Liste über verfügbare SSDs](#) (geizhals.at/eu) [gefiltert auf die, die auch etwas bringen würden]
- [Artikel zur Einführung](#) (forumdeluxx.de)
- [Alignment](#) (ocztechnologyforum.com)
- [Test von uns zu SSDs](#) (schenx.eu)