**ROMÂNIA**

**MINISTERUL APĂRĂRII NAŢIONALE**

**ACADEMIA TEHNICĂ MILITARĂ**

**„FERDINAND I”**

**FACULTATEA DE SISTEME INFORMATICE ȘI SECURITATE CIBERNETICĂ**

**Specializarea: Calculatoare şi sisteme informatice pentru apărare**

**şi securitate naţională**



**METODE DE EXTRAGERE ȘI RECUPERARE A DATELOR ȘTERSE DIN MEDIILE DE STOCARE**

CONDUCĂTOR ŞTIINŢIFIC:

**Conf. univ. dr. ing. LAURENȚIU MĂRGĂRIT**

ABSOLVENT:

**Std. sg. maj. Andreea-Elena AVRAM**

Conţine \_\_\_\_\_\_\_\_\_\_\_ file Inventariat sub nr. \_\_\_\_\_\_

Poziţia din indicator: \_\_\_\_

Termen de păstrare: \_\_\_\_\_

**BUCUREŞTI**

**2020**

LISTĂ ABREVIERI

|  |  |  |
| --- | --- | --- |
|  | FAT | File Allocation Table |
|  | NTFS | New Technology File System |
|  | EOF | End of file |
|  | FTK | Forensic Toolkit |
|  | APFS | Apple File System |
|  | MD5 | Message Digest 5 |
|  | SHA | Secure Hash Algorithm |
|  |  |  |
|  |  |  |
|  |  |  |

CUPRINS

[LISTĂ ABREVIERI 2](#_Toc60936454)

[1. INTRODUCERE 4](#_Toc60936455)

[1.1. Prezentare generală 4](#_Toc60936456)

[1.2. Importanța temei 4](#_Toc60936457)

[1.3. Metode de recuperare de date 5](#_Toc60936458)

[2. STATE OF THE ART 5](#_Toc60936459)

[2.1. Aplicații de recuperare de date 5](#_Toc60936460)

[2.1.1. EnCase 6](#_Toc60936461)

[2.1.2. FTK Imager 6](#_Toc60936462)

[2.1.3. FTK 7](#_Toc60936463)

[2.1.4. WinHex 7](#_Toc60936464)

[2.1.5. EaseUS 8](#_Toc60936465)

[2.1.6. Recuva 8](#_Toc60936466)

[2.2. Literatură relevantă recentă 8](#_Toc60936467)

[2.2.1. Scalpel: A frugal, high performance file carver – Golden G. Richard, Vasil Roussev, 2005 8](#_Toc60936468)

[2.2.2. A Comprehensive Literature Review of File Carving - Rainer Poise, Simon Tjoa, 2013 9](#_Toc60936469)

[2.2.3. The Evolution of File Carving - Anandabrata Pal, Nasir Memon, 2009 10](#_Toc60936470)

[2.2.4. Performance Analysis of File Carving Tools - Thomas Laurenson 11](#_Toc60936471)

[2.3. Concepte teoretice 13](#_Toc60936472)

[2.3.1. Termeni 13](#_Toc60936473)

[2.3.2. Suporturi de memorie existente 14](#_Toc60936474)

[2.3.3. Modul de functionare al unui Hard Disk 14](#_Toc60936475)

[2.3.4. FAT32 14](#_Toc60936476)

[2.3.5. NTFS 15](#_Toc60936477)

[2.3.6. Moduri de conectare la dispozitive de recuperare 15](#_Toc60936478)

# INTRODUCERE

Proiectul își propune realizarea unui studiu privind diversele metode de recuperare de date și implementarea unui utilitar capabil să recupereze fișiere de tip docx si pdf în absența informațiilor legate de sistemul de fișiere.

## Prezentare generală

Proiectul este format din 4 părți:

1. Prezentarea tipurilor de suporturi de memorie existente precum si modurile de conectare ale acestora la diverse echipamente destinate recuperării datelor în funcţie de interfaţa de conectare a suportului de memorie.
2. Prezentarea organizării sistemului de gestiune a fișierelor in sistemul de operare Windows.
3. Prezentarea modului de funcționare a unui sistem de recuperare de date.
4. Realizarea imaginii unui suport de memorie și analiza acesteia pentru identificarea spațiului nealocat în vederea recuperării datelor șterse.

## Importanța temei

Domeniul recuperării de date își găsește aplicabilitate într-o multitudine de arii de lucru: fie că vorbim de o situație uzuală în care un utilizator își pierde datele din diverse motive (fie șterse cu bună știință sau pierdute ca urmare a coruperi), fie că vorbim de o situație în care singurele dovezi criminalistice au fost șterse, fie că vorbim de interesele unei instituții, recuperarea de date devine un factor cheie in contextul erei informatice și prezintă numeroase direcții de viitor pentru dezvoltare și cercetare.

Este deosebit de important ca un investigator sa aibă o bună cunoștiință a detaliilor sistemelor de fișiere și a opțiunilor prezente pe piață în ceea ce privește aplicațiile de recuperare de date pentru a putea lua decizii optime și pentru a putea interpreta corect rezultatele.

## Metode de recuperare de date

Header-Based Carving – utilizeaza informatiile din anteturile fișierelor. Header-footer carving este cea mai simplă dintre metode în care se caută un șablon ce marchează începutul și sfârșitul de fișier. Ca alternative la această metodă se pot aminti *header-maximum size carving* în care, după descoperirea unui antet se calculeaza cât de departe se poate afla sfârșitul fișierului sau *header-embedded length carving* în care, unde există posibilitatea, se caută informația din antet care specifică punctul de identificare a subsolului fișierului.

File Structure Carving – se bazează pe structura internă a fișierului în care anumite informații din conținut pot ajuta la reconstrucție.

Block-Based Carving – calculează metainformațiile conținutului unui bloc de date, de obicei prin metode statistice.

File Validation – validarea aduce confirmarea că rezultatul programului este intr-adevăr un fișier valid. Așadar, un validator de format este o functie care acceptă un bloc de date și determină dacă corespunde unui anumit format de fișier.

# STATE OF THE ART

## Aplicații de recuperare de date

La momentul actual exista o multitudine de aplicații de recuperare de date pe care le vom împărți in doua categorii: cele care recuperează date pe baza informațiilor contextuale disponibile (EaseUS, Recuva) și cele care recuperează date fără informații contextuale (EnCase, FTK, WinHex).

### EnCase

EnCase este una dintre cele mai folosite solutii de recuperare de date, fiind aleasa Best Computer Forensic Solution 10 ani la rand de catre SC Magazine pentru viteza, flexibilitatea si functionalitatile sale.

Cateva dintre functionalitatile sale includ:

* Motor de indexare imbunatatit – indexare avansata, rapida si optimizata;
* Raportare usoara – sabloane personalizabile care permit utilizatorilor sa creeze rapoarte prefosionale si usor de citit;
* Extensibilitate – prin EnScripts – comenzi automate ce permit automatizarea sarcinilor pentru mai multa eficienta.
* Automatizarea fluxului de lucru – permite utilizatorilor sa navigheze usor in program;
* Suport de criptare actualizat – investigatorii pot obtine dovezi criptate fara grija coruperii datelor;
* Suport pentru Apple File System (APFS) – suporta APFS si trimite output-l ca un fisier Encase;
* Capabilitati pentru a copia volume – permite realizarea de Volume Shadow Snapshot (VSS) sau Volume Shadow Copy pentru a recupera volume intregi de date;
* Altele – Pathways, Keyword Searching, 4th Pane, Prioritization, Entropy Analysis, Email Processing, Internet Artifact Processing.

### FTK Imager

FTK Imager permite previzualizarea datelor pentru a decide daca sunt necesare investigatii ulterioare cu o unealta de criminalistica, cum ar fi AccessData Forensic Toolkit. De asemenea, este folosit si pentru a crea copii ale datelor fara a face modificari asupra continutului original.

Cateva dintre functionalitatile sale includ:

* Parsarea sistemelor de fisiere XFS – pentru colectarea de pe mediile RHEL Linux;
* Capturarea si vizualizarea imaginilor APFS – pentru produsele Mac;
* Crearea de imagini criminalistice – a unor Hard Disk-uri, CD-uri, DVD-uri, dispozitive USB, foldere sau fisiere individuale.
* Previzualizarea fisierelor, folderelor sau a imaginilor criminalistice;
* Montarea unei imagini pentru o vizualizare read-only – petru a vedea continutul exact cum a fost vazut de utilizator pe dispozitivul original;
* Observarea si recuperarea fisierelor sterse din Cosul de Gunoi care nu au fost inca suprascrise;
* Crearea hash-urilor fisierelor – pentru a verifica integritatea datelor, folosind MD5 si SHA-1.

### FTK

FTK este singura solutie care utilizeaza o baza de date single case, reducand costurile si complexitatea in cazul seturilor de date multiple case.

Spre deosebire de alte solutii, FTK ofera:

* Suport multi-threading/multi-core;
* Cautare rapida si rezultate consistente – permite schimbarea optiunilor de indexare pe parcursul procesarii;
* Integrare Qview;
* Recunoastere faciala, de imagini si de obiecte;
* Filtru dedicat pentru date mobile;
* Procesare distribuita;
* Vizualizarea datelor in diferite formate pentru a determina relatiile dintre ele.

### WinHex

WinHex este o aplicatie folosita atat ca si editor hex, dar si ca unealta pentru analiza, editarea, recuperarea si stergerea datelor.

Cateva dintre functionalitatile sale includ:

* Citirea si editarea directa a hard disk-urilor, CD-urilor, DVD-urilor si a altor medii;
* Interpretarea a 20 de tipuri de date;
* Editarea tabelelor de partitionare si a altor structuri;
* Unirea si divizarea fisierelor;
* Analizarea si compararea fisierelor;
* Cautare si inlocuire;
* Recuperare de date;
* Criptare, hashing si si realizarea de sume de control;
* Clonarea si realizarea de imagini;
* Stergerea hard disk-urilor;
* Cautarea textelor bazate pe cuvinte cheie;
* Crearea de tabele cu continutul unui hard disk.

### EaseUS

EaseUS cuprinde unele dintre cele mai avansate caracteristici existente. Un mare avantaj este reprezentat de posibilitatea de a recupera fisiere corupte sau inaccesibile dintr-un mediu de stocare extern. De asemenea, este oferita posibilitatea previzualizarii integritatii datelor inainte de a se incepe recuperarea.

### Recuva

Recuva este una dintre cele mai populare solutii gratuite de recuperare de date.

Cateva avantaje ale acestuia sunt:

* Prezinta doua moduri de lucru: Wizard si Advanced, in functie de nivelul de cunostiinte tehnice alte utilizatorului;
* Are o interfata simpla si este suportat de majoritatea versiunilor sistemelor de operare Windows si Mac
* Recuperarea este rapida;
* Posibilitatea de a rula o scanare „adanca” (deep scan) prin care se cauta date in cele mai adanci parti ale hard disk-ului;
* Suporta o gama larga de tipuri de fisiere.

## Literatură relevantă recentă

### Scalpel: A frugal, high performance file carver – Golden G. Richard, Vasil Roussev, 2005

Lucrarea este una dintre primele care vizează subiectul recuperării de date prin metoda de file carving.

Tehnica abordată este bazată pe semnăturile fișierelor (secvențe de octeți caracteristice fiecărui tip de fișier prezente în antetul si subsolul acestuia), deci se puteau recupera doar fișiere nefragmentate.

Documentul prezintă câteva aspecte legate de optimizarea procesului de file carving și cuprinde o descriere a detaliilor de design si implementare a utilitarului Scalpel – aplicatie open source cu o viteza de operare mai rapidă comparativ cu alternativele momentului respectiv.

### A Comprehensive Literature Review of File Carving - Rainer Poise, Simon Tjoa, 2013

Documentul prezintă inițial câteva aspecte teoretice legate de recuperarea logică. Recuperarea logică de date se diferențiază de cea fizică prin tipul de dispozitive necesare pentru a obține acces la informațiile de pe mediul de stocare considerat. Recuperarea logică utilizează informațiile așa cum pot fi obținute folosind echipament standard (carduri SD, stick-uri de memorie). Recuperarea fizică utilizează hardware specializat pentru obținerea de informații (ex. Interpretarea magnetismului rămas in cazul unui Hard Disk).

Lucrarea descrie o metodă de recuperare de date printr-un proces in trei pași:

1. Se construiește o imagine a memoriei și se identifică toate blocurile relevante;
2. Blocurile se împart pe categorii în funție de tipul lor;
3. Se aranjează blocurile in ordinea corectă.

În pasul 1 se prezintă modul în care metadatele ajută procesul de file carving. Metadatele sistemului de fișiere conțin informații despre dimensiunea block-urilor. Block-uri de dimensiuni mai mari înseamnă acuratețe crescută în faza de clasificare a fragmentelor. De asemenea, având aceste informații suplimentare, putem exclude anumite block-uri care nu prezintă interes și astfel reducem cantitatea de date de procesat[[1]](#footnote-1).

T. Holleboom și J. Garcia au analizat comportarea fragmentelor în relație cu spațiul rămas neocupat într-un cluster (cluster slack). Această abordare este deosebit de utilă dacă părți ale fișierului propus spre recuperare au fost deja suprascrise. Singurele informații care pot fi accessate în această situație din punct de vedere logic sunt acele rămășițe în spațiul neocupat, deoarece de obicei aceste spații raman neatinse în procesul de ștergere.

Pasul 2 constă in clasificare. După preprocesare, blocurile care nu pot fi asociate cu un fișier sunt trimise procesului de clasificare a fragmentelor, unde se determina tipul fișierelor de care aparțin. Abordările bazate pe potrivirea semnăturilor sunt potrivite pentru a identifica inceputul și sfârșitul fișierelor, dar și pentru a detecta puncte de fragmentare (fragmentation points).

În pasul al treilea, blocurile clasificate în funcție de tipul lor sunt reasamblate în ordinea corecta. Rezultatul acestui pas este fișierul întreg sau un fișier asamblat parțial.

Ca și concluzie, documentul atestă că performanța si acuratețea rămân principalele probleme ale aplicațiilor existente de file carving.

### The Evolution of File Carving - Anandabrata Pal, Nasir Memon, 2009

Documentul prezintă mai întâi câteva aspecte legate de metodele tradiționale de recuperare de date. Aceste metode se bazează pe structura sistemului de fișiere. Atunci când un fișier este șters, sistemul de operare marcheaza acea locație ca fiind disponibilă pentru a stoca informații. După ștergere, intrarea in tabela de fișiere ar putea sa fie încă prezentă, împreună cu informațiile care leagă clusterele de fișierele șterse. Un avantaj al acestei metode este ca zonele ne-șterse sunt usor de identificat și, astfel, doar zonele de pe disk nealocate trebuie parsate. Această metodă devine însă ineficientă în cazul în care nu avem informații despre aceste structuri sau ele au fost corupte.

De asemenea, este prezentat unul din modurile de lucru ale unui file carver ce se bazeaza pe structura fișierului. Având în vedere că nu sunt prezente informații legate de sistemul de fișiere, un carver trebuie mai întâi să decidă care sunt clusterele nealocate pe disk pentru a reduce volumul de date analizat. În acest scop, multe dintre aplicații sunt capabile să identifice un număr mare de fișiere care apartin de obicei sistemului de operare. Cele mai utilizate două software-uri in domeniu, Encase si Forensic Toolkit (FTK), prezintă și această opțiune.

Primele generatii de file carvers utilizau octeții de start din antet și octeții de final din subsol. Anteturile sunt folosite de sistemul de operare pentru a decide ce aplicație este necesară pentru a deschide fișierul. Un astfel de file carver va returna toate clusterele nealocate dintre aceste doua secvente.

Unele fisiere nu conțin subsol, însă conțin alte informații, cum ar fi dimensiunea fișierului (ex Windows BMP file).

Ulterior au apărut alte abordări, cum ar fi cele bazate pe “Hamiltonian Path”, “K-vertex disjoint path”, “Parallel unique path” (variație de la algoritmul Dijkstra pentru calcularea drumului cel mai scurt) sau “Shortest Path First” (cu cea mai buna acuratete, 88%, dar mai incet).

### Performance Analysis of File Carving Tools - Thomas Laurenson

Procesul de file carving poate deveni dificil și complex mai ales având în vedere diversitatea de unelte disponibile. Mulți dintre utilizatorii unor astfel de sisteme au dificultăți în a face o deosebire între diferitele soluții mai ales din perspecitiva capabilităților si limitărilor. Deși există multe cercetari, numeroase probleme încă apar printre profesioniști: ce unelte de file carving aduc cele mai bune performanțe din perspectiva procentului de fisiere recuperate, a acurateței sau a vitezei.

Documentul prezintă o comparație de performanță între 6 software-uri de recuperare de date: EnCase, FTK, WinHex, PhotoRec, Scalpel, Foremost. Printre criteriile după care au fost alese aceste opțiuni se numără numărul de tipuri de fișiere suportate, proprietățile pentru recuperare si disponibilitatea.

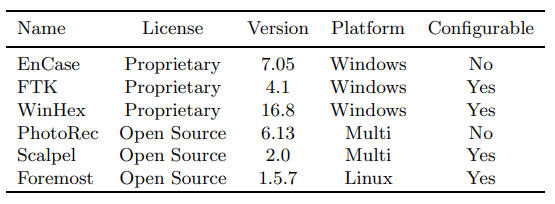
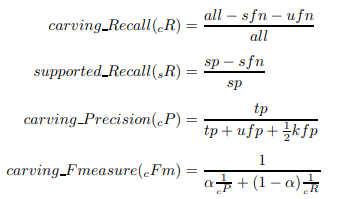


Fig. 1. Caracteristicile programelor utilizate în analiză

Procedura de testare a fost constituită de urmatoarele faze:

1. Determinarea cazurilor de adevărat pozitiv: se compară rezultatul returnat de program cu valorile din setul de date inițial. Fișierele rămase sunt verificate pentru a determina dacă ocupă aceleași blocuri precum fișierele din setul de date. Dacă oricare din aceste două condiții este îndeplinită, cazul este marcat ca fiind adevărat pozitiv.
2. Determinarea cazurilor de fals negativ: Se folosește o combinație între hashing-ul piecewise[[2]](#footnote-2) și analiza manuală pentru a determina eventualele cazuri de fals negativ.
3. Determinarea cazurilor știute de fals pozitiv: Se verifică fișierele de log pentru a identifica fișierele marcate incorecte sau corupte.
4. Determinarea cazurilor neștiute de fals pozitiv: Se marchează fișierele rămase.
5. Calcularea performanțelor: Se calculează cu ajutorul formulelor:



Rezultatele au fost centralizate în următorul tabel:

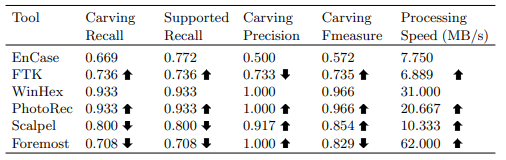


Fig. 2. Rezultatele testării

Câmpurile din tabel au urmatoarele semnificații:

* Carving Recall – Abilitatea unui utilitar de a extrage un număr cât mai mare de fișiere corecte din setul de date. Scorurile slabe indică de obicei tipuri de date sau structuri nesuportate.
* Supported Recall – Similar cu Carving recall, dar se referă exclusiv la tipurile de fișiere suportate. Scorurile slabe indică de obicei incapacitatea utilitarului.
* Carving Precision – Masoară exactitatea utilitarului. Scorurile slabe indică de obicei un număr mare de cazuri de fals pozitiv.
* Carving Fmeasure – Rezultatele de la Recall si Precision sunt combinate pentru a oferi un scor general, deci un indicativ de comparație.

Săgețile indică o creștere sau o scădere de performanță comparativ cu rezultatele obținute de S.J.J. Kloet în 2007 într-un studiu asemănător[[3]](#footnote-3).

## Concepte teoretice

### Termeni

Sector – cea mai mică cantitate de date (secvență de biti) care poate fi scrisă sau citită de pe un hard disk o dată.

Cluster (block) – grup de sectoare – cea mai mică cantitate de date care poate fi adresată folosind un sistem de fisiere.

Fragment – unul sau mai multe clustere.

Fișier continuu (contiguous file) –fișier stocat în memorie într-o secvență continuă de blocuri organizate într-o ordine logică.

Fișier fragmentat (fragmented file) – fișier stocat în memorie în două sau mai multe fragmente separate de un număr necunoscut de clustere.

Fișier defragmentat (defragmented file) – fișier stocat în memorie într-un singur fragment.

Fișier parțial (partial file) – fișier incomplet în care o parte a acestuia nu este disponibilă. Acest lucru apare atunci când o parte a fișierului a fost suprascrisă de alte date.

Fișiere încorporate (embedded files) – termen care se referă la situația în care conținutul unui fișier este stocat într-un alt fișier (ex. Un fișier JPEG este încorporat într-un document Microsoft Word, arhive ZIP, etc.).

Punct de fragmentare – locul de pe disk unde un fragment al unui fișier este urmat de un fragment al altui fișier.

File carving – proces de recuperare de date bazat pe conținutul fișierelor în scenarii în care metadatele sistemului de fișiere nu sunt disponibile.

### Tipuri de suporturi de memorie

#### HDD

#### SSD

#### Suporturi flash

### Modul de functionare al unui Hard Disk

Hard disk-ul este un dispozitiv de stocare nevolatila a datelor, ce servește un rol deosebit de important în următoarele aspecte legate de sistemele de calcul:

* **Performanță**. Viteza cu care calculatorul realizează procesul de bootare și viteza de încărcare a programelor sunt in legătura directă cu viteza hard disk-ului.
* **Capacitatea de stocare**.
* **Suport software**. Software-urile moderne solicită mai mult spațiu de stocare și viteză mai mare pentru a rula eficient.
* **Fiabilitate**. Importanța unui dispozitiv hardware poate fi măsurată prin cât de grave devin lucrurile atunci când acesta cedează. Având în vedere acest aspect, hard disk-ul este de departe cea mai importanta componentă întrucât hardware-ul poate fi înlocuit, însă datele stocate nu.

#### Specificatii tehnice

Principalele specificații tehnice ale unui hard disk sunt:

* **Capacitatea**
* **Viteza de rotație** – viteza cu care se rotește un disk, masurată în RPM (rounds per minute)
* **Timpul mediu de căutare**
* **Latența medie** – perioada de timp în care dispozitivul așteaptă ca sectorul căutat să ajungă la actuator, dupoă poziționarea corectă a acestuia
* **Timpul mediu de acces** – reprezintă o combinație a tuturor celorlalte specificații
  + Access Time = Command Overhead Time + Seek Time + Settle Time + Latency
* **Mărimea buffer-ului (Cache)**
  + Cache-ul reprezintă o zona de memorie rapidă ce reține datele accesate recent cu scopul de a scurta timpul de căutare.
  + Atunci cand datele sunt citite sau scrise în memoria principală, se creeaza o copie care va fi salvată in cache, împreună cu adresa asociată din memoria principală. Citirile ulterioare sunt monitorizate pentru a vedea daca datele cerute se află deja în cache. Daca sunt (cache hit), atunci sunt returnate imediat și căutarea în memoria principală este abandonată. Daca datele nu se află in cache (cache miss), atunci ele sunt aduse din memoria principală si salvate in cache.
  + Cea mai importantă caracteristică a memoriei cache este **Hit Rate** – câte dintre accesarile de memorie au fost satisfacute de cache. De obicei această caracteristică depinde de marimea cache-ului relativ la memoria principală.
* Zgomot și temperatură.

#### Structura fizica

Hard disk-ul este alcătuit din cinci componente principale:

* Platane – discuri neflexibile din metal acoperite cu un strat de material magnetizabil
* Brat deplasabil – cate unul pe fiecare platan. La capatul bratului se afla capul de citire/scriere
* Placă electronică de control logic
* Sistem electromecanic – blocheaza capetele de citire/scriere atunci cand discul este oprit
* Motor electric – asigură rotirea cu viteza constantă a platanelor.

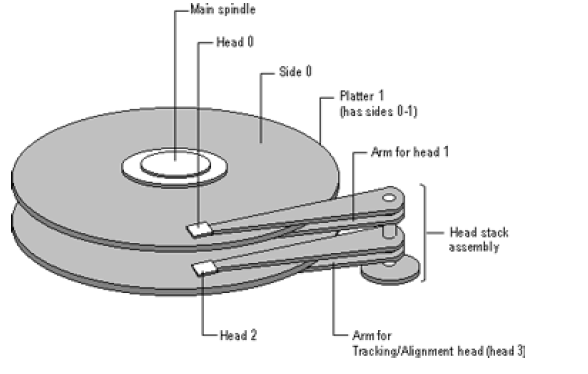


Fig. 1. Structura fizică a unui hard disk

Platanele sunt organizate in structuri specifice pentru a permite stocarea si regăsirea organizată a datelor. Fiecare platan este împărțit în benzi concentrice subțiri. Aflat într-o anumită poziție, capătul poate scrie sau citi o singură bandă circulară ce inconjoară platanul numita pistă. Secțiunile de a lungul pistei se numesc sectoare. Sectorul este cea mai mică unitate de adresare a informației stocate pe un hard disk și conține, de obicei, 512 octeți. Mai multe sectoare continue foarmează un cluster.

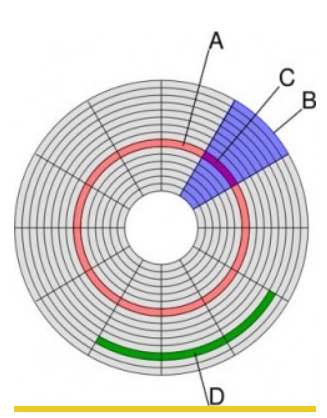


Fig.2. Organizarea platanelor

A – Pista; B – Sector de piste; C – Sector; D – Cluster.

Pentru sistemul de operare, pistele sunt mai mult structuri logice, decat fizice, si sunt stabilite în momentul în care hard disk-ul suferă procesul de formatare low level. Pistele sunt numerotate începând de la 0 (în marginea exterioară) și sunt incrementate către centru.

Stica de platane se rotește cu viteză constantă. Atunci când capul de citire/scriere se află in centrul discului, suprafața peste care se află se rotește cu o viteza mai mare decât în momentul in care acesta se află către margini. Pentru a compensa aceasta diferență, pistele din exterior sunt populate mai puțin dens decât cele din interior pentru a putea fi citită aceeasi cantitate de date în aceeasi perioadă de timp.

Spațiul de pe disk este umplut conform unor standarde. O parte a platanului conține spațiu rezervat pentru informații hardware despre poziționare și nu este accesibilă sistemului de operare. Aceste informații sunt scrise in fabrică in timpul procesului de asamblare.

#### Organizarea logica

Metoda optimă de stocare a unui fișier pe disk este într-un șir continuu de date. Cum multe fișiere sunt mai mari de 512 octeti (dimensiunea unui sector), alocarea sectoarelor pentru a stocare rămâne la alegerea sistemului de fișiere.

#### Regiuni

**MBR** (Master Boot Record)

Este primul sector fizic (cilindrul 0, head 0, sectorul 1) al primului hard disk din sistem (primul hard disk cu BIOS device number 0x80). La pornire, BIOS-ul încarcă conținutul din MBR la o adresă fixă din memorie și ii permite să ia controlul. Apoi acest cod încarcă sistemul de operare de pe o partiție bootabilă a hard disk-ului.

**DBR** (DOS boot record)

Este sectorul 1 din cilindrul 0, coloana 1 a hard disk-ului și este primul sector pe care îl vizitează sistemul de operare. Acesta conține programul de bootare si un BPB (BIOS Parameter Block). Principala sarcină a programului de bootare este de a determina dacaă primele doua fișiere din directorul root al partiției sunt fișiere de bootare a sistemului de operare. Dacă acestea există, se încarcă IO.SYS în memorie și se începe executarea lui. PBP înregistrează sectorul de început, sectorul final, formatul de stocare al fișierelo, dimensiunea directorului root, numarul FAT și dimensiunea celulei alocate.

**DIR**

Numit și FDT (File Directory Table), se află în sectorul root, imediat dupa cel de-al doilea FAT (backup FAT) și înreagistrează fiecare celulă de start a fișierelor.

**DATA**

DATA este locul efectiv unde datele sunt stocate. Se afla imediat dupa DIR, acoperind cea mai mare parte a hard disk-ului.

#### Interfete

Interfețele hard disk-ului definesc mijloacele fizice si logice prin care dispozitivul se conectează la calculator. Un calculator modern utilizează una sau mai multe dintre următoarele interfețe:

* **ATA** (Advanced Technology Attachment) – cunoscut și ca **IDE** (Integrated Drive Electronics) se împarte in două tipuri distincte: Parallel ATA (P-ATA) și Serial ATA (S-ATA).
  + **P-ATA** foloseste cabluri late cu conectori de 40 de pini pentru conectarea cu placa de bază. Cablul are de obicei 3 conectori, unul pentru placa de bază și ceilalți 2 pentru hard disk-uri. Daca două hard disk-uri sunt conectate la același controller, unul trebuie definit drept “master”, iar celălalt “slave”.
  + **S-ATA** folosește cabluri mult mai subțiri, aspect care îmbunătățește fluxul de aer în carcasă. Conectorii sunt construiți astfel încât să nu permită conectarea greșită.
* **DMA** (Direct Memory Access) - ss

### Sisteme de fișiere

#### FAT/////////////////////////////////////////////////

FAT este un sistem de fisiere dezvoltat pentru MS-DOS si este principalul (primary) sistem de fisiere pentru versiunile de consumator a Microsoft Windows. Este considerat relativ simplu si este folosit adesea pentru a distribui date intre cateva sisteme de operare care booteaza pe acelasi calculator (mediu multi-boot). Este de asemenea folosit pe solid-state memory cards si alte dispozitive similare). Marele dezavantaj este ca atunci cand fisierele sunt sterse si alte fisiere noi sunt scrise, fisierele se impart in multe parti, facand citirea si scrierea foarte incete. O solutie pentruaceasta problema este defragmentarea, insa este un proces de durata si trebuie repetat frecvent pentru a minte sistemul de fisiere “curat”.

In cadrul sistemului de fisiere FAT, cum se intampla in cazul majoritatii sistemelor de fisiere, nu se utilizeaza sectoare individuale, din motive de performanta. Folosind sectoare individuale, procesul de management al disk-urior devine dificil deoarece fisierele sunt impartite in bucati de 512 octeti. Urmarirea unui numar atat de mare de bucati este foarte consumator de timp si resurse.

Pentru a manageria fisierele intr-un mod relativ eficient, sistemul FAT grupeaza sectoarele in block-uri de dimensiuni mai mari numite clustere sau unitati de alocare. Dimensiunea clusterului nu are, insa, o dimensiune predefinita, ci este determinata deinsusi dimensiunea disk-ului. Sectoarele intr-un cluster sunt continue, deci fiecare cluster este un spatiu continuu pe disk.

Dimensiunea clusterelor si, implicit, dimensionarea partitiilor au un impac importat in performanta si utilizabilitate. Dimensiunea clusterului este determinata la momentul in care hard disk-ul este partitionat. Unele utilitare de partitionare, cum ar fi Partition Magic pot altera dimensiunea clusterului unui partitii existente.

Pentru a aloca un fisier in sistemul FAT32, mai intai fisierul este atribuit unui folder/director. Se creeaza o intrare in tabel care, impreuna cu numele fisierului, contine si numerului clusterului de inceput al fisierului, adica acel cluster unde incepe stocarea continutului fisierului. Sunt retinute si alte informatii, cum ar fi permisiunile de acces, momentul crearii sau modificarii. Sistemul FAT poate fi considerat un array de numere de clustere care indica catre urmatorul cluster al fisierului. Asadar, cand un fisier este recuperat, sistemul de fisiere cauta indexul clusterului in FAT si acceseaza urmatorul cluster al fisierului. Acesta poate indica catre un al cluster si procesul se repeta pana cand un cluster are valoarea 0xFF, indicand sfarsitul fisierului (EOF).

Atunci cand un fisier este sters, sistemul de operare atribuie valoarea 0x00 intrarilor corespondente in FAT, insa continutul pe disk inca este prezent. Desi intrarile in FAT pentru fiecare cluster sunt indicate ca nealocate, intrarea in director a fisierului inca indica spre clusterul de start, fiind utilizat un caracter special care ne arata ca aceasta intrare reprezinta un fisier sters.

Programele de recuperare verifica daca clusterul de start al fisierului este utilizat si presupune ca fisierul este stocat in clustere secventiale. Modul in care opereaza aceste programe este urmatorul: Urmaresc fiecare cluster nealocat dupa cel de inceput si le unesc. Daca intre doua clustere ale unui fisier exista un alt cluster al altui fisier, programul nu va putea face deosebirea intre cele doua si recuperarea va fi mult mai dificila fara a analiza continutul sau structura fisierelor.

//Poate niste screenshot-uri aici cu rezultatul comenzilor din cmd in care afisez dimensiunea sectoarelor si un ss din HexEdit cu DBR-ul

#### FAT32

FAT 32 este un sistem de alocate a fisierelor dezvoltat de Microsoft ce utilizeaza valori pe 32 de biti pentru intrarile din tabela FAT, permitand dimensiuni ale partitiilor pana la 2 TB. Dimensiunea maxima a unui fisier este de 4GB – 1 byte (232-1 bytes).

#### NTFS

In NTFS, informatiile despre fisiere este stocate intr-un B-Tree[[4]](#footnote-4), insa, spre deosebire de FAT, NTFS are o structura separata numita bitmap (BMP) folosita pentru a reprezenta statusul alocarii clusterelor. In BMP, fiecare cluster are asignata o valoare de 1 daca apartine unui fisier existent si 0 in rest.

In momentul stergerii, clusterele asociate in BMP sunt setate pe 0, insa, spre deosebire de FAT, legaturile dintre clustere nu sunt sterse. Asadar, recuperarea datelor este mult mai usoara. Asadar, procesul de recuperare consta in urmarirea legaturilor dintre clustere si verificarea ca acele clustere sa nu fi fost suprascrise sau alocate altui fisier.

Statisticile legate de fragmentare ale lui Simson Garfinkel arata ca rata de fragmentare a fisierelor importante din punct de vedere criminalistic (fisierele utilizator) cum ar fi e-mailuri, jpged sau doc este mai mare comparativ cu alte tipuri de fisiere. Rata de fragmentare a fisierelor jpeg este de 16%, documentele Microsoft Word au 17%, .avi au 22%, .pst au 58%. Fragmentarea apare de obicei datorita spatiului mic de pe disk, a editarii fisierelor sau a sistemului de fișiere însuși.

### Moduri de conectare la dispozitive de recuperare

In funcție de interfața de conectare a suportului de memorie

# Implementare software

## Cerinte

# Direcții vitoare

# Concluzii

# Bibliografie

1. A. Pal, H. T. Sencar, and N. D. Memon, “Detecting file fragmentation point using sequential hypothesis testing,” Digital Investigation, vol. 5, no. Supplement 1, pp. S2 – S13, 2008. [↑](#footnote-ref-1)
2. Kornblum, J.: Identifying Almost Identical Files using Context Triggered Piecewise Hashing. Digital Investigation 3(suppl.), 91–97 (2006) [↑](#footnote-ref-2)
3. Kloet, S.J.J.: Master’s Thesis: Measuring and Improving the Quality of File Carving Methods. Eindhoven University of Technology (2007) [↑](#footnote-ref-3)
4. B. Carrier , File System Forensic Analysis . Boston, MA: Pearson Education , Addison-Wesley Professional, 2005 . [↑](#footnote-ref-4)