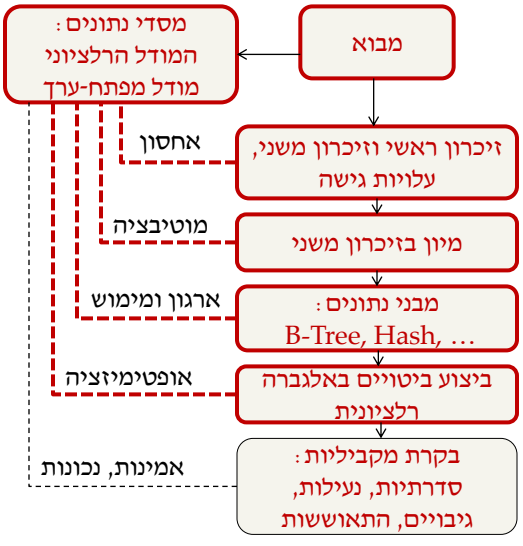
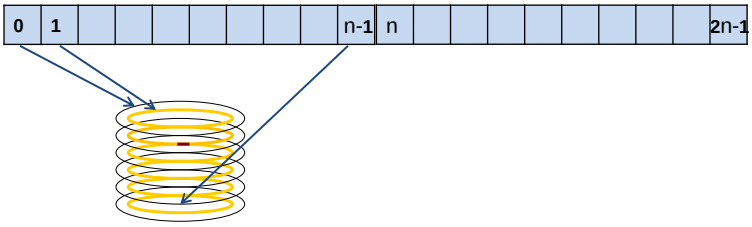
מערכות מרובות דיסקים

RAID: Redundant Array of ~~Inexpensive~~ Independent Disks

איפה אנחנו

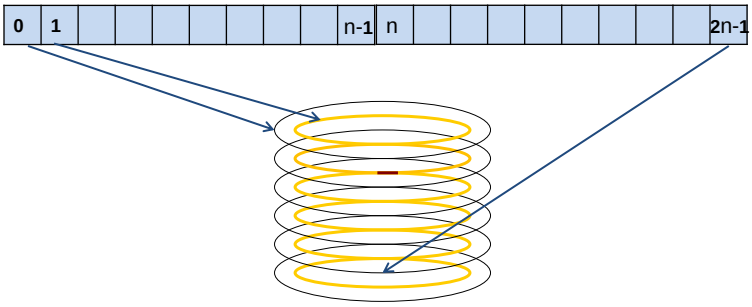


מיפוי לוגי לנפח אחסון פיזי



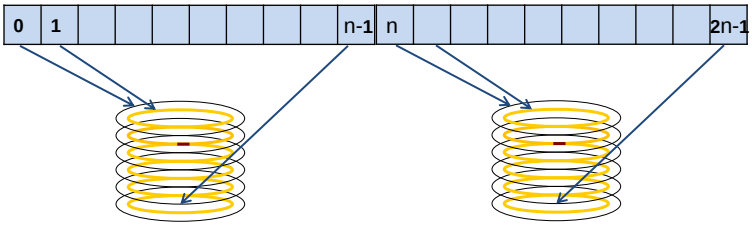
- מערכת ההפעלה ממפה בלוקים לוגיים למיקום פיזי בדיסק
- במערכות חדשות נדרש להגדיל את נפח האחסון

הגדלת נפח האחסון 1



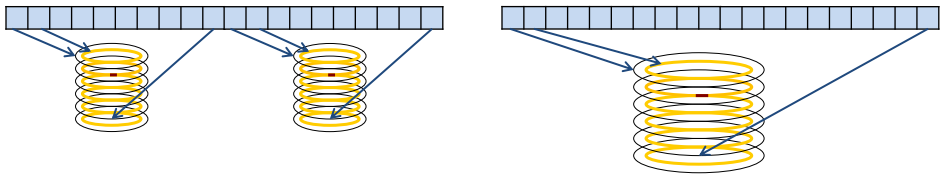
- מערכת ההפעלה ממפה בלוקים לוגיים למיקום פיזי בדיסק
- במערכות חדשות נדרש להגדיל את נפח האחסון
  - אפשרות 1: הגדלת נפח הדיסק

הגדלת נפח האחסון 2



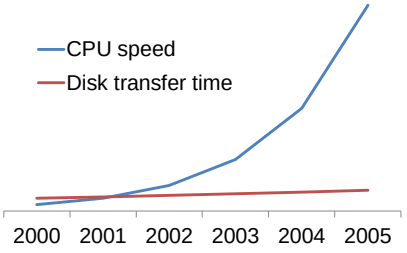
- מערכת ההפעלה ממפה בלוקים לוגיים למיקום פיזי בדיסק
- במערכות חדשות נדרש להגדיל את נפח האחסון
  - אפשרות 1: הגדלת נפח הדיסק
  - אפשרות 2: חלוקת הנתונים בין מספר דיסקים

הגדלת נפח האחסון: שיקולים נוספים



- מחיר
  - מספר דיסקים קטנים זולים מדיסק גדול
- מהירות גישה
  - ריבוי דיסקים מאפשר עבודה במקביל
- התמודדות עם תקלות
  - ההסתברות לתקלה בדיסק אחד: p
  - ההסתברות לתקלה ב-n (n>1) דיסקים מתוך G: $p_n(G) = \binom{G}{n} p^n (1-p)^{G-n}$ << p

הגדלת נפח אחסון על ידי ריבוי דיסקים



- שיפור בביצועים של מערכות מחשב:
  - מעבדים - 50% לשנה
  - זמן תנועת זרוע - 10% לשנה
  - קצב העברה - 10% לשנה

- אפליקציות חדשות (מולטימדיה) דורשות שטח אחסון רב, מהיר וזול
- אפליקציות ישנות נעשות יותר שאפתניות ודורשות יותר מהדיסקים

סוגי תקלות

תקלת מערכת: כל תקלה שאינה נובעת ממערכת הדיסקים:
הפסקת חשמל, טעות תוכנה...

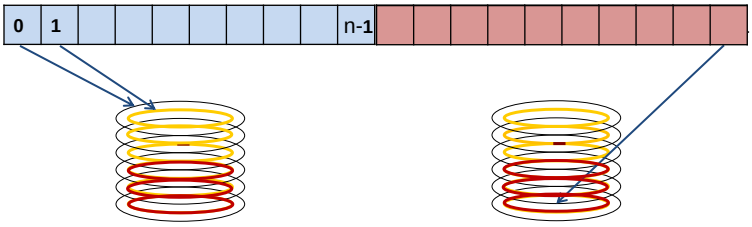
תקלת דיסק: תקלה שמקורה במערכת הדיסקים:
סקטור פגום, תקלת בקר, מחיקת דיסק...

טעות אנוש: בעיות בנתונים עצמם:
מחיקה, עדכון שגוי, דריסה...

כרגע נטפל בתקלות דיסקים בלבד ונניח:

- תקלה בדיסק מזוהה מיידית
- תקלה בדיסק משביתה את כולו - אין גישה לאף סקטור בדיסק
- אין תקלות מערכת וטעויות אנוש (בהמשך הקורס נתייחס אליהן)

יתירות: redundancy



- כדי להתמודד עם תקלות נקצה חלק מן המקום הפיזי לאינפורמציה שתאפשר שחזור נתונים במקרה של תקלה
- תקורה: החלק מנפח האחסון הפיזי שמוקצה ליתירות ולא לנתונים
  - כמה מקום להקצות?
  - איזה מידע לשמור?
  - איפה לשמור אותו?

יתירות: redundancy

- שכפול (mirroring)
  - לכל דיסק יש העתק מלא
  - תקורה: 50%
- זוגיות (parity)
  - שימוש בביט זוגיות לגיבוי קבוצה של ביטים
  - זוגיות של קבוצת מחרוזות היא שרשור הזוגיות של הביטים המתאימים
  - התקורה נקבעת על פי גודל הקבוצה

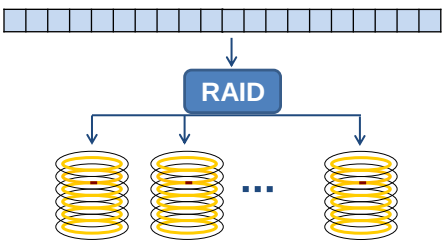
$$p(a,b,c) = a \oplus b \oplus c$$

$$a = b \oplus c \oplus p(a,b,c)$$

$$p\begin{pmatrix} 0 & 1 & 0 \\ 1, & 1, & 1 \\ 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 0 \oplus 1 \oplus 0 \\ 1 \oplus 1 \oplus 1 \\ 0 \oplus 1 \oplus 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$$

- קוד אחר לתיקון שגיאות
  - Hamming, Reed-Solomon
  - התקורה נקבעת על פי הקוד

RAID

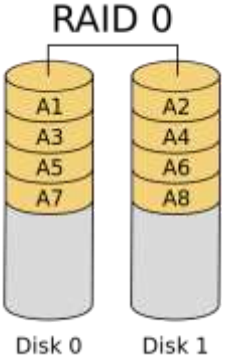


- במקור: Redundant Array of Inexpensive Disks
- בהמשך השתרש השינוי ל-Independent

- בקר ה-RAID מציג למערכת ההפעלה מרחב כתובות רציף
- מאפשר לבחור את שיטת המיפוי על פי העדפות המשתמש
- אחראי על תזמון, שכפול, וחישובי יתירות
  - ממומש בחומרה או בתוכנה

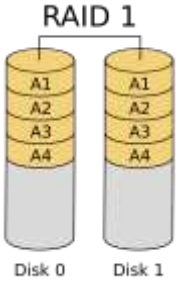
רמה 0: ללא יתירות (nonredundant)

- הנתונים נכתבים לדיסקים ללא עיבוד נוסף
  - כתיבה מהירה ביותר
  - נראה שיטות בהן הקריאה מהירה יותר
- ניצול אופטימאלי של נפח האחסון
  - 0% תקורה
- לא עמיד בפני נפילות



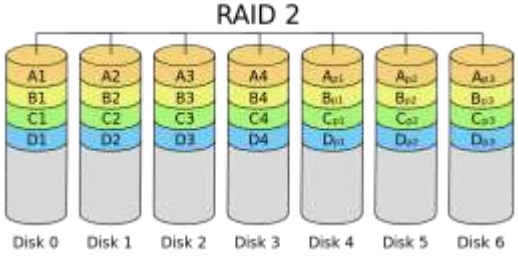
רמה 1: שכפול (mirroring)

- לכל דיסק יש העתק מדויק:
  אותו נתון מופיע בכתובת A הן בדיסק 0 והן בדיסק 1
  - ✓קריאה מהירה: נשתמש בזרוע הקרובה יותר לכתובת היעד
  - ×הכתיבה איטית: נצטרך להזיז את זרועות שני הדיסקים, ולחכות שהכתיבה השנייה תסתיים.
  - ✓התאוששות מהירה מתקלה בודדת
  - ×ניצול נמוך של נפח האחסון
    - ×50% תקורה



רמה 2: שימוש בקודים לתיקון שגיאות

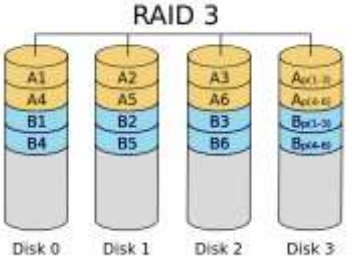
- שימוש בקודים מקובלים לזיהוי ותיקון שגיאות (Hamming code)
  - ✓חוסך מקום לעומת שכפול
- התקורה גבוהה מדי: חלק מהמידע נועד לזיהוי הטעות
  - מיותר - יודעים מתי דיסק נפל
  - בפועל לא בשימוש



רמה 3: זוגיות ברמת הביט (bit interleaved parity)

- הדיסקים מחולקים לקבוצות של G דיסקים (קבוצת תיקון שגיאות)
- לכל קבוצה מתווסף דיסק זוגיות: סה"כ G+1 דיסקים בכל קבוצה
  - יחידת האינפורמציה שכתובה על אותו הדיסק היא סיבית (ביט)
  - תקורה: $1/_{G+1}$

בקריאה ניגש לכל G+1 הדיסקים, אבל הקריאה נגמרת אחרי G הראשונים
כתיבה צריכה לעדכן גם את דיסק הזוגיות

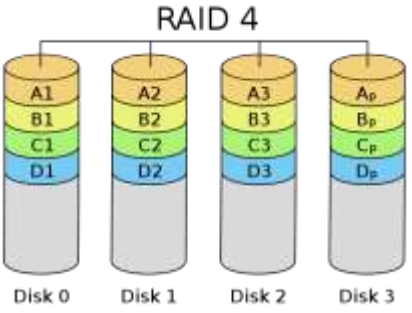


✓אלגוריתם פשוט
✓קצב העברה גבוה
×זמן גישה ארוך יחסית

רמה 3: מה צריך לקרוא?

- יחידת האינפורמציה הקטנה ביותר שניתן לקרוא מדיסק בודד היא גזרה
- אבל ב-RAID 3 כדי לקרוא גזרה, חייבים לקרוא G דיסקים
  - לכן, ביחידה הקטנה ביותר יש G גזרות
- כדי לכתוב גזרה אחת, חייבים לעדכן G+1 גזרות:
  - לקרוא G גזרות
  - לכתוב את הגרסה המעודכנת ב-G דיסקים של הקבוצה
  - לעדכן את דיסק הזוגיות
- כדי לכתוב G גזרות, אין צורך לקרוא את האינפורמציה הישנה

רמה 4: זוגיות ברמת הבלוק (block interleaved parity)



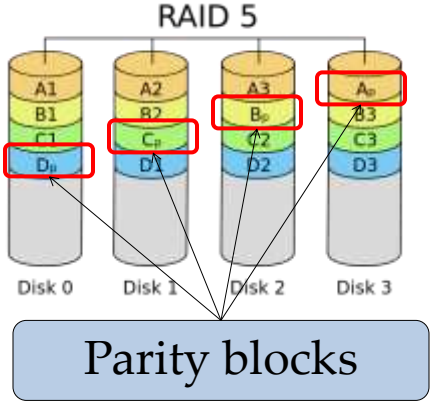
- ה striping unit הוא גזרה/מסילה (בלוק)
- קריאה ליחידה שמוכלת בבלוק ניגשת לדיסק אחד
- קריאות מדיסקים שונים יכולות להתבצע במקביל

- כתיבה ניגשת גם לדיסק הזוגיות
- דורשת 4 פעולות I/O
  - הכתיבה איטית ודיסק הזוגיות הוא hotspot

READ old_data
READ old_parity
new_parity = old_parity ⊕ old_data ⊕ new_data
WRITE new_data
WRITE new_parity

רמה 5: זוגיות מפוזרת (block interleaved distributed-parity)

- דומה לרמה 4, אך הזוגיות מפוזרת בין הדיסקים
- אם נעבור על הקובץ סדרתית, נעבור על כל הדיסקים פעם אחת לפני שנחזור לאותו דיסק פעם נוספת



- הערה: רמה 5 תמיד עדיפה על רמה 4
  - אין דיסק בודד שהוא צוואר בקבוק

סיכום ביניים: רמות RAID

רמה	יתירות	תקורה
0	אין	0
1	שכפול	$^1/_2$
2	קוד תיקון שגיאות	תלוי בקוד
3	זוגיות (ביט)	$\frac{1}{G+1}$
4	זוגיות (בלוק)	$\frac{1}{G+1}$
5	זוגיות (מבוזרת)	$\frac{1}{G+1}$

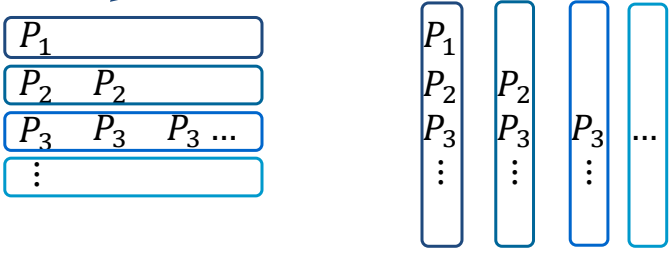
חישוב זמני גישה במערך RAID

- כאשר המערכת כוללת יתירות לצרכי התאוששות מנפילות, עדכון נתונים מחייב עדכון של מידע היתירות:
  - בשכפול כל נתון נכתב פעמיים
  - בזוגיות יש לעדכן את הערך על פי הערך הקודם והערך החדש
- נתייחס למקרים בהם הזרועות של הדיסקים נעות באופן מסונכן – כל הראשים הקוראים מגיעים לגליל הדרוש באותו הזמן
- הדיסקים עצמם אינם מסונכרנים - כל אחד מהם נמצא בגזרה אחרת
- הכתיבה נגמרת כאשר אחרון הדיסקים גומר את הכתיבה
- השהיית הסיבוב של דיסק אחד היא משתנה מקרי המפולג אחיד – נחשב את התוחלת של מקסימום של n משתנים אקראיים מפולגים אחיד

חישוב התוחלת של $X = \max\{X_1,\dots,X_n\}$

כאשר $X_1,\dots,X_n$ משתנים דיסקרטיים (בדידים), $X_i \sim U[0,1]$

$$E(X) = \sum_{i=1}^{\infty} iP(X=i) = \sum_{i=1}^{\infty}\sum_{j=1}^{i} P(X=i) = \sum_{j=1}^{\infty}\sum_{i=j}^{\infty} P(X=i) = \sum_{j=1}^{\infty} P(X \geq j)$$



חישוב התוחלת של $X = \max\{X_1,\dots,X_n\}$

כאשר $X_1,\dots,X_n$ משתנים דיסקרטיים (בדידים), $X_i \sim U[0,1]$

$$E(X) = \sum_{i=1}^{\infty} iP(X=i) = \sum_{i=1}^{\infty}\sum_{j=1}^{i} P(X=i) = \sum_{j=1}^{\infty}\sum_{i=j}^{\infty} P(X=i) = \sum_{j=1}^{\infty} P(X \geq j)$$

כאשר $X_1,\dots,X_n$ משתנים רציפים, עם פונקצית צפיפות f_X

$$E(X) = \int_0^1 x f_X(x)\,dx = \int_0^1\int_0^x f_X(x)\,dt\,dx = \int_0^1\int_t^1 f_X(t)\,dx\,dt = \int_0^1 P(X \geq t)dt$$

חישוב השהיית סיבוב

$X = \max\{X_1, \dots X_n\}$

$P(X \geq t) = 1 - P(X \leq t) = 1 - P(X_1 \leq t) \cdot \dots \cdot P(X_n \leq t) = 1 - t^n$

$$E(X) = \int_0^1 (1 - t^n)\, dt = \int_0^1 1\ dt - \int_0^1 t^n\, dt = t|_0^1 - \frac{t^{n+1}}{n+1}|_0^1 = 1 - \frac{1}{n+1}$$
$$= \frac{n}{n+1}$$

מסקנה: השהיית הסיבוב בכתיבה ל- n דיסקים היא $\frac{n}{n+1}$ סיבוב

באופן דומה: השהיית הסיבוב בקריאת הדיסק הראשון מתוך n היא $\frac{1}{n+1}$ סיבוב

סיכום ביניים: רמות RAID

רמה	יתירות	תקורה	השהיית סיבוב	
			קריאה	כתיבה
0	אין	0	$^1/_2\, T_{rot}$	
1	שכפול	$^1/_2$	R: $^1/_3\, T_{rot}$	W: $^2/_3\, T_{rot}$
2	קוד תיקון שגיאות	תלוי בקוד		
3	זוגיות (ביט)	$\frac{1}{G+1}$	R: $\frac{G}{G+1} T_{rot}$	W: $\frac{G+1}{G+2} T_{rot}$
4	זוגיות (בלוק)	$\frac{1}{G+1}$	R: $^1/_2\, T_{rot}$	W: $^2/_3\, T_{rot}$
5	זוגיות (מבוזרת)	$\frac{1}{G+1}$	R: $^1/_2\, T_{rot}$	W: $^2/_3\, T_{rot}$

אמינות (reliability)

- נחשב את ההסתברות לאובדן מידע במערכת כתוצאה מתקלת דיסק
- המדד: הזמן הממוצע לאובדן מידע
- MTTDL: Mean Time To Data Loss
- היצרן מספק את p: ההסתברות לתקלה בדיסק בשעה
- MTTF: Mean Time To Failure = 1/p
- בדיסק בודד: MTTDL = MTTF
- במערך דיסקים עם יתירות ניתן לשחזר מידע אם דיסק אחד בלבד "נפל"
- אובדן מידע מתרחש בנפילה בו זמנית של שני דיסקים או יותר
  - או בנפילת דיסק שני לפני ששחזרנו את המידע מנפילת הדיסק הראשון

זמן ממוצע לנפילה

אם ההסתברות שדיסק ייפול במשך שעה היא p, אז

MTTF = Mean Time To Failure = 1/p

כי $\text{MTTF} = \sum_{i>0} i \text{ (prob failure at time } i) = \sum_{i>0} i(1-p)^{i-1} p = \frac{1}{p}$

עבור 100 דיסקים, אם ההסתברות p קטנה, והנפילות בלתי-תלויות

P(לפחות דיסק אחד מ-100 נופל) = $1 - (1-p)^{100} \cong 100\ p$

לדיסק בודד (לפי נתוני יצרן)

MTTF = 20,000 hours = 2.3 years

$\text{MTTF(1 of 100 disks)} = \frac{\text{MTTF (1)}}{100} = 200 \text{ hours} \approx 8\frac{1}{3} \text{ days}$

נפילות של דיסקים בדרך-כלל דווקא כן תלויות...
- הפרעות ברשת החשמל משפיעות על כל הדיסקים
- הסיכוי ליפול תלוי בזמן (מקסימלי בהתחלת פעולת המערכת, וכשהמערכת מזדקנת)

שיעור נפילות שנתי

מדד אמינות מקובל כיום: Annual Failure Rate

נניח MTTF = 1,200,000 hours

$$\frac{1{,}200{,}000 \quad hours}{8760 \quad hours\ per\ year} \cong 137\ years$$

(בהנחה שהדיסק פעיל 24 שעות ביממה, 365 ימים בשנה)

$$\frac{1 \quad failure}{137 \quad years} \times 100\% \cong 0.73\% \quad \frac{failures}{year}$$

לדיסק סטנדרטי (Seagate - Barracuda 7200.11 SATA 3Gb/s 500-GB)
יש שיעור נפילות שנתי 0.34% (על פי נתוני היצרן)

הסתברות נפילת מערך דיסקים I

$p_i(G)$ ההסתברות **שבדיוק** i דיסקים מתוך G יפלו.

$$p_0(G) = (1-p)^G = 1 - Gp + \binom{G}{2}p^2 + O(G^3p^3)$$

$$p_1(G) = \binom{G}{1}p(1-p)^{G-1} = Gp[1-(G-1)p + O(G^2p^2)]$$
$$= Gp - G(G-1)p^2 + O(G^3p^3)$$

$$p_2(G) = \binom{G}{2}p^2(1-p)^{G-2} = \frac{G(G-1)}{2}p^2[1+O(Gp)]$$
$$= \frac{G(G-1)}{2}p^2 + O(G^3p^3)$$

למשל ב- RAID 1:
ההסתברות ששני הדיסקים יפלו (i=2, G=2) היא בערך p^2

הסתברות נפילת מערך דיסקים II

$p_{\geq i}(G)$ ההסתברות **שלפחות** i דיסקים מתוך G יפלו.

$$p_{\geq 2}(G) = 1 - p_0(G) - p_1(G)$$
$$= 1 - \left(1 - Gp + \frac{G(G-1)}{2}p^2\right) - (Gp - G(G-1)p^2) + O(p^3)$$
$$= \frac{G(G-1)}{2}p^2 + O(p^3) \cong p_2(G)$$

לדוגמא, ההסתברות שלפחות שני דיסקים מתוך תשעה יפלו:

$$p_{\geq 2}(9) = \binom{9}{2}p^2 + O(p^3) \cong 36p^2$$

הסתברות נפילת מערך דיסקים III

במערכת N דיסקים המחולקים לקבוצות של G

- p ההסתברות לנפילת דיסק בודד
- אובדן מידע מתרחש כאשר אובד מידע בקבוצה אחת לפחות

$$P_{Global} \cong \frac{N}{G}p_{\geq 2}(G) \cong \frac{N}{G}\frac{G(G-1)}{2}p^2 = \frac{N(G-1)}{2}p^2$$

$$MTTDL = MTTF_{Global} = {}^{1}/_{P_{Global}}$$

הסתברות נפילת מערך דיסקים: דוגמא

נתון מערך:
- 1000 דיסקי נתונים
- 125=1000/8 דיסקי זוגיות
- (1125 דיסקים סה"כ)

- המערך נופל אם שני דיסקים מתוך אחת מ-125 הקבוצות נופלים:

$$P_{Global} \cong 125 \cdot p_{\geq 2}(9) \cong 125 \times 36p^2 = 4{,}500p^2$$

- ההסתברות לנפילת דיסק בשעה היא $p = {}^{1}/_{20{,}000}$
- ההסתברות לנפילת המערך:

$$P_{Global} = \frac{4{,}500}{20{,}000^2} = \frac{1}{88{,}888\ hours}$$

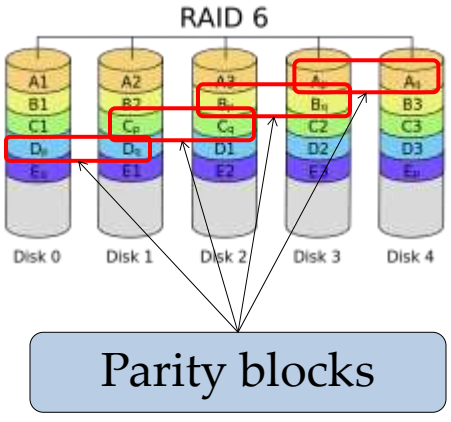
MTTDL ≈ 88,888 hours ≅ 3,700 days ≅ 10 years 4 months

סיכום: רמות RAID

רמה	יתירות	תקורה	השהיית סיבוב		ההסתברות לאובדן מידע (1/MTTDL)
			קריאה	כתיבה	
0	אין	0	$^{1}/_{2}\,T_{rot}$		$G * p$
1	שכפול	$^{1}/_{2}$	R: $^{1}/_{3}\,T_{rot}$	W: $^{2}/_{3}\,T_{rot}$	p^2
2	קוד תיקון שגיאות	תלוי בקוד			
3	זוגיות (ביט)	$^{1}/_{G+1}$	R: $\frac{G}{G+1}T_{rot}$	W: $\frac{G+1}{G+2}T_{rot}$	$\frac{G(G-1)}{2}p^2$
4	זוגיות (בלוק)	$^{1}/_{G+1}$	R: $^{1}/_{2}\,T_{rot}$	W: $^{2}/_{3}\,T_{rot}$	$\frac{G(G-1)}{2}p^2$
5	זוגיות (מבוזרת)	$^{1}/_{G+1}$	R: $^{1}/_{2}\,T_{rot}$	W: $^{2}/_{3}\,T_{rot}$	$\frac{G(G-1)}{2}p^2$

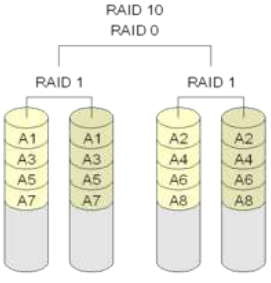
רמה 6: P + Q Redundancy

- שימוש בקוד Reed-Solomon לתיקון שגיאות
  - עמיד בפני שתי תקלות בו זמניות
  - תיקון t שגיאות ע"י הוספת 2t ביטים
  - ראה הקורס: "מבוא לתורת הצפינה"

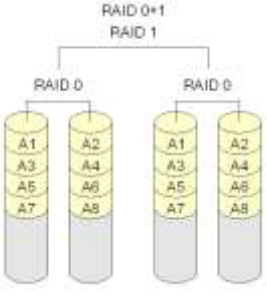


- כל כתיבה מחייבת 6 פעולות I/O
  - קריאת הנתונים
  - קריאת זוגיות מדיסק ראשון
  - קריאת זוגיות מדיסק שני
  - כתיבת הנתונים
  - כתיבת זוגיות לדיסק זוגיות ראשון
  - כתיבת זוגיות לדיסק זוגיות שני

קינון: Nested RAID



- האלמנטים במערך הם מערכי RAID בעצמם
- RAID 1+0 (a.k.a RAID 10)
  RAID 5+0 (a.k.a RAID 50)
  - MTTDL כמו שחישבנו עבור מערך המחולק לקבוצות יתירות



- RAID 0+1
  - מהו MTTDL?
  - מה היתרון על פני RAID 10?

שחזור אוטומטי

- במערכת יש דיסקים עודפים, כדי להחליף דיסקים שהתקלקלו
- ההחלפה נעשית באופן אוטומטי, ללא התערבות המפעיל
  - הקצאת דיסק חלופי
  - חישוב הנתונים שאבדו על סמך היתירות (ברקע)
  - תוך כדי השחזור ניתן לגשת לנתונים הישנים
- תקופתית, טכנאי השרות מחליף את הדיסקים שהתקלקלו מבלי להשבית את המערכת
  - כל עוד אף דיסק לא התקלקל, אפשר לנצל אותם כדי להגדיל את היתירות ולמנוע אובדן נתונים

זכרון מטמון cache

- מערכות RAID מצוידות בזיכרון אלקטרוני גדול (עד 4GB) (non-volatile) שמגובה בסוללות (מהיר פי ≈ 10,000 מהדיסק)
  - עמיד לתקלות מערכת וחומרה
  - כתיבה בזיכרון זה היא אטומית
- זיכרון המטמון מהווה חוצץ לכתיבה ב- RAID:
  - כדי לכתוב ערך לדיסק, נכתוב אותו לזיכרון המטמון
  - כאשר זיכרון המטמון מתמלא, נוריד גזרות לדיסקים
- בקריאה
  - אם הגזרה הדרושה נמצאת בזיכרון המטמון אין צורך לגשת לדיסקים
  - אחרת נטען אותה מהדיסק

☝קיצור זמן הכתיבה: מבחינת המשתמש הכתיבה מסתיימת כאשר מתעדכן המטמון.

יתרונות נוספים של המטמון

ניצול לוקליות:
אם נקרא ערך שכתבנו/קראנו לא מזמן, הערך המבוקש יהיה בזכרון המטמון

אטומיות:
הופך את הכתיבה בדיסק לאטומית (על-ידי אלגוריתמים לשחזור שנלמד בהמשך)

הקטנת מספר הכתיבות:
נחסוך כתיבות אם נכתוב שוב לגזרות שטרם העברנו לדיסק

תזמון כתיבה:
- אפשר להשתמש באלגוריתמים מתוחכמים לכתיבה, כי מרכזים פעולות
- למשל, piggy-backing
  - אם ניגשים לגליל לצורך קריאה, אפשר באותה ההזדמנות לכתוב גזרות שמיועדות לכתיבה בגליל זה הנמצאות בזכרון המטמון, ולחסוך תנועות זרוע