UNIVERSITATEA POLITEHNICA BUCUREŞTI

**Facultatea de Electronică, Telecomunicaţii si Tehnologia Informaţiei**

**Sisteme de operare**

Lucrarea :

**Sisteme de fișiere - RAID**

**Profesor coordonator :**   **Student :**

prof. univ. dr. ing. Ştefan Stăncescu Pavel Mihai-Alexandru

Grupa 433 A

Anul universitar

2014 - 2015

**Cuprins:**

# Capitolul 1: Introducere în conceptul RAID.........................................................3

# Capitolul 2: Nivelele de organizare RAID............................................................4

# Capitolul 3: Analiză comparativă a nivelelor RAID...........................................9

# Capitolul 4: Recuperarea datelor în RAID.........................................................10

# Capitolul 5: RAID Software......................................................................................11

# Capitolul 6: S.M.A.R.T................................................................................................13

# Bibliografie ..................................................................................................................18

#

**Capitolul 1:Introducere în conceptul R.A.I.D.**

Conceptul de R.A.I.D. (“Redundant Array of Inexpensive Disks”) a fost introdus la sfârșitul anilor 1980 de către un grup de cercetători, profesorii David Patterson, Randy Katz si Garth Gibson (la acea vreme sutdent) de la Universitatea Berkeley, California. Aceștia doreau obținerea unei configurații de stocare mult mai eficientă, plecând de la ideea alăturării mai multor discuri dure (HDD), unei memorii (atât volatile cât si non-volatile) și folosirea unor procesoare pentru a gestiona întreg sistemul.

Cercetările în domeniul tehnologiei RAID s-au făcut pe 3 direcții, urmărind eventualele erori ce pot apărea. Capacitatea de stocare a reprezentat prima direcție de dezvoltare. S-a expus următoarea problemă: Dat fiind un număr de N discuri, câtă memorie poate fi pusă la dispoziția utilizatorilor? Răspunsul evident ar fi N, dar dacă deținem un sistem care păstrează câte două copii ale fiecărui bloc vom obține o capacitate utilizabilă de N/2.

Fiabilitatea a reprezentat a doua axă de dezvoltare, urmărind rezolvarea problemelor de defectare a disk-urilor din diferite configurații.

Perfomanța a reprezentat cea mai importantă caracteristică în alegerea unui sistem RAID și totodată cea mai dificilă de evaluat doarecere depinde în mare parte de volmul de muncă prezent pe disk.

Conceptul de bază din spatele unui RAID a constat în alăturarea unui server cu discuri unui computer. Această operație a presupus înlocuirea controllerului de disc cu un controller RAID și copierea datelor pe noul sistem de stocare. RAID este asemănător sistemului SLED (Single Large Expensive Disk) dar cu o performanța mult mai bună și o fiabilitate excelentă. De asemenea prețurile discurilor SCSI (Small Computer System Interface), performanța acestora, capacitatea de a atașa până la șapte unități pe un singur controller și faptul că sistemul de operare percepea RAID-ul ca un singur disc fără a face modificări de software, au fost câteva atuuri greu de egalat.

 În plus, odată cu introducerea RAID, s-a introdus posibilitatea de a distribui datele pe toate unitățile pentru a permite funcționarea în paralel. În acest scop, Patterson și colaboratorii acestuia au realizat mai multe scheme fiind cunoscute ca nivelele RAID 0 pâna la RAID 5. Termenul de nivel nu implică o ierarhie, ci doar șase organizări diferite posibile. [5]

**Capitolul 2: Nivelele de organizare R.A.I.D.**

***RAID nivel 0***

 RAID nivel 0 se descrie prin vizualizarea virtuală a unui singur disc (simulat de RAID) ca fiind împarțit în benzi de sectoare K fiecare, cu sectoarele de la 0 la k – 1 fiind banda 0, sectoarele de la k la 2k – 1 fiind banda 1 și așa mai departe. Spre exemplu, pentru k = 1, fiecare bandă este un sector, pentru k = 2 o bandă are două sectoare, etc.

 Acest nivel de organizare scrie benzi consecutive peste unități folosind metoda round-robin. Procedeul de distribuire a datelor în mai multe unități se numește striping. Spre exemplu daca software-ul emite o comandă pentru citirea unui bloc de date format din patru benzi consecutive, controller-ul RAID va împarți această comandă în patru comenzi separate, una pentru fiecare disc în parte, prelucrându-le in mod paralel.

 Acest nivel lucrează cel mai bine cu cererile de date mari. Dacă o cerere este mai mare decât numărul de unități ori mărimea benzii, unele unități vor primi cereri multiple astfel încât la terminarea primei cereri, automat va începe cea de-a doua, cu întârzieri minime. În această operație controllerul se face responsabil de împărțirea cererilor, alocarea comenzilor și asamblarea corectă a rezultatelor din memorie.

 RAID nivel 0 nu lucrează la fel de eficient cu sistemele de operare care cer pe rând datele din diferite sectoare. Rezultatele vor fi corecte, dar nu mai există prelucrare paralelă și prin urmare niciun câștig de performanță Un alt dezavantaj al acestui mod de organizare constă în fiabilitatea scăzută în comparație cu un sistem SLED: dacă un RAID este format din patru discuri, fiecare cu un timp mediu de funcționare de 20.000 ore, la 5000 de ore o unitate se va defecta și toate datele vor fi pierdute. Un sistem SLED cu un timp mediu de funcționare de 20.000 ore, va fi de patru ori mai fiabil. [1]



*Figura 1.1 Reprezentare RAID nivel 0*

***RAID nivel 1***

 Nivelul RAID 1 poate fi considerat un sistem RAID autentic. Acesta multiplică toate unitățile astfel încât să existe patru discuri primare și patru discuri pentru backup. Pentru o cerere de scriere fiecare bandă este scrisă de două ori. La citire, poate fi folosită o copie, distribuind cererea peste toate unitățile.

 Pentru acest nivel, putem afirma că performanța scrierii nu este mai bună în comparație cu un SLED, dar citirea poate fi de până la de două ori mai rapidă. Fiabilitatea este excelentă: dacă o unitate cedează, copia îi preia locul. Recuperarea datelor constă în simpla instalare unei noi unități și copierea unității de backup pe aceasta.



*Figura 1.2 Reprezentare RAID nivel 1*

***RAID nivel 2***

Spre deosebire de nivelurile 0 și 1 care lucrează cu benzi de sectoare, la nivelul 2 organizarea se face în baza cuvintelor (word) , fiecare având un ordin de mărime de un octet. Divizarea fiecărui word se face într-o pereche de patru nibbles (4 biți). Prin adăugarea unui cod Hamming fiecărui nibble se pot obține cuvinte de 7 biți, dintre care biții 1,2,4 sunt de paritate.

Cele șapte unități, reprezentate în figura de mai jos, au fost sincronizate în ceea ce privește poziția de rotație, reușind astfel să creeze condițiile implementării unei codări de tip 7 biți Hamming. Un exemplu în acest sens este calculatorul CM-2: luând date pe 32-biți, la care s-au adăugat șase biți de paritate pentru a forma un cod Hamming de 38 biți pentru cuvinte, la care s-a adăugat 1 bit suplimentar de paritate a word-ului. Rezultatele au arătat că pe rând, în fiecare sector, se puteau scrie 32 sectoare de date. De asemenea, pierderea unei unități nu ar fi cauzat probleme datorită codului Hamming care ar fi permis pierderea a câte unui bit de date din cuvintele de 39-biți.

Aceast mod de organizare impune ca toate unitățile sa fie sincronizate din punct de vedere rotațional, având în vedere ca se lucrează cu un număr substanțial de discuri. De asemenea, sunt necesare multe controllere datorită controlului făcut de codul Hamming care se execută pe fiecare bit. [3]

*Figura 1.3 Reprezentare RAID nivel 2*

***RAID nivel 3***

 RAID nivel 3 este o versiune simplificată a RAID nivel 2. În această organizare, un singur bit de paritate este calculat pentru fiecare cuvânt de date și scris într-o unitate de paritate. Ca și în cazul sistemului RAID nivel 2, unitățile trebuie sa fie sincronizate exact, deoarece cuvintele de date sunt răspândite pe mai multe unități. Existența unui singur bit de paritate creează confuzia unui sistem de detectarea erorilor, fără corectarea acestora. În cazul detectării unor erori oarecare acest lucru este adevărat, dar în cazul unei cedări de unitate, bitul de paritate execută corectarea erorilor din moment ce poziția sectorului avariat este necunoscută.

 Dacă un disc cedează, controllerul presupune că toți biții sunt zero. Dacă un word are o are o eroare de paritate, se consideră că bitul de pe discul care a cedat a fost 1, urmând sa fie corectat.

 Deși, ambele niveluri RAID 2 și 3 oferă posibilitatea de a lucra cu cantități mari de date, numărul de cereri separate care poate fi prelucrat per secundă nu este mai mare decât cel al sistemului SLED.



*Figura 1.4 Reprezentare RAID nivel 3*

***RAID nivel 4***

Din punct de vedere al capacității, RAID nivel 4 folosește un disc ce conține informații despre paritate pentru toate discurile, astfel capacitatea pentru un grup RAID este (N-1). Privind fiabilitatea, această configurație tolerează defectarea unui singur disc. În cazul defectării mai multor discuri, datele nu pot fi recuperate din lipsa unei modalități de a reconstrui informațiile pierdute. Din perspectiva performanței, citirea secvențiala poate utiliza toate discurile cu excepția discului paritate, furnizând astfel o lățime de bandă de (N-1)\*S MB/s.

 Scrierea secvențiala mai este cunoscută ca scriere full-bandă. RAID nivel 4 poate efectua o astfel de operațiune. Spre exemplu: în cazul in care blocurile 0,1,2 și 3 au fost trimise către RAID ca părți ale unei cereri de scriere, noua valoare a lui P0 se poate calcula prin efectuare unui XOR peste blocurile mai sus menționate și apoi scrise în mod paralel (inclusiv blocul de paritate) în cele 5 discuri. Astfel scrierea full-bandă este cea mai eficientă modalitate pentru RAID-4 de a scrie pe disc. Din punct de vedere al performanței de scriere, chiar dacă discul discul de paritate este în mod constant în uz în timpul operațiunii, clientul nu obține avantaje de performanță de la el.

*Figura 1.5 Scriere full-bandă în RAID nivel 4*

Pentru explicarea suprascrierii unui bloc putem considera exemplul de mai sus. Suprascrierea blocului 1 se poate face direct cu riscul de a creea anumite probleme: blocul P0 nu ar mai reflecta corect valoarea parității benzii. În acest caz, P0 trebuie de asemenea actualizat. Există două modalități de a face acest lucru. Prima constă în calcularea noii parități a blocului, citirea în paralel din bandă, toate celelalte date (și anume blocurile 0,2,3) și aplicarea unui XOR celor aflate in noul bloc 1. Rezultatul va fi noul bloc de paritate. Problema cu această tehnică este aceea că se produce o scalare în funcție de numărul de discuri, fiind necesar un număr mare de operațiuni de citit pentru a calcula paritatea.

Cererile de citire aleatoare ale unui bloc vor fi răspândite în toate discurile de date ale sistemului, cu excepția celui de paritate.



*Figura 1.6 Reprezentare RAID nivel 4*

***RAID nivel 5***

O mare parte din analiza RAID nivel 5 este identică cu RAID nivel 4. Din punct de vedere al capacității efective și toleranței asupra defectărilor cele două configurări sunt la fel. La fel de performante sunt citirea și scrierea secvențială. Latența unei singure cereri este identică cu cea a unui sistem RAID-4. Performanța unei citiri aleatoare este mai bună deoarece se pot utiliza toate discurile. De asemenea, scrierea aleatoare este îmbunătățită cu mult față de RAID-4. Spre exemplu, scrierea în blocurile 1 și 10 se va transforma în cereri către discurile 1 si 4 (pentru blocul 1 și paritatea sa) și cereri către discurile 0 și 2 (pentru blocul 10 și paritatea sa). Ca urmare, putem afirma că pentru un număr mare de cereri aleatoare vom putea ocupa într-un mod uniform toate discurile. Lățimea de bandă pentru scrieri mici este de N/4 \* R MB/s. Factorul de pierderi se datorează scrierilor RAID-5 care generează încă 4 operațiuni I/O. [3]



*Figura 1.7 Reprezentare RAID nivel 5*

***RAID nivel 6***

 RAID-6 extinde RAID-5 prin adăugarea unui alt bloc de paritate, folosind astfel două blocuri de paritate.

 Această configurare nu aduce penalizări operațiunilor de citire, dar cele de scriere sunt afectate de calculele blocurilor de paritate. Performanța variază foarte mult în funcție de modul în care RAID 6 este implementat in sistemul de stocare. Mai multe metode(algoritmi) au fost folosite pentru a pune în aplicare acest nivel: calcule de dublă verificare a datelor (partiate și Reed-Solomon),”orthogonal dual parity”,”diagonal parity” .

****

*Figura 1.8 Reprezentare RAID nivel 6*

**Capitolul 3: Analiză comparativă a nivelelor R.A.I.D.**

****

*Figura 1.9 Capacitatea, fiabilitatea și performanța niv. RAID*

În cazul scrierii într-un sistem de tip oglindă, media timpului de căutare este mai mare decât atunci scrierea se efectuează pe un singur disc, datorită celor două căutări care rulează in paralel pe cele două discuri. Astfel, scrierea aleatoare în două discuri va fi în general mai puțin performantă decât scrierea aleatoare pe un singur disc.

 La momentul schimbării discului de paritate, în RAID - 4/5, prima citire a vechii parități va cauza o căutare și o rotație completă. A doua scriere în discul de paritate va genera doar rotație.

 Pentru analiza latenței s-a folosit un T ce reprezintă durata unei cereri de a fi prelucrată de către un singur disc.

 Pentru cei care doresc performanță și nu le pasă de fiabilitate, striping-ul reprezintă cea mai bună alegere. În cazul în care se caută un echilibru între performanță si fiabilitate, oglindirea este cea mai bună soluție. Pentru cei care sunt pasionați de capacități de stocare mari și fiabilitate, RAID-5 este câștigătorul, cu prețul pierderii performanței în scrierile mici.



*Figura 2.0 Comparații nivele RAID (sursa: Wikipedia)*

**Capitolul 4: Recuperarea datelor în RAID**

Sistemele RAID protejează datele împotriva unei defectări a discului, permițând matricii RAID copierea datelor de pe disc pe o unitate de rezervă în timp ce este înlocuit. O avariere a unității fizice (hardware) precum un disc fisurat sau un cablaj rupt nu trebuie confundată cu defectarea de tipsoftware, la nivel logic, cauzate de diverese fișiere corupte care nu pot fi înlăturate prin înlocuirea unității de disc.

Există două situații principale în care un hard disc va fi înlocuit într-o matrice RAID.

Prima situație este reprezentată de momentul când matricea RAID identifică discul care nu rulează în parametrii, marcându-l astfel pentru înlocuire. Dacă există spațiu disponibil, se va efectua o copiere în bloc a vechiului disc, în noua zonă de memorie, care va deveni activă.

A doua situație de înlocuire are loc atunci când unitatea hard disc se defectează înainte de o înlocuire a acesteia. Datele de pe unitatea afectată trebuie sa fie reconstruite din datele de paritate de pe restul de unități active și scrise în spații de memorie disponibile. Remedierea acestei situații durează mult mai mult datorită calculelor care trebuie să aibă loc pentru reconstrucția datelor.

 Din punct de vedere tehnic, înlocuirea unității hard disk înainte de a se defecta se face mult mai rapid, datorită posibilității de a copia datele de pe vechiul disk pe cel care se instalează. Pentru siguranță se recomandă verificarea matricii RAID dacă este capabilă de înlocuiri de acest fel.

 Pentru intervențiile post-defectare trebuie ținut cont de următorii factori care influențeaza durata procesului de înlocuire: dimensiunile unităților din setul RAID, numărul acestora, prioritatea acordată de sistem matricii pentru reconstruirea activităților.

 Prioritatea procesului de reconstruire reprezintă al doilea factor important în influențarea timpilor procesului de rescriere/recuperare de date. Cu cât este prioritizat mai mult cu atât procesul va dura mai puțin, cu mențiunea că accesul la date se va face mai greu.



*Figura 2.1 Prelucrare date RAID*

**Capitolul 5: RAID Software**

Software-ul RAID poate fi implementat în mai multe feluri:

1. Ca o soluție ,,pură,, de tip software
2. O soluție hibridă ce conține un design de hardware specific astfel încât să crească perfomanța și să reducă nivelul de utilizare al UCP. (Unitatea centrală de prelucrare)

*Modelul software.* În acest caz, implementarea RAID reprezintă o aplicație ce rulează la host fără modificări ale hardware-ului. Acest tip de software utilizează driverele hard disk ce sunt atașate sistemului calculatorului prin interfața I/O sau printr-un HBA (host bus adapter). RAID devine activ imediat ce sistemul de operare a încărcat software-ul driverului. Aceste soluții sunt de obicei integrate în sistemul de operare al serverului. Costurile mici de implementare reprezintă cel mai mare avantaj al acestei soluții.

Avantajele acestui tip de implmentare:

* Costuri scăzute: funcționalitatea RAID este prezentă în funcționalitatea sistemului de operare. Singurul cost adițional este reprezentat de drivere.
* Neprotejat la boot (nu poate administra sau proteja datele la boot): Erorile de drive sau datele corupte din timpul boot-ului duc la un sistem inoperabil.
* Performanțe adiționale sunt încărcate pe server: performanțele sistemului sunt afectate direct de către complexitatea RAID. Cu cât numărul de drivere este mai mare și complexitatea mai diversificată cu atât performanțele sunt afectate.
* Migrații limitate ale sistemelor de operare: funcționalitatea RAID poate fi limitată în sistemul curent de operare. *Figura 2.2 Reprezentare software RAID*
* Nu se poate muta matricea de discuri către alte sisteme de operare sau alte versiuni al aceluiași sistem.
* Vulnerabil aplicațiilor de tip virus: deoarece RAID funcționează ca o aplicație pe sistemul calculatorului, aceste este predispus atacurilor de tip hack.
* Problema integrității datelor din timpul cedărilor de sistem: problemele de tip hardware sau software de pe server pot afecta consistența datelor și integritatea acestora.
* Fără memorie cache pentru scrierile înapoi: RAID software rulează doar în modul de scriere înainte, dar hardware-ul RAID poate scrie invers dacă dispune de o baterie, adăugând astfel incă un nivel de protecția datelor.

*Modelul hibrid.* În timp ce software-ul este de tip RAID, hardware-ul ajută la rezolvarea câtorva probleme ale softului. Acest gen de soluții vin de obicei cu hardware adițional (HBA cu un RAID-BIOS sau doar un BIOS integrat pe placa de bază). BIOS-ul face disponibilă funcționalitatea RAID atunci când sistemul este pornit, furnizând redundanță în timpul boot, reducând astfel impactul erorilor din RAID. De asemenea, majoritatea soluțiiilor dispun de o configurare BIOS care poate fi accesată la system boot. Aceasta permite o administrare mai ușoară a matricii RAID, fără a fi necesară instalarea instalarea sau rularea unui sistem de operare de pe un hard disk sau CD-ROM.

Avantaje și dezavantaje ale modelului hibrid.

* Costuri moderate: doar un HBA (card de tip plug-in) sau o memorie adițională flash pentru BIOS-ul integrat pe placa de bază. Poate include și un accelerator XOR dacă controllerul suportă RAID 5.
* Protejat la boot: nu există impact negativ asupra datelor la pornirea sistemului.
* Interfață grafică pentru utilizatori dedicată pentru administrarea RAID.
* Migrație limitată a sistemelor de operare: funcționalitatea RAID este dependentă de sistemele de operare cu rularea driver-ului la vârful nivelului sistemului.
* Vulnerabil la atacuri: deoarece RAID rulează ca o aplicație pe sistem, virușii pot afecta datele stocate pe acesta.
* Fără memorie cache pentru scrierile înapoi: RAID software rulează doar în modul de scriere înainte, dar hardware-ul RAID poate scrie invers dacă dispune de o baterie, adăugând astfel incă un nivel de protecția datelor. [8]



 *Figura 2.3 Comparație hardware-software*

**Capitolul 6: S.M.A.R.T.**

****

*Figura 2.4 S.M.A.R.T. tools*

S.M.A.R.T. (Self-Monitoring Analysis and Reporting Technology) este produsul celor de la IBM. Acesta a fost creat pentru a monitoriza starea unui disc folosind diverse metode și senzori. Un singur hard disk ATA poate avea până la 30 de valori măsurate reprezentând parametrii de funcționare și date statistice. Potrivit S.M.A.R.T. în momentul în care este detectată o problemă, hard disk-ul ar trebui sa efectueze un backup de date de 24 ore (pentru unitățile de capacități mari), de aceea este important ca S.M.A.R.T. să recunoască problemele din datele pe care le înregistrează și să anunțe ce parametrii nu sunt conform standardelor.

 Toate hard disk-urile moderne beneficiază de această tehnologie, în variante diferite, în funcție de producător.[6]

 Un atribut S.M.A.R.T. are următoarele câmpuri:

 **Identifier** (byte) ce reprezintă sensul atributului (spre exemplu, 5 = numărul de sectoare realocate, 194 = temperatura, etc. )

 **Data** (6 bytes) conține valorile măsurate furnizate de un senzor sau un numărător. Acestdate sunt apoi prelucrate de către un algoritm.

 **Threshold** (byte) valoarea limită a atributului.

 **Value** (byte) reprezintă durata de viață a discului (pentru un hard nou, acest număr este mare, 100/200/253 în funcție de model și este în scădere).

 **Worst** (byte) cea mai mică valoare gasită într-un ciclu de funcționare anterior.

 **Status flags** indică scopul principal al atributului. Un atribut poate fi critic din punct de vedere al funcționării (precizerea unei posibile defecțiuni) sau prezentarea unei statistici ce nu afectează in mod direct unitatea disk.

***Lista atributelor S.M.A.R.T. pentru hard disk-urile serial ATA***

|  |  |  |
| --- | --- | --- |
| 1 | **Raw Read Error Rate** | Errors occured while reading raw data from a diskIndicate problem with the disk surface or the read/write heads.*Critical attribute* |
| 2 | **Throughput Performance** | General throughput performance of the hard diskIndicate problem with motor, servo or bearings. |
| 3 | **Spin Up Time** | Time needed by spindle to spin-up to full RPMIndicate problem with motor or bearings.*Critical attribute* |
| 4 | **Start/Stop Count** | Count of start/stop cycles of spindleThis value does not directly affect the condition of the drive. |
| 5 | **Reallocated Sector Count(Reallocated Sectors Count)** | Count of sectors moved to the spare areaIndicate problem with the disk surface or the read/write heads.*Critical attribute* |
| 6 | **Read Channel Margin** | Margin of a channel while reading dataThe exact function of this attribute is not specified. |
| 7 | **Seek Error Rate** | Rate of positioning errors of the read/write headsIndicate problem with servo, head. High temperature can also cause this problem.*Critical attribute* |
| 8 | **Seek Time Performance** | Average time of seek operations of the headsIndicate problem with servo.*Critical attribute* |
| 9 | **Power-On Time Count** | Total time the drive is powered onThe unit of the measure depends on the manufacturer. |
| 10 | **Spin Retry Count** | Retry count of spin start attemptsIndicate problem with motor, bearings or power supply.*Critical attribute* |
| 11 | **Drive Calibration Retry Count** | Number of attempts to calibrate a driveIndicate problem with motor, bearings or power supply. |
| 12 | **Drive Power Cycle Count** | Number of complete power on/off cyclesThis value does not directly affect the condition of the drive. |
| 13 | **Soft Read Error Rate** | Number of software read errorsThe number of uncorreactable read errors. |
| 190 | **Airflow Temperature** | Airflow temperatureThe temperature of the air inside the hard disk housing. |
| 191 | **Mechanical Shock** | Count of problems caused by mechanical shockAcceleration (for example falling) can cause mechanical shock. |
| 192 | **Power off Retract Cycle** | Count of power off cyclesThis value does not directly affect the condition of the drive. |
| 193 | **Load/Unload Cycle Count** | Count of load/unload cyclesNumber of cycles the head moved into landing zone position. |
| 194 | **HDD Temperature** | Disk temperatureThe temperature inside the hard disk housing. |
| 195 | **Hardware ECC Recovered** | Count of correctable errorsNumber of errors corrected by the internal error correcting mechanism. |
| 196 | **Reallocation Event Count** | Count of sector remap operationsNumber of all (successful and failed) remap operations.*Critical attribute* |
| 197 | **Current Pending Sector Count** | Count of unstable sectorsThese pending sectors may be remapped to the spare area.*Critical attribute* |
| 198 | **Off-Line Uncorrectable Sector Count** | Count of uncorrectable errors when reading/writingIndicate problem with the disk surface or the read/write heads.*Critical attribute* |
| 199 | **Ultra ATA CRC Error Count** | Count of errors during data transfer between disk and hostIndicate problem with the power supply or data cable. |
| 200 | **Write Error Rate** | Errors occured while writing raw data from a diskIndicate problem with the disk surface or the read/write heads. |
| 201 | **Soft Read Error Rate** | Number of software read errorsThe number of uncorreactable read errors. |
| 202 | **Data Address Mark Errors** | Number of data address mark errorsNumber of incorrect or invalid address marks. |
| 203 | **Run Out Cancel** | Number of data correction errorsInvalid error correction checksum found during error correction. |
| 204 | **Soft ECC Correction** | Number of corrected data errorsErrors corrected by the internal error correction mechanism. |
| 205 | **Thermal Asperity Rate** | Number of thermal problemsTotal number of problems caused by high temperature. |
| 206 | **Flying Height** | Head flying heightThe height of the disk heads above the disk surface. |
| 207 | **Spin High Current** | Current value during spin upThe current needed to spin up the drive. |
| 208 | **Spin Buzz** | Number of cycles needed to spin upThe number of retries during spin up because of low current available. |
| 209 | **Offline Seek Performance** | Drive performance during offline operationsThe seek performance of the drive during internal self tests. |
| 220 | **Disk Shift** | Distance of the disk has shifted relative to the spindleIncorrect disk spin can be cause by mechanical shock or high temperature. |
| 221 | **G-Sense Error Rate** | Number of mechanical errorsNumber of errors resulting from shock or vibration. |
| 222 | **Loaded Hours** | Number of powered on hoursThis value is constantly increasing (once per every hour). |
| 223 | **Load/Unload Retry Count** | Number of load/unload operationsThe number of drive head enters/leaves the data zone. |
| 224 | **Load Friction** | Mechanical friction rateThe rate of friction between mechanical parts. Indicate problem with the mechanical subsystem of the drive. |
| 226 | **Load-in Time** | Total time the heads are loadedThe time while the read/write heads are in the data zone. |
| 227 | **Torque Amplification Count** | Rate of torque increaseTorque increase during the spin up operation of the hard disk. |
| 228 | **Power-off Retract Count** | Number of power off cyclesThe number of times the head was retracted as a result of power loss. |
| 230 | **GMR Head Amplitude** | Head positioning amplitudeHead moving distances between operations. |
| 231 | **Hard Disk Temperature** | Disk temperatureThe temperature inside the hard disk housing. |
| 240 | **Head Flying Hours** | Number of head positioning hoursTime spent during the positioning of the drive heads. |
| 250 | **Read Error Retry Rate** | Number of retries during read operationsNumber of errors found during reading a sector from disk surface. |

***Lista atributelor S.M.A.R.T. pentru hard disk-urile SCSI [7]***

|  |  |
| --- | --- |
| 01 | Buffer over/under run counters |
| 02\_0000 | Write errors corrected without substantial delay |
| 02\_0001 | Write errors corrected with possible delays |
| 02\_0002 | Total write errors |
| 02\_0003 | Total write errors corrected |
| 02\_0004 | Total write errors corrected by algorythm |
| 02\_0005 | Total bytes written |
| 02\_0006 | Total uncorrected write errors |
| 03\_0000 | Read errors corrected by ECC hardware method |
| 03\_0001 | Read errors corrected with possible delays |
| 03\_0002 | Total read errors |
| 03\_0003 | Total read errors corrected |
| 03\_0004 | Total read errors corrected by algorythm |
| 03\_0005 | Total bytes read |
| 03\_0006 | Total uncorrected read errors |
| 04\_0000 | Read reverse errors corrected by ECC hardware method |
| 04\_0001 | Read reverse errors corrected with possible delays |
| 04\_0002 | Total read reverse errors |
| 04\_0003 | Total read reverse errors corrected |
| 04\_0004 | Total read reverse errors corrected by algorythm |
| 04\_0005 | Total bytes read reverse |
| 04\_0006 | Total uncorrected read reverse errors |
| 05\_0000 | Verify errors corrected without substantial delay |
| 05\_0001 | Verify errors corrected with possible delays |
| 05\_0002 | Total verify errors |
| 05\_0003 | Total verify errors corrected |
| 05\_0004 | Total verify errors corrected by algorythm |
| 05\_0005 | Total bytes verified |
| 05\_0006 | Total uncorrected verify errors |
| 06\_0000 | Non medium errors |
| 07 | Last error event |
| 0B | Last deferred error or asynchronous event |
| 0D\_0000 | Temperature |
| 0D\_0001 | Reference temperature |
| 0E\_0001 | Manufacture date (year/week) |
| 0E\_0002 | Accounting date |
| 0E\_0003 | Specified cycle count over device lifetime |
| 0E\_0004 | Accumulated start-stop cycles |
| 0F | Application client parameter data |
| 10 | Self test results log |
| 18 | Protocol specific port log |
| 37\_0000 | Blocks sent to initiator |
| 37\_0001 | Blocks received from initiator |
| 37\_0002 | Blocks read from cache and sent to initiator |
| 37\_0003 | I/O commands with size smaller than segment size |
| 37\_0004 | I/O commands with size larger than segment size |
| 3E\_0000 | Power on time |
| 3E\_0008 | Remaining minutes until next internal test |

**Bibliografie:**

**[1]** Andrew S. Tanenbaum Modern Operating Systems, 3rd Edition, 2008

**[2]** <http://pages.cs.wisc.edu/~remzi/OSTEP/file-raid.pdf>

**[3]** <http://www.eecs.berkeley.edu/Pubs/TechRpts/1987/CSD-87-391.pdf>

**[4]** http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=72DE3A334C699A2422C1400BE7C24E33?doi=10.1.1.41.3889&rep=rep1&type=pdf

**[5]** <http://ro.wikipedia.org/wiki/RAID>

**[6]** <http://en.wikipedia.org/wiki/Comparison_of_S.M.A.R.T._tools>

**[7**] <http://www.hdsentinel.com/smart/smartattr.php>

**[8]** https://www.adaptec.com/nr/rdonlyres/14b2fd84-f7a0-4ac5-a07a-214123ea3dd6/0/4423\_sw\_hwraid\_10.pdf