

ΕΠΛ222: Λειτουργικά Συστήματα  
(μετάφραση στα ελληνικά των διαφανειών του βιβλίου Operating Systems: Internals and Design Principles, 9E, William Stallings)

## Ενότητα 10 (Κεφάλαιο 12) Διαχείριση και Συστήματα Αρχείων

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

Γιώργος Α. Παπαδόπουλος  
 Τμήμα Πληροφορικής  
 Πανεπιστήμιο Κύπρου

Operating Systems  
 Internals and Design Principles  
 9th Edition  
 © Pearson

1

## Περιεχόμενα

→ **Εισαγωγικές έννοιες.**

- Οργάνωση αρχείων και πρόσβαση σε αυτά.
- Δένδρα-B.
- Κατάλογοι αρχείων.
- Διαμοίραση αρχείων.
- Ομαδοποίηση εγγραφών.
- Διαχείριση του χώρου του δίσκου.
- Διαχείριση αρχείων στο Unix.
- Διαχείριση αρχείων στο Linux.
- Διαχείριση αρχείων στα Windows.
- Διαχείριση αρχείων στο Android.

2

## Αρχεία

- Τα αρχεία (files) είναι το κεντρικό στοιχείο των περισσότερων εφαρμογών.
- Κατ' επέκταση το σύστημα διαχείρισης αρχείων είναι για τον χρήστη ένα από τα πιο σημαντικά μέρη ενός Λ.Σ. και παρέχει σε αυτόν τη δυνατότητα να χρησιμοποιεί την περιφερειακή μνήμη.
- Τα αρχεία έχουν τις εξής επιθυμητές ιδιότητες:
  - Διάρκεια ζωής που υπερβαίνει το χρόνο εκτέλεσης μίας εφαρμογής ή τη διάρκεια που ο χρήστης χρησιμοποιεί το σύστημα.
  - Δυνατότητα να χρησιμοποιούνται από και να διαμοιράζονται μεταξύ των διεργασιών.
  - Εσωτερική δομή που εξυπηρετεί τις ανάγκες των εφαρμογών που κάνουν χρήση τους και ιεραρχική σχέση μεταξύ τους.

3

## Διαχείριση αρχείων

- Επιτυγχάνεται με την ύπαρξη μίας ομάδας από βοηθητικά προγράμματα τα οποία εκτελούνται σε προνομιακή κατάσταση και ασχολούνται μόνο με την περιφερειακή μνήμη.

4

## Τυπικές πράξεις με αρχεία

- Η διαχείριση αρχείων συνεπάγεται εκτός της αποθήκευσης δεδομένων οργανωμένα σε αρχεία και την υποστήριξη μερικών βασικών λειτουργιών με αυτά:
  - Δημιουργία ενός αρχείου.
  - Διαγραφή ενός αρχείου.
  - Άνοιγμα ενός αρχείου.
  - Κλείσιμο ενός αρχείου.
  - Διάβαση από ένα αρχείο.
  - Γράψιμο σε ένα αρχείο.

5

## Δομή και οργάνωση αρχείων

- Τέσσερις βασικές έννοιες σχετίζονται με τη δομή και οργάνωση αρχείων:
  - Πεδίο (field).
  - Εγγραφή (record).
  - Αρχείο.
  - Βάση δεδομένων (database).

6

## Πεδίο και εγγραφή

- **Πεδίο:**
  - Είναι το βασικό στοιχείο των δεδομένων ενός αρχείου.
  - Περιέχει μία αυτούσια τιμή.
  - Χαρακτηρίζεται από το μήκος του και είδος δεδομένων από τα οποία αποτελείται (π.χ. αριθμός, γράμματα).
  - Ανάλογα με τη δομή του αρχείου στο οποίο ανήκουν, τα πεδία μπορεί να είναι σταθερού ή μεταβλητού μεγέθους.
- **Εγγραφή:**
  - Μία ομάδα από πεδία τα οποία συνδέονται μεταξύ τους με κάποια λογική σχέση.
  - Μία εγγραφή συνήθως αντιμετωπίζεται από τις εφαρμογές ως μονάδα πληροφορίας.

7

## Αρχείο και βάση δεδομένων

- **Αρχείο:**
  - Είναι μία συλλογή από εγγραφές που σχετίζονται μεταξύ τους με κάποια λογική σχέση.
  - Αντιμετωπίζεται ως μία οντότητα από τους χρήστες και τις εφαρμογές και γίνεται αναφορά σε αυτήν με το όνομά της.
  - Μπορεί να δημιουργηθεί και να διαγραφεί.
  - Η πρόσβαση στα περιεχόμενά του ελέγχεται με επιβολή περιορισμών.
- **Βάση δεδομένων:**
  - Είναι μία συλλογή από αρχεία που σχετίζονται μεταξύ τους με κάποια λογική σχέση.
  - Κατ' επέκταση, τα δεδομένα των αρχείων που αποτελούν μία βάση δεδομένων, επίσης σχετίζονται μεταξύ τους.
  - Υπάρχει συνήθως ένα ξεχωριστό σύστημα διαχείρισης βάσεων δεδομένων (database management system) το οποίο είναι ανεξάρτητο του Λ.Σ. και το οποίο πιθανόν να χρησιμοποιεί το σύστημα διαχείρισης αρχείων του Λ.Σ.

8

## Σύστημα διαχείρισης αρχείων

- Είναι το μέρος εκείνου του λογισμικού του συστήματος το οποίο παρέχει τις υπηρεσίες που χρειάζονται οι χρήστες και οι εφαρμογές για τη διαχείριση των αρχείων του συστήματος.
- Συνήθως, ο μοναδικός τρόπος για να έχουν οι χρήστες και οι εφαρμογές πρόσβαση στα αρχεία, είναι μέσω ενός τέτοιου συστήματος.
- Απαλλάσσει τους χρήστες και τους προγραμματιστές από την ανάγκη να δημιουργήσουν ειδικό λογισμικό σε κάθε εφαρμογή για πρόσβαση στα αρχεία της εφαρμογής και παρέχει ένα ομοιόμορφο και καλώς ορισμένο τρόπο για τον έλεγχο των αρχείων.

9

## Στόχοι ενός συστήματος διαχείρισης αρχείων

- Να ικανοποιεί τις ανάγκες και απαιτήσεις του χρήστη στη διαχείριση των δεδομένων του.
- Να εγγυάται (όσο είναι αυτό δυνατόν) ότι τα δεδομένα ενός αρχείου είναι έγκυρα.
- Να βελτιστοποιεί την απόδοση και από το σημείο αναφοράς του χρήστη σε σχέση με το χρόνο απόκρισης και από το σημείο αναφοράς του συστήματος σε σχέση με τον ρυθμό απόδοσης.
- Να παρέχει υποστήριξη για E/E για μία σειρά από συσκευές αποθήκευσης δεδομένων.
- Να ελαχιστοποιεί ή εκμηδενίζει την πιθανότητα καταστροφής ή απώλειας πληροφοριών.
- Να παρέχει μία ομάδα από τυποποιημένες ρουτίνες για εκτέλεση εντολών E/E από τον χρήστη.
- Να παρέχει υποστήριξη για E/E σε πολλούς χρήστες, σε συστήματα υποστήριξης πολλών χρηστών.

10

## Τι πρέπει να μπορεί να κάνει ένας χρήστης με ένα γενικό σύστημα διαχείρισης αρχείων

1. Να δημιουργήσει ή να διαγράψει ένα αρχείο, να γράψει σε αυτό και να διαβάσει ή να τροποποιήσει τα περιεχόμενά του.
2. Να έχει ελεγχόμενη πρόσβαση στα αρχεία άλλων χρηστών.
3. Να ελέγχει τα δικαιώματα πρόσβασης στα αρχεία του που έχουν οι υπόλοιποι χρήστες.
4. Να μπορεί να δομεί τα αρχεία του με τρόπο που να εξυπηρετεί τις ανάγκες επίλυσης ενός προβλήματος.
5. Να μεταφέρει δεδομένα μεταξύ αρχείων.
6. Να μπορεί να δημιουργεί εφεδρικά αντίγραφα των αρχείων του και να ανακτή τις πληροφορίες από αυτά όταν υπάρχει απώλεια δεδομένων.
7. Να έχει πρόσβαση στα αρχεία του με βάση συμβολικά (και με λογική σημασία σε αυτόν) ονόματα και όχι με αριθμητικούς προσδιοριστές.

11

## Τυπική αρχιτεκτονική ενός συστήματος διαχείρισης αρχείων

```

    graph TD
      UP[User Program] --> LIO[Logical I/O]
      LIO --> BISO[Basic I/O Supervisor]
      BISO --> BFS[Basic File System]
      BFS --> DDD[Disk Device Driver]
      BFS --> TDD[Tape Device Driver]
  
```

The diagram illustrates the File System Software Architecture. At the top is the User Program, which interacts with the Logical I/O layer. Below Logical I/O is the Basic I/O Supervisor, which then connects to the Basic File System. The Basic File System layer is split into two parts: the Disk Device Driver and the Tape Device Driver.

Figure 12.1 File System Software Architecture

12

## Οδηγοί συσκευών

- Οι οδηγοί συσκευών (device drivers) λειτουργούν στο χαμηλότερο επίπεδο και επικοινωνούν με τις συσκευές ή τους ελεγκτές ή τα κανάλια αυτών των συσκευών.
- Είναι υπεύθυνοι για την έναρξη των εντολών E/E σε μία συσκευή και την επεξεργασία της ολοκλήρωσης αυτών των εντολών.
- Στην περίπτωση των αρχείων, οι οδηγοί συσκευών ελέγχουν τους δίσκους και τους αναγνώστες ταινιών.
- Θεωρούνται ότι είναι μέρος του Λ.Σ.

13

## Βασικό σύστημα αρχείων

- Ονομάζεται επίσης και το φυσικό επίπεδο E/E.
- Είναι το βασικό μέσο επικοινωνίας μεταξύ του συστήματος και του περιβάλλοντος εκτός του συστήματος.
- Ασχολείται με τη μεταφορά ομάδων δεδομένων μεταξύ σκληρών δίσκων και αναγνώστων ταινιών από τη μία πλευρά και του συστήματος από την άλλη:
  - Είναι υπεύθυνο για την τοποθέτηση αυτών των δεδομένων στην περιφερειακή συσκευή ή σε buffers στην κύρια μνήμη.
  - Δεν κατανοεί τη σημασιολογία αυτών των δεδομένων ή τη δομή των αρχείων που εμπλέκονται στη μεταφορά.
- Θεωρείται ότι είναι μέρος του Λ.Σ.

14

## Επιτηρητής βασικού συστήματος E/E

- Είναι υπεύθυνος για την έναρξη και τερματισμό των εντολών E/E σε ένα αρχείο.
- Διαχειρίζεται δομές ελέγχου που ασχολούνται με:
  - E/E σε συσκευές.
  - Χρονοδρομολόγηση εντολών E/E.
  - Κατάσταση αρχείων.
- Επιλέγει τη συσκευή η οποία εμπλέκεται σε μία εντολή E/E σε κάποιο αρχείο, με βάση τα χαρακτηριστικά του αρχείου.
- Χρονοδρομολογεί την πρόσβαση σε δίσκους και αναγνώστες ταινιών για καλύτερη απόδοση του συστήματος.
  - Σε αυτό το σημείο παρέχονται buffers E/E και περιφερειακή μνήμη.
- Θεωρείται μέρος του Λ.Σ.

15

## Λογικό σύστημα E/E

- Σε σύγκριση με το βασικό σύστημα αρχείων το οποίο ασχολείται με ομάδες δεδομένων, το λογικό σύστημα E/E ασχολείται με εγγραφές και επιτρέπει την πρόσβαση σε αυτές από χρήστες και εφαρμογές.
- Παρέχει ένα γενικευμένο τρόπο εκτέλεσης εντολών E/E σε εγγραφές και διατηρεί βασικές πληροφορίες για κάθε αρχείο.

16

## Τρόποι πρόσβασης στα περιεχόμενα ενός αρχείου

- Οι τρόποι πρόσβασης στα περιεχόμενα ενός αρχείου (access methods) αποτελούν το επίπεδο εκείνο ενός συστήματος διαχείρισης αρχείων που είναι πιο κοντά στον χρήστη.
- Παρέχει έναν τυποποιημένο τρόπο επικοινωνίας μεταξύ των εφαρμογών από τη μία πλευρά και των συστημάτων αρχείων και συσκευών στα οποία βρίσκονται τα δεδομένα από την άλλη.
- Οι διαφορετικοί τρόποι πρόσβασης στα περιεχόμενα ενός αρχείου απεικονίζουν τις διαφορετικές δομές που μπορεί να έχει ένα αρχείο και τους τρόπους πρόσβασης στα περιεχόμενά του.

17

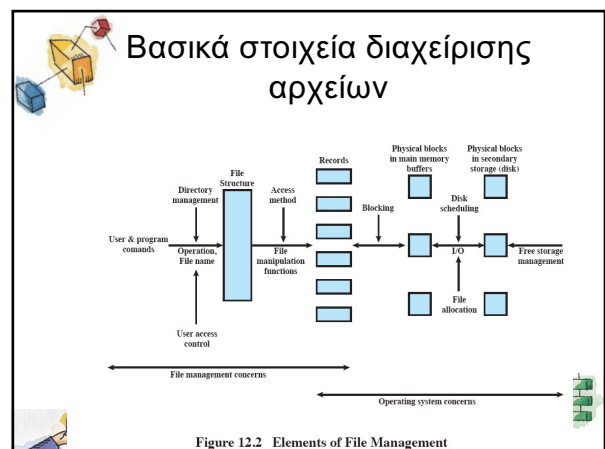


Figure 12.2 Elements of File Management

18

## Περιεχόμενα

- Εισαγωγικές έννοιες.
- - Οργάνωση αρχείων και πρόσβαση σε αυτά.
- Δένδρα-B.
- Κατάλογοι αρχείων.
- Διαμοίραση αρχείων.
- Ομαδοποίηση εγγραφών.
- Διαχείριση του χώρου του δίσκου.
- Διαχείριση αρχείων στο Unix.
- Διαχείριση αρχείων στο Linux.
- Διαχείριση αρχείων στα Windows.
- Διαχείριση αρχείων στο Android.

19

## Οργάνωση αρχείων

- Αναφέρεται στη λογική δομή των εγγραφών.
  - Η φυσική δομή εξετάζεται αργότερα.
- Αυτή καθορίζεται από τον τρόπο που γίνεται πρόσβαση στα περιεχόμενα ενός αρχείου.

20

## Κριτήρια σχετιζόμενα με την οργάνωση των αρχείων

- Μερικά σημαντικά κριτήρια είναι τα ακόλουθα:
  - Σύντομος χρόνος προσπέλασης στα περιεχόμενά τους.
  - Ευκολία ενημέρωσης.
  - Μικρό κόστος σε αποθήκευση.
  - Εύκολη συντήρηση.
  - Αξιοπιστία.
- Η σχετική προτεραιότητα αυτών των κριτηρίων εξαρτάται από το είδος των εφαρμογών που τα χρησιμοποιούν και τον τρόπο χρήσης τους, π.χ.:
  - Για σειριακή επεξεργασία των περισσότερων εγγραφών, γρήγορη πρόσβαση σε μία εγγραφή έχει μικρή αξία.
  - Τα περιεχόμενα σε ένα CD-ROM δεν τροποποιούνται, επομένως η ευκολία ενημέρωσης δεν παίζει ρόλο.
- Μερικές φορές τα κριτήρια αυτά αντικρούουν το ένα το άλλο, π.χ. η δημιουργία πολλαπλών αντιγράφων ενός αρχείου αυξάνει την ανάγκη σε μνήμη αλλά ελαττώνει το χρόνο πρόσβασης σε αυτό.

21

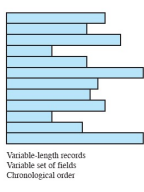
## Είδη οργάνωσης αρχείων

- Υπάρχουν πολλά, αλλά συνήθως είναι παραλλαγές των ακόλουθων:
  - Σωρός (pile).
  - Σειριακό (sequential).
  - Σειριακό με δείκτες (indexed sequential).
  - Ευρετηριασμένο (indexed).
  - Άμεσης προσπέλασης (direct/hashed access).

22

## Σωρός

- Τα δεδομένα συλλέγονται με τη χρονική σειρά που δημιουργούνται και δεν έχουν δομή.
- Ο σκοπός είναι απλά η συλλογή τους για κατοπινή επεξεργασία.
- Οι εγγραφές έχουν διαφορετικά πεδία και δομή μεταξύ τους.
- Η πρόσβαση στα περιεχόμενα γίνεται σειριακά, η μία εγγραφή μετά την άλλη.

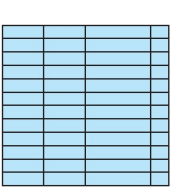


Variable-length records  
Variable set of fields  
Chronological order  
(a) Pile File

23

## Σειριακό

- Όλες οι εγγραφές έχουν ομοιόμορφη και σταθερή δομή (ίδιος αριθμός, σειρά και δομή πεδίων).
- Τα ονόματα και μέγεθος των πεδίων αποτελούν ιδιοχαρακτηριστικά του αρχείου.
- Υπάρχει πάντα ένα πεδίο κλειδί το οποίο έχει πάντα μοναδική τιμή για κάθε εγγραφή και χρησιμοποιείται για να ορίζει τη σειρά των εγγραφών στο αρχείο (αλφαβητικά ή αριθμητικά, ανάλογα με την περίπτωση).

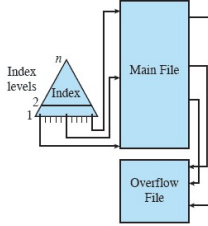


Fixed-length records  
Fixed set of fields in fixed order  
Sequential order based on key field  
(b) Sequential File

24

## Σειριακό με δείκτες

- Ισχύουν και εδώ τα βασικά χαρακτηριστικά του σειριακού αρχείου.
- Επιπλέον όμως υπάρχει ένας δείκτης για υποστήριξη τυχαίας προσπέλασης και ένα αρχείο υπερχείλισης.
- Ο δείκτης επιτρέπει τη γρήγορη πρόσβαση σε οποιαδήποτε εγγραφή.
- Στο αρχείο υπερχείλισης αποθηκεύονται προσωρινά οι νέες εγγραφές και περιοδικά τα δύο αρχεία συγχωνεύονται σε ένα.

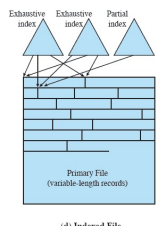


(c) Indexed Sequential File

25

## Ευρετηριασμένο

- Οι εγγραφές δεν οργανώνονται σειριακά και δεν υπάρχει ένα μοναδικό κλειδί αλλά πολλά για πολλαπλά (πιθανόν όλα) τα πεδία των εγγραφών.
- Η πρόσβαση στις εγγραφές γίνεται μόνο μέσω των κλειδίων.
- Οι εγγραφές μπορεί να είναι οποιοδήποτε στο αρχείο, αρκεί να μπορεί να γίνεται πρόσβαση σε αυτό μέσω τουλάχιστον ενός δείκτη.
- Οι εγγραφές μπορεί να έχουν μεταβλητό μήκος.
- Όταν προστίθεται μια καινούργια εγγραφή πρέπει να ανανεωθούν ανάλογα και οι δείκτες του αρχείου.
- Υπάρχουν δύο είδη δεικτών:
  - Πλήρης, που καλύπτει όλες τις εγγραφές και ο ίδιος οργανώνεται ως ένα σειριακό αρχείο για πιο εύκολη αναζήτηση.
  - Μερικός, που καλύπτει μόνο τις εγγραφές εκείνες των οποίων κάποια πεδία έχουν ενδιαφέρον.



(d) Indexed File

26

## Άμεσης προσπέλασης

- Εκμεταλλεύεται τη δυνατότητα των δίσκων να υποστηρίζουν πρόσβαση σε οποιοδήποτε μπλοκ μνήμης για το οποίο είναι γνωστή η διεύθυνσή του.
- Υπάρχει πάλι ένα κλειδί για κάθε εγγραφή.
- Το κλειδί αυτό δείχνει στη φυσική διεύθυνση της εγγραφής στο δίσκο.
- Αυτό επιτυγχάνεται με τη χρήση μίας συνάρτησης κωδικοτεμαχισμού (hashing).
- Η τεχνική αυτή είναι χρήσιμη σε αρχεία των οποίων οι εγγραφές έχουν σταθερό μήκος και χρειάζεται πολύ γρήγορη πρόσβαση σε μεμονωμένες εγγραφές.

27

## Στοιχεία απόδοσης

Table 12.1 Grades of Performance for Five Basic File Organizations (WIED87)

File Method	Space Attributes		Update Record Size		Retrieval		
	Variable	Fixed	Equal	Greater	Single record	Subset	Exhaustive
File	A	B	A	E	E	D	B
Sequential	F	A	D	F	F	D	A
Indexed sequential	F	B	B	D	B	D	B
Indexed	B	C	C	C	A	B	D
Hashed	F	B	B	F	B	F	E

A = Excellent, well suited to this purpose  $\approx O(1)$   
 B = Good  $\approx O(n \times r)$   
 C = Adequate  $\approx O(r \log n)$   
 D = Requires some extra effort  $\approx O(n)$   
 E = Possible with extreme effort  $\approx O(r \times n)$   
 F = Not reasonable for this purpose  $\approx O(r^2)$

where  
 r = size of the result  
 o = number of records that overflow  
 n = number of records in file

28

## Περιεχόμενα

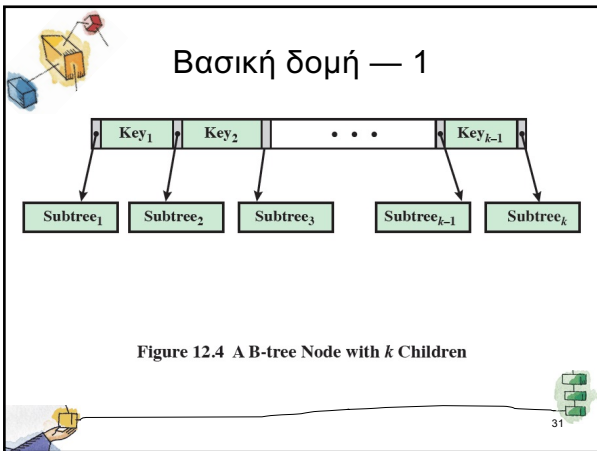
- Εισαγωγικές έννοιες.
- Οργάνωση αρχείων και πρόσβαση σε αυτά.
- **– Δένδρα-B.**
- Κατάλογοι αρχείων.
- Διαμοίραση αρχείων.
- Ομαδοποίηση εγγραφών.
- Διαχείριση του χώρου του δίσκου.
- Διαχείριση αρχείων στο Unix.
- Διαχείριση αρχείων στο Linux.
- Διαχείριση αρχείων στα Windows.
- Διαχείριση αρχείων στο Android.

29

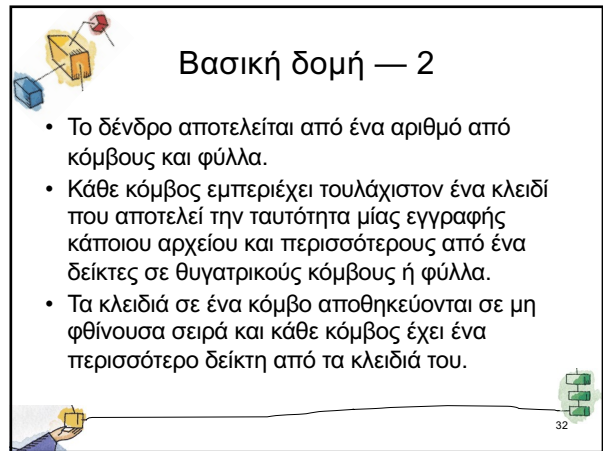
## Βασικά χαρακτηριστικά

- Είναι μία ισόβαρη δομή δένδρου όπου όλα τα κλαδιά έχουν ίσο μήκος.
- Αποτελεί μία τυπική μέθοδο οργάνωσης δεικτών για μεγάλα αρχεία και βάσεις δεδομένων.
- Χρησιμοποιείται ευρέως από μοντέρνα λειτουργικά συστήματα όπως τα OS X, Windows και Linux.
- Επιτρέπει τη γρήγορη πρόσθεση, αφαίρεση και εύρεση δεικτών.

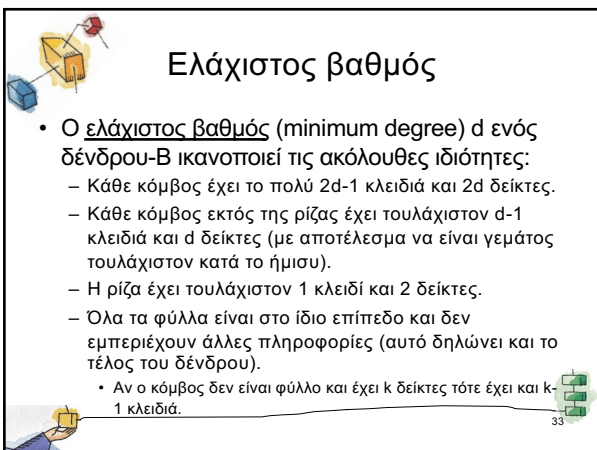
30



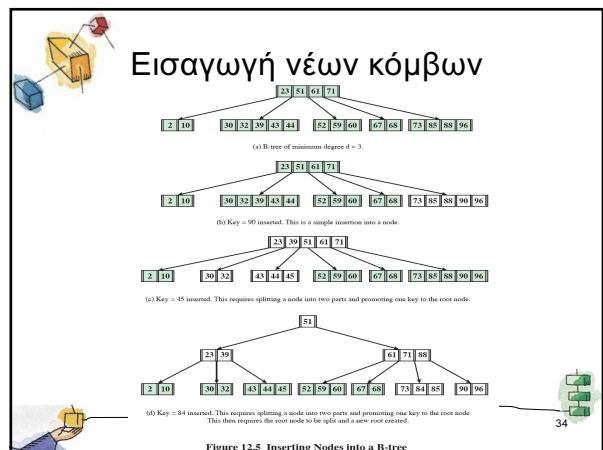
31



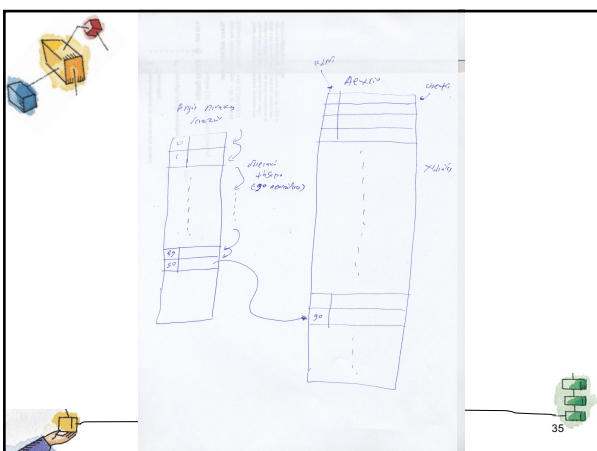
32



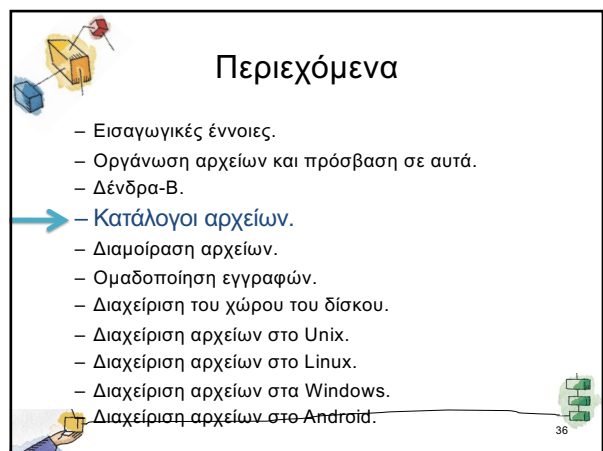
33



34



35



36

## Τι είναι ο κατάλογος και τι περιέχει

- Ένας κατάλογος αρχείων (directory) περιέχει πληροφορίες για αρχεία, όπως:
  - Ιδιοχαρακτηριστικά (attributes).
  - Χώρο στην περιφερειακή μνήμη που είναι αποθηκευμένα.
  - Ιδιοκτήτη.
- Ο ίδιος ο κατάλογος είναι ένα αρχείο το οποίο ανήκει στο Λ.Σ. και στο οποίο έχουν πρόσβαση οι σχετικές ρουτίνες διαχείρισης αρχείων του Λ.Σ.
- Παρέχει απεικόνιση μεταξύ των ονομάτων των αρχείων και καθαυτών των αρχείων.
- Αν και κάποιες από τις πληροφορίες του καταλόγου είναι διαθέσιμες στους χρήστες, αυτό επιτυγχάνεται έμμεσα μέσω της κλήσης σχετικών ρουτινών του συστήματος.

37

## Περιεχόμενα ενός καταλόγου

Basic Information	
<b>File Name</b>	Name as chosen by creator (user or program). Must be unique within a specific directory.
<b>File Type</b>	For example text, binary, load module, etc.
<b>File Organization</b>	For systems that support different organizations.
Address Information	
<b>Volume</b>	Indicates device on which file is stored.
<b>Starting Address</b>	Starting physical address on secondary storage (e.g., cylinder, track, and block number on disk).
<b>Size Used</b>	Current size of the file in bytes, words, or blocks.
<b>Size Allocated</b>	The maximum size of the file.
Access Control Information	
<b>Owner</b>	User who is assigned control of this file. The owner may be able to grant/deny access to other users and to change these privileges.
<b>Access Information</b>	A simple version of this element would include the user's name and password for each authorized user.
<b>Permitted Actions</b>	Controls reading, writing, executing, transmitting over a network.
Usage Information	
<b>Date Created</b>	When file was first placed in directory.
<b>Identity of Creator</b>	Usually but not necessarily the current owner.
<b>Date Last Read Access</b>	Date of the last time a record was read.
<b>Identity of Last Reader</b>	User who did the reading.
<b>Date Last Modified</b>	Date of the last update, insertion, or deletion.
<b>Identity of Last Modifier</b>	User who did the modifying.
<b>Date of Last Backup</b>	Date of the last time the file was backed up on another storage medium.
<b>Current Usage</b>	Information about current activity on the file, such as process or processes that have the file open, whether it is locked by a process, and whether the file has been updated in main memory but not yet on disk.

38

## Δομή ενός καταλόγου

- Ο τρόπος αποθήκευσης των περιεχομένων ενός καταλόγου διαφέρει από σύστημα σε σύστημα.
- Κάποιες από τις πληροφορίες μπορεί να είναι αποθηκευμένες σε μία εγγραφή που παίζει το ρόλο επικεφαλίδας και σχετίζεται με το κάθε αρχείο του καταλόγου.
  - Μειώνει το ποσό μνήμης που χρειάζεται ο κατάλογος.
  - Επιτρέπει την αποθήκευση όλου ή μεγάλου μέρους του στην κύρια μνήμη με επακόλουθο την αύξηση της ταχύτητας προσπέλασης στα περιεχόμενά του.
- Ο πιο απλός τρόπος είναι μία λίστα από στοιχεία, ένα στοιχείο για κάθε αρχείο του καταλόγου.
- Η δομή αυτή μπορεί να υλοποιηθεί ως ένα απλό σειριακό αρχείο, με το όνομα του κάθε αρχείου του καταλόγου να παίζει το ρόλο του κλειδιού στο σειριακό αρχείο.
- Αναγκάζει τους χρήστες να βεβαιώνονται ότι το ίδιο όνομα δεν χρησιμοποιείται για περισσότερα από ένα αρχεία.

39

## Λειτουργίες σχετιζόμενες με έναν κατάλογο

- Ένα σύστημα καταλόγων πρέπει να υποστηρίζει κάποιες βασικές λειτουργίες, όπως:
  - Αναζήτηση ενός αρχείου σε έναν κατάλογο, όταν ένας χρήστης ή μία εφαρμογή κάνει αναφορά σε αυτό το αρχείο.
  - Δημιουργία καινούργιων αρχείων.
  - Διαγραφή υπάρχοντων αρχείων.
  - Μερική ή ολική παρουσίαση των περιεχομένων ενός καταλόγου (π.χ. ονομάτων αρχείων και πιθανόν κάποιων από τα ιδιοχαρακτηριστικά των αρχείων).
  - Ενημέρωση των περιεχομένων του καταλόγου (π.χ. αν αλλάξουν κάποια ιδιοχαρακτηριστικά ενός αρχείου).

40

## Σύστημα οργάνωσης καταλόγων δύο επιπέδων

- Σε αυτό το σύστημα υπάρχει ένας κατάλογος για κάθε χρήστη και ένας πρωτεύον κατάλογος (master directory).
  - Ο πρωτεύον κατάλογος έχει ένα στοιχείο για κάθε χρήστη.
  - Παρέχει τη διεύθυνση και πληροφορίες ελέγχου πρόσβασης στα περιεχόμενα του καταλόγου του χρήστη.
- Ο κατάλογος του χρήστη είναι μία απλή λίστα των αρχείων του χρήστη.
- Επιτρέπει την επιβολή περιορισμών στην πρόσβαση σε αρχεία.
- Όμως δεν επιτρέπει στους χρήστες να έχουν ιεραρχική δομή σε ομάδες αρχείων.

41

## Ιεραρχική ή δενδροειδής οργάνωση καταλόγων

- Υπάρχει ένας πρωτεύον κατάλογος, κάτω από τον οποίον υπάρχει ένας αριθμός από καταλόγους χρηστών.
- Κάθε κατάλογος χρήστη με τη σειρά του μπορεί να εμπεριέχει άλλους υποκαταλόγους ή/και αρχεία.
- Κάθε κατάλογος αποθηκεύεται ως ένα σειριακό αρχείο.
- Αν ο κατάλογος είναι μεγάλος και κατ'επέκταση η αναζήτηση πληροφοριών σε αυτόν είναι χρονοβόρα με τη σειριακή αναζήτηση, χρησιμοποιείται εναλλακτικά η μέθοδος του κωδικοτεμαχισμού.

Figure 12.6 Tree-Structured Directory

42

## Ονοματολογία

- Οι χρήστες πρέπει να έχουν τη δυνατότητα να αναφέρονται στα αρχεία ενός συστήματος με χρήση συμβολικών ονομάτων.
- Κάθε αρχείο πρέπει να έχει ένα μοναδικό όνομα έτσι ώστε η χρήση ενός ονόματος να είναι σαφές σε ποιο αρχείο αναφέρεται.
- Όμως είναι πολύ δύσκολο για τον χρήστη να παρέχει πάντα μοναδικά ονόματα, ειδικά αν το σύστημα χρησιμοποιείται και από άλλους χρήστες.
- Η χρήση της δενδροειδούς οργάνωσης επιτρέπει στους χρήστες να μην χρειάζονται μοναδικά ονόματα, νοούμενου ότι το ίδιο όνομα δεν χρησιμοποιείται για περισσότερα από ένα αρχεία στον ίδιο κατάλογο.
- Αρχεία που βρίσκονται σε διαφορετικούς καταλόγους μπορούν να έχουν το ίδιο όνομα, διότι η σειρά των καταλόγων που πρέπει να διατρηθούν για να βρεθεί το κάθε αρχείο είναι μοναδική.

43

## Παράδειγμα καταλόγου δενδροειδούς οργάνωσης

Figure 12.7 Example of Tree-Structured Directory

44

## Τρέχων κατάλογος

- Η εύρεση κάποιου αρχείου γίνεται μέσω του ονόματος διαδρομής (path name) που είναι η ακολουθία των ονομάτων των καταλόγων από το σημείο αναφοράς μέχρι τον κατάλογο όπου βρίσκεται το αρχείο.
- Αν το σημείο αναφοράς είναι ο πρωτεύων κατάλογος, τότε το όνομα διαδρομής είναι απόλυτο (absolute path name).
- Θα ήταν άβολο για ένα χρήστη, κάθε φορά που θα έπρεπε να αναφερθεί στο όνομα ενός αρχείου, να πρέπει να συμπεριλαμβάνει το απόλυτο όνομα διαδρομής.
- Γι' αυτό, ο κάθε χρήστης (ή διεργασία) σχετίζεται με ένα τρέχοντα κατάλογο, αναφορά σε αρχεία γίνεται με βάση τον κατάλογο αυτόν και σε αυτήν την περίπτωση το όνομα διαδρομής είναι σχετικό (relative path name).
- Επομένως, η αναφορά στα αρχεία του τρέχοντος καταλόγου γίνεται με απλή χρήση του ονόματος του κάθε αρχείου.

45

## Περιεχόμενα

- Εισαγωγικές έννοιες.
- Οργάνωση αρχείων και πρόσβαση σε αυτά.
- Δένδρα-B.
- Κατάλογοι αρχείων.
- **– Διαμοίραση αρχείων.**
- Ομαδοποίηση εγγραφών.
- Διαχείριση του χώρου του δίσκου.
- Διαχείριση αρχείων στο Unix.
- Διαχείριση αρχείων στο Linux.
- Διαχείριση αρχείων στα Windows.
- Διαχείριση αρχείων στο Android.

46

## Διαμοίραση αρχείων

- Σε ένα σύστημα υποστήριξης πολλαπλών χρηστών, σχεδόν πάντα υπάρχει η ανάγκη να διαμοιράζονται κάποια αρχεία μεταξύ ενός αριθμού χρηστών.
- Εδώ πρέπει να εξετασθούν δύο θέματα:
  - Δικαιώματα πρόσβασης (access rights) για τον κάθε χρήστη σε διαμοιραζόμενα αρχεία.
  - Διαχείριση ταυτόχρονων προσβάσεων στα αρχεία.


47

## Δικαιώματα πρόσβασης

- Ένα σύστημα διαχείρισης αρχείων πρέπει να παρέχει στους χρήστες ένα αριθμό από επιλογές στον έλεγχο πρόσβασης στα περιεχόμενα των αρχείων τους.
- Συνήθως, μεμονωμένοι χρήστες ή ομάδες χρηστών έχουν συγκεκριμένα δικαιώματα πρόσβασης σε κάποιο αρχείο.
- Υπάρχει μία σειρά από διαφορετικά δικαιώματα που λειτουργεί ιεραρχικά, με την έννοια ότι η κατοχή ενός δικαιώματος υπονοεί κατοχή και σε όλα τα προηγούμενα.


48






## Είδη δικαιωμάτων — 1

- **Κανένα.**
  - Ο χρήστης πιθανόν δεν γνωρίζει καν την ύπαρξη του αρχείου.
- **Γνώση.**
  - Ο χρήστης μπορεί μόνο να διαπιστώσει την ύπαρξη του αρχείου και ποιος είναι ο ιδιοκτήτης του.
- **Εκτέλεση.**
  - Ο χρήστης μπορεί να φορτώσει και να εκτελέσει το αρχείο αλλά δεν μπορεί να κάνει οτιδήποτε άλλο, συμπεριλαμβανομένης και της αντιγραφής.
- **Διάβασμα.**
  - Ο χρήστης μπορεί να διαβάσει το αρχείο, όπως επίσης να το αντιγράψει και να το εκτελέσει.




49




## Είδη δικαιωμάτων — 2

- **Προσάρτηση.**
  - Ο χρήστης μπορεί να προσθέσει δεδομένα στο αρχείο αλλά δεν μπορεί να τροποποιήσει ή αφαιρέσει υπάρχοντα δεδομένα σε αυτό.
- **Ενημέρωση.**
  - Ο χρήστης μπορεί να προσθέσει, αφαιρέσει ή τροποποιήσει τα περιεχόμενα του αρχείου.
- **Αλλαγή δικαιωμάτων.**
  - Ο χρήστης μπορεί να τροποποιήσει τα δικαιώματα πρόσβασης στο αρχείο από άλλους χρήστες.
- **Διαγραφή.**
  - Ο χρήστης μπορεί να διαγράψει το αρχείο.




50




## Κατηγορίες χρηστών

- **Ιδιοκτήτης.**
  - Συνήθως (αλλά όχι κατ' ανάγκη) ο χρήστης που δημιούργησε το αρχείο και ο οποίος έχει όλα τα δικαιώματα πρόσβασης σε αυτό το αρχείο, τα οποία μπορεί εν μέρει ή συνολικά να τα δώσει και σε άλλους χρήστες.
- **Συγκεκριμένοι χρήστες.**
  - Αυτοί που έχουν συγκεκριμένα δικαιώματα πρόσβασης στο αρχείο.
- **Ομάδες χρηστών.**
  - Ομάδες από χρήστες που δεν αναφέρονται ξεχωριστά ο καθένας αλλά ως μέλη μίας ομάδας έχουν κάποια δικαιώματα.
- **Όλοι.**
  - Ο οποιοσδήποτε χρήστης που έχει δικαίωμα στα δημόσια αρχεία του συστήματος.




51

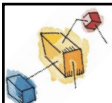


## Ταυτόχρονη πρόσβαση στα αρχεία

- Στην περίπτωση που περισσότεροι από ένας χρήστες προσπαθήσουν ταυτόχρονα να έχουν πρόσβαση σε ένα αρχείο, τότε δημιουργείται θέμα πρόσβασης σε κρίσιμο τμήμα.
- Σε τέτοια περίπτωση, ένας χρήστης μπορεί να κλειδώσει ολόκληρο το αρχείο.
- Μία πιο ευέλικτη προσέγγιση, είναι ο κάθε χρήστης να μπορεί να κλειδώνει την εγγραφή την οποία θέλει να ενημερώσει, οπότε πολλαπλοί χρήστες μπορούν ταυτόχρονα να έχουν πρόσβαση σε διαφορετικές εγγραφές του ίδιου αρχείου.
- Εδώ θα πρέπει να αντιμετωπίζονται τα γνωστά προβλήματα αμοιβαίου αποκλεισμού, αδιέξοδου, κλπ.




52

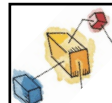


## Περιεχόμενα

- Εισαγωγικές έννοιες.
- Οργάνωση αρχείων και πρόσβαση σε αυτά.
- Δένδρα-B.
- Κατάλογοι αρχείων.
- Διαμοίραση αρχείων.
- **Ομαδοποίηση εγγραφών.**
  - Διαχείριση του χώρου του δίσκου.
  - Διαχείριση αρχείων στο Unix.
  - Διαχείριση αρχείων στο Linux.
  - Διαχείριση αρχείων στα Windows.
  - Διαχείριση αρχείων στο Android.




53



## Εγγραφές και μπλοκ

- Η λογική μονάδα πρόσβασης στα περιεχόμενα ενός αρχείου είναι η εγγραφή.
- Όμως, η μονάδα αποθήκευσης πληροφοριών στο δίσκο είναι το μπλοκ.
- Επομένως, για τη μεταφορά εγγραφών μεταξύ κύριας και περιφερειακής μνήμης, οι εγγραφές πρέπει να οργανωθούν ως μπλοκ.
- Ένα θέμα είναι το μέγεθος του μπλοκ και το αν αυτό θα πρέπει να είναι σταθερό ή μεταβλητό. Στην περίπτωση σταθερού μεγέθους σπαταλιέται χώρος αλλά η χρήση μεταβλητού μεγέθους είναι δύσκολο να υλοποιηθεί.
- Επειδή λοιπόν συνήθως χρησιμοποιείται η πρώτη μέθοδος το επόμενο ερώτημα που πρέπει να απαντηθεί είναι ποιο είναι το μέγεθος του μπλοκ.
  - Αν είναι σχετικά μεγάλο, τότε σε κάθε πράξη E/E μεταφέρονται περισσότερες εγγραφές που είναι ωφέλιμο όταν η προσπάθεια στο αρχείο είναι σειριακή.
  - Αν είναι σχετικά μικρό, τότε ωφελεί την τυχαία προσπάθεια όπου κατά κανόνα χρειάζεται η μεταφορά λίγων εγγραφών.



54

### Αποθήκευση εγγραφών σε μπλοκ

- Υπάρχουν τρεις βασικές τεχνικές:
  - Αποθήκευση σε μπλοκ εγγραφών σταθερού μεγέθους (fixed blocking).
  - Αποθήκευση σε μπλοκ εγγραφών μεταβλητού μεγέθους με δυνατότητα σπασίματος (spanned blocking).
  - Αποθήκευση σε μπλοκ εγγραφών μεταβλητού μεγέθους χωρίς δυνατότητα σπασίματος (unspanned blocking).

55

### Εγγραφές σταθερού μεγέθους

- Όλες οι εγγραφές έχουν σταθερό μέγεθος και ένας αριθμός από ολόκληρες εγγραφές αποθηκεύεται σε κάθε μπλοκ.
- Πιθανόν να υπάρχει αχρησιμοποίητος χώρος στο τέλος κάθε μπλοκ, που αποτελεί ένα είδος εσωτερικού κατακερματισμού.

56

### Αποθήκευση εγγραφών σταθερού μεγέθους

Fixed Blocking

- Data
- Gaps due to hardware design
- Waste due to block fit to track size
- Waste due to record fit to block size
- Waste due to block size constraint from fixed record size

57

### Εγγραφές μεταβλητού μεγέθους με δυνατότητα σπασίματος

- Οι εγγραφές δεν έχουν σταθερό μέγεθος και αποθηκεύονται σε μπλοκ χωρίς σπατάλη χώρου.
- Αν μία εγγραφή δεν χωράει ολόκληρη σε ένα μπλοκ, τότε σπάει σε δύο κομμάτια με το δεύτερο κομμάτι να αποθηκεύεται στο επόμενο στη σειρά μπλοκ.
- Η συσχέτιση των δύο κομματιών της εγγραφής γίνεται με τη χρήση ενός δείκτη από το τέλος του πρώτου κομματιού στο δεύτερο.

58

### Αποθήκευση εγγραφών μεταβλητού μεγέθους με σπάσιμο

Variable Blocking: Unspanned

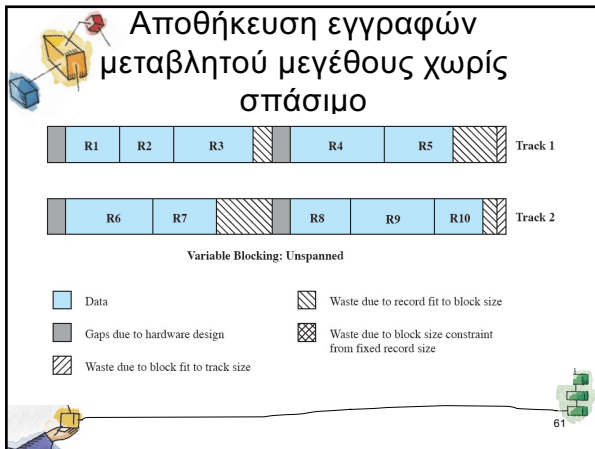
- Data
- Gaps due to hardware design
- Waste due to block fit to track size
- Waste due to record fit to block size
- Waste due to block size constraint from fixed record size

59

### Εγγραφές μεταβλητού μεγέθους χωρίς δυνατότητα σπασίματος

- Οι εγγραφές και πάλι δεν έχουν σταθερό μέγεθος αλλά η αποθήκευση μίας εγγραφής πρέπει να είναι ολόκληρη μέσα σε ένα μπλοκ.
- Κατ' επέκταση, υπάρχει και πάλι πρόβλημα εσωτερικού κατακερματισμού αν μία εγγραφή δεν χωράει ολόκληρη μέσα στο διαθέσιμο χώρο ενός μπλοκ.

60



61

### Περιεχόμενα

- Εισαγωγικές έννοιες.
- Οργάνωση αρχείων και πρόσβαση σε αυτά.
- Δένδρα-B.
- Κατάλογοι αρχείων.
- Διαμοίραση αρχείων.
- Ομαδοποίηση εγγραφών.
- Διαχείριση του χώρου του δίσκου.
- Διαχείριση αρχείων στο Unix.
- Διαχείριση αρχείων στο Linux.
- Διαχείριση αρχείων στα Windows.
- Διαχείριση αρχείων στο Android.

62

### Διαχείριση του χώρου του δίσκου

- Στην περιφερειακή μνήμη (δηλαδή τον δίσκο), ένα αρχείο αποτελείται από μία ομάδα μπλοκ.
- Το Λ.Σ. είναι υπεύθυνο για την κατανομή των μπλοκ του δίσκου σε αρχεία.
- Αυτό θέτει τα εξής ερωτήματα:
  - Πως διαμοιράζεται ο διαθέσιμος χώρος στα αρχεία.
  - Πως το σύστημα γνωρίζει που βρίσκεται ο διαθέσιμος χώρος για να τον διαμοιράσει στα αρχεία.

63

### Διαμοίραση του χώρου στα αρχεία

1. Όταν δημιουργηθεί ένα νέο αρχείο, του δίνεται από την αρχή όλος ο χώρος στον δίσκο που χρειάζεται;
2. Ο χώρος του δίσκου δίνεται στα αρχεία ως ένα ή περισσότερα συνεχόμενα κομμάτια, γνωστά και ως μερίδες (portions), όπου το μέγεθος μιας μερίδας μπορεί να είναι από ένα μπλοκ μέχρι ολόκληρο το αρχείο.
  - Ποιο είναι το ιδανικό μέγεθος μιας μερίδας;
3. Τι είδους δομές δεδομένων χρειάζονται για να γνωρίζει το σύστημα ποιες μερίδες έχουν δοθεί σε ποια αρχεία (π.χ. ο πίνακας FAT (File Allocation Table) σε κάποια Λ.Σ.);

64

### Κατανομή χώρου του δίσκου σε ένα αρχείο

- Η πρώτη μέθοδος (στατική κατανομή κατά τη δημιουργία του αρχείου) προϋποθέτει ότι γνωρίζουμε εκ των προτέρων το μέγιστο μέγεθος του αρχείου.
- Μερικές φορές αυτό είναι δυνατόν (π.χ. από τη μετάφραση ενός προγράμματος, όταν ένα αρχείο αντιγράφεται, κλπ.).
- Γενικώς όμως είναι αδύνατο να γίνει ακριβής πρόβλεψη του μέγιστου μεγέθους ενός αρχείου.
- Κατ' επέκταση, οι χρήστες και οι προγραμματιστές εφαρμογών τείνουν να υπερεκτιμούν το μέγεθος ενός αρχείου, κάτι το οποίο οδηγεί σε σπατάλη χώρου αποθήκευσης.
- Η δεύτερη μέθοδος της σταδιακής δυναμικής κατανομής θέτει το ερώτημα του πόσο χώρο δίνουμε αρχικά για την αποθήκευση του αρχείου και αν αυτός είναι ο ίδιος σε μέγεθος κάθε φορά.

65

### Παράγοντες που επηρεάζουν το μέγεθος της μερίδας

- Η επιλογή δύναται να επηρεάσει την απόδοση σε επίπεδο αρχείου ή την απόδοση σε επίπεδο συστήματος και σχετίζεται με τους ακόλουθους παράγοντες:
  - Η αποθήκευση δεδομένων σε συνεχόμενες θέσεις αυξάνει την απόδοση προσπέλασης στα δεδομένα ενός αρχείου.
  - Η ύπαρξη ενός μεγάλου αριθμού από μερίδες, αυξάνει το μέγεθος των πινάκων που χρειάζονται για να αποθηκεύσουν τις σχετικές πληροφορίες.
  - Η χρήση μερίδων σταθερού μεγέθους (π.χ. μπλοκ) απλοποιεί την ανακατανομή του χώρου αποθήκευσης.
  - Η χρήση μερίδων μεταβλητού μεγέθους ή σταθερού μικρού μεγέθους ελαττώνει την σπατάλη του χώρου αποθήκευσης.
- Όπως γίνεται συνήθως σε αυτές τις περιπτώσεις, πρέπει να γίνεται συνδυασμός των υπέρ και των κατά.

66

## Μέγεθος μερίδας

- Το αποτέλεσμα είναι η ύπαρξη δύο βασικών προσεγγίσεων:
  - Το μέγεθος της μερίδας είναι μεταβλητό και αρκετά μεγάλο για την αποθήκευση όλου του αρχείου.
    - Καλύτερη απόδοση, μικρότερη σπατάλη χώρου και μικρότερο μέγεθος των σχετικών πινάκων.
    - Δυσκολία επαναχρησιμοποίησης του χώρου που καταμεμήθηκε.
    - Δημιουργία κατακερματισμού στον ελεύθερο χώρο που πρέπει να αντιμετωπισθεί με χρήση μιας εκ των τεχνικών που χρησιμοποιούνται στα μεταβλητά τμήματα (πρώτη / καλύτερη / πιο κοινή τοποθέτηση).
  - Το μέγεθος της μερίδας είναι ένα μπλοκ (η ελάχιστη μονάδα παροχής μνήμης σε επίπεδο δίσκου).
    - Περισσότερη ευελιξία.
    - Ανάγκη για μεγαλύτερους πίνακες και πιο πολύπλοκες δομές δεδομένων.

67

## Κατανομή μπλοκ σε αρχεία

- Το βασικό ζήτημα στην υλοποίηση της αποθήκευσης των αρχείων στο δίσκο είναι να γνωρίζουμε κάθε στιγμή ποια μπλοκ του δίσκου ανήκουν σε ποιο αρχείο.
- Εδώ υπάρχουν οι εξής μέθοδοι:
  - Συνεχής κατανομή (contiguous allocation).
  - Κατανομή συνδεδεμένης λίστας (chained allocation).
  - Κατανομή συνδεδεμένης λίστας με δείκτες (indexed allocation).

68

## Συνεχής κατανομή

- Αποθηκεύει κάθε αρχείο σαν ένα συνεχόμενο μπλοκ δεδομένων.
- Είναι εύκολη στην υλοποίηση (χρειάζεται να θυμόμαστε μόνο το πρώτο μπλοκ κάθε αρχείου και το μέγεθος του αρχείου) και έχει καλή απόδοση.
- Όμως δεν είναι ευέλικτη (ιδιαίτερα αν δεν γνωρίζουμε το μέγιστο πιθανό μέγεθος του αρχείου) και οδηγεί σε (εξωτερικό) κατακερματισμό του δίσκου και ανάγκη συμπίεσής του (compaction) περιοδικά.
- Τα μπλοκ δίνονται εκ των προτέρων και το μέγεθος της μερίδας είναι μεταβλητό.

69

## Παράδειγμα συνεχούς κατανομής

File Name	Start Block	Length
File A	2	3
File B	9	5
File C	18	8
File D	30	2
File E	26	3

Figure 12.9 Contiguous File Allocation

70

## Αντιμετώπιση εξωτερικού κατακερματισμού με συμπίεση

File Name	Start Block	Length
File A	0	3
File B	3	5
File C	8	8
File D	19	2
File E	16	3

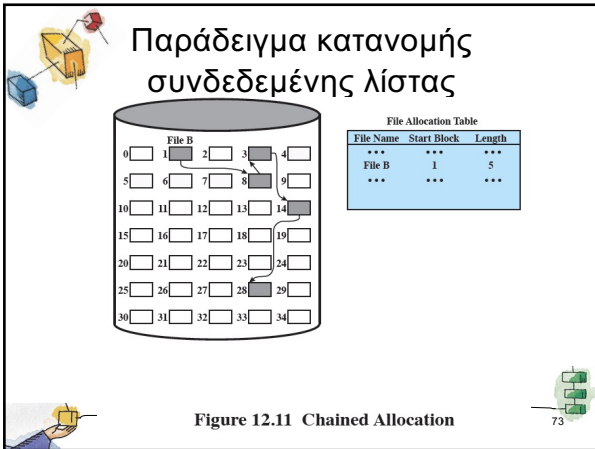
Figure 12.10 Contiguous File Allocation (After Compaction)

71

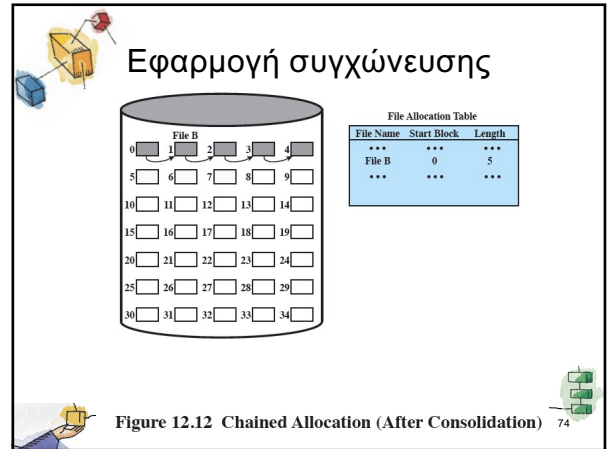
## Κατανομή συνδεδεμένης λίστας

- Αποθηκεύει ένα αρχείο κάνοντας χρήση μίας ομάδας μπλοκ, όχι κατ' ανάγκη συνεχόμενων.
  - Η πρώτη λέξη του κάθε μπλοκ είναι ένας δείκτης στο επόμενο.
- Είναι σχετικά εύκολη η υλοποίησή της (χρειάζεται η αποθήκευση της διεύθυνσης του πρώτου μπλοκ κάθε αρχείου).
- Αλλά η χρήση δεικτών δεσμεύει κάποιο ποσό μνήμης.
- Επίσης η τυχαία προσπέλαση είναι πιο αργή δημιουργώντας την ανάγκη συνώνευσης (consolidation) των μπλοκ σε συνεχόμενες θέσεις μνήμης.
- Τα μπλοκ δίνονται δυναμικά ένα-ένα όποτε παρίσταται ανάγκη και κατ' επέκταση δεν δημιουργείται εξωτερικός κατακερματισμός.
- Είναι κατάλληλη για σειριακά αρχεία.

72



73



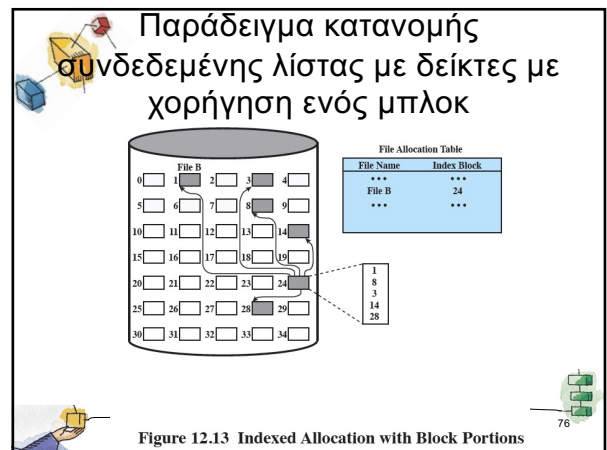
74

### Κατανομή συνδεδεμένης λίστας με δείκτες

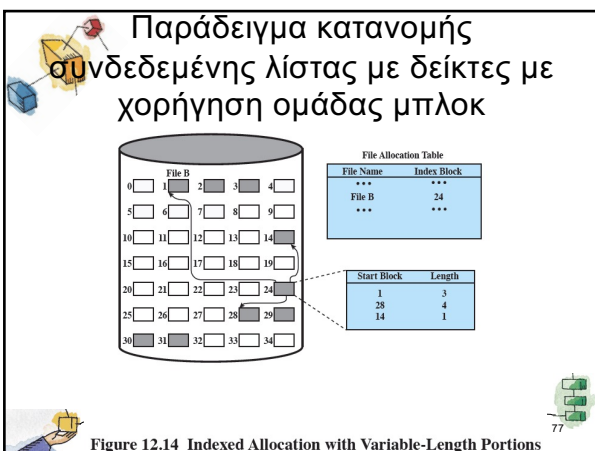
- Απαλείφει τα προβλήματα των προηγούμενων μεθόδων με τη χρήση ενός πίνακα για κάθε αρχείο, κάθε εγγραφή του οποίου είναι ένας δείκτης στο αντίστοιχο μπλοκ του αρχείου.
- Τα μπλοκ μπορεί να δίνονται ένα κάθε φορά που παρίσταται ανάγκη (παραλλαγή της δεύτερης μεθόδου) ή σε ομάδες περιοδικά.
  - Για τη δεύτερη αυτή περίπτωση το μέγεθος της ομάδας μπορεί να είναι σταθερό ή μεταβλητό (παραλλαγές της πρώτης περίπτωσης).
  - Επίσης στη δεύτερη περίπτωση υποστηρίζεται καλύτερα η τοπικότητα των αναφορών.
- Η τυχαία προσπέλαση τώρα είναι πιο εύκολη.
- Επίσης δεν δημιουργείται εξωτερικός κατακερματισμός.
- Όμως ο πίνακας αυτός (που αποθηκεύεται σε κάποιο μπλοκ και δεν είναι μέρος του FAT) πρέπει να βρίσκεται εξ' ολοκλήρου στην κύρια μνήμη.
- Και εδώ μπορεί να γίνεται περιοδικά συγχώνευση των μπλοκ.

Figure 12.13 Indexed Allocation with Block Portions

75



76




77

### Σύγκριση μεθόδων

	Contiguous	Chained	Indexed	
Preallocation?	Necessary	Possible	Possible	
Fixed or variable size portions?	Variable	Fixed blocks	Fixed blocks	Variable
Portion size	Large	Small	Small	Medium
Allocation frequency	Once	Low to high	High	Low
Time to allocate	Medium	Long	Short	Medium
File allocation table size	One entry	One entry	Large	Medium


Figure 12.14 Indexed Allocation with Variable-Length Portions

78




## Διαχείριση των ελεύθερων μπλοκ

- Πέρα από την ανάγκη διαχείρισης μνήμης που είναι δεσμευμένη από αρχεία, χρειάζεται διαχείριση και η ελεύθερη μνήμη του δίσκου.
- Για να χορηγούμε μνήμη σε κάποιο αρχείο, πρέπει να γνωρίζουμε ποια μπλοκ του δίσκου είναι ελεύθερα.
- Επομένως, χρειαζόμαστε ένα πίνακα κατανομής δίσκου επιπρόσθετα του πίνακα κατανομής αρχείων.
- Για τη δημιουργία ενός τέτοιου πίνακα υπάρχει ένας αριθμός από προσεγγίσεις.




79




## Χάρτες bits

- Με βάση αυτή τη μέθοδο, υπάρχει ένα διάνυσμα με ένα bit για κάθε μπλοκ του δίσκου.
- Αν ένα bit έχει τιμή 0 το αντίστοιχο μπλοκ είναι ελεύθερο, ενώ αν είναι 1 το μπλοκ χρησιμοποιείται από κάποιο αρχείο.
- Η μέθοδος αυτή είναι εύκολη στην υλοποίηση, χρειάζεται λίγο χώρο μνήμης (και επομένως το διάνυσμα μπορεί να βρίσκεται εξ' ολοκλήρου στην κύρια μνήμη) και είναι εύκολο να βρεθεί ένα ή περισσότερα (γειτνιάζοντα) ελεύθερα μπλοκ.




80




## Λίστα ελεύθερων μερίδων

- Οι ελεύθερες μερίδες (δηλαδή ομάδες από συνεχόμενα ελεύθερα μπλοκ) συνδέονται με δείκτες σε μία λίστα και με χρήση μίας τιμής που δηλώνει το μέγεθος της μερίδας (δηλαδή τον αριθμό των συνεχόμενων μπλοκ από τα οποία αποτελείται).
- Χρησιμοποιεί πολύ λίγο χώρο γιατί το μόνο που χρειάζεται να αποθηκευτεί είναι ο δείκτης στην πρώτη μερίδα και το μέγεθός της.
- Είναι κατάλληλη με όλες τις τεχνικές κατανομής μπλοκ που εξετάσαμε προηγουμένως:
  - Αν χρειάζεται να καταμετρηθεί ένα μπλοκ, επιλέγεται το πρώτο στη λίστα και τροποποιείται ανάλογα ο δείκτης και η τιμή μεγέθους της (πρώτης) μερίδας.
  - Αν χρειάζεται να καταμετρηθεί μία ομάδα από μπλοκ, τότε εξετάζεται το μέγεθος της κάθε μερίδας και επιλέγεται η πρώτη που ικανοποιεί τις ανάγκες σε αριθμό μπλοκ.
- Η μέθοδος αυτή δυνατόν να οδηγήσει σε κατακερματισμό με τη δημιουργία πολλών μεμονωμένων ελεύθερων μπλοκ.
- Επίσης, πριν να μπορεί να χρησιμοποιηθεί ένα καταμετρηθέν μπλοκ, πρέπει να διαβασθεί ο δείκτης στο επόμενο ελεύθερο μπλοκ, κάτι που έχει σημαντικό κόστος στο χρόνο δημιουργίας ενός αρχείου αν χρειάζεται η κατανομή πολλών μπλοκ μαζί.
- Τέλος, το σβήσιμο αρχείων που είναι κατακερματισμένα παίρνει μεγάλο χρονικό διάστημα.




81

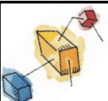


## Ευρετηρίαση ελεύθερου χώρου

- Θεωρεί ότι ο ελεύθερος χώρος είναι ένα αρχείο και χρησιμοποιεί ένα πίνακα ευρετηρίασης (index table) για πρόσβαση σε αυτόν.
- Για λόγους καλύτερης απόδοσης, η ευρετηρίαση δεν γίνεται σε επίπεδο μεμονωμένων μπλοκ αλλά μερίδων (συνεχόμενης ομάδας ελεύθερων μπλοκ).
- Κατ' επέκταση, στον πίνακα υπάρχει ένα στοιχείο για κάθε ελεύθερη μερίδα του δίσκου.
- Είναι κατάλληλη με όλες τις τεχνικές κατανομής μπλοκ που εξετάσαμε προηγουμένως.




82




## Λίστα ελεύθερων μπλοκ

- Κάθε μπλοκ έχει ένα μοναδικό αριθμό με σειριακή αύξηση.
- Σε κάποιο χώρο στο δίσκο αποθηκεύεται μία λίστα με τους αριθμούς των μπλοκ που είναι ελεύθερα.
- Ανάλογα με το μέγεθος του δίσκου, η αναπαράσταση ενός αριθμού χρειάζεται 24 ή 32 bits.
- Κατ' επέκταση το μέγεθος της λίστας είναι 24 ή 32 φορές μεγαλύτερο από το μέγεθος του αντίστοιχου χάρτη bits και η αποθήκευσή της πρέπει να γίνει στο δίσκο και όχι στην κύρια μνήμη.




83



## Τόμος

- Ο όρος αυτός χρησιμοποιείται συχνά με διαφορετική ερμηνεία από διαφορετικά Λ.Σ. και συστήματα διαχείρισης αρχείων, αλλά ουσιαστικά ο **τόμος** (volume) είναι ένας λογικός δίσκος.
- Συγκεκριμένα, είναι μία συλλογή από τομείς (sectors) στην περιφερειακή μνήμη, που μπορούν να χρησιμοποιηθούν για αποθήκευση από το Λ.Σ. ή τις εφαρμογές.
- Οι τομείς ενός τόμου δεν χρειάζονται να βρίσκονται σε συνεχόμενες φυσικές θέσεις στην αποθηκευτική συσκευή αλλά θα πρέπει να δίνεται αυτή η εντύπωση στο Λ.Σ. ή τις εφαρμογές.
- Ένας τόμος μπορεί να είναι σύνθετος, δηλαδή να αποτελείται από τη συγχώνευση μικρότερων τόμων.



84

## Περιεχόμενα

- Εισαγωγικές έννοιες.
- Οργάνωση αρχείων και πρόσβαση σε αυτά.
- Δένδρα-B.
- Κατάλογοι αρχείων.
- Διαμοίραση αρχείων.
- Ομαδοποίηση εγγραφών.
- Διαχείριση του χώρου του δίσκου.
- Διαχείριση αρχείων στο Unix.
- Διαχείριση αρχείων στο Linux.
- Διαχείριση αρχείων στα Windows.
- Διαχείριση αρχείων στο Android.

85

## Είδη αρχείων στο Unix — 1

- Συνηθισμένο.
  - Έχει μέγεθος από 0 μέχρι οποιονδήποτε αριθμό από μπλοκ.
  - Περιέχει οποιασδήποτε μορφής δεδομένα.
  - Δημιουργείται από ένα χρήστη ή μία εφαρμογή ή ένα βοηθητικό πρόγραμμα.
  - Το σύστημα αρχείων δεν του επιβάλει μία συγκεκριμένη εσωτερική δομή αλλά το χειρίζεται ως μία σειρά από bytes.
- Κατάλογος.
  - Περιέχει μία λίστα από ονόματα αρχείων και δείκτες σε μία δομή ελέγχου των αρχείων αυτών (inode).
  - Οι κατάλογοι είναι ιεραρχικά οργανωμένοι μεταξύ τους.
  - Ουσιαστικά είναι ένα συνηθισμένο αρχείο στο οποίο όμως οι χρήστες έχουν δικαίωμα πρόσβασης μόνο για διάβασμα ενώ το σύστημα έχει δικαίωμα πρόσβασης για γράψιμο.

86

## Είδη αρχείων στο Unix — 2

- Ειδικό.
  - Δεν περιέχει δεδομένα αλλά παρέχει ένα μηχανισμό για την αντιστοίχιση φυσικών συσκευών με ονόματα αρχείων.
  - Τα ονόματα αυτά χρησιμοποιούνται για την πρόσβαση σε περιφερειακές συσκευές, όπως τερματικά και εκτυπωτές.
  - Κάθε συσκευή E/E έχει ένα τέτοιο αντίστοιχο αρχείο.
- Ονοματικός ενδοαγωγός (named pipes).
  - Επιτρέπει την επικοινωνία μεταξύ διεργασιών.
  - Είναι ένας προσωρινός χώρος που επιτρέπει σε μία διεργασία να εξάγει πληροφορίες σε αυτόν οι οποίες μετά θα εισαχθούν σε μία άλλη διεργασία.
- Σύνδεσμος (link).
  - Είναι ένας εναλλακτικό όνομα για κάποιο προϋπάρχον αρχείο.
- Συμβολικός σύνδεσμος (symbolic link).
  - Είναι ένα αρχείο δεδομένων που περιέχει το όνομα του αρχείου με το οποίο συνδέεται.

87

## Inode

- Το **inode** (index node) είναι μία δομή ελέγχου που περιέχει τις πιο σημαντικές πληροφορίες που χρειάζεται το Unix για κάποιο αρχείο (όπως τα ιδιοχαρακτηριστικά του αρχείου, δικαιώματα πρόσβασης σε αυτό, κλπ.).
- Αν και πολλαπλά ονόματα αρχείων μπορούν να σχετίζονται με ένα inode, ένα εν ενεργεία inode μπορεί να σχετίζεται με ακριβώς ένα αρχείο και κάθε αρχείο ελέγχεται από ακριβώς ένα inode.
- Η ακριβής δομή ενός inode δεν είναι η ίδια για όλες τις παραλλαγές του Unix.

88

## Δομή του inode για κάποιο αρχείο στο Free BSD

- Είδος και δικαιώματα πρόσβασης στο αρχείο.
- Ο ιδιοκτήτης του αρχείου και πρόσβαση σε αυτό από ομάδες χρηστών.
- Πότε δημιουργήθηκε και πότε έγινε τελευταία φορά πρόσβαση σε αυτό για διάβασμα ή γράψιμο.
- Μέγεθος σε bytes.
- Μία ομάδα από δείκτες σε μπλοκ, όπως εξηγείται παρακάτω.
- Ο αριθμός και μέγεθος των μπλοκ από τα οποία αποτελείται το αρχείο (συνήθως το δεύτερο είναι το ίδιο με αυτό του συστήματος).
- Ο αριθμός των καταλόγων που έχουν αναφορά στο αρχείο.
- Ιδιοχαρακτηριστικά του αρχείου.
- Ένας μοναδικός αριθμός που δημιουργείται κάθε φορά που ένα inode σχετίζεται με ένα αρχείο και το οποίο χρησιμοποιείται για να ανιχνεύει αναφορές σε διαγραμμένα αρχεία.

89

## Απεικόνιση της δομής του inode στο FreeBSD

inode
owner (2)
timestamp (4)
size
direct (0)
direct (1)
...
direct (12)
single indirect
double indirect
triple indirect
block count
reference count
flags (2)
generation number
blocksize
extended attr size
extended attribute blocks

The diagram illustrates how these fields point to data blocks. 'direct' fields point directly to 'Data' blocks. 'single indirect' points to a 'Pointers' block, which then points to 'Data' blocks. 'double indirect' points to a 'Pointers' block, which points to other 'Pointers' blocks, which in turn point to 'Data' blocks. 'triple indirect' follows a similar pattern with three levels of pointers.

90

### Κατανομή χώρου σε ένα αρχείο

- Η μονάδα κατανομής χώρου σε ένα αρχείο είναι το μπλοκ.
- Η κατανομή γίνεται δυναμικά όταν δημιουργείται η ανάγκη και όχι με προκατανομή.
- Τα μπλοκ ενός αρχείου δεν χρειάζεται να είναι σε συνεχόμενες θέσεις στο δίσκο.
- Χρησιμοποιείται η ευρετηρίαση για να γνωρίζει το σύστημα ποια μπλοκ ανήκουν σε ποια αρχεία και το ευρετήριο είναι αποθηκευμένο στη δομή inode του κάθε αρχείου.
- Σε όλες τις παραλλαγές του Unix, η δομή ελέγχου inode αποτελείται από ένα αριθμό από άμεσους δείκτες και τρεις έμμεσους δείκτες.

91

### Μεγέθη αρχείων υποστηριζόμενα από το FreeBSD inode

- Το inode περιέχει 120 bytes πληροφοριών για διευθύνσεις μνήμης που είναι οργανωμένα σε 15 δείκτες των 64 bits ο καθένας. Το μέγεθος του μπλοκ είναι 4K και κάθε μπλοκ αποθηκεύει 512 διευθύνσεις μπλοκ.
- Οι πρώτοι 12 (άμεσοι) δείκτες δείχνουν στα πρώτα 12 μπλοκ του αρχείου. Αν το αρχείο είναι μεγαλύτερο, τότε χρησιμοποιούνται οι υπόλοιποι (έμμεσοι) δείκτες.

Level	Number of Blocks	Number of Bytes
Direct	12	48K
Single Indirect	512	2M
Double Indirect	$512 \times 512 = 256K$	1G
Triple Indirect	$512 \times 256K = 128M$	512G

92

### Σχέση καταλόγων και inodes στο UNIX

- Ένας κατάλογος περιέχει:
  - Μία ομάδα από ονόματα αρχείων ή άλλων υποκαταλόγων.
  - Δείκτες στα inodes των αρχείων του καταλόγου.
  - Κάθε στοιχείο στον πίνακα ενός καταλόγου αποτελείται από το όνομα ενός αρχείου ή υποκαταλόγου και έναν ακέραιο αριθμό που είναι δείκτης στο αντίστοιχο στοιχείο του πίνακα του inode.

Figure 12.16 UNIX Directories and Inodes

93

### Περιεχόμενα

- Εισαγωγικές έννοιες.
- Οργάνωση αρχείων και πρόσβαση σε αυτά.
- Δένδρα-B.
- Κατάλογοι αρχείων.
- Διαμοίραση αρχείων.
- Ομαδοποίηση εγγραφών.
- Διαχείριση του χώρου του δίσκου.
- Διαχείριση αρχείων στο Unix.
- Διαχείριση αρχείων στο Linux.
- Διαχείριση αρχείων στα Windows.
- Διαχείριση αρχείων στο Android.

94

### Ιδεατό σύστημα αρχείων στο Linux

- Το ιδεατό σύστημα αρχείων (Virtual File System) στο Linux παρέχει ένα ομοιόμορφο τρόπο παρουσίασης του συστήματος αρχείων στο χρήστη.
- Θεωρεί ότι τα αρχεία είναι αντικείμενα της περιφερειακής μνήμης του συστήματος και έχουν κοινές βασικές ιδιότητες ανεξάρτητα από την αρχιτεκτονική του συστήματος.
- Καλύπτει όλες τις δυνατές λειτουργίες ενός τέτοιου συστήματος.

95

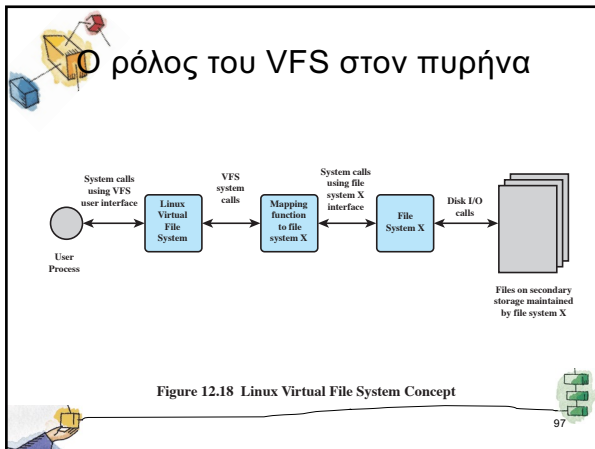
### Βασικά στοιχεία της στρατηγικής VFS

- Μία διεργασία του χρήστη καλεί μία ρουτίνα του συστήματος, π.χ. για διάβασμα.
- Το VFS μετατρέπει την κλήση σε μία άλλη κλήση που λαμβάνει υπ' όψη τις ιδιαιτερότητες του κάθε συστήματος αρχείων.

Figure 12.17 Linux Virtual File System Context

96





97

## Τα βασικά είδη αντικειμένων στο VFS

- Το VFS είναι αντικειμενοστρεφές και τα αντικείμενα υλοποιούνται ως δομές δεδομένων στη C.
- Κάθε αντικείμενο περιέχει δεδομένα και δείκτες σε συναρτήσεις που εκτελούνται σε αυτά τα δεδομένα και είναι υλοποιημένες από το σύστημα αρχείων.
- Υπάρχουν 4 βασικά είδη αντικειμένων, ανάλογα με αν αναπαριστούν τόμο, inode, κατάλογο ή αρχείο.

98

## Περιεχόμενα

- Εισαγωγικές έννοιες.
- Οργάνωση αρχείων και πρόσβαση σε αυτά.
- Δένδρα-B.
- Κατάλογοι αρχείων.
- Διαμοίραση αρχείων.
- Ομαδοποίηση εγγραφών.
- Διαχείριση του χώρου του δίσκου.
- Διαχείριση αρχείων στο Unix.
- Διαχείριση αρχείων στο Linux.
- ➔ Διαχείριση αρχείων στα Windows.
- ➔ Διαχείριση αρχείων στο Android.

99

## Το σύστημα NTFS

- Η διαχείριση αρχείων στα Windows γίνεται από το σύστημα NTFS (New Technology File System), το οποίο υποστηρίζει:
  - Δυνατότητα επανόρθωσης μετά από λάθη στο σύστημα ή βλάβη στο δίσκο.
  - Ασφάλεια των πληροφοριών.
  - Χρήση μεγάλης χωρητικότητας δίσκων και μεγάλου μεγέθους αρχεία.
  - Πολλαπλές ροές δεδομένων που επιτρέπει τη χρήση αρχείων και από άλλα Λ.Σ.
  - Αποθήκευση πληροφοριών αναφορικά με τις αλλαγές που γίνονται σε αρχεία και τόμους.
  - Συμπίεση και κωδικοποίηση των δεδομένων.

100

## Βασικά στοιχεία του NTFS

- Τομέας.
  - Η μικρότερη φυσική μονάδα αποθήκευσης στο δίσκο.
  - Έχει μέγεθος σχεδόν πάντα 512 bytes.
- Σύμπλεγμα (cluster).
  - Ένας ή περισσότεροι τομείς, συνεχόμενοι στον ίδιο δίσκο του δίσκου.
- Τόμος.
  - Λογικό διαχωριστικό σε ένα δίσκο, αποτελούμενο από ένα ή περισσότερα συμπλέγματα.
  - Περιέχει αρχεία, το σύστημα διαχείρισης αυτών των αρχείων και ότι ελεύθερος χώρος υπάρχει για μελλοντική χρήση από αρχεία.
  - Μπορεί να είναι μέρος ενός δίσκου ή να χρησιμοποιεί και περισσότερους από ένα δίσκους.
  - Το μεγαλύτερο μέγεθός του μπορεί να είναι  $2^{64}$  bytes.

101

## Τυπικές τιμές μεγεθών

Volume Size	Sectors per Cluster	Cluster Size
512Mbyte	1	512bytes
512Mbyte – 1 Gbyte	2	1K
1–2 Gbyte	4	2K
2–4 Gbyte	8	4K
4–8 Gbyte	16	8K
8–16 Gbyte	32	16K
16–32 Gbyte	64	32K
>32Gbyte	128	64K

102

## Δομή τόμου στο NTFS

- Κάθε στοιχείο του τόμου είναι ένα αρχείο και κάθε αρχείο είναι μία ομάδα από ιδιοχαρακτηριστικά.
  - Ακόμα και τα περιεχόμενα ενός αρχείου θεωρούνται ως ένα ιδιοχαρακτηριστικό.
  - Το πρώτο μέρος της δομής ενός τόμου έχει πληροφορίες για τη διάταξη του τόμου και για το αρχικό διάβασμα του τόμου από το Λ.Σ.
  - Το MFT περιέχει πληροφορίες για τα αρχεία και καταλόγους του τόμου και είναι οργανωμένο ως μία ομάδα από σειρές σε μία σχεσιακή βάση δεδομένων.
  - Τα αρχεία συστήματος είναι συνήθως 1 MB σε μέγεθος και δείχνουν μεταξύ άλλων τη χρήση του τόμου από συμπλέγματα, τα είδη των ιδιοχαρακτηριστικών που υποστηρίζει ο τόμος, κλπ.

Figure 12.19 NTFS Volume Layout

103

## Αρχιτεκτονική του NTFS

Figure 12.20 Windows NTFS Components

104

## Περιεχόμενα

- Εισαγωγικές έννοιες.
- Οργάνωση αρχείων και πρόσβαση σε αυτά.
- Δένδρα-B.
- Κατάλογοι αρχείων.
- Διαμοίραση αρχείων.
- Ομαδοποίηση εγγραφών.
- Διαχείριση του χώρου του δίσκου.
- Διαχείριση αρχείων στο Unix.
- Διαχείριση αρχείων στο Linux.
- Διαχείριση αρχείων στα Windows.

[Διαχείριση αρχείων στο Android.](#)

105

## Οργάνωση του συστήματος αρχείων

- Το Android χρησιμοποιεί τις δυνατότητες του Linux και η οργάνωση των αρχείων του Android είναι παρόμοια με αυτής του Linux, με κάποιες ιδιαιτερότητες:
  - Ο κατάλογος του συστήματος εμπεριέχει τα βασικά μέρη του Λ.Σ. συμπεριλαμβανομένων του εκτελέσιμου κώδικα, βιβλιοθηκών, κλπ. Επίσης περιλαμβάνει έναν αριθμό από βασικές εφαρμογές, όπως `AlarmClock`, `Calculator` και `Camera`.
  - Ο κατάλογος δεδομένων χρησιμοποιείται από τις εφαρμογές για την αποθήκευση των αρχείων τους. Η εγκατάσταση μίας νέας εφαρμογής, τοποθετεί το `.apk` της εφαρμογής στο `/data/app`, σχετικές βιβλιοθήκες στο `/data/data/<όνομα εφαρμογής>` και δημιουργεί άλλα σχετικά αρχεία.
  - Ο κατάλογος `cache` χρησιμοποιείται για την προσωρινή αποθήκευση δεδομένων.
  - Ο κατάλογος `mnt/sdcard` σχετίζεται με εξωτερική κάρτα μνήμης.

106

## Η τυπική οργάνωση του δέντρου αρχείων

Figure 12.21 Typical Directory Tree of Android

107

## SQLite

- Η SQL (Structured Query Language) είναι η πιο διαδεδομένη γλώσσα για σχεσιακές βάσεις δεδομένων, δημιουργήθηκε από την IBM στα μέσα της δεκαετίας του '70 και αποτελεί ευρέως διαδεδομένο πρότυπο.
- Η SQLite που υποστηρίζεται από το Android είναι ειδικά σχεδιασμένη για ενσωματωμένα συστήματα και συστήματα με περιορισμένο χώρο μνήμης.
- Όλη η βιβλιοθήκη της SQLite μπορεί να υλοποιηθεί σε λιγότερο από 400 KB χώρο μνήμης.
- Μέρη της βιβλιοθήκης που δεν χρειάζονται μπορούν να αφαιρεθούν στο στάδιο της μεταγλώττισης και τότε οι ανάγκες σε μνήμη μειώνονται σε λιγότερο από 190 KB.
- Αντίθετα με άλλα συστήματα διαχείρισης βάσεων δεδομένων, η SQLite δεν είναι ξεχωριστή διεργασία που καλείται από μία εφαρμογή, αλλά η βιβλιοθήκη της ενώνεται με την εφαρμογή και αποτελεί αναπόσπαστο κομμάτι αυτής.

108

ΕΠΛ222: Λειτουργικά Συστήματα  
(μετάφραση στα ελληνικά των διαφανειών του βιβλίου Operating Systems: Internals and Design Principles, 9E, William Stallings)

## Τέλος Ενότητας 10

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

Γιώργος Α. Παπαδόπουλος  
Τμήμα Πληροφορικής  
Πανεπιστήμιο Κύπρου

