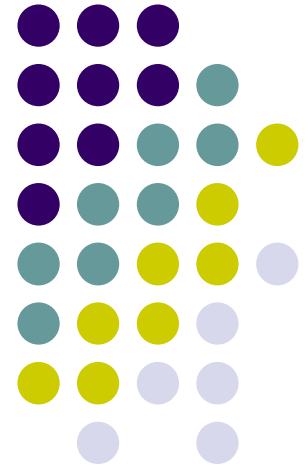




ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΥΠΟΛΟΓΙΣΤΙΚΕΣ ΤΕΧΝΙΚΕΣ ΓΙΑ ΣΥΣΤΗΜΑΤΑ ΜΕΤΑΔΟΣΗΣ ΠΛΗΡΟΦΟΡΙΑΣ



**Τεχνολογίες κατανεμημένων αντικειμένων
για κατανεμημένη παράλληλη επεξεργασία**

Web Site: <https://helios.ntua.gr/course/view.php?id=973>
<http://mfol.ece.ntua.gr/computational-techniques-for-information-transmission-systems/>



Τι είναι κατανεμημένο σύστημα;

- Ένα κατανεμημένο σύστημα είναι μια συλλογή ανεξάρτητων υπολογιστών που εμφανίζεται στους χρήστες της ως ένα ενιαίο σύστημα (Andrew Tanenbaum)
- Ένα κατανεμημένο σύστημα είναι ένα σύστημα του οποίου οι συνιστώσες βρίσκονται σε δικτυωμένους υπολογιστές και επικοινωνούν και συντονίζουν τις ενέργειές τους μόνο με τη διαβίβαση μηνυμάτων.



Χαρακτηριστικά κατανεμημένων συστημάτων

- Ετερογενείς υπολογιστές (επεξεργαστής, μνήμη, OS κλπ)
- Οι λεπτομέρειες της επικοινωνίας δεν είναι ορατές στο χρήστη (δηλ. δεν γίνονται αντιληπτές από τον χρήστη)
- Η αλληλεπίδραση του χρήστη με το σύστημα και με τις εφαρμογές είναι η ίδια, ανεξάρτητα από την τοποθεσία
- Κλιμάκωση
- Διαθεσιμότητα



Πλεονεκτήματα

- Πλήρης αξιοποίηση των δυνατοτήτων ενός υπολογιστή
- Περισσότερη υπολογιστική ισχύς
- Η κατανομή του φόρτου εργασίας μειώνει την εμφάνιση προβλημάτων
 - Κάθε υπολογιστής είναι ανεξάρτητος και τα προβλήματά του δεν επηρεάζουν τους υπόλοιπους υπολογιστές
 - Αποφυγή SPF (Single Point of Failure)



Μειονεκτήματα

- Όλοι οι υπολογιστές στο Internet μπορεί να είναι ισχυροί εφόσον συνδυαστούν, αλλά υπάρχει η ανάγκη κάποιας οντότητας που θα τους “συνδυάζει” και θα τους “συντονίζει”
- Η επικοινωνία με συμμετέχοντες υπολογιστές μπορεί να είναι αργή
- Τα δεδομένα θα πρέπει να καταλαμβάνουν όσο το δυνατόν λιγότερο bandwidth και χώρο στο δίσκο
- Τα δεδομένα θα πρέπει να είναι ανεξάρτητα μεταξύ τους



Μειονεκτήματα

- Ορισμένες υπολογιστικές διεργασίες εξαρτώνται από πολλές μεταβλητές
 - τσιπ επεξεργαστή του υπολογιστή θα πρέπει να προσφύγει σε όλους τους άλλους υπολογιστές
 - Έλλειψη αποτελεσματικής επικοινωνίας
 - αρνητικά αποτελέσματα για την ταχύτητα της διαδικασίας υπολογισμού
- Σοβαρά ζητήματα ασφάλειας
 - για τον οργανισμό που χρησιμοποιεί κατακεκομμένα συστήματα
 - για τον ίδιο τον χρήστη, όταν ο ίδιος ο υπολογιστής του εκτελεί μέρος μίας συγκεκριμένης εργασίας ενός κατακεκομμένου δικτύου



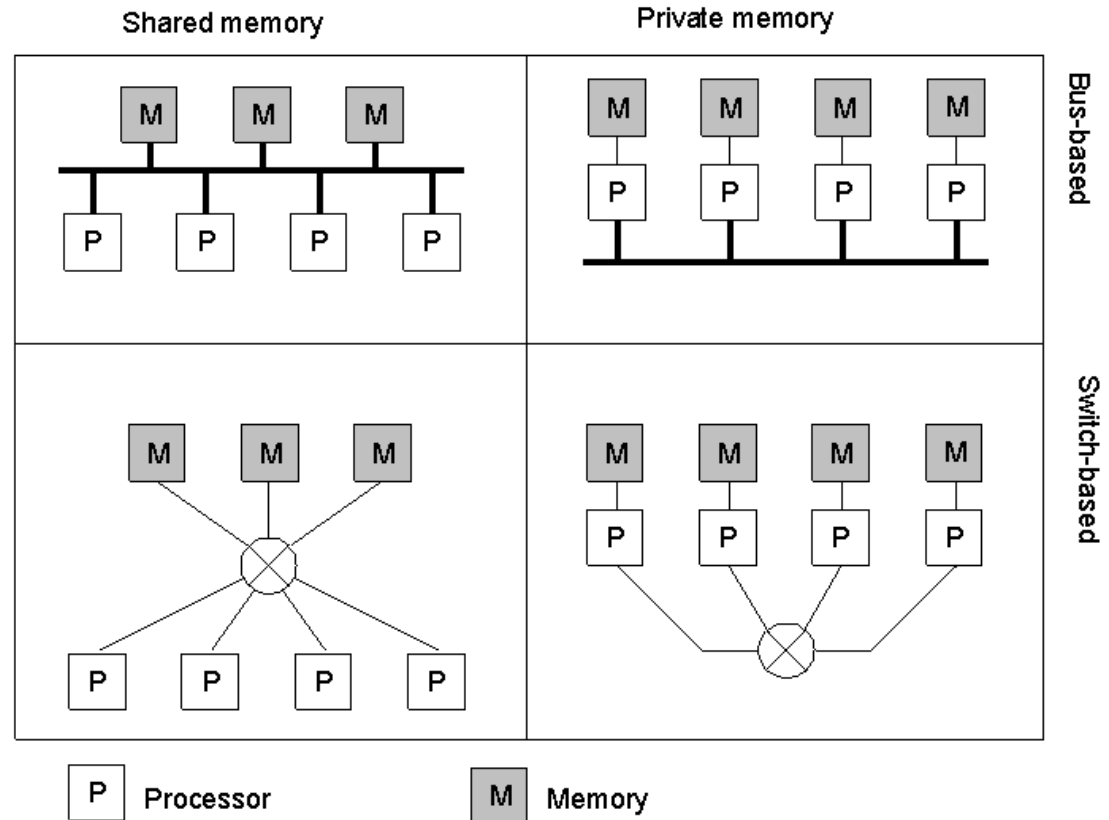
Μειονεκτήματα

- Στην περίπτωση ενός τοπικού κατανεμημένου δικτύου - όπου οι υπολογιστές βρίσκονται μέσα σε μια συγκεκριμένη περιοχή, όπως ένα δωμάτιο - το κόστος της συντήρησης μπορεί να ανέλθει στα ύψη
 - Κάθε υπολογιστής έχει τα δικά του προβλήματα και σφάλματα
 - Αν αυτοί οι υπολογιστές δεν χρησιμοποιούνται ταυτόχρονα και για κάτι άλλο, (π.χ. ως σταθμοί εργασίας σε μια βιβλιοθήκη) το κατανεμημένο σύστημα δεν είναι αποτελεσματικό από πλευράς κόστους
 - Αλλά και πάλι...Trade off: Μπορούμε να εμπιστευτούμε τον τυχαίο χρήστη του υπολογιστή π.χ. μίας βιβλιοθήκης?!?!???



Ζητήματα Υλικού

- Εισαγωγή στον τρόπο οργάνωσης κατανεμημένων συστημάτων
 - Πώς διασυνδέονται
 - Πώς επικοινωνούν





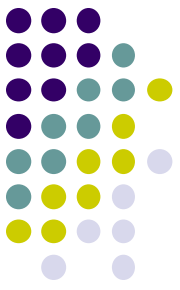
Μοιραζόμενη & Ιδιωτική Μνήμη

- Πολυεπεξεργαστές -- Multiprocessors
 - Ένας φυσικός χώρος διευθύνσεων μοιραζόμενος από όλες τις CPUs
 - Η CPU A γράφει 37 στη διεύθυνση 1000
 - Η CPU B διαβάζει από τη διεύθυνση 1000 και παίρνει 37
 - π.χ. πολλαπλοί επεξεργαστές σε μια πλακέτα με μοιραζόμενη μνήμη
- Πολύ-υπολογιστές -- Multicomputers
 - Κάθε μηχανήμα έχει τη δική του ιδιωτική μνήμη
 - Η CPU A γράφει 37 στη δική της διεύθυνση 1000
 - Η CPU B διαβάζει από τη δική της διεύθυνση 1000 και παίρνει οτιδήποτε τυχαίνει να βρίσκεται εκεί. Δεν επηρεάζεται από το τι έγραψε η άλλη CPU
 - Για παράδειγμα PCs που συνδέονται μέσω του ενός δικτύου



Bus-based & Switch-based

- Bus architecture διασύνδεσης δικτύου
 - Ένα και μόνο δίκτυο, backplane, bus, καλώδιο ή άλλο μέσο που συνδέει όλα τα μηχανήματα
 - Παράδειγμα
 - καλωδιακή τηλεόραση
- Switched architecture
 - Ξεχωριστά καλώδια από μηχάνημα σε μηχάνημα με χρήση πολλών διαφορετικών μοντέλων καλωδίωσης
 - Διακίνηση μηνυμάτων μέσω καλωδίων μέσω μιας έμμεσης απόφασης switching σε κάθε βήμα για τη δρομολόγηση του μηνύματος μέσω ενός από τα εξερχόμενα καλώδια
 - π.χ. παγκόσμιο δημόσιο τηλεφωνικό σύστημα



Συστήματα ετερογενών πολυεπεξεργαστών

- Τα περισσότερα κατανεμημένα συστήματα είναι τέτοιου τύπου
- Οι υπολογιστές είναι ετερογενείς ως προς τον τύπο του επεξεργαστή, το μέγεθος της μνήμης, I/O bandwidth, κτλ.
- Δίκτυα διασύνδεσης μπορεί να είναι ετερογενή
- Πολλοί & μεγάλης-κλίμακας ετερογενείς πολύ-υπολογιστές παρουσιάζουν έλλειψη μιας γενικής εικόνας συστήματος
 - δεν είναι δυνατή η υπόθεση ότι είναι διαθέσιμη η ίδια επίδοση ή οι ίδιες υπηρεσίες παντού
- Συνεπώς ανάγκη για σύνθετο/εξεζητημένο λογισμικό
 - Απόκρυψη από τους προγραμματιστές εφαρμογών του τι συμβαίνει στο επίπεδο του υλικού (διαφάνεια -- transparency)

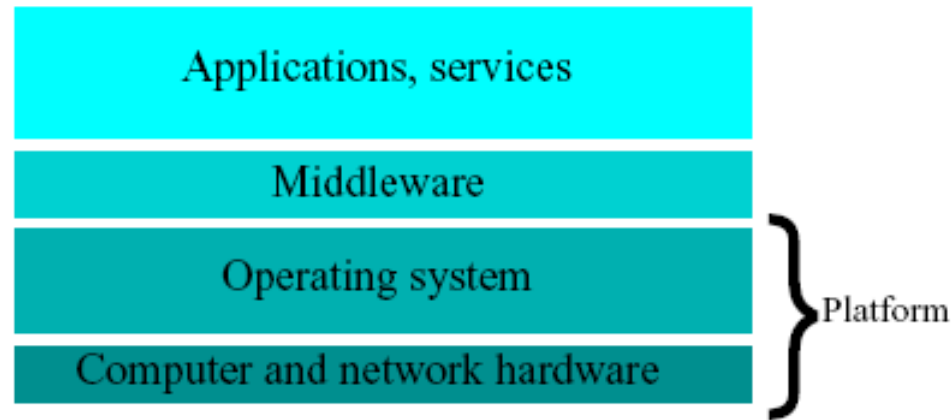


Ζητήματα Λογισμικού

- Λογισμικό κατανεμημένων συστημάτων
 - λειτουργεί σαν διαχειριστής πόρων για το υποκείμενο υλικό
 - κρύβει την πολυπλοκότητα και την ετερογένεια του υποκείμενου υλικού



Στρώματα υπηρεσιών λογισμικού και υλικού στα Κατανεμημένα Συστήματα



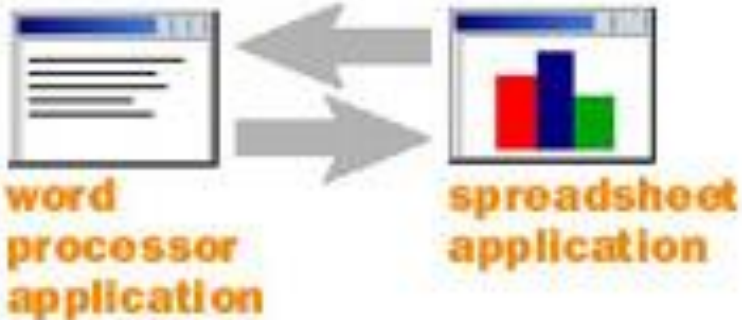
- **Middleware**

- Όρος που αναφέρεται σε ένα στρώμα λογισμικού που βρίσκεται μεταξύ του δικτύου/πλατφόρμας και της εφαρμογής
- Στόχος του middleware είναι να αποκρύψει την ετερογένεια διαφορετικών υπολογιστών, λειτουργικών συστημάτων και εφαρμογών

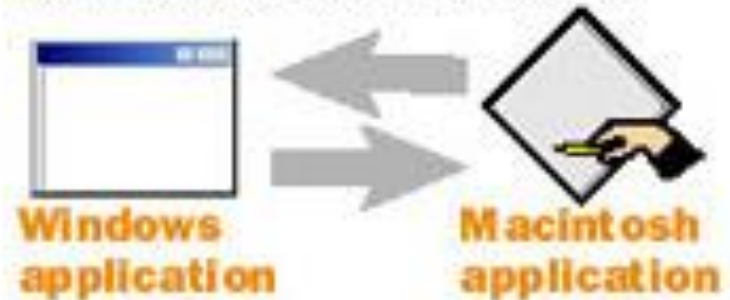


Middleware

Middleware allows two applications to communicate with each other through a standard language.



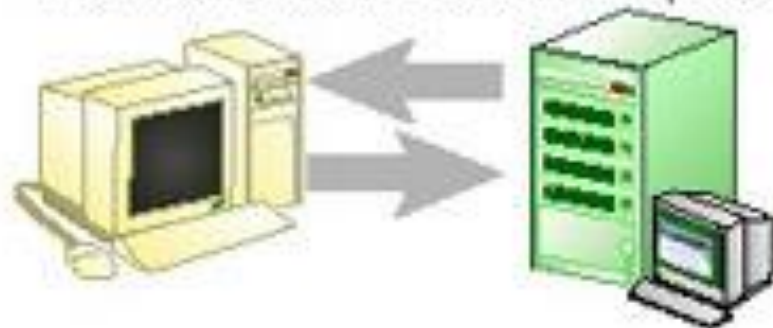
Middleware allows applications across different platforms to communicate with each other.

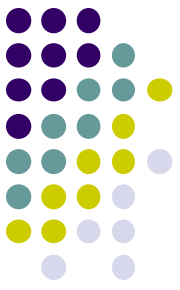


Middleware also allows legacy applications and modern applications to communicate.



Middleware takes care of transactions between servers, data conversion, authentication, and communications between computers.





Μοντέλα Middleware

- Απομακρυσμένη κλήση μεθόδων (Remote Procedure Calls-RPC)
 - Επιτρέπει σε μια διεργασία να καλέσει διαφανώς μια άλλη διεργασία σε ένα απομακρυσμένο μηχάνημα
- Κατανεμημένα αντικείμενα (Distributed objects-Remote Object Invocation):
 - Αντικειμενοστρεφές ανάλογο της RPC
 - Επιτρέπει την κλήση μεθόδων απομακρυσμένων αντικειμένων, οι οποίες είναι αυστηρά οι μέθοδοι που εκθέτουν τα καθορισμένα μέσω κατάλληλης γλώσσας, interfaces των αντικειμένων.



Κάθε μοντέλο πρέπει να αντιμετωπίσει τα εξής θέματα:

- Επικοινωνία
- Διεργασίες και ο συγχρονισμός τους
- Διεργασίες και η αλληλεπίδρασή τους
- Ονομασία
- Ανοχή βλαβών
- Ασφάλεια

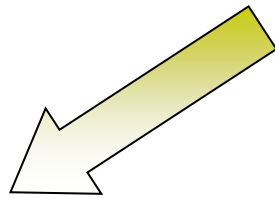


Κατανεμημένα αντικείμενα

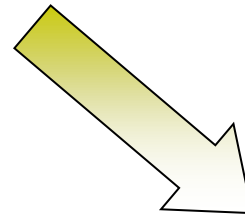
Τα αντικείμενα τα οποία συνθέτουν την εφαρμογή εκτελούνται σε διαφορετικά περιβάλλοντα εκτέλεσης

Το κάθε αντικείμενο έχει ως στόχο του σε μία δεδομένη στιγμή του προγράμματος να εκτελέσει μία συγκεκριμένη εργασία.

Το αντικείμενο αυτό χρειάζεται:



Να έχει τη γνώση (κώδικα)
για το πώς θα εκτελεστεί η διεργασία



Να έχει πρόσβαση στους απαραίτητους πόρους
(π.χ αρχεία, βάσεις δεδομένων κλπ)
για την εκτέλεση της εργασίας



Αρχιτεκτονικές κατακεμημένων αντικειμένων

- Αρχιτεκτονική CORBA (Common Object Request Broker Architecture)
 - Επικοινωνία αντικειμένων με βάση τις διεπαφές που ορίζονται από τη γλώσσα IDL (Interface Definition Language)
 - Υποστήριξη για ανάπτυξη αντικειμένων σε διάφορες γλώσσες προγραμματισμού
 - Βασικό στοιχείο: Το αντικείμενο ORB (Object Request Broker), το οποίο αναλαμβάνει την εκτέλεση μεθόδων απομακρυσμένων αντικειμένων
- Remote Method Invocation (RMI)
- Αρχιτεκτονική DCOM (Distributed Component Model)
 - Επικοινωνία κατακεμημένων αντικειμένων σε συστήματα με λειτουργικά συστήματα της Microsoft
 - Πλεονέκτημα: Η εκμετάλλευση της ήδη υπάρχουσας υποδομής για την ανάπτυξη κατακεμημένων εφαρμογών
- Υπηρεσίες Ιστού (Web Services)
- Κινητοί Πράκτορες (mobile agents)
- Πλέγμα (Grid)



Αρχιτεκτονικές καταναμημένων αντικειμένων

API based RPC methods

- IBM: Application-to-application communications (APPC)
- SUN: SUN RPC/Network File System (NFS) for workstations
- Open Systems Foundation: Distributed Computing Environment (DCE) RPC

Object-Oriented RPC methods

- Object Management Group: Common Object Request Architecture (CORBA)

