



# Λειτουργικά Συστήματα

Ενότητα 2α: Δομή, Υπηρεσίες, Εισαγωγή στις Διεργασίες

Αθηνά Βακάλη  
Τμήμα Πληροφορικής



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



# Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



# Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «Ανοικτά Ακαδημαϊκά Μαθήματα στο Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



# Λειτουργικό Σύστημα

---

- Ένα πρόγραμμα που ελέγχει την εκτέλεση των προγραμμάτων εφαρμογών.
- Λειτουργεί ως ενδιάμεσο μεταξύ εφαρμογών και του υλικού μέρους του υπολογιστή.



# Στόχοι του Λειτουργικού Συστήματος

## Ευκολία

- Κάνει πιο εύκολη τη χρήση ενός υπολογιστή.

## Αποτελεσματικότητα

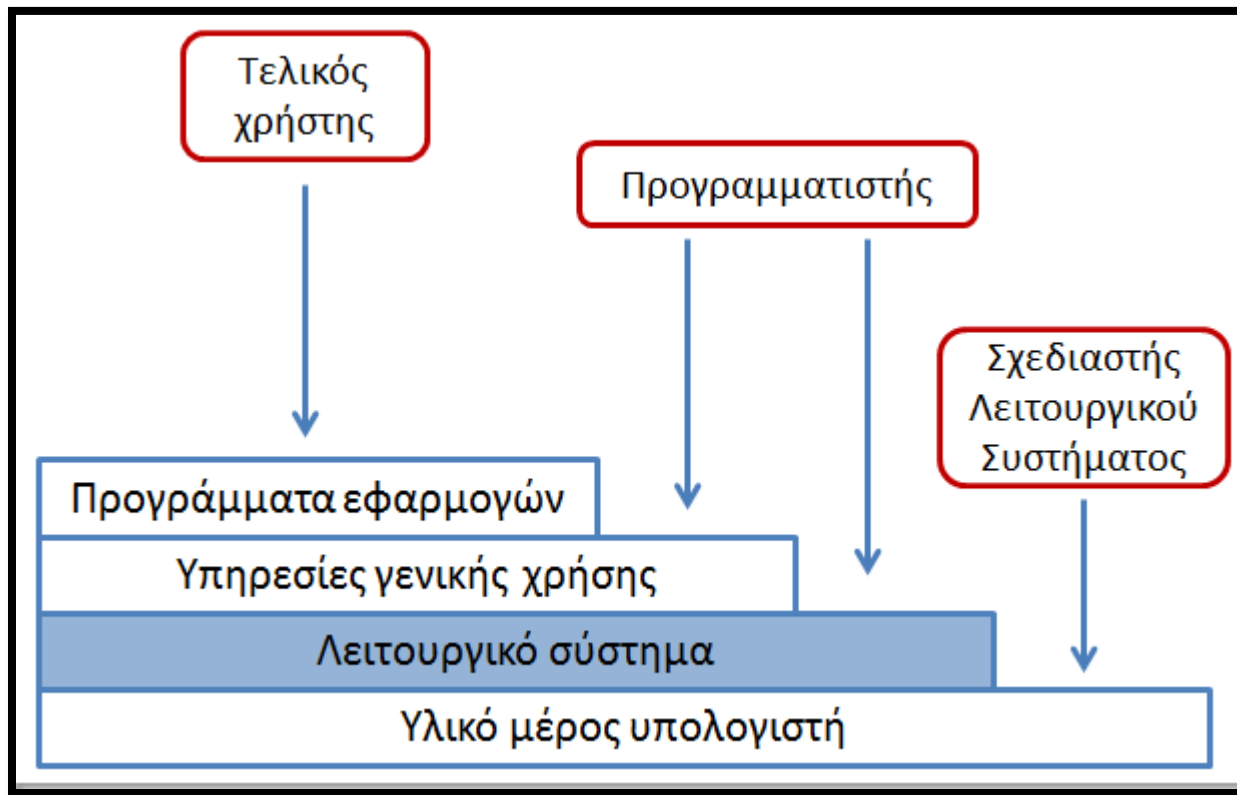
- Επιτρέπει τους πόρους του λειτουργικού συστήματος να χρησιμοποιούνται με αποτελεσματικό τρόπο.

## Ικανότητα εξέλιξης

- Επιτρέπει αποτελεσματική ανάπτυξη, έλεγχο και εισαγωγή νέων λειτουργιών συστήματος χωρίς να παρεμποδίζεται η διαδικασία εξυπηρέτησης.



# Επίπεδα και όψεις ενός Υπολογιστικού Συστήματος



Επίπεδα και Απόψεις ενός υπολογιστικού συστήματος



# Υπηρεσίες που παρέχονται από ένα Λειτουργικό Σύστημα (1/3)

- **Ανάπτυξη προγράμματος**  
π.χ. Κειμενογράφοι (editors) και απο-σφαλματωτές (debuggers).
- **Εκτέλεση προγράμματος**
- **Προσπέλαση σε συσκευές Εισόδου/Εξόδου (E/E)**
- **Ελεγχόμενη πρόσβαση σε αρχεία**
- **Σύστημα προσπέλασης**



# Υπηρεσίες που παρέχονται από ένα Λειτουργικό Σύστημα (2/3)

## Ανίχνευση σφάλματος και απόκριση

- Εσωτερικά και εξωτερικά σφάλματα υλικού τμήματος
  - Σφάλμα μνήμης
  - Δυσλειτουργία συσκευής
- Σφάλματα λογισμικού
  - Αριθμητική υπερχείλιση
  - Προσπέλαση σε απαγορευμένη θέση μνήμης
- Αδυναμία επικύρωσης της αίτησης μίας εφαρμογής





# Υπηρεσίες που παρέχονται από ένα Λειτουργικό Σύστημα (3/3)

## Λογιστική

- Συλλογή στατιστικών χρήσης.
- Παρακολούθηση παραμέτρων επίδοσης.
- Χρήσιμη για τη μελέτη μελλοντικών επεκτάσεων.
- Χρήσιμη για τις ανάγκες χρέωσης.



# Το Λειτουργικό Σύστημα ως διαχειριστής πόρων (1/2)

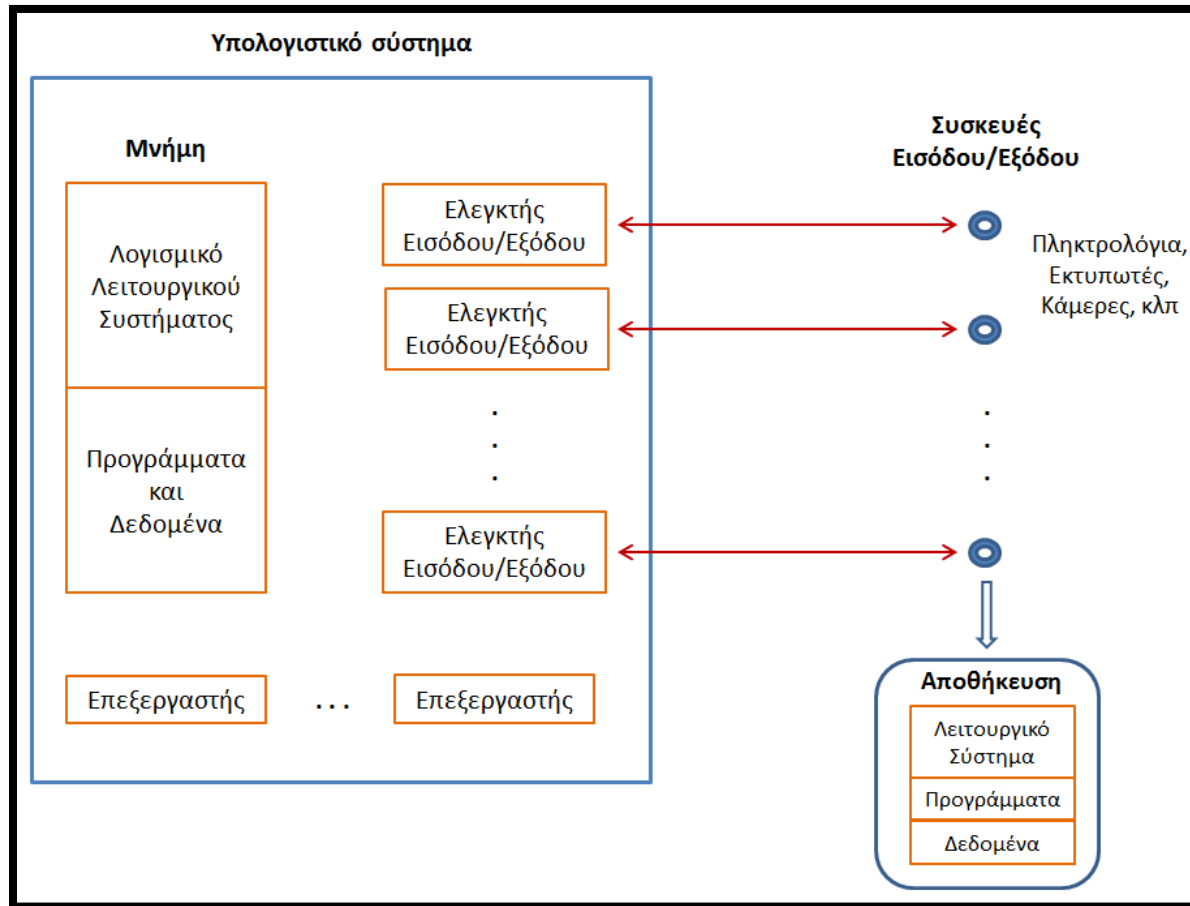
Λειτουργεί με τον ίδιο τρόπο με τον οποίο λειτουργεί ένα συνηθισμένο λογισμικό προγράμματος.

- Είναι ένα πρόγραμμα που εκτελείται.

Το λειτουργικό σύστημα εκχωρεί τον έλεγχο του επεξεργαστή για να εκτελεστούν άλλα προγράμματα.



# Το Λειτουργικό Σύστημα ως διαχειριστής πόρων (2/2)



## Το λειτουργικό σύστημα ως διαχειριστής πόρων

# Δυσκολίες στο σχεδιασμό λογισμικού συστήματος

- **Συγχρονισμός**

Εξασφάλιση ότι μία διεργασία που περιμένει για Είσοδο/Εξοδο έλαβε σήμα.

- **Αμοιβαίος αποκλεισμός**

- **Λειτουργία προγράμματος**

- **Αδιέξοδα**



# Σημαντικότερα θέματα

---

- Διεργασίες
- Διαχείριση μνήμης
- Προστασία και ασφάλεια πληροφορίας
- Δρομολόγηση και διαχείριση πόρων
- Δομή συστήματος



# Διαχείριση Διεργασιών

**Διεργασία: ένα πρόγραμμα που εκτελείται.**

Απαιτεί συγκεκριμένους πόρους (CPU χρόνος, μνήμη, αρχεία, συσκευές εισόδου/εξόδου) για να ολοκληρώσει μία εργασία.

Σχετικά με τη διαχείριση διεργασιών το Λειτουργικό Σύστημα έχει την ευθύνη των εξής ενεργειών:

- Δημιουργία/διαγραφή διεργασίας
- Αναστολή/επανάληψη διεργασίας
- Παροχή μηχανισμών για:
  - συγχρονισμό διεργασιών
  - επικοινωνία διεργασιών



# Διεργασία (1/3)

- Ένα πρόγραμμα σε εκτέλεση.
- Ένα στιγμιότυπο προγράμματος που εκτελείται σε έναν επεξεργαστή.
- Η οντότητα η οποία μπορεί να ανατεθεί και να εκτελεστεί σε έναν επεξεργαστή.
- Μία μονάδα δραστηριότητας που χαρακτηρίζεται από ένα μοναδικό ακολουθιακό νήμα εκτέλεσης, μία τρέχουσα κατάσταση και ένα συνδεδεμένο σύνολο πόρων του συστήματος.



# Διεργασία (2/3)

Αποτελείται από τρία συστατικά στοιχεία:

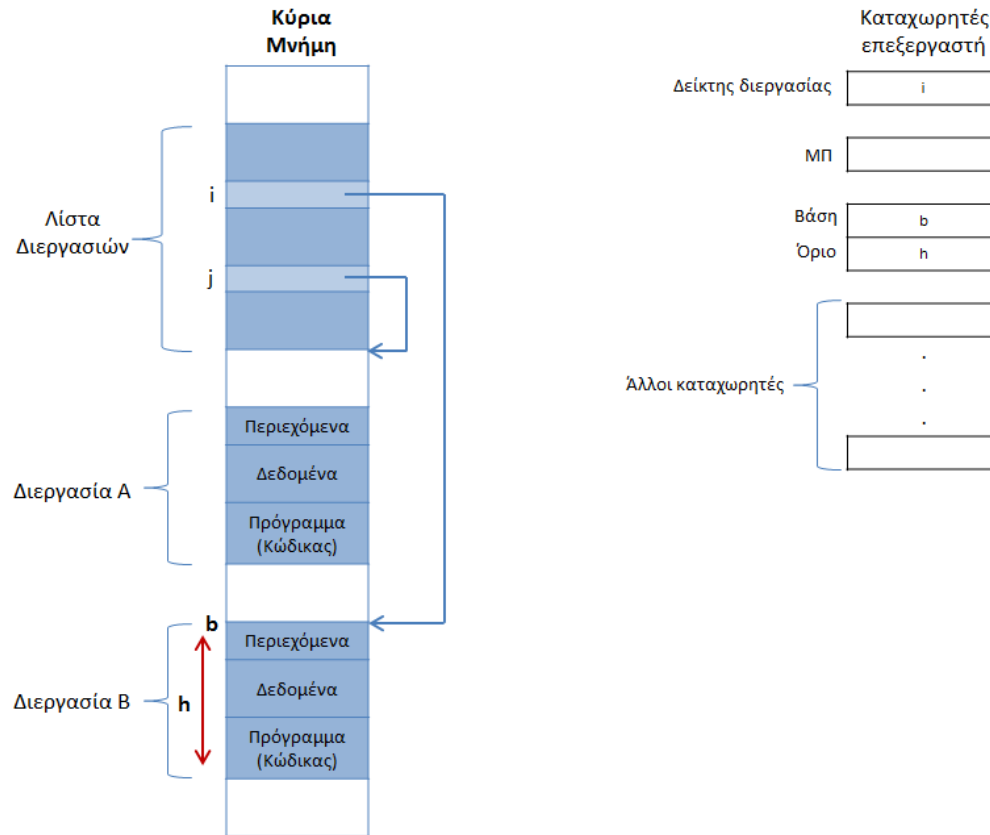
- Ένα εκτελέσιμο πρόγραμμα.
- Συνδεδεμένα δεδομένα που είναι απαιτούμενα από το πρόγραμμα.
- Το εκτελέσιμο περιεχόμενο του προγράμματος (**κατάσταση διεργασίας**).

Όλες οι διεργασίες που χρειάζεται το Λειτουργικό Σύστημα για να διαχειριστεί τη διεργασία.





# Διεργασία (3/3)



## Τυπική υλοποίηση διεργασίας

# Διεργασίες και Διαχείριση Κύριας Μνήμης

**Μνήμη:** ένας πίνακας από λέξεις ή bytes, καθένα με τηδική του διεύθυνση. Αποτελεί την “αποθήκη” μεταξύ Κεντρικής Μονάδας Επεξεργασίας (CPU) και Είσοδο/Εξοδο (I/O) συσκευών.

**Κύρια Μνήμη:** αποτελεί ένα μη-διατηρήσιμο μέσο αποθήκευσης. Χάνει τα δεδομένα της μόλις κλείσει το σύστημα.

Σχετικά με τη διαχείριση της **Κύριας Μνήμης** το λειτουργικό σύστημα έχει την ευθύνη των εξής λειτουργιών:

- Ποια κομμάτια μνήμης είναι σε τρέχουσα χρήση και από ποιον.
- Ποια διεργασία θα φορτωθεί στη μνήμη μόλις ελευθερωθεί χώρος μνήμης.
- Κατανομή και απο-κατανομή χώρου μνήμης σύμφωνα με τις απαιτήσεις.



# Διαχείριση Δευτερεύουσας Μνήμης

**Δευτερεύουσα Μνήμη:** είναι ένα μόνιμο μέσο αποθήκευσης όπου φυλάσσονται τα δεδομένα, υποστηρικτικά της Κύριας Μνήμης.

Τα περισσότερα υπολογιστικά συστήματα χρησιμοποιούν δίσκους ως πρωταρχικό άμεσο (online) αποθηκευτικό μέσο, για τα προγράμματα και τα δεδομένα.

Σχετικά με τη διαχείριση της **Δευτερεύουσας Μνήμης** το λειτουργικό σύστημα έχει την ευθύνη των εξής ενεργειών:

- Διαχείριση ελεύθερου χώρου μνήμης.
- Κατανομή χώρου αποθήκευσης.
- Χρονο-δρομολόγηση δίσκων.



# Διαχείριση Συστήματος Εισόδου/Εξόδου

Το **σύστημα Εισόδου/Εξόδου** αποτελείται από:

- Σύστημα επαναποθήκευσης ενδιάμεσης μνήμης.
- Γενικό σύστημα προσαρμογής οδηγών συσκευών.
- Προγράμματα οδηγών για συγκεκριμένες συσκευές.

## Διαχείριση Αρχείων

Σχετικά με τη διαχείριση των **Αρχείων** το λειτουργικό σύστημα έχει την ευθύνη των εξής ενεργειών:

- Δημιουργία/διαγραφή αρχείου.
- Δημιουργία/διαγραφή καταλόγου.
- Υποστήριξη πρωταρχικών ενεργειών διαχείρισης αρχείων και καταλόγων.
- Απεικόνιση αρχείων στη δευτερεύουσα μνήμη.
- Εφεδρικό αντίγραφο αρχείου σε μόνιμο αποθηκευτικό μέσο μνήμης.



# Σύστημα Προστασίας

Το **σύστημα προστασίας** αφορά στο μηχανισμό ελέγχου της πρόσβασης σε προγράμματα, χρήστες, διεργασίες του συστήματος και των πόρων των χρηστών.

Διάκριση (μη)-εξουσιοδοτημένης χρήσης.

Καθορισμός ελέγχων που θα επιβληθούν.



# Κατανεμημένο Σύστημα/Δικτύωση

**Σύνολο επεξεργαστών που δε δια-μοιράζονται μνήμη ή χρονισμό.**

Κάθε επεξεργαστής έχει τη δική του τοπική μνήμη.

Οι επεξεργαστές του συστήματος συνδέονται μέσω δικτύου επικοινωνίας.

Ένα κατανεμημένο σύστημα παρέχει δυνατότητα πρόσβασης των χρηστών σε πόρους συστήματος:

- Επιτάχυνση υπολογισμών.
- Αύξηση διαθεσιμότητας δεδομένων.
- Αύξηση αξιοπιστίας.



# Σύστημα Μεταγλώττισης Εντολών (1/2)

Πολλές εντολές δίνονται στο λειτουργικό σύστημα με εντολές ελέγχου (**control statements**) και αφορούν:

- Δημιουργία/διαχείριση διεργασιών.
- Χειρισμό εισόδου/εξόδου.
- Διαχείριση δευτερεύουσας μνήμης.
- Διαχείριση κύριας μνήμης.
- Πρόσβαση στο σύστημα αρχείων.
- Προστασία.
- Δικτύωση.



# Σύστημα Μεταγλώτισης Εντολών (2/2)

Το πρόγραμμα ανάγνωσης και μεταγλώτισης των εντολών ελέγχου καλείται:

- Μεταγλωτιστής καρτών-ελέγχου.
- Μεταγλωτιστής εντολών γραμμής.
- Κέλυφος (shell στο UNIX).

Ο ρόλος του είναι να παίρνει και να εκτελεί την επόμενη εντολή.





# Κλήσεις Συστήματος

Οι κλήσεις συστήματος παρέχουν το ενδιάμεσο μεταξύ ενός εκτελούμενου προγράμματος και του λειτουργικού συστήματος:

- Διαθέσιμες ως εντολές γλώσσας assembly.
- Γλώσσες προγραμματισμού συστήματος (C, PL/360).

**Μέθοδοι για το πέρασμα παραμέτρων** μεταξύ ενός εκτελούμενου προγράμματος και του λειτουργικού συστήματος:

- Πέρασμα παραμέτρων μέσω καταχωρητών.
- Αποθήκευση παραμέτρων σε έναν πίνακα στη μνήμη και πέρασμα της διεύθυνσης του πίνακα ως παράμετρο σε έναν καταχωρητή.
- Αποθήκευση (push) παραμέτρων σε στοίβα μέσω του προγράμματος και ανάκτηση (pop) από τη στοίβα μέσω του λειτουργικού συστήματος.



# Προγράμματα Συστήματος

Τα προγράμματα συστήματος παρέχουν ένα βολικό περιβάλλον για την ανάπτυξη και εκτέλεση προγραμμάτων.

Διακρίνονται σε προγράμματα για :

- Διαχείριση Αρχείων.
- Πληροφόρηση κατάστασης.
- Τροποποίηση αρχείων.
- Υποστήριξη γλώσσας προγραμματισμού.
- Φόρτωση και εκτέλεση προγράμματος.
- Επικοινωνίες.
- Προγράμματα εφαρμογών.



# Δομή Συστήματος (Απλή Περίπτωση)

## MS-DOS

Έχει γραφεί έτσι ώστε να παρέχει τη μεγαλύτερη δυνατή λειτουργικότητα σε ελάχιστο χώρο. Δε χωρίζεται σε επιμέρους κομμάτια. Έχει κάποια δομή, αλλά τα προσαρμοστικά ενδιάμεσα και τα επίπεδα λειτουργικότητας του δεν διαχωρίζονται με σαφήνεια.

## UNIX

Το αρχικό UNIX είχε περιορισμένη δόμηση λόγω της λειτουργικότητας του υλικού μέρους.

Το λειτουργικό σύστημα UNIX αποτελείται από δύο ξεχωριστά μέρη:

- Τα προγράμματα συστήματος
- Τον πυρήνα (kernel) που αποτελείται από κάθε τι πάνω από το υλικό και κάτω από το ενδιάμεσο των κλήσεων συστήματος. Παρέχει το σύστημα αρχείων, τη χρονοδρομολόγηση της CPU, τη διαχείριση μνήμης και άλλες λειτουργίες λειτουργικού συστήματος.



# Δομή Συστήματος (Περίπτωση Επιπέδων) (1/2)

Το λειτουργικό σύστημα χωρίζεται σε έναν αριθμό επιπέδων. Κάθε επίπεδο “χτίζεται” πάνω στο χαμηλότερο επίπεδο.

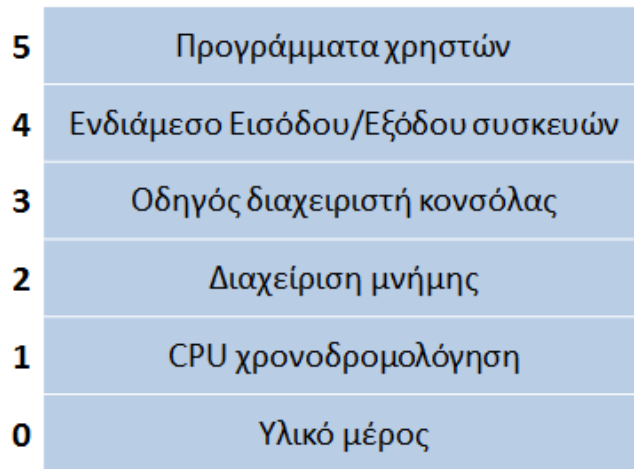
Το επίπεδο βάσης (επίπεδο 0) είναι το υλικό, ενώ το μέγιστο επίπεδο (επίπεδο N) είναι το ενδιάμεσο χρηστών (user interface). Κάθε επίπεδο χρησιμοποιεί τις υπηρεσίες των χαμηλότερών του επιπέδων.



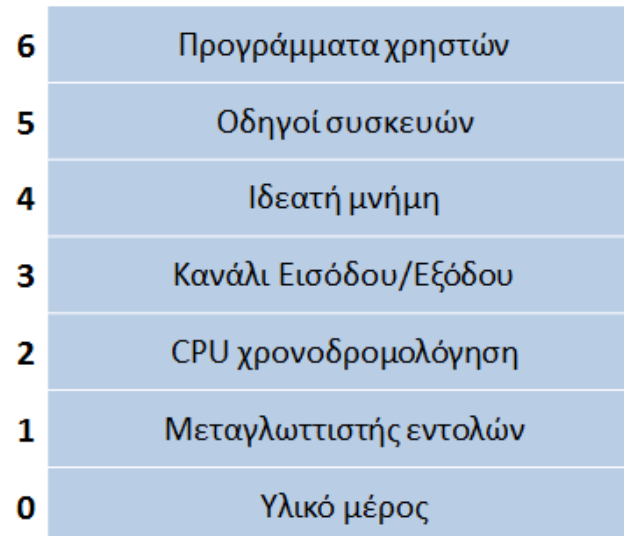
# Δομή Συστήματος (Περίπτωση Επιπέδων) (2/2)

Το Λειτουργικό Σύστημα χωρίζεται σε έναν αριθμό επιπέδων. Κάθε επίπεδο “χτίζεται” πάνω στο χαμηλότερο επίπεδο.

Το επίπεδο βάσης (επίπεδο 0) είναι το υλικό, ενώ το μέγιστο επίπεδο (επίπεδο N) είναι το ενδιάμεσο χρηστών (user interface). Κάθε επίπεδο χρησιμοποιεί τις υπηρεσίες των χαμηλότερών του επιπέδων.



THE (Technische Hogeschool Eindhoven)  
Δομή επιπέδων



Venus  
Δομή επιπέδων



# Δομή Συστήματος

- Βλέπουμε το σύστημα ως μία σειρά από επίπεδα.
- Κάθε επίπεδο εκτελεί ένα σχετικό υποσύνολο των λειτουργιών.
- Κάθε επίπεδο βασίζεται στο επόμενο χαμηλότερο επίπεδο για την εκτέλεση πιο πρωταρχικών λειτουργιών.
- Κάθε πρόβλημα αποσυνθέεται σε έναν αριθμό από πιο διαχειριζόμενα υποπρόβλήματα.



# Ιεραρχία Σχεδίασης Λειτουργικού Συστήματος (1/2)

Επίπεδο	Όνομα	Αντικείμενα	Παραδείγματα Λειτουργιών
13	Κέλυφος	Περιβάλλον προγραμματισμού	Εντολές στη γλώσσα του κελύφους
12	Διεργασίες χρήστη	Διεργασίες χρήστη	Quit, kill, suspend, resume
11	Κατάλογοι	Κατάλογοι	Create, destroy, attach, detach, search, list
10	Συσκευές	Εξωτερικές συσκευές, όπως εκτυπωτές, οθόνες	Open, close, read, write
9	Σύστημα αρχείων	Αρχεία	Create, destroy, open, close, read, write
8	Επικοινωνίες	Διασωληνώσεις (Pipes)	Create, destroy, open, close, read, write
7	Ιδεατή μνήμη	Μπλοκ, σελίδες	Read, write, fetch
6	Τοπική δευτερεύουσα μνήμη	Μπλοκ δεδομένων, Κανάλια δεδομένων	Read, write, allocate, free
5	Πρωταρχικές διεργασίες	Πρωταρχικές διεργασίες, σημαφόρος	Suspend, resume, wait, signal

# Ιεραρχία Σχεδίασης Λειτουργικού Συστήματος (2/2)

## Επίπεδα Υλικού

Επίπεδο	Όνομα	Αντικείμενα	Παραδείγματα Λειτουργιών
4	Διακοπές	Προγράμματα διαχείρισης διακοπών	Invoke, mask, unmask, retry
3	Διαδικασίες	Διαδικασίες, στοίβα κλήσης	Mark stack, call, return
2	Σύνολο εντολών	Στοίβα υπολογισμών, διερμηνέας μικρο-προγραμμάτων	Load, store, add, subtract, branch
1	Ηλεκτρικά κυκλώματα	Καταχωρητές, πύλες, αρτηρίες	Clear, transfer, activate, complement





# Ιδεατές μηχανές

Μία ιδεατή μηχανή (**virtual machine**) αποτελεί τη λογική κατάληξη της δομής επιπέδων. Θεωρεί το υλικό μέρος και το λειτουργικό σύστημα σαν να είναι όλα υλικό μέρος.

Το λειτουργικό σύστημα δημιουργεί τη “ψευδαίσθηση” των πολλαπλών διεργασιών, καθεμία από τις οποίες εκτελείται στο δικό της επεξεργαστή με τη δική του (ιδεατη) μνήμη.

Οι πόροι του Η/Υ δια-μοιράζονται ώστε να δημιουργούνται ιδεατές μηχανές.

- Η χρονο-δρομολόγηση της CPU δημιουργεί την εντύπωση ότι οι χρήστες έχουν τους δικούς τους επεξεργαστές.
- Ο ετεροχρονισμός και το σύστημα αρχείων παρέχουν ιδεατούς αναγνώστες καρτών και ιδεατούς εκτυπωτές γραμμής.
- Ένα τερματικό χρήστη δια-μοιραζόμενου χρόνου αποτελεί την κονσόλα του χειριστή της ιδεατής μηχανής.



# Υλοποίηση Συστήματος (1/2)

Το λειτουργικό σύστημα παλιότερα συντάσσονταν **μόνο** σε γλώσσα assembly. Τώρα μπορούν να γραφούν σε γλώσσες υψηλού επιπέδου, το οποίο επιτρέπει:

- Γρηγορότερη σύνταξη.
- Δημιουργία περισσότερο συμπαγή λειτουργικού συστήματος.
- Ευκολότερη διόρθωση/κατανόηση.

Το λειτουργικό σύστημα που έχει συνταχθεί σε γλώσσα υψηλού επιπέδου είναι πολύ ευκολότερο να εγκατασταθεί στα μηχανήματα.



# Υλοποίηση Συστήματος (2/2)

## Γέννηση Συστήματος (SYSGEN)

- Τα λειτουργικά συστήματα σχεδιάζονται για να εκτελούνται από οποιοδήποτε σύστημα μηχανών.
- Το πρόγραμμα **SYSGEN** παρέχει τις συγκεκριμένες προδιαγραφές του υλικού μέρους του συστήματος.
- Εκκίνηση (booting) του Η/Υ με φόρτωση του πυρήνα (kernel).
- Πρόγραμμα εκκινητή (**Bootstrap program**).
- Κώδικας που φυλάσσεται στη ROM και βρίσκει τον πυρήνα, τον φορτώνει στη μνήμη και ξεκινά την εκτέλεσή του.



# Στόχοι Σχεδίασης Συστήματος

## Σύστημα

Εύκολος σχεδιασμός/  
υλοποίηση/υποστήριξη

Ευέλικτο

Αξιόπιστο

Ελεύθερο λαθών

Ευφύες

## Χρήστης

Ευκολία στη χρήση

Ευκολία στην εκμάθηση

Αξιόπιστο

Ασφαλές

Γρήγορο

## Μηχανισμοί και Πολιτικές

Οι μηχανισμοί καθορίζουν το **ΠΩΣ** θα γίνει κάτι.

Οι πολιτικές αποφασίζουν **ΤΙ** θα γίνει.

Η διάκριση των μηχανισμών από τις πολιτικές αποτελεί μία πολύ σημαντική αρχή του συστήματος. Επιτρέπει τη μέγιστη ευελιξία σε περίπτωση αλλαγής των πολιτικών του συστήματος.



# Χαρακτηριστικά Σύγχρονων Λειτουργικών Συστημάτων (1/5)

## Αρχιτεκτονική μικροπυρήνα

Αναθέτει μόνο λίγες λειτουργίες στον πυρήνα συμπεριλαμβανομένου:

- Χώρου διευθύνσεων.
- Διαδικρασιακής επικοινωνίας (inter-process communication – IPC).
- Βασικής δρομολόγησης.



# Χαρακτηριστικά Σύγχρονων Λειτουργικών Συστημάτων (2/5)

## Πολυνηματική (Multithreading)

Η διεργασία χωρίζεται σε νήματα τα οποία μπορούν να εκτελούνται ταυτόχρονα.

## Νήμα (Thread)

- Διαχωρίσιμο τμήμα της εργασίας.
- Εκτελείται σειριακά και είναι διακοπτόμενο (interruptable).

**Διεργασία** είναι μία συλλογή από ένα ή περισσότερα νήματα.



# Χαρακτηριστικά Σύγχρονων Λειτουργικών Συστημάτων (3/5)

## Συμμετρική Πολυεπεξεργασία (Symmetric multiprocessing, SMP)

- Υπάρχουν πολλοί επεξεργαστές.
- Αυτοί οι επεξεργαστές διαμοιράζονται την ίδια κύρια μνήμη και τις υπηρεσίες εισόδου/εξόδου.
- Όλοι οι επεξεργαστές μπορούν να εκτελούν τις ίδιες λειτουργίες (“συμμετρική”).



# Χαρακτηριστικά Σύγχρονων Λειτουργικών Συστημάτων (4/5)

## Κατανεμημένο Λειτουργικό Σύστημα (Distributed operating systems)

- Παρέχει τη ψευδαίσθηση ενός μοναδικού χώρου για την κύρια μνήμη, ενός μοναδικού χώρου για τη δευτερεύουσα μνήμη.
- Χρησιμοποιείται για ένα κατεναμημένο σύστημα αρχείων (distributed file system).





# Χαρακτηριστικά Σύγχρονων Λειτουργικών Συστημάτων (5/5)

## Αντικειμενοστραφής σχεδίαση (Object-oriented design)

- Χρησιμοποιείται για τη σπονδυλωτή προέκταση ενός μικρού πυρήνα.
- Επιτρέπει στους προγραμματιστές να ρυθμίζουν το λειτουργικό σύστημα χωρίς να διασπάσουν την ακεραιότητα του συστήματος.



# UNIX (1/2)

Αρχή από MULTICS (**MULTI**plexed Information and **C**omputing **S**ervice) - M.I.T., Bell, G.E.

UNICS (**Un**iplexed Information and **C**omputer **S**ervice).

Το Hardware “περιπλείεται” από το λειτουργικό σύστημα.

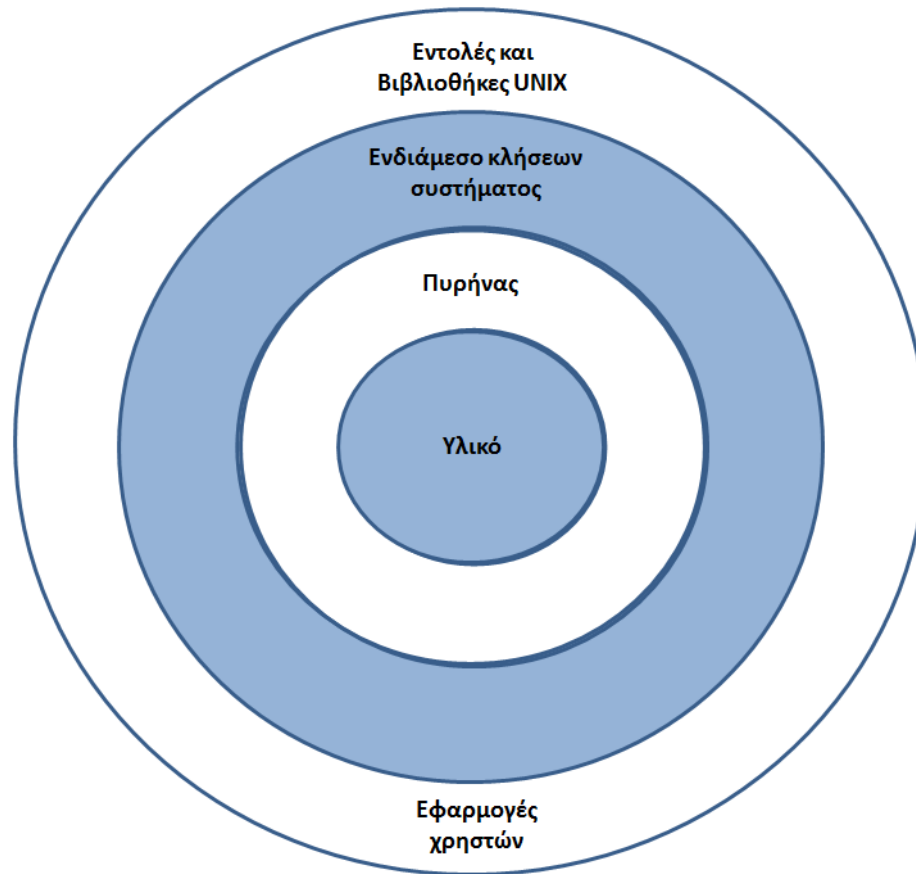
Το λειτουργικό σύστημα είναι “συνώνυμο” του πυρήνα (kernel).

Περιλαμβάνει έναν αριθμό από υπηρεσίες και ενδιάμεσα χρηστών:

- Shell
- C compiler



# UNIX (2/2)



**Γενική αρχιτεκτονική του UNIX**



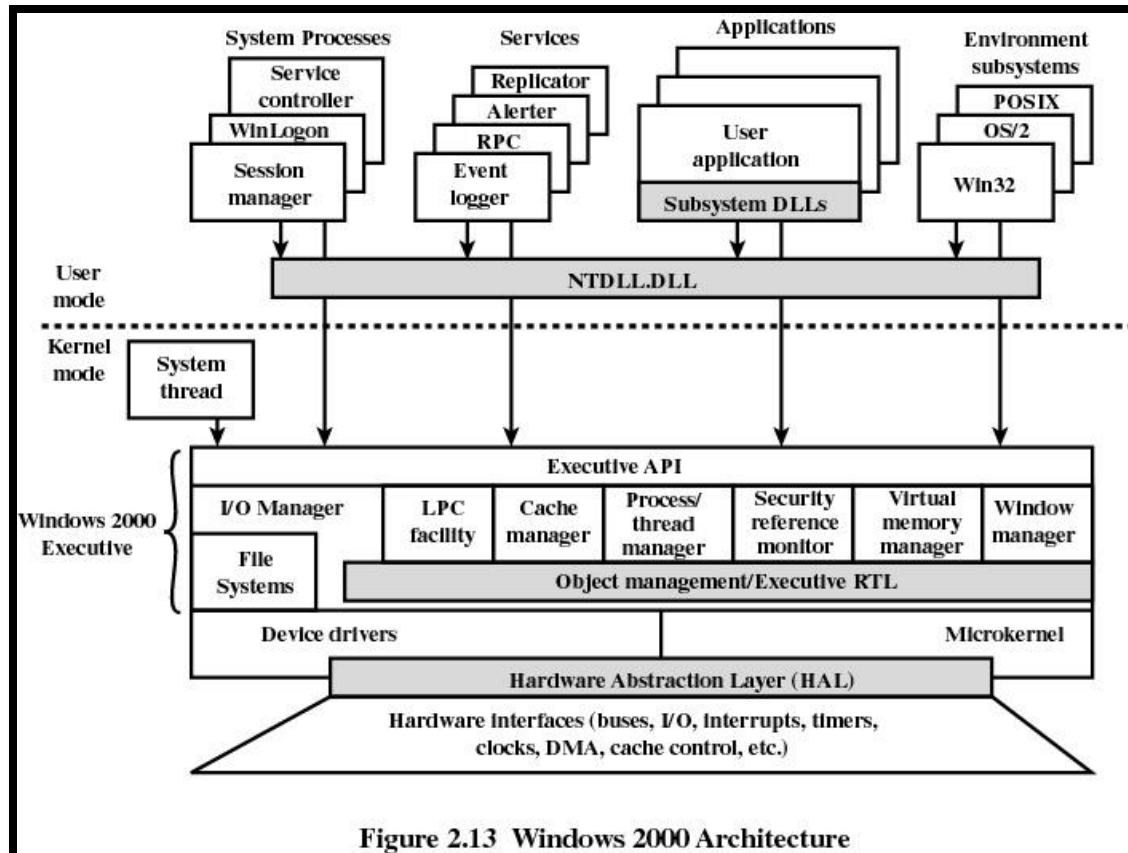
# Windows 2000 Αρχιτεκτονική (1/2)

---

- Σπονδυλωτή δομή για λόγους ευελιξίας.
- Εκτέλεση σε ένα εύρος από διαφορετικές πλατφόρμες.
- Υποστήριξη εφαρμογών που έχουν εγγραφεί για ένα εύρος λειτουργικών συστημάτων.



# Windows 2000 Αρχιτεκτονική (2/2)



Αρχιτεκτονική των Windows 2000 [1]



# JavaOS (1/2)

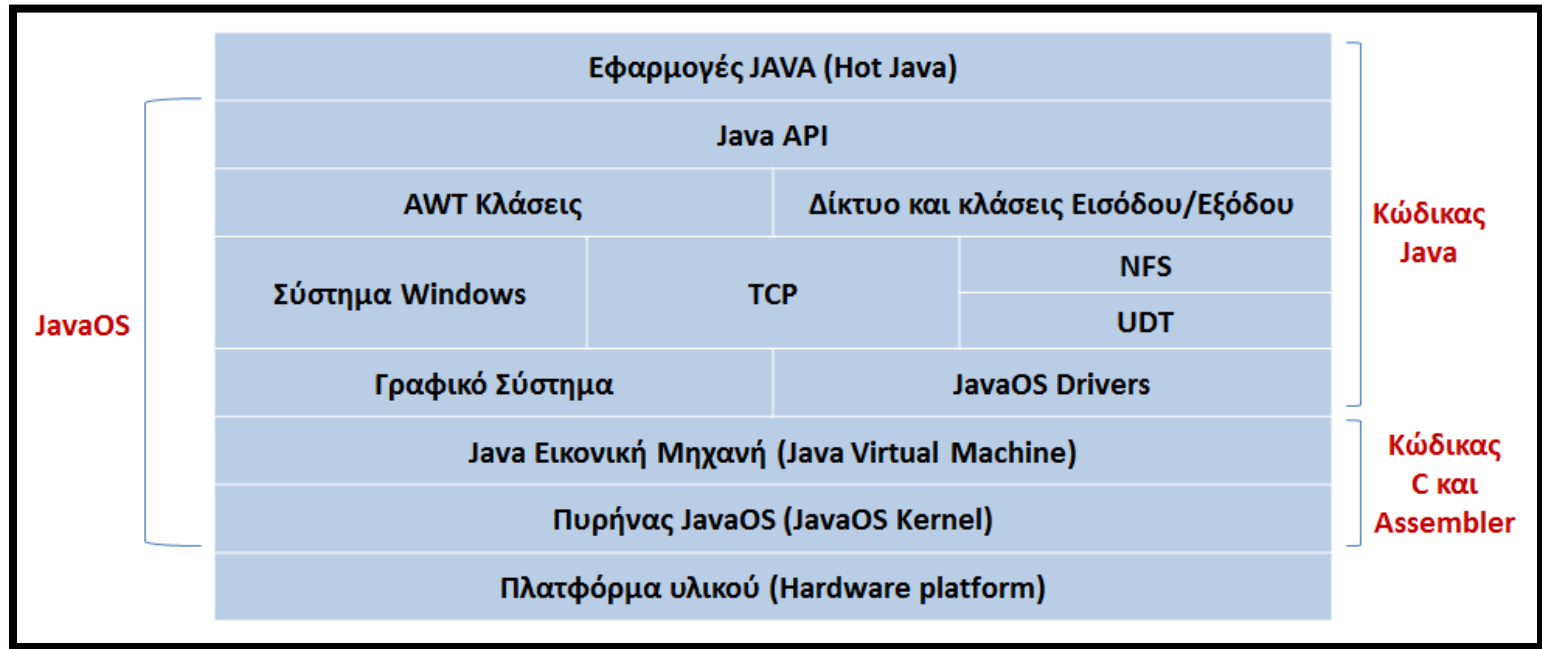
**JavaOS:** ένα λειτουργικό σύστημα που σχεδιάστηκε για να εκτελεί αποδοτικά Java εφαρμογές απευθείας σε διάφορες υπολογιστικές πλατφόρμες στις οποίες δεν υπάρχει εγκατάσταση άλλου λειτουργικού συστήματος.

Έχει συγγραφεί με χρήση assembler της εκάστοτε πλατφόρμας και κώδικα σε C και παρέχει τη λειτουργικότητα που απαιτείται από την ιδεατή μηχανή της **Java Virtual Machine** (JVM).

Το JVM παρέχει το περιβάλλον εκτέλεσης της Java για συστατικά μέρη υψηλότερου επιπέδου (graph system, window system, device drivers, networking και I/O support που μπορούν να συνταχθούν σε απλή Java). Αυτές οι υπηρεσίες αποτελούν το λεγόμενο **JAVA API** (**Application Programming Interface**).



# JavaOS (2/2)



## JavaOS



# Αναφορές

[1]. William Stallings, “Operating systems, Internals and Design Principles”, 5<sup>th</sup> edition, Κεφάλαιο 2, Σελίδα 83.

[2]. Stallings William, “Operating systems: Internal and Design Principles”, 4<sup>th</sup> edition, Publishing as Prentice Hall, 2000.

[3] Alan C. Shaw, "The logical design of operating systems ", Englewood Cliffs, N.J., Prentice-Hall.

[4] Raymond W. Turner, "Operating systems: design and implementation", New York : Macmillan Publ.

## **Εργασίες και ιστότοποι για υποστήριξη αντικειμένου νέων τεχνολογιών (ΕΝΔΕΙΚΤΙΚΑ):**

Wubi Ubuntu Windows installer: <http://www.ubuntu.com/download/desktop/windows-installer>.

Ubuntu 12.04: <http://www.ubuntu.com/>.

Vistual box 4.2: <https://www.virtualbox.org/>

Λειτουργικό σύστημα Windows: <http://windows.microsoft.com/>.







# Τέλος Ενότητας



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

