

## Ενότητα 9 (Κεφάλαιο 11) Διαχείριση Συσκευών Ε/Ε και Αποθήκευσης

Οι διαφάνειες αυτές έχουν συμπληρωματικό και επεξηγηματικό χαρακτήρα και σε καμία περίπτωση δεν υποκαθιστούν το βιβλίο

Γιώργος Α. Παπαδόπουλος  
Τμήμα Πληροφορικής  
Πανεπιστήμιο Κύπρου



1

## Περιεχόμενα

- Συσκευές Ε/Ε.
- Οργάνωση των λειτουργιών Ε/Ε.
- Ο ρόλος του Λ.Σ. στη διαχείριση των συσκευών Ε/Ε.
- Χρήση προσωρινής μνήμης στις λειτουργίες Ε/Ε.
- Διαχείριση δίσκου.
- Συστήματα Raid.
- Μνήμη cache στο δίσκο.
- Ε/Ε στο UNIX SVR4.
- Ε/Ε στο LINUX.
- Ε/Ε στα Windows.

2

## Κατηγορίες συσκευών Ε/Ε

- Προβληματική πτυχή της σχεδίασης Λ.Σ.
  - Δύσκολη η εύρεση ομοιομορφων λύσεων λόγω του εύρους των διαφορετικών συσκευών και εφαρμογών.
- Υπάρχουν τρεις κατηγορίες συσκευών:
  - Αυτές που διαχειρίζονται δεδομένα αναγνώσιμα από τον άνθρωπο και χρησιμοποιούνται για την επικοινωνία ανθρώπου-μηχανής:
    - Οθόνες, ποντίκια, πληκτρολόγια, εκτυπωτές.
  - Αυτές που διαχειρίζονται δεδομένα αναγνώσιμα από τη μηχανή και χρησιμοποιούνται για την επικοινωνία μεταξύ ηλεκτρονικών συσκευών:
    - Ελεγκτές, ανιχνευτές, USB.
  - Αυτές που χρησιμοποιούνται για επικοινωνία:
    - Μετατροπείς.

3

3

## Διαφορές μεταξύ συσκευών Ε/Ε

- Ανάλογα με το είδος τους διαφέρουν στα ακόλουθα:
  - Ρυθμός μεταφοράς δεδομένων (data rate).
  - Εφαρμογή.
  - Πολυπλοκότητα του ελέγχου τους.
  - Μονάδα μεταφοράς δεδομένων.
  - Αναπαράσταση δεδομένων.
  - Είδη σφαλμάτων.

4

4

## Ρυθμός μεταφοράς δεδομένων

- Υπάρχει μεγάλη διαφορά μεταξύ διαφόρων κατηγοριών συσκευών αναφορικά με αυτήν την παράμετρο.

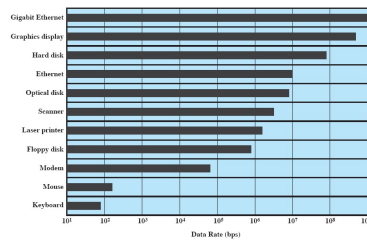


Figure 11.1 Typical I/O Device Data Rates

5

5

## Εφαρμογή

- Ο τρόπος που χρησιμοποιείται μία συσκευή επηρεάζει το λογισμικό της συσκευής, τις πολιτικές του Λ.Σ. στη χρήση της και την υποστήριξη βοηθητικών υπηρεσιών.
  - Ο δίσκος χρειάζεται ειδικό λογισμικό διαχείρισης αρχείων για την αποθήκευσή τους και ειδικό υλικό και λογισμικό για υποστήριξη ιδεατής μνήμης.
  - Τερματικές οθόνες που χρησιμοποιούνται από διαχειριστές συστημάτων (system administrators) δύναται να έχουν μεγαλύτερη προτεραιότητα.

6

6

## Πολυπλοκότητα ελέγχου συσκευής

- Ο έλεγχος ενός εκτυπωτή είναι σχετικά απλός.
- Ο έλεγχος ενός δίσκου είναι σχετικά πολύπλοκος.
- Σε κάποιο βαθμό, το λογισμικό της κάθε συσκευής αναλαμβάνει να χειριστεί μέρος αυτής της πολυπλοκότητας.

7

## Μονάδα μεταφοράς δεδομένων

- Τα δεδομένα μπορούν να μεταφερθούν ως:
  - Ακολουθία χαρακτήρων (π.χ. τερματικό, ποντίκι).
  - Ομάδα (ή μπλοκ) σταθερού μεγέθους (π.χ. δίσκος, ταινία).

8

## Αναπαράσταση δεδομένων

- Διαφορετικές συσκευές χρησιμοποιούν διαφορετικούς τρόπους αναπαράστασης δεδομένων (π.χ. κωδικοποίηση χαρακτήρων, ισοτιμίας).

9

## Είδη σφαλμάτων

- Τα είδη σφαλμάτων που δημιουργούνται από τη χρήση των συσκευών διαφέρουν σημαντικά από το ένα είδος συσκευής στο άλλο, σε θέματα όπως:
  - Τρόπος που παρουσιάζονται.
  - Επίπτωση που έχουν στο περιβάλλον τους.
  - Τρόποι αντιμετώπισής τους.

10

## Περιεχόμενα

- Συσκευές E/E.
- Οργάνωση των λειτουργιών E/E.
- Ο ρόλος του Λ.Σ. στη διαχείριση των συσκευών E/E.
- Χρήση προσωρινής μνήμης στις λειτουργίες E/E.
- Διαχείριση δίσκου.
- Συστήματα Raid.
- Μνήμη cache στο δίσκο.
- E/E στο UNIX SVR4.
- E/E στο LINUX.
- E/E στα Windows.

11

## Τεχνικές για εκτέλεση εντολών E/E

- Προγραμματιζόμενη E/E.
- Χρήση διακοπών.
- Απευθείας προσπέλαση μνήμης.

Table 11.1 I/O Techniques

	No Interrupts	Use of Interrupts
I/O-to-memory transfer through processor	Programmed I/O	Interrupt-driven I/O
Direct I/O-to-memory transfer		Direct memory access (DMA)

12

### Εξέλιξη των τεχνικών εκτέλεσης εντολών E/E — 1

1. Η ΚΜΕ ελέγχει άμεσα μία περιφερειακή συσκευή. Τυπικό παράδειγμα είναι οι συσκευές που ελέγχονται από μικροεπεξεργαστές.
2. Προστίθεται ένας ελεγκτής E/E.
  - Η ΚΜΕ χρησιμοποιεί προγραμματιζόμενη E/E χωρίς διακόπτες.
  - Κατ' επέκταση, δεν χρειάζεται να γνωρίζει τις ιδιαιτερότητες της κάθε συσκευής.

13

### Εξέλιξη των τεχνικών εκτέλεσης εντολών E/E — 2

3. Ελεγκτής E/E με υποστήριξη διακοπών.
  - Βελτιώνεται η απόδοση γιατί η ΚΜΕ δεν χρειάζεται να σπαταλάει χρόνο περιμένοντας την ολοκλήρωση μίας εντολής E/E.
4. Απευθείας προσπέλαση μνήμης (direct memory access, DMA).
  - Τα δεδομένα μεταφέρονται σε ομάδες από και προς την κύρια μνήμη χωρίς τη χρήση της ΚΜΕ.

Η ΚΜΕ εμπλέκεται μόνο στην αρχή και στο τέλος της διαδικασίας.

14

### Εξέλιξη των τεχνικών εκτέλεσης εντολών E/E — 3

5. Ο ελεγκτής είναι μία ξεχωριστή μονάδα επεξεργασίας που εκτελεί ένα ρεπερτόριο ειδικών εντολών για E/E.
  - Η ΚΜΕ δίνει εντολή στον ελεγκτή να εκτελέσει ένα πρόγραμμα E/E στη μνήμη.
  - Ο ελεγκτής φέρνει στη μνήμη και εκτελεί τις σχετικές εντολές χωρίς την εμπλοκή της ΚΜΕ.
  - Η ΚΜΕ ενημερώνεται μόλις ολοκληρωθεί αυτή η διαδικασία.
6. Επεξεργαστής E/E.
  - Έχει δική του μνήμη και ουσιαστικά είναι ένας εξειδικευμένος (για E/E) υπολογιστής από μόνος του.
  - Μπορεί να χειριστεί ένα μεγάλο αριθμό από συσκευές E/E με ελάχιστη εμπλοκή της ΚΜΕ.
  - Χρησιμοποιείται συχνά στον έλεγχο της επικοινωνίας μεταξύ του Η/Υ και τερματικών.

15

### Απευθείας προσπέλαση μνήμης

- Η ΚΜΕ μεταβιβάζει την εκτέλεση των πράξεων E/E στην μονάδα του συστήματος που υλοποιεί αυτή την τεχνική.
- Η μονάδα αυτή μεταφέρει τα δεδομένα κατ' ευθείαν από ή προς τη μνήμη.
- Μόνο στο τέλος της μεταφοράς, η μονάδα χρησιμοποιεί ένα διακόπτη για να ενημερώσει την ΚΜΕ ότι η μεταφορά έχει ολοκληρωθεί.

Figure 11.2 Typical DMA Block Diagram

16

### Χρήση ενός διαύλου επικοινωνίας

(a) Single-bus, detached DMA

- Όλες οι εμπλεκόμενες οντότητες χρησιμοποιούν τον ίδιο δίαυλο επικοινωνίας.
- Η μονάδα DMA, δρώντας ως επιβοηθητικός επεξεργαστής, χρησιμοποιεί προγραμματιζόμενη E/E για να ανταλλάξει δεδομένα μεταξύ μνήμης και μονάδων E/E.
- Για κάθε μεταφορά δεδομένων χρειάζονται δύο κύκλοι: ένας για την αίτηση μεταφοράς και ένας για την ίδια τη μεταφορά.

17

### Ενσωματωμένη DMA και E/E

(b) Single-bus, Integrated DMA-I/O

- Ενσωματώνοντας τη μονάδα DMA με τις μονάδες E/E, επιτρέπει τη δημιουργία ενός διαύλου άμεσης επικοινωνίας μεταξύ των δύο, αποφεύγοντας τη χρήση του κυρίως διαύλου του συστήματος και μειώνοντας τον αριθμό των κύκλων μεταφοράς δεδομένων.

18

## Χρήση ανεξάρτητου διαύλου E/E

• Με τη χρήση ενός αποκλειστικού διαύλου επικοινωνίας μεταξύ της μονάδας DMA και των μονάδων E/E, επιτυγχάνεται μείωση των διασυνδέσεων E/E στη μονάδα DMA σε μία, εύκολη επέκταση του συστήματος με επιπλέον μονάδες και χρήση του διαύλου του συστήματος μόνο για μεταφορά δεδομένων μεταξύ DMA και μνήμης.

19

## Περιεχόμενα

- Συσκευές E/E.
- Οργάνωση των λειτουργιών E/E.
- - Ο ρόλος του Λ.Σ. στη διαχείριση των συσκευών E/E.
- Χρήση προσωρινής μνήμης στις λειτουργίες E/E.
- Διαχείριση δίσκου.
- Συστήματα Raid.
- Μνήμη cache στο δίσκο.
- E/E στο UNIX SVR4.
- E/E στο LINUX.
- E/E στα Windows.

20

## Στόχος 1<sup>ος</sup>: Αποδοτικότητα

- Οι περισσότερες συσκευές E/E είναι πολύ αργές σε σύγκριση όχι μόνο με την ΚΜΕ αλλά και την κύρια μνήμη.
- Η χρήση πολυπρογραμματισμού επιτρέπει σε κάποιες διεργασίες να είναι υπό αναστολή περιμένοντας την ολοκλήρωση της εκτέλεσης εντολών E/E την ίδια ώρα που άλλες διεργασίες εκτελούνται.
- Όμως παρόλο που στις μέρες μας η κύρια μνήμη είναι μεγάλη (και κατ' επέκταση ο βαθμός πολυπρογραμματισμού είναι επίσης μεγάλος), θα καταλήξουμε και πάλι σε μία κατάσταση που οι περισσότερες διεργασίες στη μνήμη θα είναι υπό αναστολή.
- Επομένως υπάρχει η ανάγκη μεταφοράς διεργασιών μεταξύ κύριας και περιφερειακής μνήμης, που αποτελεί πράξη E/E.
- Κατ' επέκταση η βελτίωση του χρόνου εκτέλεσης εντολών E/E είναι μεγάλης σημασίας για την συνολική απόδοση του συστήματος, κυρίως όταν εμπλέκεται ο δίσκος.

21

## Στόχος 2<sup>ος</sup>: Ομοιόμορφη χρήση συσκευών E/E

- Για λόγους εύκολης χρήσης και αποφυγής σφαλμάτων είναι επιθυμητό να χειριζόμαστε όλες τις συσκευές E/E ομοιόμορφα.
  - Αυτό ισχύει και στον τρόπο που βλέπουν οι διεργασίες τις συσκευές και στον τρόπο που το Λ.Σ. διαχειρίζεται τις συσκευές και τη λειτουργία τους.
- Αυτό επιτυγχάνεται με την απόκρυψη των ιδιαιτεροτήτων της κάθε συσκευής σε χαμηλού επιπέδου ρουτίνες.
- Αν και είναι δύσκολο να υπάρξει απόλυτη γενίκευση σε αυτόν τον τομέα, ο στόχος επιτυγχάνεται σε γενικές γραμμές με τη χρήση λογικών στρωμάτων και αρθρωτού σχεδιασμού όπου οι χαμηλού επιπέδου λεπτομέρειες που είναι διαφορετικές για κάθε συσκευή απομονώνονται στα χαμηλά στρώματα, ενώ τα υψηλότερα παρέχουν γενικές και ομοιογενείς λογικές λειτουργίες όπως άνοιγμα/κλείσιμο συσκευής, διάβασμα/γράψιμο, κλπ.

22

## Ιεραρχικός σχεδιασμός

- Όπως είδαμε και στην πρώτη ενότητα, η χρήση μίας ιεραρχικής φιλοσοφίας στο σχεδιασμό των Λ.Σ. οδηγεί στη δημιουργία Λ.Σ. οργανωμένα σε στρώματα και στο διαχωρισμό των λειτουργιών ενός Λ.Σ. ανάλογα με τα χαρακτηριστικά τους, πολυπλοκότητά τους και επίπεδο αφαιρετικότητας.
- Το κάθε στρώμα υλοποιεί ένα μέρος των λειτουργιών που πρέπει να υποστηρίξει ένα Λ.Σ.
- Βασίζεται στο επόμενο χαμηλότερο επίπεδο για να υλοποιήσει πιο βασικές λειτουργίες και να αποκρύψει τις λεπτομέρειές τους.
- Παρέχει υπηρεσίες στο επόμενο υψηλότερο επίπεδο.
- Ιδανικά, αλλαγές σε κάποιο επίπεδο δεν θα πρέπει να επηρεάζουν άλλα επίπεδα.

23

## Πολυστρωματική οργάνωση περιφερειακών συσκευών

- Λογικό επίπεδο.
  - Χειρίζεται τις συσκευές με χρήση γενικευμένων εντολών.
- Επίπεδο συσκευής.
  - Μετατρέπει τις γενικευμένες εντολές και δεδομένα (χαρακτήρες, εγγραφές, κλπ.) σε ειδικές (για τα χαρακτηριστικά της κάθε συσκευής) εντολές E/E, εντολές σε κανάλια, κλπ.
  - Κάνει χρήση προσωρινής μνήμης (buffering).
  - Εδώ υλοποιούνται οι οδηγοί συσκευών.
- Χρονοδρομολόγηση και έλεγχος.
  - Υλοποιεί τη σειρά εκτέλεσης των ειδικών εντολών και ελέγχει αυτές καθαυτές τις συσκευές (π.χ. χειριστές διακοπών, αναφορά στην κατάσταση των συσκευών).

24

### Πολυστρωματική οργάνωση μονάδων επικοινωνίας

- Παρόμοια με την προηγούμενη.
- Το λογικό επίπεδο αντικαθίσταται με μία αρχιτεκτονική επικοινωνίας.
- Η αρχιτεκτονική αυτή μπορεί με τη σειρά της να είναι πολυστρωματική.
  - Κλασικό παράδειγμα είναι το πρωτόκολλο επικοινωνίας TCP/IP.

(b) Communications port

25

### Πολυστρωματική οργάνωση συστήματος αρχείων

- Διαχείριση καταλόγων (directories).
  - Σε αυτό το επίπεδο τα συμβολικά ονόματα αρχείων μετατρέπονται σε προσδιοριστές που αναφέρονται στα αρχεία άμεσα ή μέσω ενός περιγραφέα αρχείων ή πίνακα δεικτών.
  - Υποστηρίζει εντολές στο επίπεδο χρήστη για διαχείριση των καταλόγων.
- Σύστημα αρχείων.
  - Σχετίζεται με τη λογική δομή των αρχείων, υποστηρίζει λογικές πράξεις σε αρχεία και τρόπους πρόσβασης σε αυτά.
- Φυσική οργάνωση.
  - Μετατρέπει τη λογική δομή των αρχείων σε φυσικές διευθύνσεις μνήμης και χρησιμεύει περιφερειακή μνήμη σε αυτά.

(c) File system

26

### Περιεχόμενα

- Συσκευές E/E.
- Οργάνωση των λειτουργιών E/E.
- Ο ρόλος του Λ.Σ. στη διαχείριση των συσκευών E/E.
- **Χρήση προσωρινής μνήμης στις λειτουργίες E/E.**
- Διαχείριση δίσκου.
- Συστήματα Raid.
- Μνήμη cache στο δίσκο.
- E/E στο UNIX SVR4.
- E/E στο LINUX.
- E/E στα Windows.

27

### Προσωρινή μνήμη

- Οι διεργασίες πρέπει να αναμένουν την ολοκλήρωση μίας εντολής E/E που έχουν ζητήσει να εκτελεστεί πριν συνεχίσουν την εκτέλεσή τους.
- Αυτό επιβάλλει οι σελίδες που εμπλέκονται στη μεταφορά δεδομένων να παραμείνουν κλειδωμένες στη μνήμη, γιατί διαφορετικά υπάρχει πιθανότητα απώλειας δεδομένων ή να περιέλθει η διεργασία που έδωσε την εντολή E/E σε αδιέξοδο (π.χ. αν η διεργασία απομακρυνθεί από τη μνήμη πριν ξεκινήσει η εκτέλεση της εντολής E/E).
- Είναι πιο αποδοτικό να εκτελούνται οι μεταφορές δεδομένων προς τη μνήμη πριν ζητηθούν και από τη μνήμη κάποιο χρονικό διάστημα μετά που θα έχει ζητηθεί η μεταφορά τους.
- Αυτό επιτυγχάνεται με τη χρήση προσωρινής μνήμης (buffering).

28

### Είδη προσωρινής μνήμης

- Μεταφορά δεδομένων σε ομάδες (block oriented).
  - Οι πληροφορίες αποθηκεύονται σε ομάδες σταθερού μεγέθους.
  - Κάθε φορά μεταφέρεται μία ομάδα.
  - Αναφορά σε δεδομένα γίνεται με χρήση του αριθμού της κάθε ομάδας.
  - Τυπικά παραδείγματα είναι ο δίσκος και το USB.
- Μεταφορά δεδομένων ως ακολουθίες χαρακτήρων (stream oriented).
  - Τα δεδομένα μεταφέρονται ως ακολουθίες από bytes χωρίς ομαδοποιημένη δομή.
  - Τυπικά παραδείγματα είναι τα τερματικά, οι εκτυπωτές, τα ποντίκια και οι συναφείς συσκευές και γενικώς οι περισσότερες μη αποθηκευτικές συσκευές.

29

### Μη χρήση προσωρινής μνήμης

- Το Λ.Σ. έχει άμεση πρόσβαση σε μία συσκευή E/E.

(a) No buffering

30

### Χρήση μονής προσωρινής μνήμης

- Το Λ.Σ. χρησιμοποιεί ως προσωρινή μνήμη ένα μέρος του χώρου στην κύρια μνήμη που κατέχει το ίδιο.

(b) Single buffering

31

31

### Χρήση μονής προσωρινής μνήμης για μεταφορά μίας ομάδας δεδομένων

- Για συσκευές όπου τα δεδομένα μεταφέρονται σε ομάδες (μπλοκ):
  - Τα δεδομένα μεταφέρονται κατ' αρχήν στην προσωρινή μνήμη.
  - Μόλις ολοκληρωθεί αυτή η διαδικασία, η διεργασία μεταφέρει την ομάδα στο χώρο μνήμης του χρήστη και ζητεί την επόμενη ομάδα δεδομένων.
- Λόγω του ότι η επόμενη ομάδα ζητείται πριν χρειασθεί, η τεχνική αυτή ονομάζεται προδιάβασμα (reading ahead) ή αναμενόμενη είσοδος (anticipated input).
- Συνήθως η πρόβλεψη αυτή είναι σωστή λόγω του ότι τα δεδομένα πολύ συχνά χρησιμοποιούνται με σειριακό τρόπο και μόνο στο τέλος της επεξεργασίας θα μεταφερθεί μία ομάδα δεδομένων χωρίς να χρειάζεται.

(b) Single buffering

32

32

### Χρήση μονής προσωρινής μνήμης για μεταφορά μίας ακολουθίας δεδομένων

- Για συσκευές όπου τα δεδομένα μεταφέρονται ως ακολουθίες χαρακτήρων, αυτή η μεταφορά μπορεί να γίνει μία γραμμή ή ένα byte κάθε φορά.
- Συσκευές όπου τα δεδομένα μεταφέρονται μία γραμμή κάθε φορά είναι τα παλιά τερματικά όπου οι χαρακτήρες διαβάζονταν κάθε φορά που ο χρήστης πάταγε το πλήκτρο Return.
- Η μεταφορά δεδομένων με ένα byte κάθε φορά εφαρμόζεται για συσκευές όπου το κάθε πάτημα ενός πλήκτρου είναι σημαντικό (π.χ. ποντίκια) ή για ελεγκτές και αισθητήρες όπου κάθε byte έχει ξεχωριστή σημασία.

(b) Single buffering

33

33

### Χρήση διπλής προσωρινής μνήμης

- Χρήση δύο χώρων προσωρινής μνήμης.
- Όσο μία διεργασία μεταφέρει δεδομένα από ή προς τον ένα χώρο, το Λ.Σ. γεμίζει ή αδειάζει τον άλλο.

(c) Double buffering

34

34

### Χρήση κυκλικής προσωρινής μνήμης

- Χρήση ενός αριθμού από προσωρινούς χώρους μνήμης.
- Είναι χρήσιμο στην περίπτωση που μία συσκευή μεταφέρει πολύ συχνά δεδομένα.

(d) Circular buffering

35

35

### Αποτελεσματικότητα της χρήσης προσωρινής μνήμης

- Αν και η χρήση προσωρινής μνήμης βοηθάει την αντιμετώπιση μεγάλης κίνησης δεδομένων σε εντολές E/E, εν τούτοις κάποια στιγμή ο χώρος αυτός θα γεμίσει αν η μέση ανάγκη σε E/E μίας διεργασίας είναι μεγαλύτερη από τη χωρητικότητα της προσωρινής μνήμης.
- Παρόλα αυτά, σε ένα περιβάλλον πολυπρογραμματισμού όπου εκτός των εντολών E/E υπάρχουν και άλλες δραστηριότητες, η τεχνική αυτή ανεβάζει την απόδοση του Λ.Σ. γενικώς και των διεργασιών του ειδικότερα.

(d) Circular buffering

36

36

## Περιεχόμενα

- Συσσκευές Ε/Ε.
- Οργάνωση των λειτουργιών Ε/Ε.
- Ο ρόλος του Λ.Σ. στη διαχείριση των συσκευών Ε/Ε.
- Χρήση προσωρινής μνήμης στις λειτουργίες Ε/Ε.
- Διαχείριση δίσκου.
- Συστήματα Raid.
- Μνήμη cache στο δίσκο.
- Ε/Ε στο UNIX SVR4.
- Ε/Ε στο LINUX.
- Ε/Ε στα Windows.

37

## Ιεράρχηση τεχνολογιών αποθήκευσης δεδομένων

➤ The memory hierarchy

The memory hierarchy

38

## Ιστορική εξέλιξη τεχνολογιών δίσκου

- Μέχρι το 1955 τα δεδομένα ήταν αποθηκευμένα σε διάτρητες κάρτες και μαγνητικές ταινίες.
- Το 1956 κυκλοφορεί ο πρώτος σκληρός δίσκος, IBM 350 RAMAC, χωρητικότητας 5 MB.
- Το 1973 κυκλοφορεί από την IBM η τεχνολογία «Winchester» (μοντέλο 3340, χωρητικότητα 30 MB), με το δίσκο να είναι προστατευμένος σε μεταλλική θήκη.
- Από τα μέσα της δεκαετίας του '70 κυκλοφορούν οι εύκαμπτοι δίσκοι (floppy disks), μεγέθους 8, 5.25 και 3.5 ιντσών και χωρητικότητας από 110 KB μέχρι 2 MB.
- Τα τελευταία 20 χρόνια η χωρητικότητα των δίσκων έχει αυξηθεί από μερικές εκατοντάδες MB σε μερικές εκατοντάδες GB.

39

## Δομή του δίσκου

40

## Επιφάνεια του δίσκου

41

## Αποθήκευση δεδομένων στο δίσκο

42

### Διαφορετικές μέθοδοι αποθήκευσης δεδομένων στο δίσκο

(a) Constant angular velocity

(b) Multiple zoned recording

43

### Μορφοποίηση του δίσκου

- Η μορφοποίηση (formatting) του δίσκου δημιουργεί μία ομάδα από ομόκεντρους κυκλικούς διαύλους εγγραφής (tracks).
- Κάθε τέτοιος διαύλος αποτελείται από μία ομάδα τομέων (sectors), οι οποίοι χωρίζονται μεταξύ τους με κενά.
- Υπάρχουν δύο τρόποι να οργανωθούν οι τομείς σε ένα δίσκο:
  - Η τεχνική της σταθερής γωνιακής ταχύτητας (constant angular velocity) δημιουργεί τον ίδιο αριθμό από τομείς ανά διαύλο, είναι απλή στην υλοποίηση αλλά σπαταλάει χώρο στους εξωτερικούς διαύλους του δίσκου.
  - Η τεχνική των πολλαπλών ζωνών εγγραφής (multiple zone recording) χωρίζει το δίσκο σε ένα αριθμό από ζώνες (συνήθως 16). Σε κάθε ζώνη ο αριθμός των bits ανά τομέα είναι σταθερός αλλά αυξάνει από ζώνη σε ζώνη από μέσα προς τα έξω. Κατ' επέκταση, ο αριθμός των τομέων επίσης αυξάνει από ζώνη σε ζώνη. Η τεχνική αυτή παρέχει μεγαλύτερη πυκνότητα και επομένως μεγαλύτερη χωρητικότητα στο δίσκο αλλά είναι πιο πολύπλοκη στην υλοποίηση και δεν έχει σταθερό χρόνο προσπέλασης στα δεδομένα από διαύλο σε διαύλο.

44

43

44

### Λειτουργία του δίσκου

- Ο δίσκος περιστρέφεται με σταθερή ταχύτητα.
- Για να γίνει ανάγνωση από ή εγγραφή στο δίσκο, η κεφαλή πρέπει να τοποθετηθεί πάνω από τον επιθυμητό διάυλο και στην αρχή του επιθυμητού τομέα σε αυτόν το διάυλο.

45

### Χρονικές παράμετροι σχετιζόμενες με τη λειτουργία του δίσκου

- Οι λεπτομέρειες της μεταφοράς δεδομένων από και προς ένα δίσκο εξαρτάται από πολλούς παράγοντες σχετιζόμενους με το σύστημα, το Λ.Σ., τον ελεγκτή του δίσκου και τα κανάλια Ε/Ε.
  - Ένα γενικό διάγραμμα φαίνεται κατωτέρω:

Figure 11.6 Timing of a Disk I/O Transfer

46

45

46

### Χρονικές παράμετροι σχετιζόμενες με την απόδοση του δίσκου — 1

47

### Χρονικές παράμετροι σχετιζόμενες με την απόδοση του δίσκου — 2

- Ο χρόνος ανάγνωσης ή εγγραφής ενός μπλοκ δίσκου καθορίζεται από 3 παράγοντες:
  - Τον χρόνο αναζήτησης (seek time), δηλαδή το χρονικό διάστημα που χρειάζεται η κεφαλή ανάγνωσης/εγγραφής να τοποθετηθεί πάνω από τον κατάλληλο διάυλο εγγραφής του δίσκου.
  - Την καθυστέρηση περιστροφής (rotational / latency time), δηλαδή το χρονικό διάστημα που χρειάζεται να φτάσει ο κατάλληλος τομέας κάτω από την κεφαλή (ένας δίσκος περιστρέφεται με ταχύτητα μεταξύ 3600-15000 rpm).
  - Τον πραγματικό χρόνο μεταφοράς (transfer / transmission time).

48

47

48



### Χρονικές παράμετροι σχετιζόμενες με την απόδοση του δίσκου — 3

- Το άθροισμα του χρόνου αναζήτησης και καθυστέρησης περιστροφής αποτελεί τον χρόνο προσπέλασης (access time), δηλαδή το χρονικό διάστημα που χρειάζεται η κεφαλή για να φτάσει στο σημείο του δίσκου από το οποίο θα διαβάσει ή στο οποίο θα γράψει.
- Η παράμετρος εκείνη που καθορίζει το ρυθμό απόδοσης του δίσκου είναι ο χρόνος αναζήτησης. Με τον όρο διαχείριση δίσκου εννοούμε κυρίως την προσπάθεια ελαχιστοποίησης των κινήσεων του βραχίονα του δίσκου ώστε να τοποθετηθεί πάνω από τον κατάλληλο διάυλο εγγραφής, με άλλα λόγια την ελαχιστοποίηση του χρόνου αναζήτησης.
- Στους δίσκους της εποχής μας ο μέσος χρόνος αναζήτησης είναι κάτω από 10 ms.
- Σημειώτεον ότι η καθυστέρηση περιστροφής μπορεί να ελαχιστοποιηθεί με την επιλογή του τομέα εκείνου που βρίσκεται πιο κοντά στον βραχίονα (κάτι ανάλογο της μεθόδου SSTF).

49

### Πολιτικές χρονοδρομολόγησης (της κεφαλής του) δίσκου

- Για να μπορούμε να συγκρίνουμε τις διαφορετικές πολιτικές ή αλγόριθμους χρονοδρομολόγησης στο δίσκο, θεωρούμε ένα συγκεκριμένο σενάριο στο οποίο ο δίσκος έχει 200 αυλάκια και η κεφαλή του δίσκου βρίσκεται αρχικά στο αυλάκι 100, με κατεύθυνση προς τα αυλάκια με αύξοντα αριθμό.
- Οι αιτήσεις για διάβασμα ή γράψιμο στο δίσκο, με τη χρονική σειρά που εμφανίζονται, είναι στα ακόλουθα αυλάκια:

55, 58, 39, 18, 90, 160, 150, 38, 184.

50

### Τυχαία χρονοδρομολόγηση

- Ο αλγόριθμος της τυχαίας χρονοδρομολόγησης (random scheduling) απλά διαλέγει κάποια από τις αιτήσεις για E/E που εκκρεμούν.
- Επειδή και η συμπεριφορά του βραχίονα θα είναι και αυτή τυχαία τα αποτελέσματα θα είναι τα χειρότερα δυνατά.
- Ο αλγόριθμος αυτός χρησιμοποιείται μόνο σαν σημείο σύγκρισης και εκτίμησης των αποτελεσμάτων που έχουν οι υπόλοιποι αλγόριθμοι.

51

### Με βάση τις προτεραιότητες

- Ο αλγόριθμος των προτεραιοτήτων εξυπηρετεί την αίτηση E/E που ανήκει στη διεργασία με την υψηλότερη προτεραιότητα.
- Σκοπός του είναι κυρίως η βελτιστοποίηση άλλων παραμέτρων του Λ.Σ. (όπως χρόνου απόκρισης) και όχι της χρήσης του δίσκου.
- Κατ' επέκταση, ο αλγόριθμος αυτός είναι εκτός του ελέγχου του συστήματος διαχείρισης του δίσκου.
- Είναι κατάλληλος μόνο για μικρές εργασίες σε δέσμες ή διαλογικές (οι οποίες συνήθως έχουν υψηλή προτεραιότητα).
- Όμως μεγάλες εργασίες πιθανόν να αντιμετωπίσουν προβλήματα παρατεταμένης στέρσης.
- Επιπλέον, δύναται να οδηγήσει και σε αντίδραση των χρηστών οι οποίοι εσκεμμένα πιθανόν να σπάσουν τις εργασίες τους σε μικρότερες για να ξεγελάσουν το σύστημα.
- Έχει άσχημα αποτελέσματα σε συστήματα βάσεων δεδομένων.

52

### Πρώτη Εισερχόμενη, Πρώτη Εξερχόμενη

- Ο αλγόριθμος της πρώτης εισερχόμενης, πρώτης εξερχόμενης (FIFO), είναι ο πιο απλός και εξυπηρετεί τις αιτήσεις με τη χρονική σειρά που έχουν δημιουργηθεί.
- Το πλεονέκτημα είναι ότι ο αλγόριθμος αυτός είναι δίκαιος προς όλες τις αιτήσεις.
- Όμως έχει καλά αποτελέσματα μόνο αν οι αιτήσεις είναι λίγες και αναφέρονται σε χώρους του δίσκου που γειτονεύουν. Στη γενική περίπτωση έχει τα ίδια αποτελέσματα με της τυχαίας επιλογής.

(a) FIFO

53

### Τελευταία Εισερχόμενη, Πρώτη Εξερχόμενη

- Μία παραλλαγή του προηγούμενου αλγόριθμου είναι η Τελευταία Εισερχόμενη, Πρώτη Εξερχόμενη (last-in-first-out).
- Βασίζεται στο ότι υπάρχουν πολλές πιθανότητες οι πιο πρόσφατες αιτήσεις να ανήκουν στον ίδιο χρήστη που μπορεί να θέλει να προσπελάσει τον ίδιο χώρο μνήμης (π.χ. σειριακό αρχείο).
- Έχει καλά αποτελέσματα μόνο για τέτοια σενάρια.

54

## Συντομότερη Αναζήτηση Πρώτη

- Ο αλγόριθμος της Συντομότερης Αναζήτησης Πρώτη (shortest service time first) εξυπηρετεί την πιο κοντινή αίτηση.
- Ελαχιστοποιεί το χρόνο αναζήτησης από αίτηση σε αίτηση αλλά όχι κατ' ανάγκη και συνολικά (δηλαδή το μέσο χρόνο αναζήτησης).
- Στην περίπτωση που υπάρχουν δύο αιτήσεις ίσης απόστασης (μπροστά και πίσω) διαλέγεται τυχαία μία από αυτές.
- Τείνει να εξυπηρετεί καλύτερα αιτήσεις που αναφέρονται στις μεσαίες περιοχές μεταξύ των εσωτερικών και εξωτερικών διαύλων εγγραφής.
- Είναι κατά κάποιο τρόπο το αντίστροφο του αλγόριθμου SPN στη χρονοδρομολόγηση διεργασιών και μπορεί και αυτός να προκαλέσει παρατεταμένη στέρηση μερικών αιτήσεων E/E αν υπάρχουν συνεχώς καινούργιες αιτήσεις οι οποίες προτιμούνται έναντι των παλαιότερων.

(b) SSTF

55

## Σάρωση Προς την Ίδια Κατεύθυνση

- Ο αλγόριθμος της Σάρωσης Προς την Ίδια Κατεύθυνση (SCAN) διαλέγει την πιο κοντινή αίτηση που βρίσκεται στην κατεύθυνση προς την οποία σαρώνει (προς το κέντρο ή την περιφέρεια του δίσκου).
- Μόνο όταν εξυπηρετηθούν όλες οι αιτήσεις προς την ίδια κατεύθυνση, αντιστρέφεται η φορά της σάρωσης.
- Ο αλγόριθμος αυτός έχει σχεδόν τα ίδια αποτελέσματα με τον προηγούμενο χωρίς όμως να δημιουργεί προβλήματα παρατεταμένης στέρησης.
- Τείνει όμως και αυτός να εξυπηρετεί καλύτερα αιτήσεις που αναφέρονται στις μεσαίες περιοχές μεταξύ των εσωτερικών και εξωτερικών διαύλων εγγραφής καθώς επίσης και τις αιτήσεις που δημιουργήθηκαν πρόσφατα.
- Ο αλγόριθμος αυτός λέγεται και αλγόριθμος του ανελκυστήρα.

(e) SCAN

56

## Κυκλική Σάρωση

- Ο αλγόριθμος της Κυκλικής Σάρωσης (C-SCAN) ουσιαστικά σαρώνει προς μία κατεύθυνση μόνο, συνήθως από έξω προς τα μέσα.
- Μόλις ο βραχίονας ολοκληρώσει το σάρωμα, μετατοπίζεται γρήγορα στον εξωτερικό δίαυλο και επαναρχίζει το σάρωμα.
- Ο αλγόριθμος αυτός επιλύει το πρόβλημα της μη ικανοποιητικής εξυπηρέτησης των αιτήσεων που αναφέρονται στις εσωτερικές και εξωτερικές περιοχές του δίσκου.

(d) C-SCAN

57

## LOOK και C-LOOK

- Θεωρητικά οι αλγόριθμοι SCAN και C-SCAN μετακινούν το βραχίονα μέχρι το τέλος της εσωτερικής ή εξωτερικής επιφάνειας του δίσκου, ακόμα και αν από κάποιο σημείο και μετά δεν υπάρχει κάποια αίτηση για εξυπηρέτηση.
- Στην πράξη όμως, η μετακίνηση του βραχίονα προς μία κατεύθυνση τερματίζεται στην τελευταία αίτηση προς εκείνη την κατεύθυνση.
- Αυτές οι παραλλαγές ονομάζονται LOOK και C-LOOK.
- Σημειωτέον, ότι στο παράδειγμα του βιβλίου, αν και γίνεται αναφορά σε SCAN/C-SCAN, ουσιαστικά χρησιμοποιείται η λογική του LOOK/C-LOOK.

58

## Σάρωση N Βημάτων

- Ο αλγόριθμος της Σάρωσης N Βημάτων (N-step-SCAN) κάνει χρήση του απλού αλγόριθμου της σάρωσης αλλά μόνο για μία ομάδα από αιτήσεις μεγέθους N.
- Κατά τη διάρκεια της σάρωσης οι υπόλοιπες αιτήσεις που καταφθάνουν μπαίνουν στην επόμενη ομάδα που θα εξυπηρετηθεί μόλις ολοκληρωθεί η εξυπηρέτηση της τρέχουσας ομάδας για την οποία γίνεται η σάρωση.
- Αν η επόμενη ομάδα έχει παραπάνω αιτήσεις από N, τότε θα εξυπηρετηθούν μόνο οι πρώτες N.
- Η μέθοδος αυτή τείνει να είναι δίκαιη προς όλες τις αιτήσεις ανεξάρτητα του χρόνου άφιξής τους.
- Αν ο αριθμός N είναι μικρός ο αλγόριθμος συμπεριφέρεται σαν τον FIFO ενώ για μεγάλες τιμές η απόδοσή του είναι ανάλογη του SCAN.

59

## F-SCAN

- Είναι παραλλαγή του προηγούμενου αλγόριθμου, όπου γίνεται χρήση 2 ουρών.
- Μόλις ξεκινήσει η σάρωση, όλες οι αιτήσεις βρίσκονται σε μία από τις δύο ουρές και η άλλη ουρά είναι άδεια.
- Κατά τη διάρκεια της σάρωσης, όλες οι νέες αιτήσεις εισέρχονται στη δεύτερη ουρά.
- Όταν ολοκληρωθεί η τρέχουσα σάρωση, εξυπηρετούνται οι αιτήσεις της άλλης ουράς, κοκ.
- Επομένως μπορεί να θεωρηθεί ως παραλλαγή του αλγόριθμου N-SCAN όπου το N είναι μεταβλητό.

60

## Απόδοση των διαφορετικών αλγόριθμων

Η σύγκριση της απόδοσης των διαφορετικών αλγόριθμων εστιάζεται στο μέσο χρόνο αναζήτησης για κάθε αλγόριθμο.

(a) FIFO (starting at track 100)		(b) SSTF (starting at track 100)		(c) SCAN (starting at track 100, in the direction of increasing track number)		(d) C-SCAN (starting at track 100, in the direction of increasing track number)	
Next track accessed	Number of tracks traversed	Next track accessed	Number of tracks traversed	Next track accessed	Number of tracks traversed	Next track accessed	Number of tracks traversed
55	45	90	10	150	50	150	50
58	3	58	32	160	10	160	10
39	19	55	3	184	24	184	24
18	21	39	16	90	94	18	106
90	72	38	1	58	32	38	20
160	70	18	20	55	3	39	1
150	10	150	132	39	16	55	16
38	112	160	10	38	1	58	3
184	146	184	24	18	20	90	32
Average seek length	55.3	Average seek length	27.5	Average seek length	27.8	Average seek length	35.8

61

## Κατηγοριοποίηση των αλγορίθμων χρονοδρομολόγησης δίσκου

Table 11.3 Disk Scheduling Algorithms

Name	Description	Remarks
Selection according to requestor		
RSS	Random scheduling	For analysis and simulation
FIFO	First in first out	Fairest of them all
PRI	Priority by process	Control outside of disk queue management
LIFO	Last in first out	Maximize locality and resource utilization
Selection according to requested item		
SSTF	Shortest service time first	High utilization, small queues
SCAN	Back and forth over disk	Better service distribution
C-SCAN	One way with fast return	Lower service variability
N-step-SCAN	SCAN of N records at a time	Service guarantee
FSCAN	N-step-SCAN with N = queue size at beginning of SCAN cycle	Load sensitive

62

## Περιεχόμενα

- Συσκευές E/E.
- Οργάνωση των λειτουργιών E/E.
- Ο ρόλος του Λ.Σ. στη διαχείριση των συσκευών E/E.
- Χρήση προσωρινής μνήμης στις λειτουργίες E/E.
- Διαχείριση δίσκου.
- **Συστήματα Raid.**
- Μνήμη cache στο δίσκο.
- E/E στο UNIX SVR4.
- E/E στο LINUX.
- E/E στα Windows.

63

## Πολλαπλοί δίσκοι

- Η απόδοση της εξυπηρέτησης των αιτήσεων E/E μπορεί να βελτιωθεί με τη χρήση πολλαπλών δίσκων και το διαμοιρασμό των αιτήσεων σε αυτούς.
- Αυτό οδηγεί στην παράλληλη (και επομένως πιο γρήγορη) εκτέλεση των αιτήσεων.
- Επιπλέον, αν τα ίδια δεδομένα βρίσκονται αποθηκευμένα ταυτόχρονα σε περισσότερους από ένα δίσκους, δεν κινδυνεύουν να χαθούν αν ένας δίσκος καταστραφεί.

64

## Πλεονάζουσα Διάταξη Ανεξάρτητων Δίσκων

- Η τεχνική της Πλεονάζουσας Διάταξης Ανεξάρτητων Δίσκων (Redundant Array of Independent Disks – RAID) κάνει χρήση πολλαπλών δίσκων.
- Μία ομάδα από φυσικούς δίσκους αντιμετωπίζεται από το Λ.Σ. ως μία λογική μονάδα δίσκου.
- Τα δεδομένα διαμοιράζονται σε διαφορετικούς δίσκους και πιθανόν να υπάρχουν αντίγραφα τους σε περισσότερους από ένα δίσκους.
- Ανάλογα με την λειτουργικότητά τους, τα συστήματα RAID κατηγοριοποιούνται σε 7 επίπεδα (0-6).

65

## RAID 0 (χρήση λωρίδων)

- Δεν θεωρείται γνήσιο μέλος της οικογένειας RAID γιατί δεν υπάρχουν εφεδρικά αντίγραφα των δεδομένων.
- Η καταστροφή ενός δίσκου συνεπάγεται απώλεια δεδομένων.
- Παράλληλη και κατ' επέκταση γρήγορη εξυπηρέτηση αιτήσεων E/E, ειδικά αν μία αίτηση αποτελείται από πρόσβαση σε λογικά συνεχόμενες λωρίδες (π.χ. 4,5,6,7).

(a) RAID 0 (non-redundant)

66

### RAID 1 (χρήση κατοπτρισμού δεδομένων)

- Υπάρχουν εφεδρικά αντίγραφα των δεδομένων.
- Κάθε λογική λωρίδα αντικατοπτρίζεται σε δύο φυσικούς δίσκους.
- Μία αίτηση ανάγνωσης μπορεί να εξυπηρετηθεί από οποιονδήποτε δίσκο (ανάλογα με το σε ποιον από τους δύο θα έχουμε το μικρότερο χρόνο αναζήτησης και περιστροφής).
- Μία αίτηση εγγραφής απαιτεί την ενημέρωση και των δύο αντιγράφων των εμπλεκόμενων δεδομένων, αλλά μπορεί να γίνει παράλληλα.
- Αν καταστραφεί ο ένας από τους δίσκους, τα δεδομένα μπορούν να ανακτηθούν από τον άλλο.

(b) RAID 1 (mirrored)

67

### RAID 2 (χρήση κώδικα Hamming)

- Όλοι οι δίσκοι συμμετέχουν στην εκτέλεση μίας εντολής E/E.
- Τα δεδομένα (όπως και προηγουμένως) είναι οργανωμένα σε (πολύ μικρές αυτή τη φορά) λωρίδες του ενός byte.
- Για την αντιμετώπιση λαθών χρησιμοποιείται κώδικας Hamming που διορθώνει λάθη του ενός bit και μπορεί να ανιχνεύσει λάθη των δύο bits.

(c) RAID 2 (redundancy through Hamming code)

68

### RAID 3 (χρήση ισοτιμίας παρεμβλλόμενου bit)

- Παρόμοιο με το RAID 2 αλλά όλα τα bits αποθηκεύονται σε ένα δίσκο.
- Για την αντιμετώπιση λαθών αντί να γίνεται χρήση κώδικα Hamming, χρησιμοποιείται ένα απλό bit ισοτιμίας (parity bit).

(d) RAID 3 (bit-interleaved parity)

69

### RAID 4 (χρήση ισοτιμίας μπλοκ)

- Ο κάθε δίσκος λειτουργεί ανεξάρτητα από τους υπόλοιπους.
- Τα δεδομένα χωρίζονται σε ομάδες (μπλοκ), που είναι μεγαλύτερα σε μέγεθος από τις λωρίδες.
- Για κάθε ομάδα μπλοκ υπάρχει ένα bit ισοτιμίας.

(e) RAID 4 (block-level parity)

70

### RAID 5 (χρήση καταναμημένης ισοτιμίας)

- Παρόμοιο με το RAID 4 αλλά τα bit ισοτιμίας είναι καταναμημένα.

(f) RAID 5 (block-level distributed parity)

71

### RAID 6 (χρήση διπλής εφεδρικότητας)

- Χρήση δύο bits εφεδρικότητας.
- Μπορεί να αντιμετωπίσει δύο λάθη.

(g) RAID 6 (dual redundancy)

72

## Σύγκριση των επιπέδων RAID

Table 11.4 RAID Levels

Category	Level	Description	Disks required	Data availability	Large I/O data transfer capacity	Small I/O request rate
Striping	0	Nonredundant	$N$	Lower than single disk	Very high	Very high for both read and write
Mirroring	1	Mirrored	$2N$	Higher than RAID 2, 3, 4, or 5; lower than RAID 6	Higher than single disk for read; similar to single disk for write	Up to twice that of a single disk for read; similar to single disk for write
Parallel access	2	Redundant via Hamming code	$N + n$	Much higher than single disk; comparable to RAID 2, 4, or 5	Highest of all listed alternatives	Approximately twice that of a single disk
	3	Bit-interleaved parity	$N + 1$	Much higher than single disk; comparable to RAID 2, 4, or 5	Highest of all listed alternatives	Approximately twice that of a single disk
Independent access	4	Block-interleaved parity	$N + 1$	Much higher than single disk; comparable to RAID 2, 3, or 5	Similar to RAID 0 for read; significantly lower than single disk for write	Similar to RAID 0 for read; significantly lower than single disk for write
	5	Block-interleaved distributed parity	$N + 1$	Much higher than single disk; comparable to RAID 2, 3, or 4	Similar to RAID 0 for read; lower than single disk for write	Similar to RAID 0 for read; generally lower than single disk for write
	6	Block-interleaved dual distributed parity	$N + 2$	Highest of all listed alternatives	Similar to RAID 0 for read; lower than RAID 5 for write	Similar to RAID 0 for read; significantly lower than RAID 5 for write

73

## Περιεχόμενα

- Συσκευές E/E.
- Οργάνωση των λειτουργιών E/E.
- Ο ρόλος του Λ.Σ. στη διαχείριση των συσκευών E/E.
- Χρήση προσωρινής μνήμης στις λειτουργίες E/E.
- Διαχείριση δίσκου.
- Συστήματα Raid.
- Μνήμη cache στο δίσκο.
- E/E στο UNIX SVR4.
- E/E στο LINUX.
- E/E στα Windows.

74

## Μνήμη cache

- Χρήση ενός μέρους της πολύ γρήγορης (ή κρυφής) μνήμης (cache memory) για αποθήκευση των περιεχομένων των πιο συχνά χρησιμοποιούμενων τομέων του δίσκου (disk cache).
- Όταν πρόκειται να εκτελεσθεί μία αίτηση E/E σε κάποιο τομέα του δίσκου, εξετάζεται πρώτα αν ο συγκεκριμένος τομέας βρίσκεται στη μνήμη cache.
  - Αν ναι, η αίτηση εξυπηρετείται από τη μνήμη cache αλλιώς ο ζητούμενος τομέας μεταφέρεται σε αυτήν.
- Η ιδέα εδώ είναι ότι λόγω της τοπικότητας των αναφορών, υπάρχει μεγάλη πιθανότητα να υπάρχουν αρκετές αιτήσεις στον ίδιο τομέα.

75

## Οργάνωση μνήμης cache

- Υπάρχουν δύο θέματα που πρέπει να αντιμετωπισθούν στη χρήση μνήμης cache.
- Το πρώτο είναι πως ακριβώς γίνεται η προσπέλαση των δεδομένων που βρίσκονται εκεί.
- Υπάρχουν δύο πιθανοί τρόποι:
  - Μεταφορά των δεδομένων αυτών στη "συνθησισμένη" κύρια μνήμη πριν να χρησιμοποιηθούν.
  - Απ' ευθείας χρήση τους από τη μνήμη cache με τη χρήση δεικτών.
- Η δεύτερη λύση φαίνεται να είναι καλύτερη λόγω του ότι δεν ξοδεύεται χρόνος στη μεταφορά των δεδομένων από τη μνήμη cache στην κύρια μνήμη. Επίσης, επιτρέπει την κοινή χρήση των δεδομένων από πολλές διεργασίες χρησιμοποιώντας μοντέλα ταυτοχρονισμού τύπου αναγνωστών-εγγραφών.

76

## Αντικατάσταση τομέων στη μνήμη cache

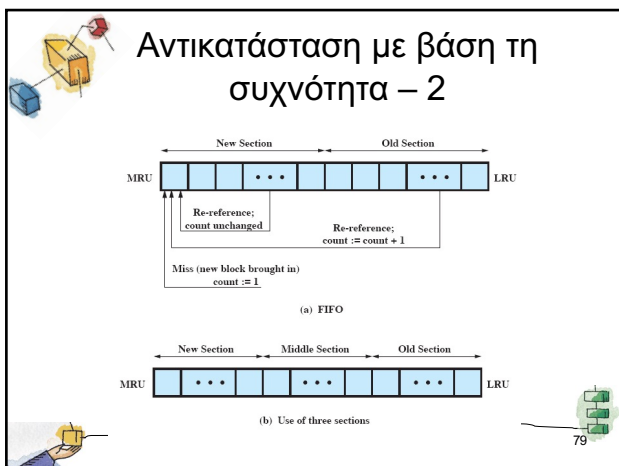
- Το δεύτερο πρόβλημα έχει να κάνει με το ποια πολιτική αντικατάστασης τομέων του δίσκου θα υλοποιηθεί (αυτό είναι ένα παρόμοιο πρόβλημα με εκείνο της αντικατάστασης σελίδων).
  - Ο καλύτερος αλγόριθμος φαίνεται να είναι ο LRU (Λιγότερο Πρόσφατα Χρησιμοποιούμενου Τομέα) λόγω της τοπικότητας αναφορές στα δεδομένα.
    - Εδώ αντικαθίσταται ο τομέας εκείνος που βρίσκεται στη μνήμη χωρίς να έχει γίνει αναφορά σε αυτόν για το μεγαλύτερο χρονικό διάστημα.
  - Ένας άλλος αλγόριθμος με καλά αποτελέσματα είναι αυτός του Λιγότερο Αναφερθέντα Τομέα (LFU, Least Frequently Used).
    - Εδώ αντικαθίσταται ο τομέας εκείνος για τον οποίον έχουν γίνει οι λιγότερες αναφορές (εδώ γίνεται χρήση ενός μετρητή).
    - Προσοχή όμως πρέπει να δίνεται εδώ στο αν η συχνότητα αναφορές είναι ομοιογενής χρονικά αλλιώς η εικόνα που παρουσιάζεται είναι πλάσματική.
    - Αυτό συμβαίνει όταν κάποιος τομέας χρησιμοποιείται σε σχετικά αραιά χρονικά διαστήματα, αλλά όταν χρησιμοποιείται γίνονται πολλές αναφορές σε αυτόν, λόγω της τοπικότητας των αναφορών.
- Σε αυτές τις περιπτώσεις ο μετρητής θα έχει υψηλή τιμή, η οποία όμως δεν αντικατοπτρίζει την πραγματική συχνότητα χρήσης του τομέα.

77

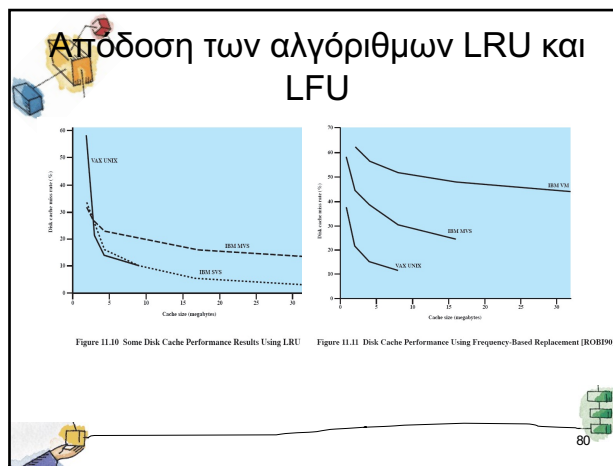
## Αντικατάσταση με βάση τη συχνότητα - 1

- Οι τομείς οργανώνονται σε ένα σωρό (stack), με τους τομείς που έχουν αναφερθεί πιο πρόσφατα να βρίσκονται στην κορυφή του σωρού (νέο τμήμα) και τους υπόλοιπους τομείς στο υπόλοιπο μέρος του σωρού (παλιό τμήμα).
- Κάθε φορά που ένας τομέας αναφέρεται, μεταφέρεται στην κορυφή του σωρού, αν δεν είναι ήδη εκεί. Στην πρώτη περίπτωση, αυξάνεται και ο μετρητής αναφοράς κατά 1.
- Σταδιακά, η κορυφή του σωρού θα έχει τους τομείς που χρησιμοποιούνται πιο συχνά, επομένως στην περίπτωση που πρέπει να αντικατασταθεί κάποιος τομέας, επιλέγεται ένας από αυτούς που βρίσκονται χαμηλά στο σωρό.
- Πρακτικές μετρήσεις έδειξαν ότι η διαμοίραση του σωρού σε δύο περιοχές συχνά δεν έχει τα επιθυμητά αποτελέσματα, δηλαδή το χρονικό διάστημα για τη μετακίνηση ενός τομέα από το νέο στο παλιό τμήμα δεν είναι αρκετός για να είναι σίγουρο ότι δεν θα χρειασθεί στο μέλλον.
- Η τεχνική αυτή βελτιώνεται με την προσθήκη ενός ενδιάμεσου τρίτου χώρου για την αντικατάσταση τομέων που βρίσκονται μόνο στο παλιό τμήμα.

78



79



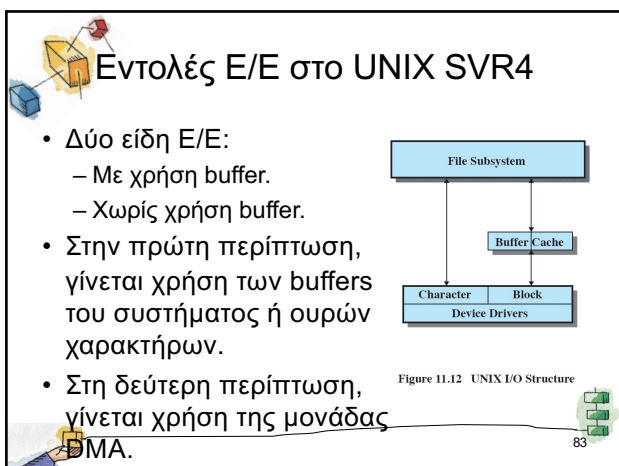
80

- ## Περιεχόμενα
- Συσκευές E/E.
  - Οργάνωση των λειτουργιών E/E.
  - Ο ρόλος του Λ.Σ. στη διαχείριση των συσκευών E/E.
  - Χρήση προσωρινής μνήμης στις λειτουργίες E/E.
  - Διαχείριση δίσκου.
  - Συστήματα Raid.
  - Μνήμη cache στο δίσκο.
  - E/E στο UNIX SVR4.
  - E/E στο LINUX.
  - E/E στα Windows.

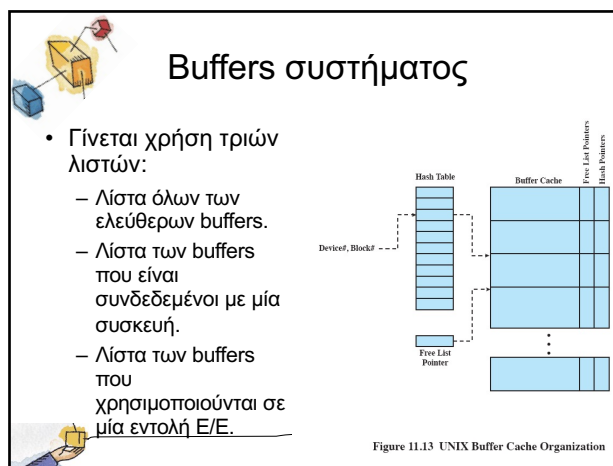
81

- ## Συσκευές ως αρχεία
- Κάθε συσκευή αντιμετωπίζεται ως ένα ειδικό αρχείο.
    - Ο χειρισμός του γίνεται από το σύστημα διαχείρισης αρχείων.
    - Παρέχει ομοιόμορφη πρόσβαση στις συσκευές από χρήστες και διεργασίες.
  - Η χρήση μίας συσκευής για είσοδο ή έξοδο αντιμετωπίζεται από το σύστημα ως αίτηση για διάβασμα ή γράψιμο στο αρχείο που σχετίζεται με την εν λόγω συσκευή.

82



83



84

## Μνήμη cache χαρακτήρων

- Χρησιμοποιείται για συσκευές που χειρίζονται χαρακτήρες.
  - Π.χ. θερματικά ή εκτυπωτές.
- Μία ουρά χαρακτήρων είτε γράφεται από τη συσκευή E/E και διαβάζεται από τη διεργασία είτε αντίστροφα.
- Σε κάθε περίπτωση γίνεται χρήση του μοντέλου του παραγωγού-καταναλωτή, όπου ο κάθε χαρακτήρας διαβάζεται μόνο μία φορά και μετά καταστρέφεται.
- Αντίθετα, στην περίπτωση των buffers, γίνεται χρήση του μοντέλου των αναγνωστών-εγγραφών όπου τα περιεχόμενα ενός buffer μπορούν να διαβασθούν πολλαπλές φορές.

85

## E/E χωρίς buffers

- Η εκτέλεση εντολών E/E χωρίς buffers γίνεται μεταξύ μίας συσκευής και μίας διεργασίας μέσω της μονάδας DMA.
  - Είναι πολύ γρήγορη.
  - Η διεργασία κλειδώνεται στη μνήμη και δεν μπορεί να απομακρυνθεί από αυτή, μέχρις ότου ολοκληρωθεί η εκτέλεση της εντολής.
  - Η συσκευή δεσμεύεται αποκλειστικά από τη διεργασία που εκτελεί την εντολή E/E και δεν μπορεί να χρησιμοποιηθεί από άλλη διεργασία.

86

## Κατηγοριοποίηση των τεχνικών εκτέλεσης εντολών E/E

Table 11.5 Device I/O in UNIX

	Unbuffered I/O	Buffer Cache	Character Queue
Disk drive	X	X	
Tape drive	X	X	
Terminals			X
Communication lines			X
Printers	X		X

87

## Περιεχόμενα

- Συσκευές E/E.
- Οργάνωση των λειτουργιών E/E.
- Ο ρόλος του Λ.Σ. στη διαχείριση των συσκευών E/E.
- Χρήση προσωρινής μνήμης στις λειτουργίες E/E.
- Διαχείριση δίσκου.
- Συστήματα Raid.
- Μνήμη cache στο δίσκο.
- E/E στο UNIX SVR4.
- E/E στο LINUX.
- E/E στα Windows.

88

## Ομοιότητες Linux με Unix

- Το Linux και το Unix (π.χ. SVR4) είναι παρόμοια στην εκτέλεση εντολών E/E.
- Όπως και το Unix, έτσι και το Linux συσχετίζει ένα αρχείο για κάθε συσκευή.
- Επιπλέον αναγνωρίζονται οι συσκευές ανάλογα με το αν χειρίζονται χαρακτήρες ή μπλοκ χαρακτήρων ή είναι συσκευές δικτύου.

89

## Χρονοδρομολόγηση του δίσκου

- Το Linux χρησιμοποιεί την πολιτική του ανεγκυστήρα στη χρονοδρομολόγηση του δίσκου, που είναι παραλλαγή του αλγόριθμου LOOK.
- Γίνεται χρήση μίας ουράς με αιτήσεις E/E.
- Οι αιτήσεις μπαίνουν σε σειρά με βάση τον αριθμό του τομέα στον οποίον ζητούν πρόσβαση.
- Οι αιτήσεις εξυπηρετούνται με τη σειρά που η κεφαλή του δίσκου μεταφέρεται από ένα τομέα στον επόμενο.
- Στο Linux 2.6 ο βασικός αυτός αλγόριθμος έχει επαυξηθεί με δύο επιπλέον αλγόριθμους:
  - Τον αλγόριθμο με βάση τις προθεσμίες.
  - Τον αλγόριθμο προβλεπτικής εκτέλεσης εντολών E/E.

90

## Χρονοδρομολογητής με βάση προθεσμίες

- Κάθε αίτηση E/E εισέρχεται στην πρώτη ουρά και στην αντίστοιχη δεύτερη ή τρίτη.
- Η κάθε αίτηση έχει προθεσμία 0.5 δ. για ανάγνωση και 5 δ. για εγγραφή.
- Ο επιλογέας συνήθως εκτελεί τις αιτήσεις της πρώτης ουράς (όταν μία αίτηση εκτελείται απομακρύνεται και από την αντίστοιχη δεύτερη ή τρίτη ουρά).
- Όμως όταν η πρώτη αίτηση στην κορυφή μίας από τις άλλες δύο ουρές είναι πιο παλιά από την προθεσμία της, ο επιλογέας την επιλέγει μαζί με τις αμέσως επόμενες αιτήσεις αυτής της ουράς (και πάλι οι αιτήσεις που επιλέγονται απομακρύνονται και από την πρώτη ουρά).

91

## Προβλεπτική εκτέλεση εντολών E/E

- Η πολιτική αυτή συμπληρώνει την προηγούμενη.
- Κάθε φορά που εκτελείται μία εντολή διαβάσματος, ο επιλογέας αδρανοποιεί το σύστημα χρονοδρομολόγησης για 6 χιλιοδευτερόλεπτα.
- Η ιδέα είναι ότι λόγω της τοπικότητας των αναφορών, μπορεί να υπάρχουν και άλλες εντολές διαβάσματος στον τομέα της πρώτης εντολής.
- Σε αυτήν την περίπτωση, οι εντολές αυτές εξυπηρετούνται αμέσως.
- Αλλιώς, εφαρμόζεται κανονικά η προηγούμενη πολιτική.

92

## Ο χρονοδρομολογητής NOOP

- Είναι ο πιο απλός από τους χρονοδρομολογητές του Linux.
- Απλά τοποθετεί τις αιτήσεις σε μία ουρά FIFO χωρίς να κάνει οτιδήποτε άλλο (NOOP=no operation).
- Είναι χρήσιμος για συσκευές που δεν χρησιμοποιούν δίσκο (όπως στικάκια μνήμης ή εξιδεικευμένες συσκευές που έχουν δικό τους μηχανισμό χρονοδρομολόγησης).

93

## Ο χρονοδρομολογητής CFQ

- Ο CFQ (Completely Fair Queuing I/O Scheduler), αναπτύχθηκε το 2003 και είναι αυτός που χρησιμοποιείται κατά κανόνα στο Linux.
- Εγγυάται δίκαιη κατανομή του εύρους ζώνης E/E του δίσκου για όλες τις διεργασίες.
- Κάθε διεργασία έχει τη δική της ουρά για αιτήσεις E/E και κάθε ουρά έχει το δικό της κβάντο.
- Οι αιτήσεις σε αυτές τις ουρές εξυπηρετούνται εκ περιτροπής.
- Όταν ο χρονοδρομολογητής εξυπηρετεί μία ουρά και δεν υπάρχουν σε αυτήν άλλες αιτήσεις, περιμένει για κάποιο προκαθορισμένο χρονικό διάστημα μήπως εμφανιστούν και άλλες αιτήσεις και μετά εξυπηρετεί την επόμενη ουρά.
- Σε περίπτωση που υπάρχουν νέες αιτήσεις όταν περιμένει, αυτές θα εξυπηρετηθούν και με αυτόν τον τρόπο βελτιστοποιείται η λειτουργία του χρονοδρομολογητή.

94

## Οι χρονοδρομολογητές του Linux

Figure 11.14 Linux I/O Schedulers

95

## Χρήση μνήμης cache για σελίδες

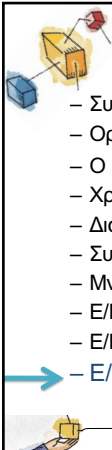
- Μέχρι το Linux 2.2, χρησιμοποιούσαν διαφορετικές μνήμες cache για τη μεταφορά σελίδων, πρόσβαση σε αρχεία και εκτέλεση εντολών E/E.
- Από το Linux 2.3 και μετά, όλες οι μεταφορές δεδομένων μεταξύ δίσκου και μνήμης γίνεται με χρήση μίας μνήμης cache.
- Αυτό επιτρέπει την πιο γρήγορη αναφορά σε σελίδες αλλά και την πιο οργανωμένη μεταφορά τους στο δίσκο.

96



## Περιεχόμενα

- Συσσκευές Ε/Ε.
- Οργάνωση των λειτουργιών Ε/Ε.
- Ο ρόλος του Λ.Σ. στη διαχείριση των συσκευών Ε/Ε.
- Χρήση προσωρινής μνήμης στις λειτουργίες Ε/Ε.
- Διαχείριση δίσκου.
- Συστήματα Raid.
- Μνήμη cache στο δίσκο.
- Ε/Ε στο UNIX SVR4.
- Ε/Ε στο LINUX.
- Ε/Ε στα Windows.



97

## Διαχειριστής εντολών Ε/Ε στα Windows

- Υπεύθυνος για όλες τις εντολές Ε/Ε.
- Παρέχει ένα ομοιόμοφο τρόπο πρόσβασης σε όλες τις συσκευές του συστήματος.
- Λειτουργεί στενά με 4 μέρη του πυρήνα, όπως φαίνεται παράπλευρα.

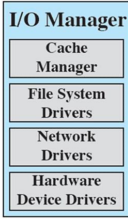


Figure 11.15 Windows I/O Manager

98

## Ασύγχρονη και σύγχρονη εκτέλεση εντολών Ε/Ε

- Τα Windows προσφέρουν δύο είδη εκτέλεσης των εντολών Ε/Ε:
  - Ασύγχρονη.
  - Σύγχρονη.
- Προτιμάται η πρώτη επιλογή όπου είναι δυνατόν γιατί είναι πιο αποδοτική.

99

## Σύστημα RAID

- Τα Windows χρησιμοποιούν την τεχνική RAID για οποιοδήποτε αριθμό δίσκων
- Υποστηρίζονται οι τεχνικές RAID 0, 1 και RAID 5.
- Στην περίπτωση του RAID 1 οι δύο δίσκοι μπορεί να κάνουν χρήση του ίδιου ελεγκτή δίσκων ή διαφορετικού ο καθένας από αυτούς.

100

## Σκιώδη αντίγραφα

- Η τεχνική των σκιωδών αντιγράφων (shadow copies) χρησιμοποιείται για δεδομένα πριν αυτά σβηστούν.
- Αν ο χρήστης σβήσει κατά λάθος ένα αρχείο, το σύστημα του παρέχει τη δυνατότητα να ανακτήσει το αρχείο σε μία παλαιότερη μορφή του.
- Η τεχνική αυτή είναι επίσης χρήσιμη στην αρχειοθέτηση των δεδομένων σε ένα δίσκο.

101


## ΕΠΛ222: Λειτουργικά Συστήματα

(μετάφραση στα ελληνικά των διαφανειών του βιβλίου Operating Systems: Internals and Design Principles, 9/E, William Stallings)

### Τέλος Ενότητας 9

Οι διαφάνειες αυτές έχουν συμπληρωματικό και επεξηγηματικό χαρακτήρα και σε καμία περίπτωση δεν υποκαθιστούν το βιβλίο

Γιώργος Α. Παπαδόπουλος  
Τμήμα Πληροφορικής  
Πανεπιστήμιο Κύπρου



102