



ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ ΙΙ (Θ)

Ενότητα 6: ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ ΙΙ

- Νικολαΐδης Αθανάσιος
- Διδάκτορας Ανάπτυξης Τεχνικών Προστασίας Πληροφορίας Εικόνας
- ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΤΕ

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 ΤΕΙ Κεντρικής Μακεδονίας» 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.



Ενότητα 6

ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ Ι

Νικολαΐδης Αθανάσιος
Διδάκτορας Ανάπτυξης Τεχνικών
Προστασίας Πληροφορίας Εικόνας

Περιεχόμενα ενότητας

1. Ευρυζωνική Πρόσβαση ΜΕΤΑΛΛΟΥΡΓΙΚΕΣ
2. Ασύρματη Ευρυζωνική Πρόσβαση
3. Ευρυζωνικά Ασύρματα Δίκτυα
4. Ανάπτυξη της Ευρυζωνικότητας
5. Τεχνικές Προκλήσεις Ευρυζωνική Πρόσβαση
6. Τεχνικές Προκλήσεις- Ασύρματο ραδιοκυματικό κανάλι
7. Τεχνικές Προκλήσεις- Παρεμπόδιση λόγω μεγάλων εμποδίων
– Διακύμανση περιβάλλουσας
8. Τεχνικές Προκλήσεις- Διασυμβολική Παρεμβολή
9. Τεχνικές Προκλήσεις- Διασπορά συχνότητας λόγω κίνησης
10. Τεχνικές Προκλήσεις- Παρεμβολή

Σκοποί ενότητας

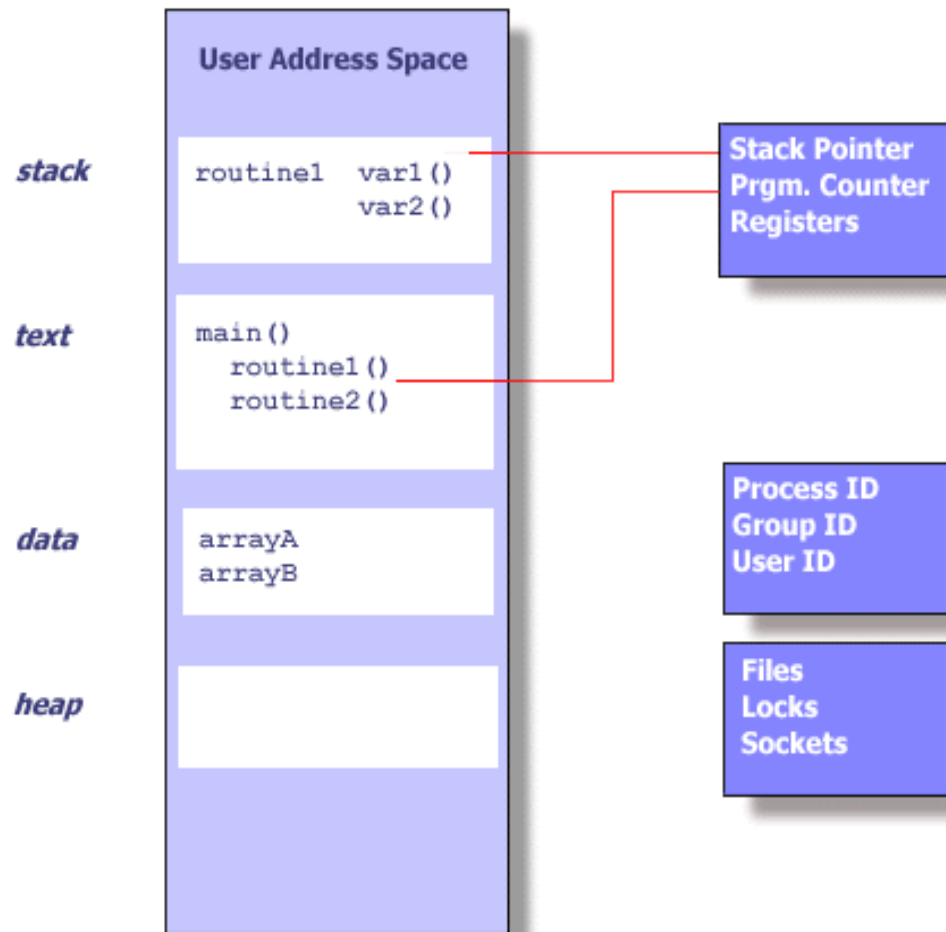
Νήματα

- Οι διεργασίες έχουν τα παρακάτω συστατικά:
 - Ένα χώρο διευθύνσεων
 - Συμφραζόμενα της ΚΜΕ (CPU context)... ή *νήμα ελέγχου* (thread of control)
- Σε πολυεπεξεργαστικά συστήματα θα είχε νόημα για μια διεργασία να έχει πολλά νήματα ελέγχου
 - Η κλήση `fork` ενός νήματος παράγει νέο νήμα και όχι μνήμη
 - Πολλαπλά νήματα ελέγχου θα μπορούσαν να τρέχουν στον ίδιο χώρο μνήμης ακόμη και σε μονοεπεξεργαστικά συστήματα

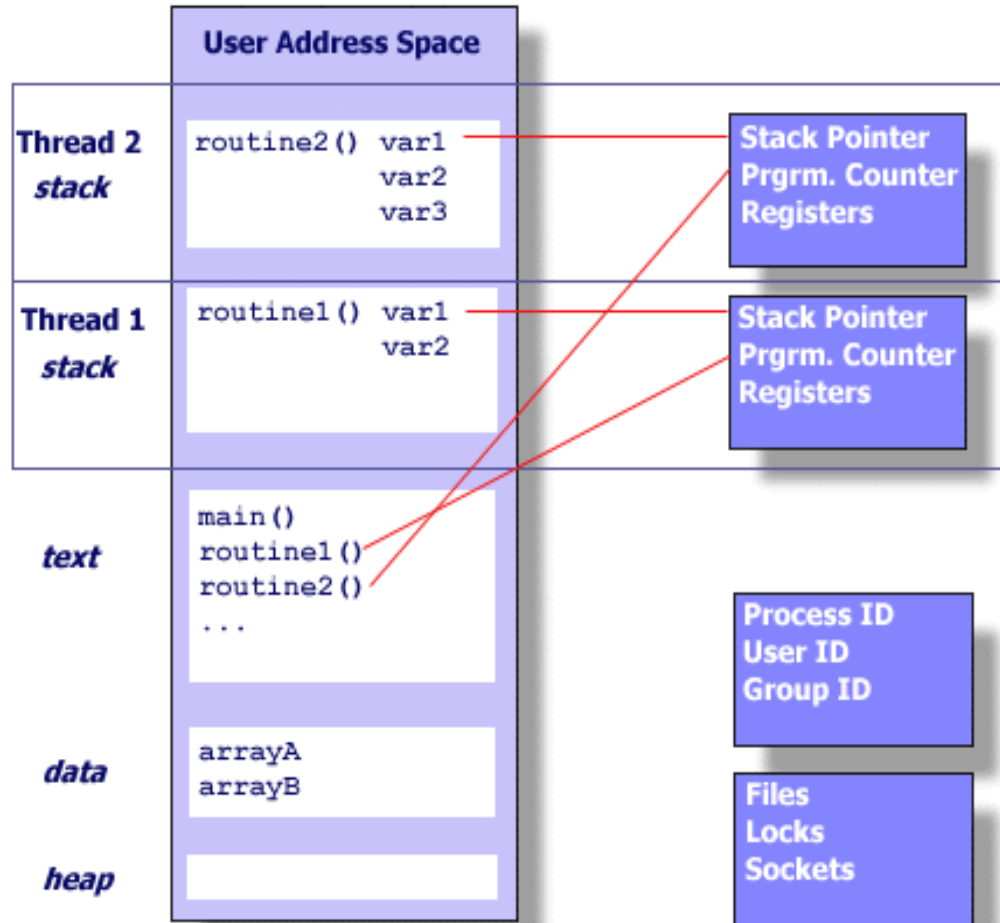
Νήματα

- Κάθε νήμα μοιράζεται με μηδέν ή περισσότερα νήματα το χώρο διευθύνσεων μιας διεργασίας
- Τα νήματα έχουν το δικό τους:
 - PC, SP, κατάσταση καταχωρητών, στοίβα
- Μπορούμε να δούμε μια παραδοσιακή διεργασία ως ένα χώρο διευθύνσεων μνήμης με ένα μοναδικό νήμα

Μοναδικό νήμα σε μία διεργασία



Πολλαπλά νήματα σε ένα χώρο διευθύνσεων



Τί είναι ένα νήμα;

- Ένα νήμα εκτελεί ένα ρεύμα από εντολές
 - Είναι μια αφαίρεση για τη ροή ελέγχου
- Πρακτικά, είναι μια κατάσταση ΚΜΕ και μια στοίβα
 - Του ανατίθεται μια ΚΜΕ από τον χρονοπρογραμματιστή
 - Εκτελείται στα συμφραζόμενα ενός χώρου διευθύνσεων

Περίληψη της ιδιωτικής, ανά νήμα, κατάστασης

- Τι ορίζει την κατάσταση μιας συγκεκριμένης ροής ελέγχου σε ένα εκτελούμενο πρόγραμμα:
 - Στοίβα (τοπικές παράμετροι)
 - Δείκτης στοίβας
 - Καταχωρητές
 - Ιδιότητες χρονοπρογραμματισμού (π.χ. προτεραιότητα)

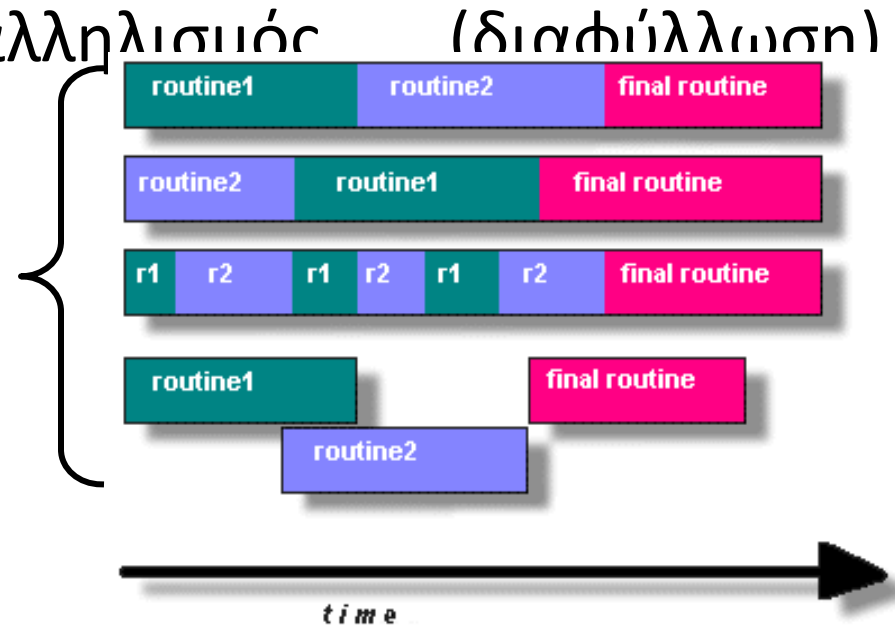
Διαμοιραζόμενη κατάσταση μεταξύ νημάτων

- Τι σχετίζεται με το στιγμιότυπο ενός εκτελούμενου προγράμματος:
 - UID, GID, PID
 - Χώρος διευθύνσεων
 - Κώδικας
 - Δεδομένα (εκτός στοίβας σφαιρικές μεταβλητές)
 - Σωρός (δυναμικά δεδομένα)
 - Ανοιχτά αρχεία, υποδοχές, κλειδώματα
- Αλλαγές που γίνονται από ένα νήμα στη διαμοιραζόμενη κατάσταση είναι ορατές από τα άλλα (απαιτείται συγχρονισμός)

Πώς προγραμματίζουμε με χρήση νημάτων;

- Χωρίζουμε το πρόγραμμα σε ρουτίνες για να εκτελούνται παράλληλα
 - Αληθινός ή ψεύδο-παραλληλισμός (διαφύλλιση)

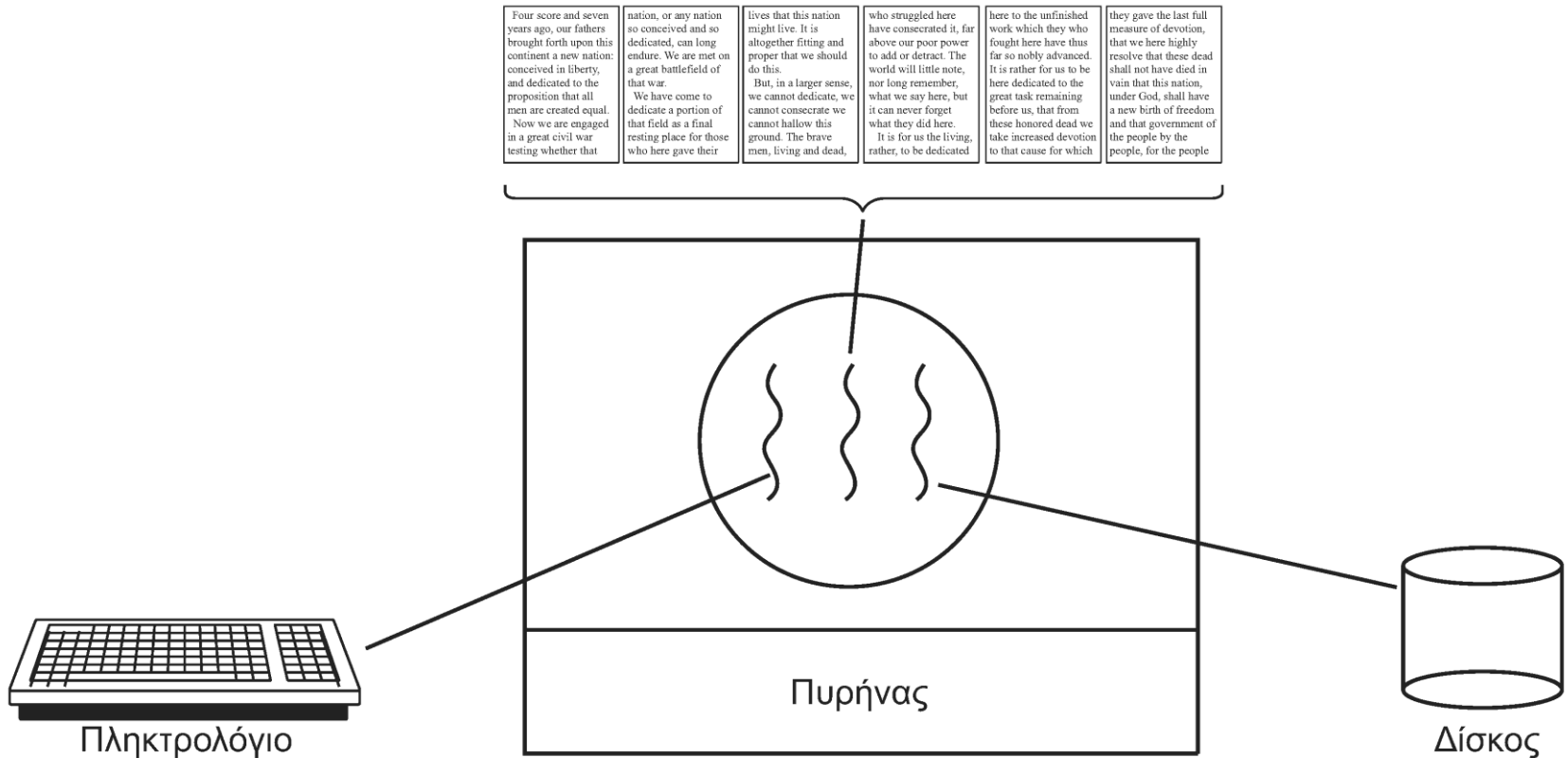
Εναλλακτικές
στρατηγικές για εκτέλεση
πολλαπλών ρουτινών



Γιατί προγραμματισμός με νήματα;

- Χρησιμοποιούνται πολλαπλές ΚΜΕ παράλληλα
- Μικρό κόστος δημιουργίας/καταστροφής νημάτων
- Επικάλυψη υπολογισμών και μπλοκαρίσματος σε μονή ΚΜΕ
 - Μπλοκάρισμα λόγω Ε/Ε
 - Υπολογισμοί και επικοινωνία
- Ταχύτερη εκτέλεση εφαρμογής (όταν υπάρχει και πολλή χρήση Ε/Ε)

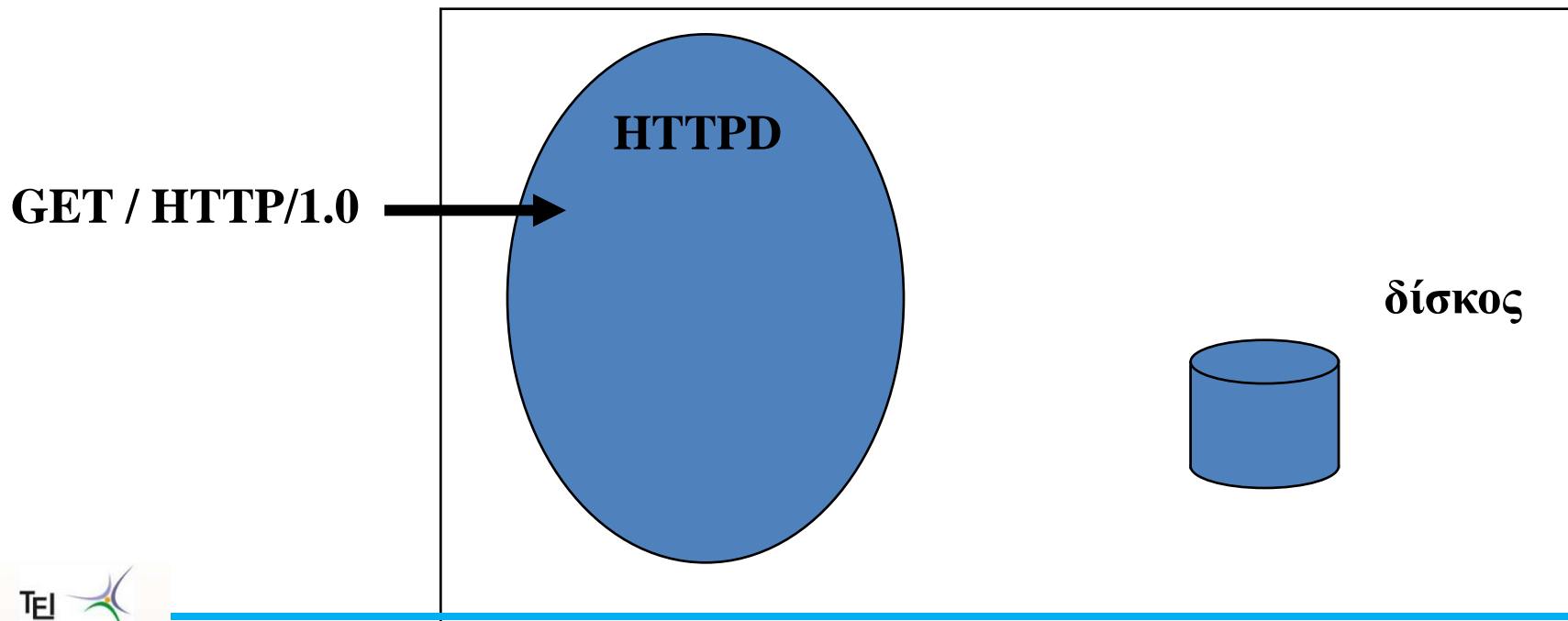
Χρήση νημάτων



Ένας επεξεργαστής κειμένου με τρία νήματα

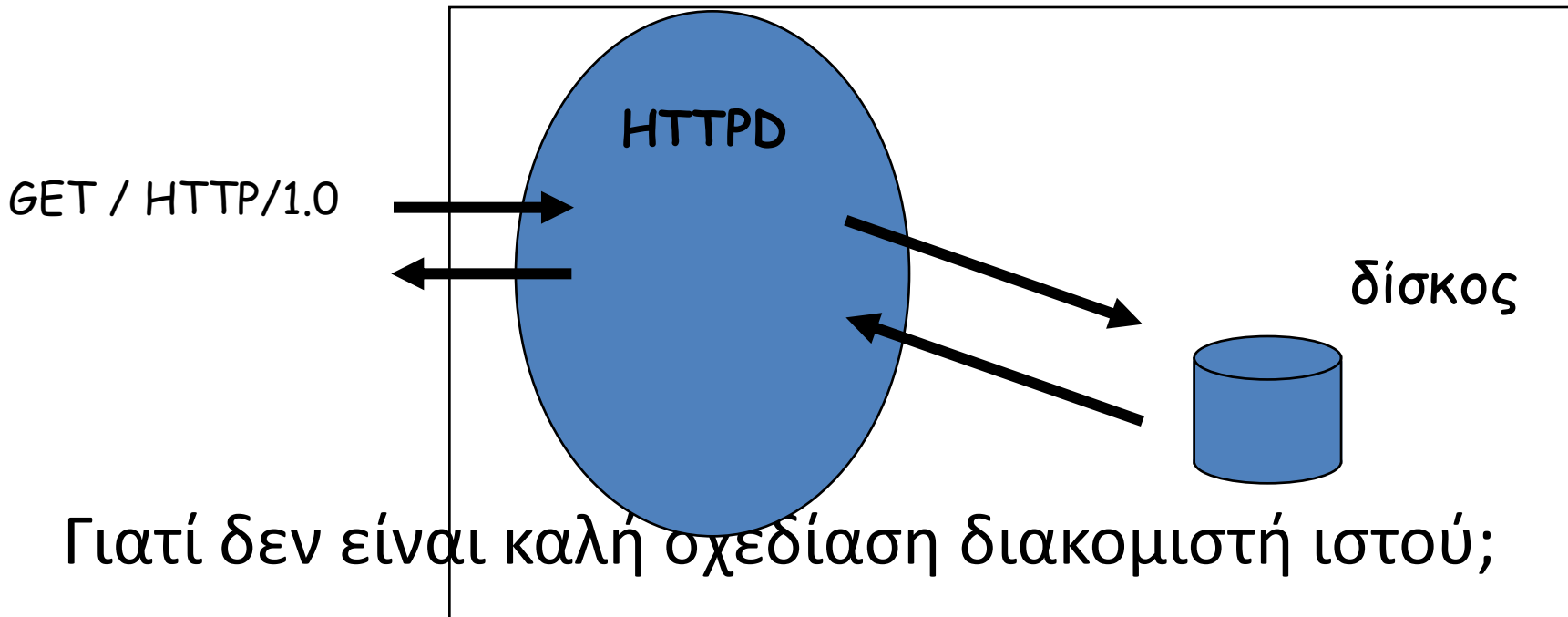
Διεργασίες εναντίον νημάτων - παράδειγμα

- Μια διεργασία παγκόσμιου ιστού



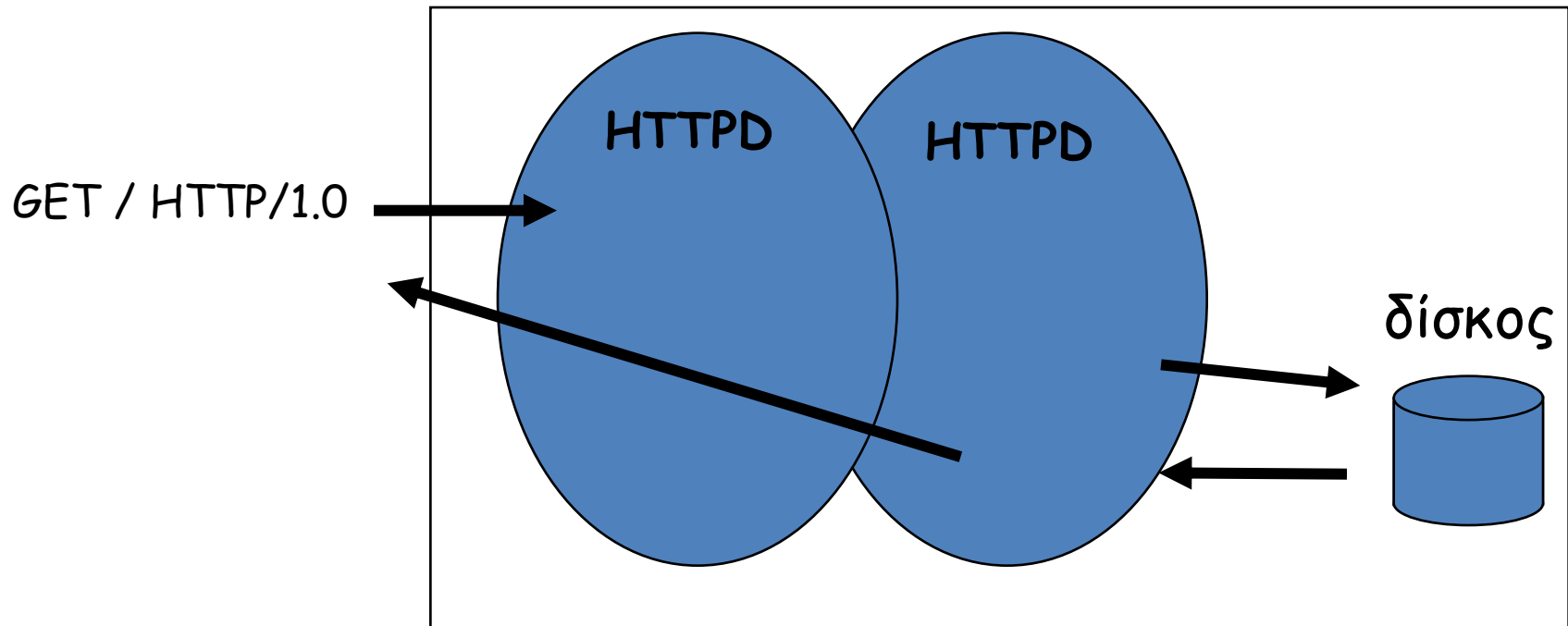
Διεργασίες εναντίον νημάτων - παράδειγμα

- Μια διεργασία παγκόσμιου ιστού



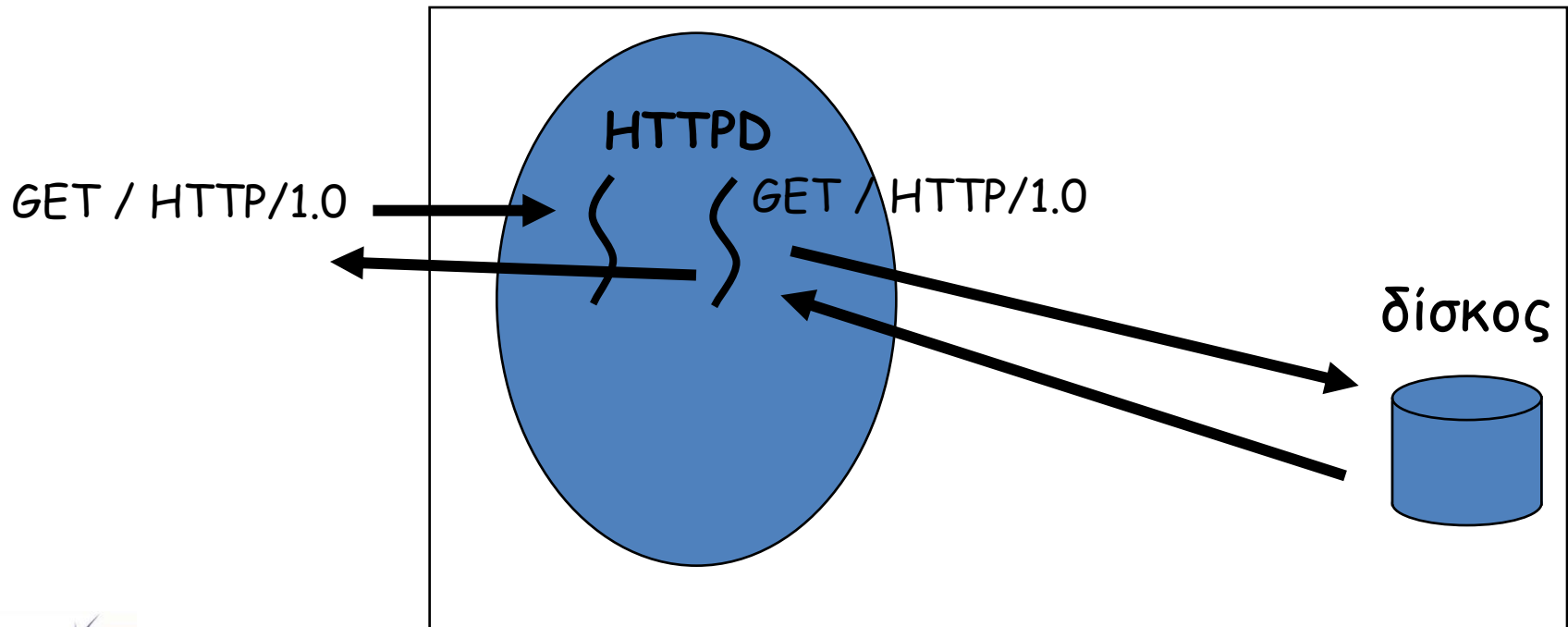
Διεργασίες εναντίον νημάτων - παράδειγμα

- Μια διεργασία παγκόσμιου ιστού



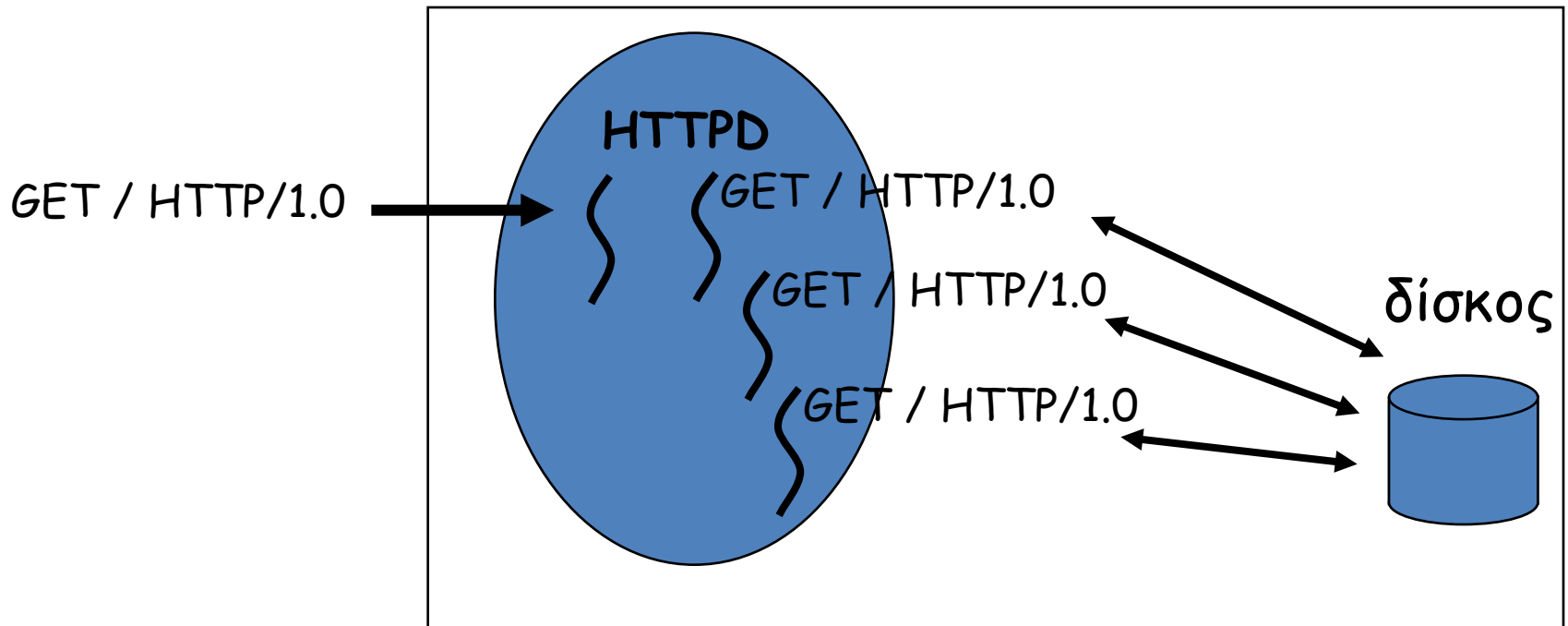
Διεργασίες εναντίον νημάτων - παράδειγμα

- Μια διεργασία παγκόσμιου ιστού

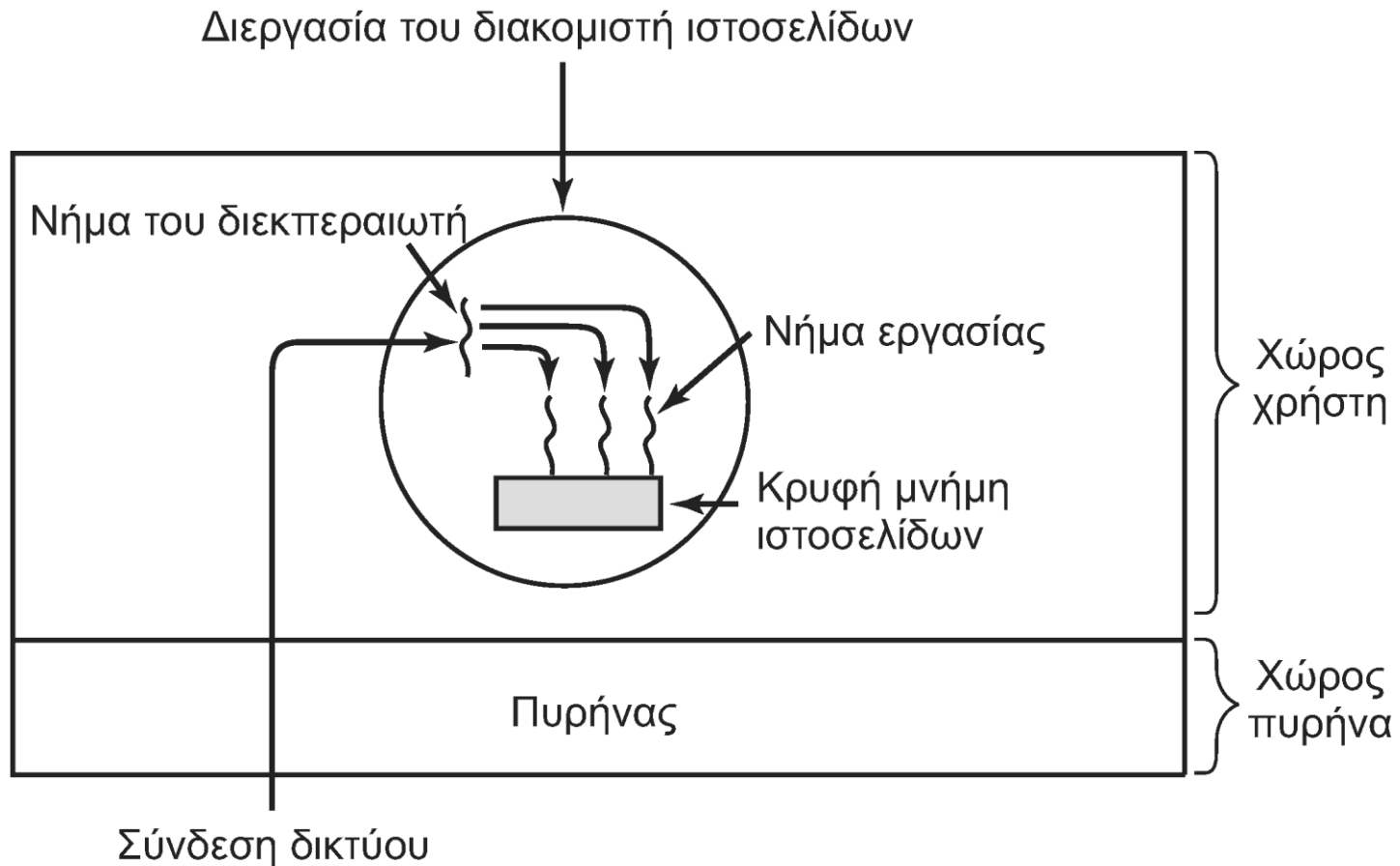


Διεργασίες εναντίον νημάτων - παράδειγμα

- Μια διεργασία παγκόσμιου ιστού



Νήματα σε έναν διακομιστή ιστού



Ένας πολυνηματικός διακομιστής ιστού

Χρήση νημάτων

```
while (TRUE) {  
  get_next_request(&buf);  
  handoff_work(&buf);  
}
```

(a)

```
while (TRUE) {  
  wait_for_work(&buf)  
  look_for_page_in_cache(&buf, &page);  
  if (page_not_in_cache(&page)  
      read_page_from_disk(&buf, &page);  
  return_page(&page);  
}
```

(b)

- Χονδρικό περίγραμμα του κώδικα της προηγούμενης διαφάνειας

(a) Νήμα διεκπεραιωτή

(b) Νήμα εργασίας

Επιλογές δόμησης συστήματος

Μοντέλο	Γνωρίσματα
Νήματα	Παραλληλία, κλήσεις συστήματος που μπλοκάρουν (ανασταλτικές)
Διεργασία με ένα νήμα	Δεν υπάρχει παραλληλία, κλήσεις συστήματος που μπλοκάρουν (ανασταλτικές)
Μηχανή πεπερασμένων καταστάσεων	Παραλληλία, κλήσεις συστήματος που δεν μπλοκάρουν (μη ανασταλτικές), διακοπές

Τρεις τρόποι να δομηθεί ένας διακομιστής

Κοινά μοντέλα προγραμματισμού νημάτων

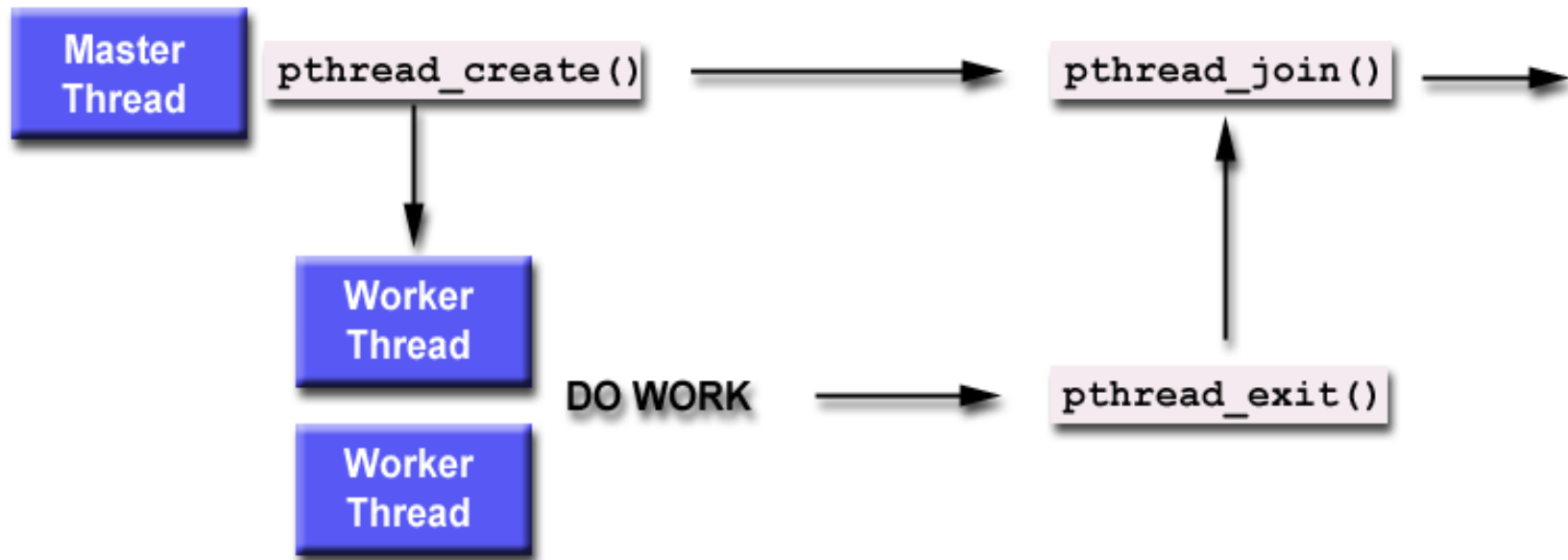
- Διαχειριστή/εργάτη
 - Ο διαχειριστής χειρίζεται την Ε/Ε και αναθέτει δουλειά στους εργάτες
 - Οι εργάτες μπορούν να δημιουργούνται δυναμικά ή να παρέχονται από μια δεξαμενή νημάτων
- Διοχέτευση
 - Κάθε νήμα χειρίζεται διαφορετικό στάδιο μιας γραμμής συναρμολόγησης
 - Τα νήματα παραδίδουν δουλειά το ένα στο άλλο σε μια σχέση παραγωγού-καταναλωτή

Πώς μοιάζει ένα τυπικό πολυνηματικό API;

- Στάνταρ νήματα POSIX (Pthreads)
- Το πρώτο νήμα υπάρχει στη `main()`, τυπικά δημιουργεί τα άλλα

Κλήση νήματος	Περιγραφή
<code>Pthread_create</code>	Δημιουργία νέου νήματος
<code>Pthread_exit</code>	Τερματισμός του καλούντος νήματος
<code>Pthread_join</code>	Αναμονή για έξοδο συγκεκριμένου νήματος
<code>Pthread_yield</code>	Αποδέσμευση της CPU ώστε να μπορεί να ξεκινήσει ένα άλλο νήμα
<code>Pthread_attr_init</code>	Δημιουργία και απόδοση αρχικών τιμών στη δομή χαρακτηριστικών ενός νήματος
<code>Pthread_attr_destroy</code>	Διαγραφή της δομής χαρακτηριστικών ενός νήματος

Χρησιμοποιώντας τις συναρτήσεις create, join και exit



Παράδειγμα με Pthreads

```
#include <pthread.h>
#include <stdio.h>
#define NUM_THREADS 5

void *PrintHello(void *threadid)
{
    printf("\n%d: Hello World!\n", threadid);
    pthread_exit(NULL);
}

int main (int argc, char *argv[])
{
    pthread_t threads[NUM_THREADS];
    int rc, t;
    for(t=0; t<NUM_THREADS; t++)
    {
        printf("Creating thread %d\n", t);
        rc = pthread_create(&threads[t], NULL, PrintHello, (void *)t);
        if (rc)
        {
            printf("ERROR; return code from pthread_create() is %d\n", rc);
            exit(-1);
        }
    }
    pthread_exit(NULL);
}
```

Έξοδος προγράμματος

```
Creating thread 0
Creating thread 1
0: Hello World!
1: Hello World!
Creating thread 2
Creating thread 3
2: Hello World!
3: Hello World!
Creating thread 4
4: Hello World!
```

Υπέρ και κατά των νημάτων

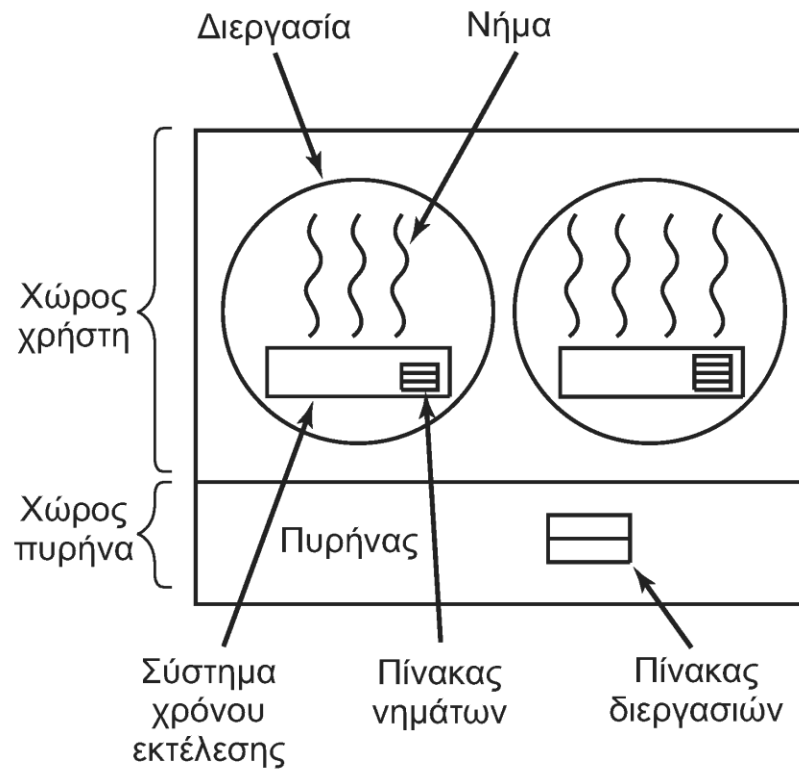
- Υπέρ:
 - Επικάλυψη E/E με υπολογισμούς
 - Φθηνότερες μεταγωγές περιβάλλοντος
 - Καλύτερη αντιστοίχιση σε πολυεπεξεργαστές διαμοιραζόμενης μνήμης
- Κατά:
 - Πιθανές αλληλεπιδράσεις νημάτων
 - Πολυπλοκότητα αποσφαλμάτωσης
 - Πολυπλοκότητα πολυνηματικού προγραμματισμού
 - Προς τα πίσω συμβατότητα με υπάρχοντα κώδικα

Νήματα επιπέδου χρήστη

- Πολλαπλοί ιδεατοί απαριθμητές προγράμματος πάνω από έναν πραγματικό με προνομιούχο ή μή κώδικα
 - Τα νήματα μπορούν να υλοποιηθούν στο επίπεδο του Λ.Σ. ή του χρήστη
- Υλοποιήσεις επιπέδου χρήστη:
 - Ο χρονοπρογραμματιστής τρέχει ως κώδικας χρήστη (βιβλιοθήκη νημάτων)
 - Διαχειρίζεται συμφραζόμενα νημάτων στο χώρο χρήστη
 - Το υποκείμενο Λ.Σ. βλέπει μόνο μια παραδοσιακή διεργασία από πάνω.

Νήματα επιπέδου χρήστη

Ο κώδικας μεταγωγής μεταξύ νημάτων είναι στο χώρο χρήστη

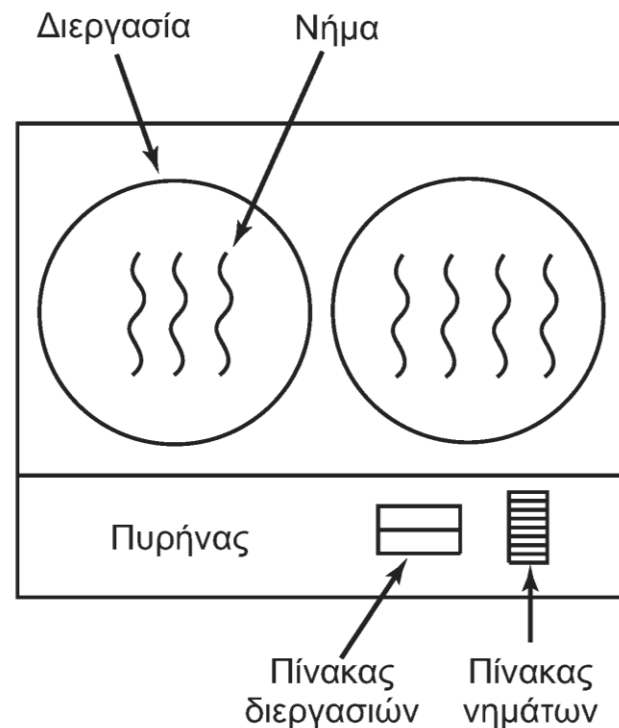


Νήματα επιπέδου χρήστη

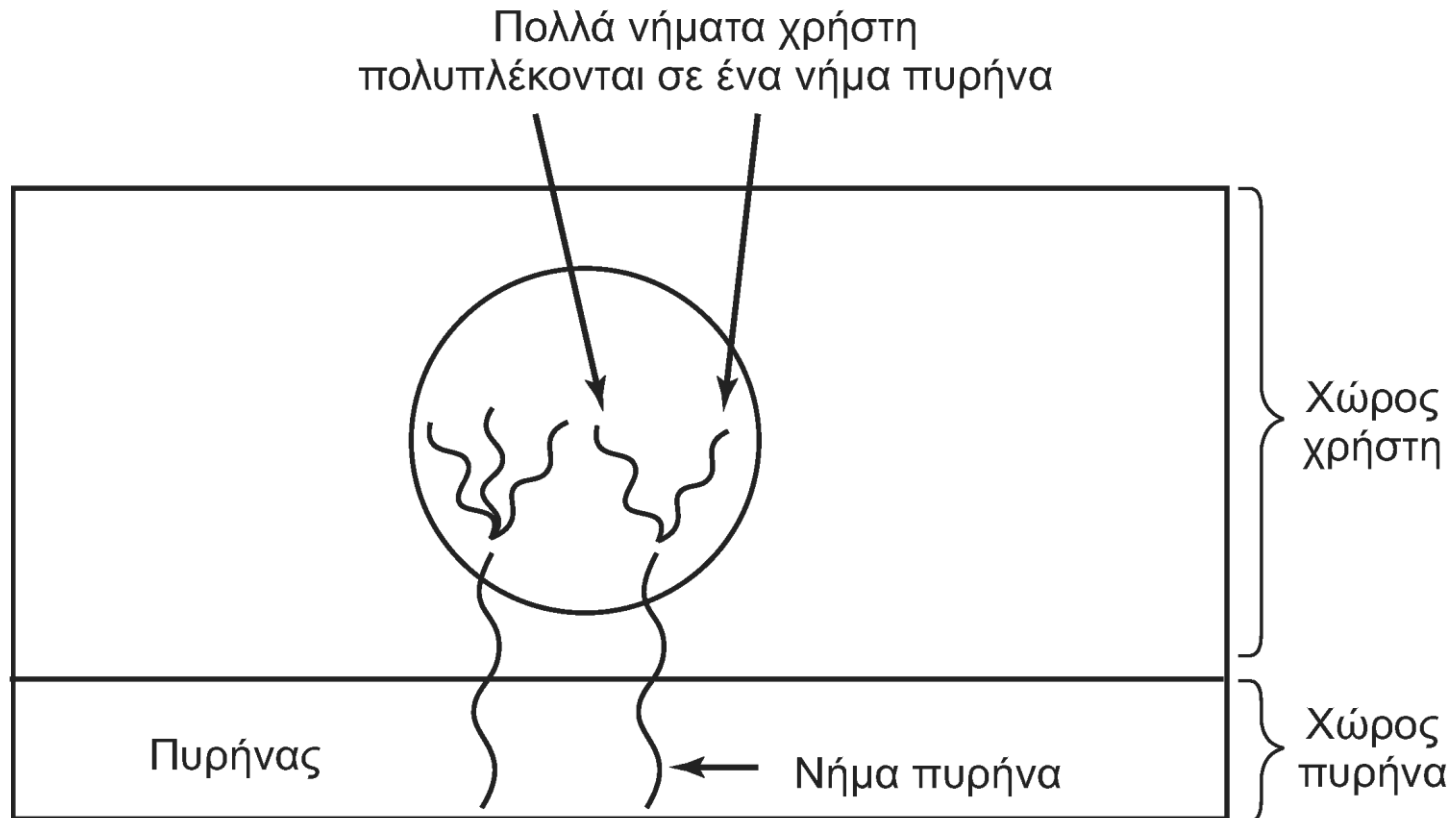
- Πλεονεκτήματα:
 - Οικονομική μεταγωγή περιβάλλοντος μεταξύ νημάτων της ίδιας διεργασίας
 - Κλήση διαδικασίας και όχι συστήματος
 - Πολιτική χρονοδρομολόγησης προγραμματιζόμενη από το χρήστη
- Μειονεκτήματα:
 - Πώς χειριζόμαστε τις ανασταλτικές κλήσεις συστήματος;
 - Πώς επικαλύπτουμε υπολογισμούς και E/E;

Νήματα επιπέδου πυρήνα

Ο κώδικας μεταγωγής μεταξύ νημάτων είναι στον πυρήνα



Υβριδική υλοποίηση



Τέλος Ενότητας

