

# 40. SYSTÉMY SOUBORŮ, ALOKACE A EVIDENCE DISKOVÉHO PROSTORU

Soubor - abstraktní jednotka obsahující data

- má jednoznačnou identifikaci - jméno

- metadata o něm jsou buď přímo v adresáři nebo ve speciální struktuře (I-nod nebo B+stom)

Grub MFT  
a NTFS

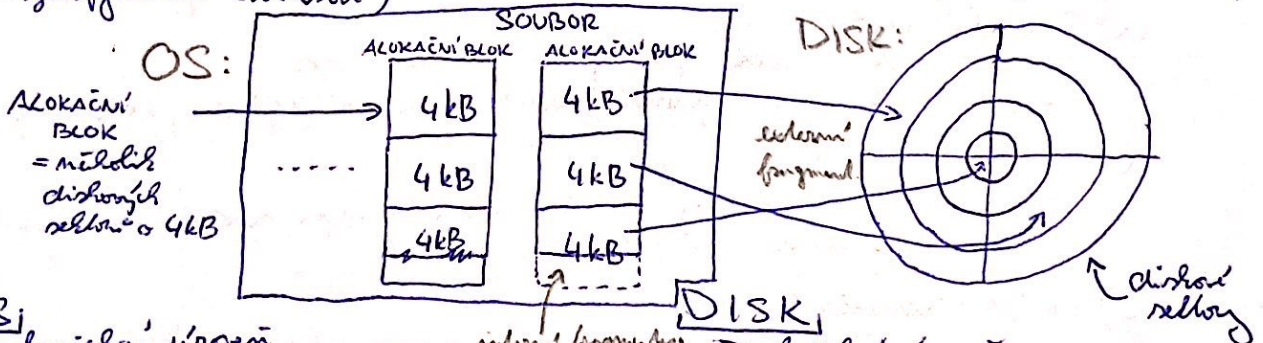
**Fyzická úroveň** - soubor z hlediska disku  
(z pohledu disku)  
- řápis a člen souboru se realizují po diskových sektorech  
- identifikují se fyzickou adresou toho sektoru  
- nemá žádnou ochranu

**Logická úroveň** - soubor jako soubor a ne jako místo na disku  
(z pohledu OS)  
- řápis a člen je po blocích / sektorech / alokačních blocích - jeden soubor má několik alokačních bloků  
- identifikují se jménem  
- je tam ochrana - vlastník, skupina, přístupová práva, ...

**diskový sektor** - nejmenší možná jednotka se kterou musí disk pracovat  
- řápis a člen musí disk po diskových sektorech  
- dříve 512 B, pak 1024 B ... až nyní 4096 B

**alokační blok** - nejmenší možná jednotka se kterou musí pracovat OS  
- skupina diskových sektorů první daného počtu  
→ obvykle  $2^N$  diskových sektorů

⇒ OS ukládá soubor po alokačních blocích a ten je rozdělen na diskové sektory po kterých disk zapisuje (OS pracuje s alokačními bloci a pojmenovávají soubory - soubor ukládá a kde po blocích X disk čte a alokační bloci na diskové sektory a po nich čte a zapisuje a identifikuje se fyzickou adresou)



**OS:**  
⇒ logická úroveň  
- soubor identifikovaný jménem  
- rozdělen na alokační bloci  
- řápis a člen po blocích  
- ochrana: vlastník, skupina, přístupová práva

interní fragmentace

⇒ fyzická úroveň  
- alokační bloci rozloženy na diskové sektory a ty identifikovaný fyzickou adresou  
- bez ochrany  
- člen a řápis po diskových

→ řeší se tím že se ze většího počtu alokačních bloků sestaví (nejmenší) soubor  
- vzniká interní a externí fragmentace

⇒ používají se nástroje pro defragmentaci

①



## System souborů (FS) poskytuje:

- 1) strukturu souborů - ukládání struktury souborů a jejich vnitřní struktury
- 2) abstrakci diskového prostoru
- 3) souborové operace - operace nad soubory (vytváření, mazání, přeměna, kopírování)
- 4) ochrana - práva, skupiny, minimální práva, řízení, ...)

### 1) Struktura souborů

- A) Logická struktura - na úrovni jádra OS
- bez struktury - pole slov
  - logické souborové formáty - rychlý přístup, interní fragmentace
  - logické souborové formáty - pomalý přístup
  - index - seznamy souborů - souborové formáty / formáty s klíčem  
- vyhledávání ze podle klíče (B strom)
- B) Vnitřní struktura - na úrovni vstupů a výstupů
- soubor je pole abstrakčních bloků
  - abstrakční blok je minimální jednotka na disku
  - souvisí s abstrakcí diskového prostoru
  - musí být mapování logické adresy (OS) → fyzické adresy (disk)
  - ↳ OS čte / zapisuje soubor → disk čte / zapisuje diskové sektory

#### Abstrakce ~~souborů~~ abstrakčních bloků

- je to způsob, jakým se rozumí abstrakce z hlediska OS a tedy ne přímo na disku ale v tom logickém prostoru - viz abstrakční bloky na předchozí stránce
- nejedná se o fyzickou abstrakci na disku ale o abstrakci těch abstrakčních bloků pro soubor z hlediska OS ⇒ Logická úroveň (viz předchozí stránka)

#### a) Souvislé abstrakce

- celý soubor v celku jako jeden abstrakční blok
- rychlý přístup ale nutná prealokace místa dopředu, externí fragmentace

#### b) Lineární seznam

- lineární (jednosměrný nebo obousměrný) seznam abstrakčních bloků
- nutná fragmentace ale přístup je pomalý

#### c) Index - seznamy abstrakce

- první bloky - index obsahující seznam abstrakčních bloků
- jednoduchý ale pomalý přístup
- poslední bloky

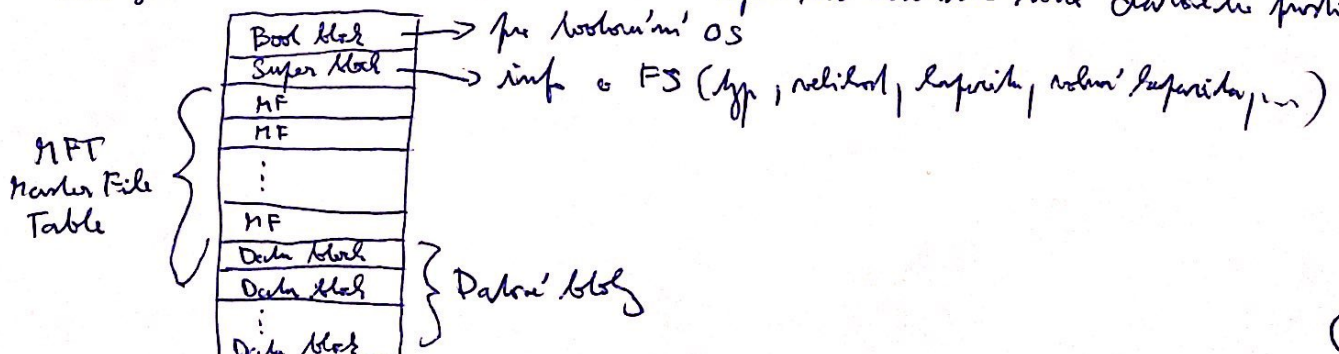


## Alokace diskového prostoru

- OS musí mít přehled o volném souboru na disku
  - musí mít také přehled o volném místě
  - a těchto znalostí volí minimální a relativní jednotky pro alokaci
- Jak velké zvolit bloky pro alokaci na disku?
    - malé - lépe se využívá místo na disku a nedochází k velké fragmentaci
      - je to proměnlivé
    - velké - rychlejší ale větší fragmentace
  - Jak vybrat volné místo pro alokaci?
    - souvislá alokace a také alokace po prvních blocích evincují volné místo a tak se pouze nahledne do té evidence
    - alokace proměnné délky - musí také evincovat ty volné úseky ale také je musí dělit a spojovat aby byla malou interní fragmentací a aby se snížila externí fragmentace
  - pro optimalizaci operací disk je dobré mít bloky co nejblíže
  - blok jednotky souboru je dobré mít blízko sebe

## Příklady FS:

- UFS - Unix File System
  - alokační blok = diskový sektor → 2°
  - volné bloky se hledají index - seznamem
  - využívá I-naly a tímto jejich přímými a nepřímými odkazy na diskové jednotky
- BSD
  - vychází z UFS
  - alokační bloky jsou stále větší než diskový sektor
  - volné bloky se hledají listovou mapou
- EXT2 / EXT3
  - vychází z BSD
  - alokační bloky jsou trochu menší
  - také listová mapa
- NTFS (New Technology File System) → jsou to tam metody
  - všechny soubory a adresáře jsou poprány pomocí MFT (Master File Table)
  - což je tabulka I-naly a nachází se opět na začátku toho diskového prostoru



# Funkce FS

- strukturují soubory [ struktura všech souborů  
interní struktura souborů
- alokuje místo na disku - alokuje diskový prostor
- poskytuje operace nad soubory
- ochrana



## 1) Strukturování souborů

- logická struktura
  - struktura z pohledu té logické ~~struktury~~ abstrakce
  - z pohledu OS
    - a) bez struktury souborů
    - b) logické párování s souborovými jmény
    - c) — " — pomocí klíčů
    - d) indexování souborů - indexují se klíčem
- fyzická struktura
  - soubor identifikován jménem
  - skládá se z abstraktních bloků a fyzických sektorů
  - musí být mapování abstraktních bloků → diskové sektory

- jak abstraktní fyzické abstraktní bloky? jak velké abstraktní bloky?

- 1) celý soubor jako jeden blok
- 2) soubor jako lineární seznam abstraktních bloků
- 3) index - seznam abstraktních bloků - index obsahuje seznam abstraktních bloků

## 2) Abstraktní diskový prostor

- OS musí vědět jak je disk organizován a co je volné a co plně
- přes umístění a velikost bloků
- pro jeden soubor je třeba mít bloky blízko sebe - což je externí fragmentace  
mnozí registry
- Velikost bloků:
  - velké - rychlejší operace ale neefektivní využití prostoru
  - malé - efektivní využití prostoru ale pomalé

Umístění bloků: - kam to abstraktní? kde je místo?

- v abstraktní po fyzické bloky se podle nové evidence kládou  
soubor plně a kládou volné
- ~~je~~ v abstraktní po fyzické bloky se musí shánět a redistribuovat  
bloky a pak vložit evidence