

## Auslagerungsdatei in Windows bei 1024MB (oder mehr) RAM ?

*Bislang vertrat ich vehement den Standpunkt "warum soll ich mein Betriebssystem auf der langsamen Festplatte herumkratzen lassen, wenn ich doch genügend schnellen (und teuren) RAM zur Verfügung habe!"*

*Ich habe nun diese dafür erforderlichen Prozesse einmal studiert und bin zu dem Ergebnis gekommen, das ich mich gewaltig irrte.*

*Hier "kratzt" nichts "auf der Festplatte herum", wenn ich diese Auslagerungsdatei größer mache – und es bringt absolut nichts, wenn ich diese auf einer anderen Festplatte ablege.*

*Meinen Irrtum und die sich (für mich) daraus ergebenden Konsequenzen will ich mal hier darlegen.*

Ich hab mich erst etwas mit der Theorie beschäftigt und dabei das gelernt:

Windows verwendet für den Speicher das sogenannte "[Paging](#)-Verfahren". Damit wird der Speicher in Seiten zusammengefaßt. Diese werden dann in einer Indexliste geführt.

Die Seiten- plus die Indexadresse ergeben dann "virtuelle Adresse".

Eine "Virtuelle" Adressierung kann durchaus höhere Speicheradressen ansprechen, als überhaupt reell (physikalisch) vorhanden sind.

Für das Auslagern nun verwendet Windows eine Algorithmus, "NRU"

"Not recently used"

Die verwendeten Speicherseiten haben ein "R-bit" für "read" und ein "M-bit" für "memory".

Diese Bits "R" und "M" werden nun in vorgegebenen Abständen gelöscht.

Leere Seiten, also welche ohne diese bits, werden ausgelagert.

Wird ein Prozeß gestartet, wird diesem Prozeß ein vorgegebener Speicherbereich "[Working Set](#)" zugeteilt. Dieser ist selten exakt angepaßt, er kann zu groß oder zu klein sein – und deshalb versucht Windows diesen Bereich dynamisch immer wieder den Gegebenheiten anzupassen.

Wenn jedoch der tatsächliche Speicherbedarf immer weiter ansteigt, hat Windows ein Problem.

Es markiert eigentlich in Benutzung befindliche Seiten, die gerade leer sind, als "Free Page", freie Seiten.

Reicht das noch immer nicht, werden auch in Verwendung befindliche, bereits beschriebene Seiten, bei denen aber R und M leer sind, verwendet und als "Zero Page" deklariert.

Diese werden nun, wenn der RAM- Speicherplatz nicht mehr ausreicht, ausgelagert.

Denn obwohl diese einige Zeit lang nicht benötigt wurden, ist deren Notwendigkeit ja noch immer möglich.

Somit wird RAM- Speicherplatz frei und Windows kann arbeiten. Es hat sich aber gemerkt, wo die ausgelagerten Teile liegen – das ist die "Virtuelle Adresse". Die Auslagerungsdatei, "[Pagefile](#)" – siehe oben "Paging".

Gibt es nun immer noch Probleme mit dem erforderlichen RAM- Platz, greift Windows auch auf Seiten zu und lagert diese aus, bei denen nur eines der beiden R- oder M- bits leer ist. Auch, wenn diese Daten (fast) sofort wieder benötigt werden.

Erst wenn alle enthaltenen Seiten auch über ein R- und M- bit verfügen, ist der Speicher gefüllt.

Es ist natürlich klar und logisch, daß dies bei wachsendem Speicherplatz immer seltener geschieht.

Es gibt allerdings auch Prozesse, die geben Seiten ihres "[Workingsets](#)" zurück. und das, obwohl in ihnen noch Daten stehen – oder auch die Bits R und M nicht gesetzt sind.

Diese als "Standby" geltenden Seiten können eigentlich überschrieben werden. Es kann allerdings auch sein, daß diese vom Speichermanager ausgelagert werden, weil ein Seitenfehler, "[Page fault](#)" vorliegt.

Diese Standby- Seiten, (Seiten = pages) ebenso wie zero pages werden nun vom Taskmanager als "free pages", "freie Seiten" deklariert. Aber sie sind es nicht wirklich! Haben wir nun keine Auslagerungsdatei vorgesehen, ist also keine "pagefile" vorhanden, werden die oben genannten Seiten einfach überschrieben.

Benötigt nun ein Prozeß den alten Inhalt einer solchen überschriebenen Seite, und dieser ist nicht mehr vorhanden, tritt zwangsläufig ein "page fault" auf.

Ich nehme allerdings an, daß dies ab 1,5GB RAM nur noch selten, über 2GB kaum noch eintritt.

Nun werden ohne pagefile diese Seiten eben eiskalt gelöscht. Ein Nutzen ist dabei nicht zu erkennen, wohl aber ein ziemlicher Nachteil:

Anwendungen, bearbeitungsintensive Anwendungen, die ihr Workingset in einer Tour verändern oder wenn mehrere solcher Sets benötigt werden, können teilweise recht gewaltige Performance- Einbrüche erfahren. muß doch die aus den "page fault" resultierende Unterbrechung repariert bzw. kompensiert werden.

Eine fehlende oder zu kleine pagefile wird nach meinem nunmehrigen Verständnis davon profitieren. Eine, die viel Speicher benötigt (Battlefield2 ist da ein prädestiniertes Beispiel, ebenso Film- und Videobearbeitung) leidet da erheblich drunter.

Wenn also Windows nicht mehr auslagern kann, bekommt eben der laufende Prozeß nur ein kleines, vom RAM zu fassendes working set.

Es geht aber weiter: in die Pagefile wird ja nicht nur ausgelagert, wenn der RAM voll ist. Es werden durch den Taskmanager eben auch Dateien ausgelagert, auf die nie oder selten zugegriffen wird. Weil sie eben selten oder nie benötigt, aber immer mitgeschleppt werden. Diese nebensächlichen Dateien blockieren nun –gleichberechtigt mit wichtigen Dateien- wertvollen RAM.

### **Ich gewinne nun diese theoretischen Erkenntnisse:**

- unabhängig vom verbauten RAM sollte ich nun immer 1GB Auslagerungsdatei einstellen.

### **Ist das nun wirklich so?**

Das wollte ich nachprüfen und habe Tests gemacht. Auf jedem beliebigen PC nachvollziehbare Tests.

Vorraussetzung bei allen Versuchen gleich – der vollkommen gleiche PC.

Also

- Prozessor Athlon 3200+,
- Grafikkarte Radeon 9600,
- 1024 (2x512MB) RAM, FSB400, CI2,5, (in meinem Haupt- PC sind es 3x512MB)
- 2 Seagate- Festplatten.

Diese Einstellungen verwendete ich:

**Test 1:** 64MB Auslagerungsdatei auf Partition C

**Test 2:** 1536MB Auslagerungsdatei auf Partition C

**Test 3:** 1536MB Auslagerungsdatei auf die erste Partition der Slave- Festplatte

1.)

***Es werden 3 mpeg- Videos zu einer Video- DVD erstellt.***

### **Test 1 Test 2 Test 3**

72:10 min 72:36 min 72:49 min

2.)

**Es wird die Zahl Pi mit dem Rechenprogramm "SuperPI" berechnet (Wert: 8M)**

9:28 min 9:16 min 9:28 min

3.)

**Es wird eine Paßwortsuche mit "Unrarit" durchgeführt**

192:47 min 192:12 min 192:58 min

Ich habe auch noch diesen benchmark- Test mit "MCCWinstone" gemacht. Da dieser aber komplett englisch ist und ich bei Meldungen dieses Programms in englischer Sprache nicht erkennen kann, ob dies eine wichtige Meldung ist oder ein unbedeutender Hinweis, sage ich hier nichts dazu. Es wären nur Vermutungen – genau das, was ich bekämpfe. Wissen durch Vermutungen, Annahmen und Meinungen zu ersetzen.

### Und was folgt für mich daraus?

Nicht in einem Falle wurden wirkliche, gravierende Abweichungen erkannt. Die ganzen, teils erbittert geführten Diskussionen um den Standort und die Größe einer Auslagerungsdatei werden schon ab ca. 1GB RAM vollkommen bedeutungslos.

Die aus der Theorie anscheinend resultierenden Erkenntnisse sind insofern fehlerhaft, das diese gesamte Auslagerung mit Überschreibung und neu suchen wohl doch nur in Ausnahmefällen relevant ist.

Windows XP scheint intelligenter zu sein, als man gemeinhin annimmt.

### Ich gewinne diese Erkenntnisse:

Eine große Auslagerungsdatei ist unnütz – bringt keinerlei Vorteile. Wohl aber Nachteile:

- sie kann ziemliche, oft schwer zu beseitigende Fragmentationen erzeugen.
- sie kann (möglicherweise) jede Menge Dateiteilen mit herumschleppen. Wenn man sie nicht löscht beim Runterfahren – was diesen Vorgang eben beträchtlich verlangsamt.

Eine Auslagerungsdatei auf einer anderen Festplatte?

Hab ich mal vor längerer Zeit, vor diesem Test gemacht – weil mir die Gründe *sympathisch* erschienen.

Dabei müssen die beiden Festplatten als einheitliches Ganzes betrachtet werden. Schon die Wiederherstellung der Systempartition kann Probleme bereiten – selbst erlebt:

"Der Benutzer kann nicht angemeldet werden"

Weil eben dieses System auf Einträge in der Pagefile zurückgreifen wollte.

Also auch hier – keinerlei erkennbare Vorteile – wohl aber mögliche Nachteile.

Ich muß dazu sagen – ich bin froh, ein Neuling zu sein. Ich speichere eben "Ansichten", die mir entweder logisch erscheinen –siehe meine theoretischen Betrachtungen – oder aber einfach "sympathisch" sind nicht stur als Dogma ab. Sondern hinterfrage diese und prüfe selbige nach. Und korrigiere eine Erkenntnis sogar dankbar, wenn man mir deren Fehlerhaftigkeit nachweist.

Mir ist klar, das Dogmatiker, denen es unmöglich ist, einen einmal öffentlich vertretenen Standpunkt, eine Meinung, zu revidieren, da allerhand Gegenmeinungen – nicht etwa Gegenargumente bringen werden.

Die manchmal angesprochene Fragmentierung dieser Pagefile – nun, da meine Auslagerungsdatei nur 128MB beträgt, wird der Anteil des dadurch verursachten Defragmentierungsgrades auch recht gering bleiben. Und zum anderen – diese gesamten Manipulationen der Auslagerungsdatei sollen doch der Performance dienen?

Ja, ganz genau deswegen betreibe ich regelmäßig Systemhygiene, unter anderem mit "O&O Defrag".

Jürgen Kirsten

\*\*\*\*\*