

## DEFRAGMENTACIJA

### Fragmentacija

Fragmentacija je definirana kot shranjevanje vsebine posamezne datoteke ali procesa na več različnih fizičnih lokacij na disku.

Če imamo hipotetičen disk s tremi datotekami A, B in C in je datoteka B izbrisana, se sprostí prostor in pride do fragmentacije nezasedenega prostora – disk postane fragmentiran. Datotečni sistem bi sicer lahko takoj defragmentiral disk, vendar bi to nepričakovano upočasnilo delovanje sistema. Na disk nato shranimo novo datoteko D, ki je ravno prave velikosti za prvi nezaseden blok, ki ga je pustila izbrisana datoteka B. Če se datoteka D poveča, jo je potrebno shraniti na naslednji nezasedeni blok, saj poleg nje ni prostora. Lahko bi sicer zamaknili datoteko C v desno, da bi bilo dovolj prostora ali pa prestavili celotno datoteko D na konec, vendar pa to postane nepraktično, ko je na disku veliko datotek oz. so te datoteke zelo velike. Zato se drugi del datoteke D shrani na konec zasedenega prostora in označi kot drugi extent datoteke D.

Ko na disk shranjujemo in iz njega brišemo datoteke postane vse več datotek razdeljenih na extente, ki so razmetani po celotnem disku. Ko hočemo posamezno datoteko prebrati mora bralno-pisalna glava skakati med več različnimi fizičnimi lokacijami na trdem disku, da prebere vsak fragment, kar lahko zelo upočasni dostop. Za rešitev tega problema uporabljamo proces defragmentacije. Pri SSD-jih fragmentacija ponavadi ni problem, saj nimajo gibljivih delov. Močno fragmentiran SSD pa je lahko vseeno počasnejši kot bi sicer bil, saj mora operacijski sistem vsakič, ko dostopa do pretrganega LBA (Logical Block Address) razpona, poslati nov vhodno-izhodni ukaz, kar traja dlje, če je datoteka bolj fragmentirana. Vendar pa SSD verjetno ne bo opazno počasnejši, dokler ne postane zelo močno fragmentiran, zato je morebitna defragmentacija zelo redko potrebna.

### Oblike

Fragmentacija se na splošno deli na notranjo in zunanjo.

Pri večini datotečnih sistemov je ob formatiranju diska potrebno določiti alokacijsko enoto (Windows – cluster, Linux – blok). Če se na primer odločimo za 4KB in potem na disk želimo shraniti datoteko, ki je 5 KB, bomo za njo porabili dve alokacijski enoti, vendar pa bo druga le delno zasedena. Gre za zapravljen prostor, ki ga ni mogoče dodeliti nobeni drugi datoteki. Temu se reče **notranja fragmentacija**. Defragmentacijska orodja ne morejo počistiti notranje fragmentacije. Lahko se jo kvečjemu zmanjša, če vnaprej vemo približno kakšne velikosti datotek bomo shranjevali na disk in glede na to izberemo najbolj idealno velikost alokacijske enote ob formatiranju diska.

**Zunanja fragmentacija** se nadaljno deli na fragmentacijo datotek, kar pomeni, da so datoteke shranjene na več različnih fizičnih lokacijah na disku, torej imajo več extentov, in na fragmentacijo nezasedenega prostora, kar pomeni, da je nedodeljen prostor na disku razporejen na več fizičnih lokacij. Pri računanju procenta fragmentacije na disku se ponavadi upošteva le fragmentacija datotek. Poznamo še eno vrsto notranje fragmentacije – razpršitev datotek. Nanaša na to, da procesor rad dostopa do sorodnih datotek zaporedno. Če so programske datoteke, ki se nanašajo druga na drugo, razpršene po disku, potem je lahko zagon programov počasnejši, zato je npr. defragmentacijsko orodje v Windows 9x seriji optimiziralo programe, tako da je vse programske datoteke in njihove odvisnosti razporedilo zaporedno v takem vrstnem redu v kakršnem bi jih program naložil v pomnilnik.

## Preprečevanje

Problem fragmentacije lahko datotečni sistem poskuša čim bolj zmanjšati že sproti ob branju in pisanju na disk, npr. tako da rezervira buffer prostor okrog datotek, ki se bodo v prihodnosti morda povečale ali pa da takoj dodeli prostor, če je končna velikost datoteke že znana (Windows swap file, BitTorrent protokol). O zakasneni alokaciji malo pozneje.

Po drugi strani pa lahko sistem počaka, da fragmentacija doseže nek določen procent in nato samodejno v ozadju požene defragmentacijsko orodje.

## Defragmentacija

Defragmentacijska orodja uporabljajo proces defragmentacije, ki je definirana kot premikanje extentov datotek po disku, tako da se združijo v neprekinjen blok. To ponavadi terja dve operaciji kopiranja: eno, da prestavimo posamezne extente na stran na nek nezaseden prostor, in drugo, da jih prestavimo na njihovo ustrezno mesto. Takšen sistem se uporablja, zato da so datoteke vedno shranjene nekje na disku v primeru, da pride do izpada elektrike. Reorganizacija datotek seveda ne spremeni njihove logične lokacije z vidika uporabnika. Obstajajo razni pristopi do programiranja algoritma defragmentacije: lahko prerazporedimo fragmente drugih datotek, da sprostimo dovolj prostora, za vsako datoteko lahko poiščemo nepretrgano zaporedje blokov, ki najbolj ustreza njeni velikosti (best fit), itd. Nekateri programi (MyDefrag) tudi predstavijo pogosto dostopane datoteke in programe na zunanji obroč trdega diska, kjer naj bi bil dostop hitrejši.

## Windows

**FAT** (File Allocation Table) datotečni sistem je bil prvotno razvit za diskete in je bil šele pozneje prirejen za uporabo na trdih diskih v sistemih PC in MS-DOS. Na disketah ni bilo potrebno skrbeti za postavitev datotek ob pisanju. Prva datoteka je bila preprosto shranjena na začetek, druga tik ob njej, itd. Če so se prvotne datoteke povečale so vedno postale fragmentirane, saj ob njih ni bilo prostora, kamor bi se lahko razširile. Če je bila katera vmesna datoteka izbrisana, se je naslednja delno shranila v ravnokar sproščen blok in na konec zasedenega prostora. Ko se je FAT začel uporabljati tudi na trdih diskih in so ti postajali vse večji, je začelo prihajati do opazne upočasnitve. V sistemih Windows 95 in 98, kjer je bilo veliko dostopov do diska in je dostopanje do fragmentiranih datotek močno upočasnilo delovanje, je bilo zato vključeno orodje za defragmentacijo imenovano Defrag, ki ga je bilo potrebno ročno pognati.

**NTFS** sistem je delno posnemal datotečni sistem ext (Extended File System) od Linuxa, ki ne shranjuje datotek tesno skupaj na začetku diska, temveč jih shranjuje enakomerno po celotnem disku, tako da je med posameznimi datotekami veliko prostora za shranjevanje novih ali širjenje že obstoječih datotek. NTFS predvideva in rezervira prostor za nove datoteke preden so zapisane na disk. Ob shranjevanju datoteke okrog nje dodeli majhen buffer prostor, vendar pa čez čas vseeno pride do fragmentacije, zato je od Windowsa 2000 naprej v sistem vključeno defragmentacijsko orodje, ki se samodejno zažene v ozadju.

## Linux in macOS

Linux datotečni sistemi **ext2**, **ext3** in **ext4** poleg zgoraj omenjene uporabljajo veliko tehnik, ki tako minimirajo fragmentacijo, da defragmentacija ni potrebna tudi po več letih uporabe diska. Če se

izkaže, da datoteka potrebuje več prostora, jo Linux poskusi prestaviti na novo lokacijo, kjer bo dovolj prostora za celotno datoteko. Datotečna sistema ext4 in reiser4 uporabljata tudi tehniko zakasnjene alokacije oz. allocate-on-flush, pri kateri se le zmanjša števec prostora na disku, vendar pa se prostor ne dodeli datoteki, temveč se le-ta obdrži v pomnilniku, kolikor dolgo je mogoče. Ko je datoteko potrebno zares shraniti na disk, je njena končna velikost ponavadi zanesljivo znana in je manj možnosti, da bo prišlo do fragmentiranja. Če pa začne prihajati do fragmentacije, datotečni sistem sam poskuša prerazporediti datoteke, tako da jo čimbolj zmanjša. Do problema pri naštetih načinih shranjevanja datotek lahko pride, če primankuje prostora na disku ali pa če delamo z zelo velikimi datotekami.

MacOS datotečni sistemi **APFS** (Apple File System) in **HFS+** (Hierarchical File System Plus) uporabljajo podobne tehnike kot Linuxovi. Poleg tega se poskušajo izogibati uporabi ravnokar sproščenega prostora, saj je tak prostor ponavadi majhen in bi hitreje prišlo do fragmentiranja. Ob dostopu do posameznih datotek tudi poskuša le-te sproti defragmentirati, če so manjše od 20MB.

Za Linux ne obstaja veliko defragmentacijskih orodij – najbolj razširjeno je e4defrag. Ker je fragmentiranih datotek ponavadi zelo malo, jih ta program preprosto začasno prestavi nekam na konec diska in jih nato prepíše nazaj pravilno razporejene ter ne uporablja bolj naprednih algoritmov. Za Windows sisteme obstaja poleg Microsoftovega orodja tudi veliko third-party defragmentacijskih orodij. Vse so spisane v C-ju in uporabljajo Windows API funkcije za delo z diskom, edina razlika med njimi so uporabljeni algoritmi.

## Windows API

Na Windowsu je programiranje defragmentacijskega orodja mogoče s pomočjo treh kontrolnih kod, ki omogočajo, da aplikacija najde prazne clusterje, določi lokacije clusterjev datoteke, ki jo želimo defragmentirati in jih nato premakne. Te kode tudi poskrbijo, da ne pride do konfliktov z drugimi procesi, ki prav tako dostopajo do diska med defragmentacijo. Najprej uporabimo kodo FSCTL\_GET\_VOLUME\_BITMAP, da najdem prostor na pogonu, ki je dovolj velik, da sprejme celotno datoteko. Če je potrebno, premaknemo druge datoteke, da sprostimo dovolj prostora. V idealnem primeru bi bilo dovolj prostora za prvim extentom fragmentirane datoteke in ne bi bilo potrebno prestaviti nobene druge datoteke. Nato uporabimo kodo FSCTL\_GET\_RETRIEVAL\_POINTERS, da dobimo zemljevid trenutne razporeditve datoteke na disku. Nato se sprehodimo čez vrnjeno strukturo ter uporabimo FSCTL\_MOVE\_FILE, da sproti prestavimo vsak cluster na izbrano mesto. Vmes bo morda potrebno obnoviti bitmap ali retrieval strukturi, saj se lahko spremenita, ko drugi procesi pišejo na disk.

## Viri

<https://docs.microsoft.com/en-us/windows/win32/fileio/defragmenting-files>

<https://www.linux.org/threads/allocation-methods.8787/>

<https://support.apple.com/sl-si/guide/disk-utility/dsku19ed921c/mac>

<https://discussions.apple.com/thread/7063348>

<https://www.minitool.com/lib/disk-fragmentation.html#4.-external-fragments-and-internal-fragments-1744>

<https://www.linux-magazine.com/Online/Features/Defragmenting-Linux>

<https://www.howtoforge.com/tutorial/linux-filesystem-defrag/>

<https://www.howtogeek.com/362864/why-mac-hard-drives-don%E2%80%99t-need-defragmenting/>

<https://www.overclock.net/threads/yes-file-system-fragmentation-does-affect-ssd-read-speed.1538878/>

<http://www.hanselman.com/blog/the-real-and-complete-story-does-windows-defragment-your-ssd>

<https://sourceforge.net/projects/ultradefrag/>

<https://github.com/tytso/e2fsprogs/blob/master/misc/e4defrag.c>

<https://opensource.apple.com/source/ntfs/ntfs-91.20.2/newfs/layout.h.auto.html>