

Κατανεμημένα Συστήματα

Μάθημα #1



Κατανεμημένα Συστήματα – Θεωρία

(Σκοπός - Αντικείμενο)



- Σκοπός του μαθήματος είναι η μελέτη του σχεδιασμού και της υλοποίησης κατανεμημένων συστημάτων.
- Θα μελετηθούν έννοιες σχετικές με το υλικό και το λογισμικό πάνω στα οποία οικοδομείται ένα τέτοιο σύστημα.
- Θα μελετηθούν θεμελιώδεις αρχές/τεχνικές όπως η αποδοτική επικοινωνία και ο συγχρονισμός τμημάτων του συστήματος μέσω διαφόρων μηχανισμών καθώς και η διαχείριση διεργασιών.

Κατανεμημένα Συστήματα – Θεωρία (Περίγραμμα – Οργάνωση / Επικοινωνία)



- Βασικές Έννοιες Κατανεμημένων Συστημάτων - Κατανεμημένη Επεξεργασία, Ενδιάμεσο Λογισμικό, Υπηρεσίες Ενδιάμεσου Λογισμικού
- Οργάνωση Κατανεμημένων Συστημάτων - Μοντέλο Πελάτη-Διακομιστή (client-server), Ομότιμα Συστήματα (P2P systems)
- Επικοινωνία στα Κατανεμημένα Συστήματα - Κλήση Απομακρυσμένων Διαδικασιών, Απομακρυσμένα Αντικείμενα, Κλήση Απομακρυσμένων Μεθόδων, Μηνυματοστρεφής και Ρευματοστρεφής Επικοινωνία

Κατανεμημένα Συστήματα – Θεωρία (Περίγραμμα – **Αλγοριθμικά Θέματα**)



- Διάταξη Γεγονότων, Συγχρονισμός, Λογικά Ρολόγια, Αλγόριθμοι Εκλογής Αρχηγού, Αλγόριθμοι Αμοιβαίου Αποκλεισμού, Κατανεμημένες Συναλλαγές
- Ανοχή σε βλάβες, Μοντέλα Αστοχιών, Συγκάλυψη αστοχιών μέσω υπερεπάρκειας, Συγκάλυψη Αστοχιών και Αναπαραγωγή
- Συμφωνία σε Κατανεμημένα Συστήματα με Σφάλματα, Κατανεμημένη Δέσμευση, Ανάκαμψη από Σφάλματα

Κατανεμημένα Συστήματα – Θεωρία

(Περίγραμμα – Άλλα Θέματα)



- Κατανεμημένα Συστήματα Αρχείων, Υπηρεσίες Ονομάτων και Καταλόγου
- Υπολογισμός σε περιβάλλον συστοιχίας (cluster computing). Λογισμικό συστήματος, απαιτήσεις δικτυακής επικοινωνίας, ανάθεση-δρομολόγηση εργασιών, τεχνικές και εργαλεία προγραμματισμού, κ.α.
- Εισαγωγή στην Υπολογιστική Νέφους (ορισμοί, μοντέλα διανομής, παρεχόμενες υπηρεσίες, το μοντέλο του MapReduce)

Κατανεμημένα Συστήματα

(Εργαστηριακά Μαθήματα – Ασκήσεις)



- ❖ Τα μοντέλο επικοινωνίας πελάτη-διακομιστή (internet domain sockets)
- ❖ Ενδιάμεσο λογισμικό - Unix RPC / JRPC / Java RMI / CORBA / XML RPC κ.α.
- ❖ Ενδεικτικές ασκήσεις σε HTCondor και Hadoop/MapReduce

Κατανεμημένα Συστήματα (Αξιολόγηση)

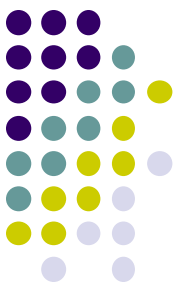


$$\begin{aligned} &\text{Τελικός_Βαθμός_Μαθήματος} = \\ &(\text{70\%} \times \text{Βαθμός_Θεωρητικού_Μέρους}) + \\ &(\text{30\%} \times \text{Βαθμός_Εργαστηριακού_Μέρους}) \end{aligned}$$

A. Διαδικασία Αξιολόγησης **Θεωρητικού Μέρους**:
Τελική γραπτή εξέταση στο τέλος του εξαμήνου

B. Διαδικασία Αξιολόγησης **Εργαστηριακού Μέρους**:
Εκπόνηση εργαστηριακών ασκήσεων / εργασιών και
προφορική εξέταση

Κατανεμημένα Συστήματα – Θεωρία (Βιβλιογραφία)



- Παρουσιάσεις μαθήματος (eclass)
- Tanenbaum A.S., Van Steen M., Κατανεμημένα Συστήματα: Αρχές και Παραδείγματα, 1η έκδοση, Εκδόσεις Κλειδάριθμος, 2006.
- Coulouris G., Dollimore J., Kindberg T., Blair G., Κατανεμημένα Συστήματα, 5η έκδοση, 2011. [Μετάφραση - DA VINCI M.E.Π.Ε., Επιμέλεια: Κωνσταντίνος Κοντογιάννης, 2018]
- Κάβουρας Ι.Κ., Μήλης Ι.Ζ., Ξυλωμένος Γ.Β., Ρουκουνάκη Α.Α., Κατανεμημένα Συστήματα με Java, 3η έκδοση, Εκδόσεις Κλειδάριθμος, 2011.
- Kai Hwang, Jack Dongarra, and Geoffrey Fox, Distributed and Cloud Computing, Morgan Kaufmann, 2011.

Κατανεμημένα Συστήματα



- Η ανάγκη **ευέλικτης χρήσης κοινών πόρων** σε τοπική κλίματα (επιχειρήσεις και οργανισμοί) οδήγησαν στα πρώτα ΚΣ τοπικής εμβέλειας.
- Η ανάγκη για αποφυγή του «**μοναδικού**» σημείου βλάβης και για απομακρυσμένη συνεργασία μεταξύ ερευνητικών κέντρων/οργανισμών οδήγησαν στο διαδίκτυο.
- Δύο εξελίξεις της τεχνολογίας επέτρεψαν την εμφάνιση Κατανεμημένων Συστημάτων (ΚΣ)
 - ισχυροί και φθηνοί μικροεπεξεργαστές
 - δίκτυα δεδομένων ψηλών ταχυτήτων (LANs, WANs)
- Ο **παγκόσμιος ιστός** σε συνδυασμό με τη ραγδαία εξάπλωση του προσωπικού υπολογιστή οδήγησαν σε πιο αποκεντρωμένα (και χαλαρά συνδεδεμένα) ΚΣ σε παγκόσμια κλίμακα.
- Τα ΚΣ **επεκτείνονται** πλέον και σε ενσωματωμένα και φορητά συστήματα.

Ορισμός Κατανεμημένου Συστήματος #1



- Μια συλλογή από αυτόνομους υπολογιστές που συνδέονται μεταξύ τους μέσω ενός δικτύου και χρησιμοποιούν ειδικά σχεδιασμένο λογισμικό για την παροχή ενοποιημένων υπολογιστικών υπηρεσιών
- Οι διεργασίες που εκτελούνται από τους δικτυωμένους υπολογιστές, επικοινωνούν μεταξύ τους και συντονίζουν τις κινήσεις τους μέσω της ανταλλαγής μηνυμάτων
- Πρόκειται περί ενός πληροφοριακού συστήματος πολλαπλών αυτόνομων υπολογιστικών στοιχείων, τα οποία συνεργάζονται για την επίτευξη ενός κοινού στόχου (Burns & Willings)

Ορισμός Κατανεμημένου Συστήματος #2



- Κατανεμημένο σύστημα είναι ένα σύνολο από ανεξάρτητους υπολογιστές το οποίο παρουσιάζεται στους χρήστες σαν ένα λογικό ενιαίο σύστημα.
- Ο ορισμός έχει 2 πλευρές:
 - Το **υλικό** - τα μηχανήματα είναι αυτόνομα
 - Το **λογισμικό** - οι χρήστες θεωρούν ότι έχουν να κάνουν με ένα ενιαίο σύστημα. Οι διαφορές των υπολογιστών μεταξύ τους ή ο τρόπος με τον οποίο επικοινωνούν δεν απασχολούν τους χρήστες
- “Γνωρίζεις ότι έχεις ένα κατανεμημένο σύστημα όταν η δυσλειτουργία ενός αγνώστου συστήματος δεν σου επιτρέπει να κάνεις τη δουλειά σου” – Leslie Lamport

Χαρακτηριστικά Κατανεμημένων Συστημάτων



- Οι διαφορές μεταξύ των διαφόρων υπολογιστών και οι τρόποι με τους οποίους αυτοί επικοινωνούν παραμένουν **κρυφοί** για τους χρήστες
- Το ίδιο ισχύει και για την **εσωτερική οργάνωση** των κατανεμημένων συστημάτων.
- Οι χρήστες και οι εφαρμογές μπορούν να αλληλεπιδρούν με ένα κατανεμημένο σύστημα με **συνεπή και ομοιόμορφο τρόπο**, ανεξάρτητα από το που και πότε πραγματοποιείται η αλληλεπίδραση
- Τα κατανεμημένα συστήματα θα πρέπει να επιτρέπουν με σχετική ευκολία **την επέκτασή τους** ή την προσαρμογή της κλίμακας του μεγέθους τους

Τομείς Εφαρμογής



Ιατρική, βιολογία και γενετική

(διάγνωση νοσημάτων μέσω
επεξεργασίας ιατρικών
εικόνων)

Αγορές χρήματος

(τραπεζικές εφαρμογές,
χρηματοοικονομικές
αναλύσεις κ.α.)

Καταναεμημένα
Συστήματα

Επιστημονικό και Εκπαιδευτικό κλάδο

(ερευνητικά ινστιτούτα,
δημόσια έρευνα κ.α.)

Βιομηχανικό και Κατασκευαστικό τομέα

(αυτοκινητοβιομηχανία,
retail and logistics κ.α.)



Παραδείγματα

- Διαδίκτυο, παγκόσμιος ιστός, κινητή τηλεφωνία.
- Τηλε* (εκπαίδευση, ιατρική, συνεργασία, ...).
- Ενδο-και δια-επιχειρησιακά συστήματα (όμιλος εταιρειών, τράπεζες, ηλεκτρονικό επιχειρείν, ...).
- Αυτόματος έλεγχος (παραγωγή, μεταφορές, ...).
- Ενοποιημένοι υπολογιστικοί και αποθηκευτικοί πόροι σε παγκόσμια κλίμακα (grid computing).
- Ομότιμα συστήματα υπηρεσιών (peer-to-peer).
- Δίκτυα αισθητήρων (sensor networks).
- Έξυπνα κτήρια, συστήματα και περιβάλλοντα αλληλεπίδρασης (pervasive/ubiquitous computing).
- Cloud Architectures & Internet of Things



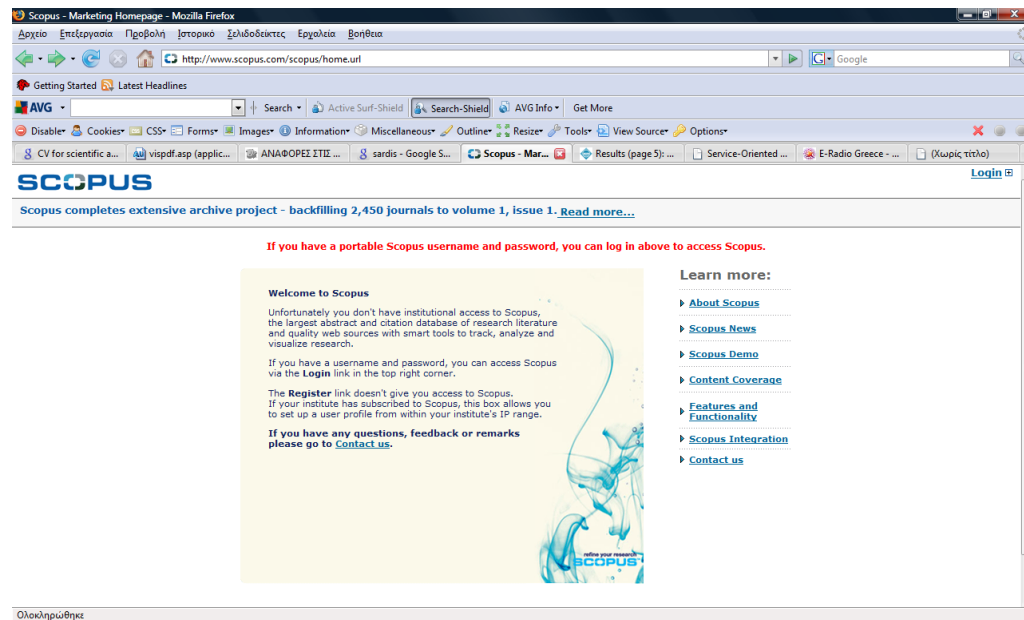
Εφαρμογή: Ο Παγκόσμιος Ιστός

- Αποτελεί ένα τεράστιο σύστημα πληροφοριών βασιζόμενο σε έγγραφα, όπου κάθε έγγραφο έχει ένα δικό του όνομα με τη μορφή μιας διεύθυνσης URL
- Από λογική άποψη φαίνεται σαν να υπάρχει ένας και μόνο διακομιστής
- Από φυσική άποψη ο Ιστός είναι κατανεμημένος σε έναν τεράστιο αριθμό διακομιστών, κάθε ένας από τους οποίους χειρίζεται έναν αριθμό εγγράφων του Ιστού.
- Το όνομα του διακομιστή που χειρίζεται ένα έγγραφο είναι **κωδικοποιημένο** μέσα στη διεύθυνση URL αυτού του εγγράφου

Εφαρμογή: Ο Παγκόσμιος Ιστός



- Ο Παγκόσμιος Ιστός (WWW) παρέχει ένα απλό και ομοιόμορφο μοντέλο κατανεμημένων εγγράφων.
- Για να δει ο χρήστης ένα έγγραφο χρειάζεται μόνο να ενεργοποιήσει μια αναφορά, και το έγγραφο εμφανίζεται στην οθόνη.
- Η δημοσίευση ενός εγγράφου είναι απλή. Χρειάζεται μόνο να δώσουμε ένα μοναδικό όνομα με την μορφή μιας διεύθυνσης URL η οποία παραπέμπει σε ένα τοπικό αρχείο που περιέχει το περιεχόμενο του εγγράφου.



Κεντριοποιημένη vs. Κατανεμημένης οργάνωσης



- **Κεντριοποιημένη λύση:** ένας κεντρικός υπολογιστής, τερματικά σε κάθε κόμβο
- **Κατανεμημένη λύση:** Κάθε κόμβος έχει τα δικά του δεδομένα, δικό του επεξεργαστή



Κατανεμημένη επεξεργασία

- Πολλοί υπολογιστικοί κόμβοι συνεργάζονται χαλαρά (*loosely coupled nodes*)
- Στόχος η κατανομή υπηρεσιών, εφαρμογών, επεξεργαστικής ισχύος, ή δεδομένων
- Δεύτερος στόχος είναι η ταχύτητα
- Η επικοινωνία δεν είναι συνήθως συχνή
- Παρότι οι κόμβοι είναι ετερογενείς το δίκτυο είναι τυποποιημένο
- Οι κόμβοι προστίθενται στο δίκτυο με βάση τη γεωγραφική τους θέση

Κατανεμημένη επεξεργασία: Παράδειγμα



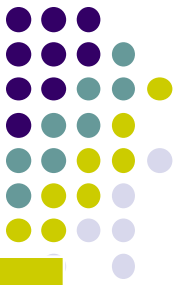
- Τραπεζικές Συναλλαγές σε Υποκαταστήματα
- Διαχείριση Αυτόματων Ταμειακών Μηχανών (ΑΤΜ) μιας τράπεζας που καλύπτει όλη την Ελλάδα
- Τοπική και κεντρική επεξεργασία σε εφαρμογές Ηλεκτρονικού Εμπορίου
- Τοπική και κεντρική επεξεργασία σε Τηλε-Ιατρικές εφαρμογές
- Επεξεργασία σε κόμβους αισθητήρων (sensor nodes) στο πλαίσιο ενός ολοκληρωμένου ΑΔΑ
-



Κατανεμημένη επεξεργασία

- **Θέματα**
 - **Κατανομή δεδομένων: διαχείριση πολλαπλών αντιγράφων**
 - **Διαχείριση ταυτοχρονισμού: κλείδωμα, αδιέξοδα**
 - **Δίκτυο: ταχύτητα, τυποποίηση, επίπεδα ασφάλειας**
 - **Ανοχή σε σφάλματα**

Πλεονεκτήματα



□ Έναντι Κεντριοποιημένων Συστημάτων

Χαρακτηριστικό	Περιγραφή
Οικονομία	Οι μικροεπεξεργαστές παρέχουν καλύτερο λόγο κόστους / απόδοσης από τα κεντρικά υπολογιστικά συστήματα
Ταχύτητα	Ένα κατανεμημένο σύστημα μπορεί να έχει συνολικά περισσότερη υπολογιστική ισχύ από ένα κεντρικό υπολογιστικό σύστημα
Κατανομή	Κάποιες εφαρμογές αναφέρονται σε ξεχωριστές υπολογιστικές μονάδες
Αξιοπιστία	Ακόμα και αν μια μονάδα καταρρεύσει το σύστημα μπορεί να συνεχίσει να λειτουργεί
Διαδοχική Ανάπτυξη	Μπορεί να προστεθεί υπολογιστική ισχύς μέσω διαδοχικών αυξήσεων

□ Έναντι Ανεξάρτητων Υπολογιστών

Χαρακτηριστικό	Περιγραφή
Διαμοιρασμός δεδομένων	Επιτρέπει την πρόσβαση πολλών χρηστών σε μια κοινή βάση δεδομένων
Διαμοιρασμός συσκευών	Επιτρέπει σε πολλούς χρήστες να διαμοιράζονται διάφορα περιφερειακά
Επικοινωνία	Καθιστά την επικοινωνία ανθρώπου με άνθρωπο ευκολότερη (π.χ. ηλεκτρονικό ταχυδρομείο)
Ευελιξία	Διανέμει το φόρτο εργασίας στις μονάδες με περισσότερο αποτελεσματικό τρόπο αναφορικά με το κόστος

Κατανεμημένη επεξεργασία



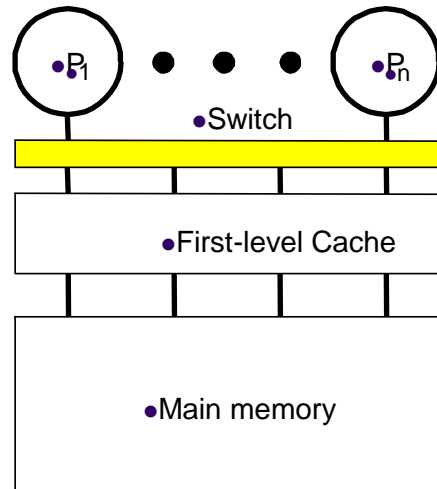
Πλεονεκτήματα	Μειονεκτήματα
Ευελιξία	Δυσκολότερη διαχείριση
Τοπική αυτονομία	Δυσκολότερη ασφάλεια
Αξιοπιστία	Τεχνικό προσωπικό με γνώσεις ολοκλήρωσης συστημάτων
Διαθεσιμότητα	
Επίδοση	



Παράλληλα - Κατανεμημένα Συστήματα

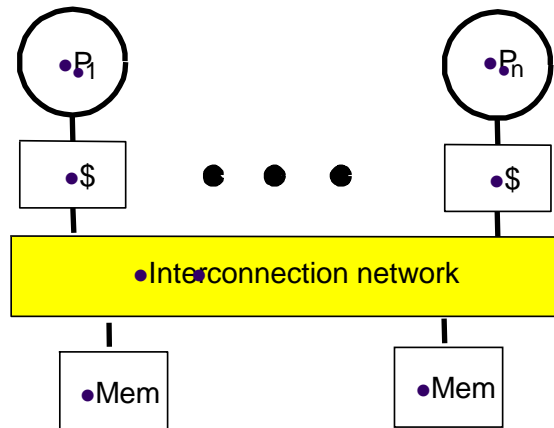
	Παράλληλα Συστήματα	Κατανεμημένα Συστήματα
Στόχος	Επιτάχυνση	1. Τοπολογική κατανομή εφαρμογών, επεξεργαστών ή δεδομένων 2. Ταχύτητα
Δίκτυο	Έμφαση στην ταχύτητα	Ευρείας χρήσης (π.χ. ISDN, Οπτικές ίνες, μισθωμένες γραμμές, Ethernet)
Εφαρμογές	Επιστημονικές εξομοιώσεις	Υπηρεσίες, Βάσεις δεδομένων
Αποστάσεις	<1m	>1m συχνά >1Km
Τοπολογία	Γεωμετρική	Αυθαίρετη
Κόμβοι	Ομοιογενείς	Πιθανώς ετερογενείς

Πολυεπεξεργαστές



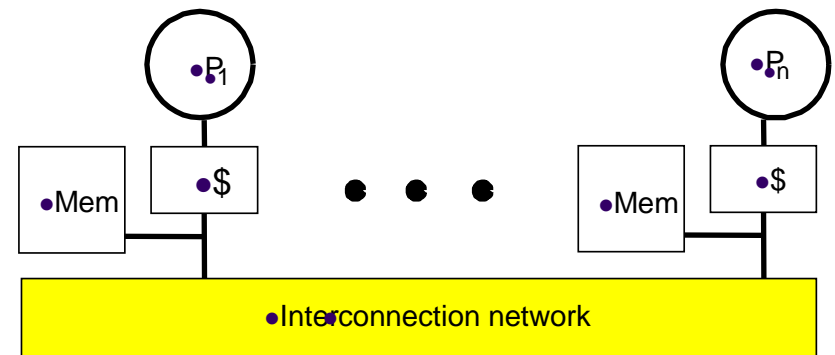
•Shared Cache

•Scale



•Centralized Memory
•(Uniform Memory Access -UMA)

•Distributed Memory
•(Non Uniform Memory Access - UMA)

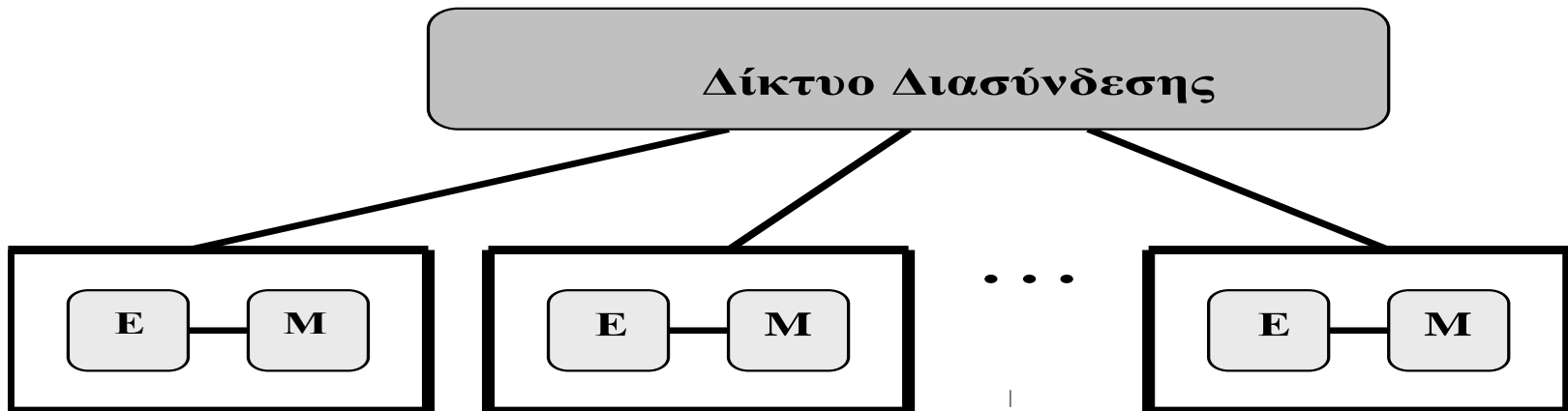


Πολυυπολογιστές (Multicomputers)

Συστήματα Κατανεμημένης Μνήμης



- Κάθε επεξεργαστής έχει τη δική του τοπική μνήμη. Δεν υπάρχει κοινή μνήμη στην οποία έχουν πρόσβαση όλοι οι επεξεργαστές.
- Η επικοινωνία των επεξεργαστών γίνεται με πέρασμα μηνυμάτων (message-passing) μέσω δικτύου διασύνδεσης.

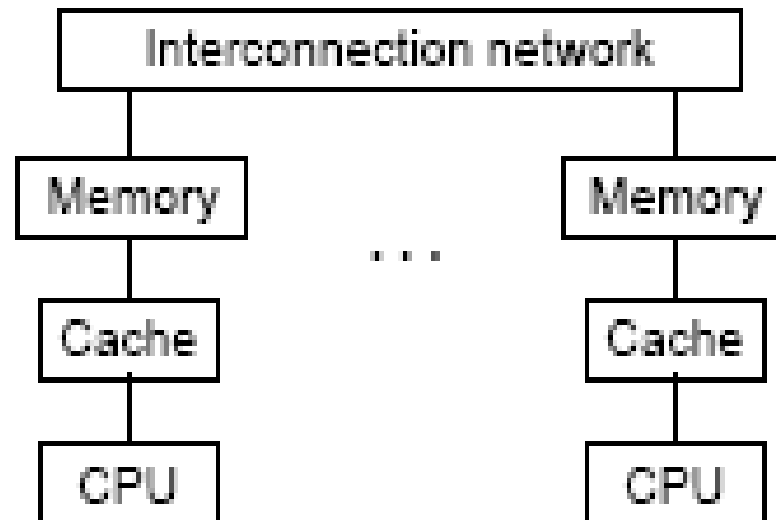


E : Επεξεργαστής

M : Μνήμη

Πολυυπολογιστές (Multicomputers)

Συστήματα Κατανεμημένης Μνήμης



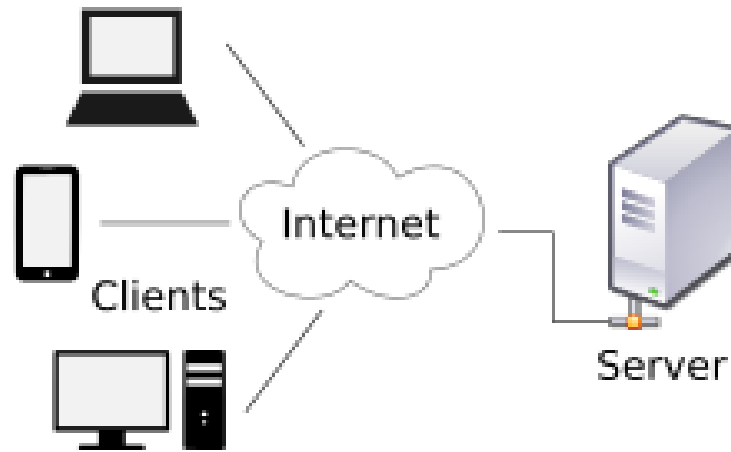
Structure of distributed-memory machines.



Κεντριοποιημένο μοντέλο

- Παραδοσιακό σύστημα time-sharing
- Η επικοινωνία δεν γίνεται μέσω δικτύου
- Δεν κλιμακώνεται εύκολα
 - Όριο στον αριθμό CPUs
 - Δύσκολη διαχείριση πρόσβασης σε κοινόχρηστους πόρους

Μοντέλο πελάτη-εξυπηρετητή



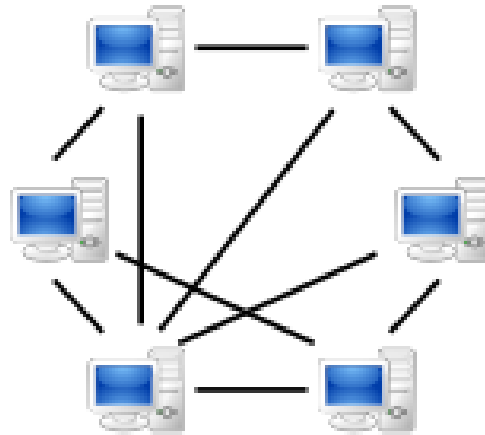
- Ο πελάτης στέλνει αιτήσεις στον εξυπηρετητή
- Ο εξυπηρετητής παρέχει πόρους ή υπηρεσίες
- Οι πελάτες δεν έχουν καμία μεταξύ τους επικοινωνία
Πλεονέκτημα: Εύκολη υλοποίηση - διαχείριση
Μειονέκτημα: Ένα σημείο αποτυχίας, δύσκολη κλιμάκωση
- E-mail, www, ftp, DNS, κλπ.



Fat και Thin clients

- **Thin client:** Ο πελάτης εκτελεί τη λιγότερο δυνατή επεξεργασία. Η βαριά επεξεργασία εκτελείται στον εξυπηρετητή
 - Λιγότερες απαιτήσεις σε hardware και τεχνολογία
 - Καθόλου διαχειριστικό κόστος
 - Καθυστέρηση δικτύου
- **Fat client:** Το μεγαλύτερο μέρος της επεξεργασίας εκτελείται από τον πελάτη
 - Υψηλές δυνατότητες τοπικά
 - Μικρότερη εξάρτηση από το δίκτυο
 - Μεγαλύτερο κόστος διαχείρισης-συντήρησης

Το μοντέλο ομότιμων κόμβων (Peer-to-Peer / P2P)

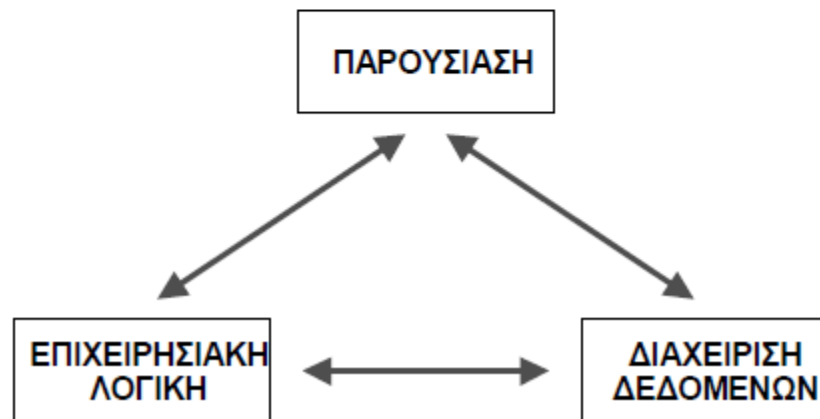


- Όλοι οι κόμβοι είναι ισότιμοι (clients & servers)
- Επικοινωνούν μεταξύ τους
Robustness, scalability, αυτό-οργάνωση
Δύσκολη διαχείριση, ασφάλεια
- SETI@home, Gnutella, BitTorrent, JXTA, skype, κλπ.



Διατάξεις λογισμικού

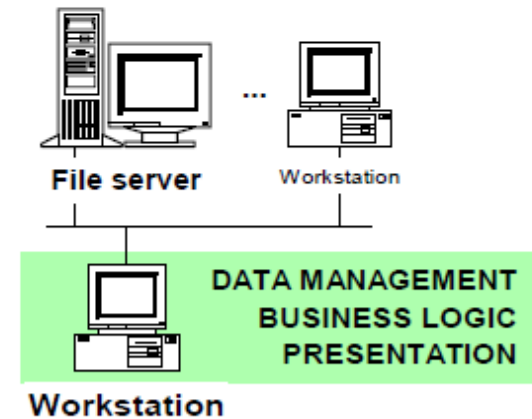
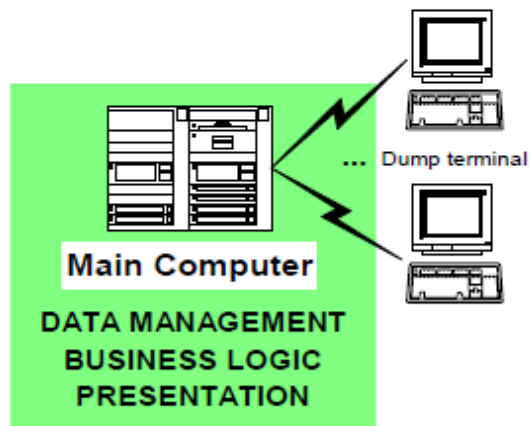
- Το σύγχρονο λογισμικό είναι **κατανεμημένο**.
- **Διάταξη λογισμικού (deployment)**
είναι η κατάτμηση μιας εφαρμογής σε ανεξάρτητα λειτουργικά τμήματα και η ανάθεση αυτών σε διατιθέμενους υπολογιστικούς πόρους.
- Για να μπορεί να καθοριστεί μια διάταξη, θα πρέπει να διακρίνουμε ένα κριτήριο διάκρισης των εργασιών που επιτελεί το λογισμικό.





Διατάξεις λογισμικού

- Η μονολιθική διάταξη

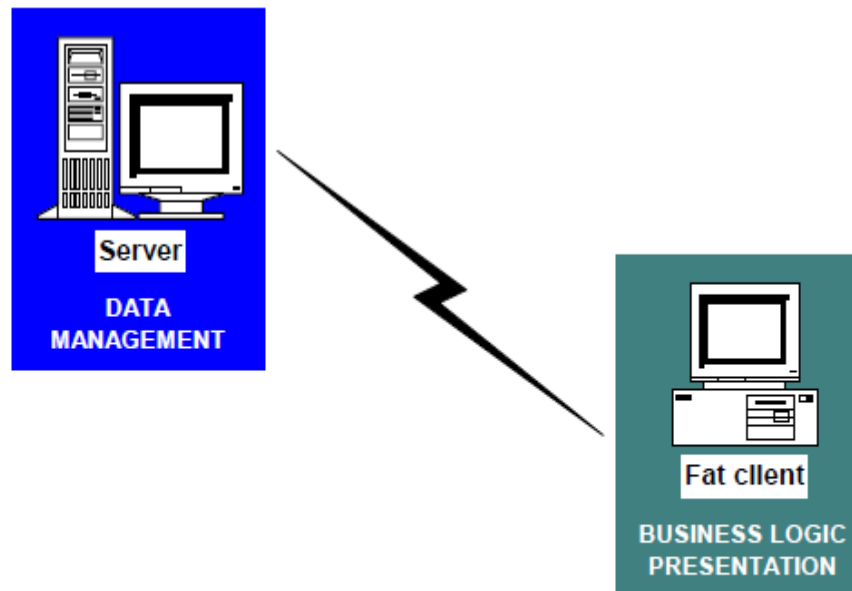




Διατάξεις λογισμικού

- Η διάταξη πελάτη – εξυπηρετητή (client – server)

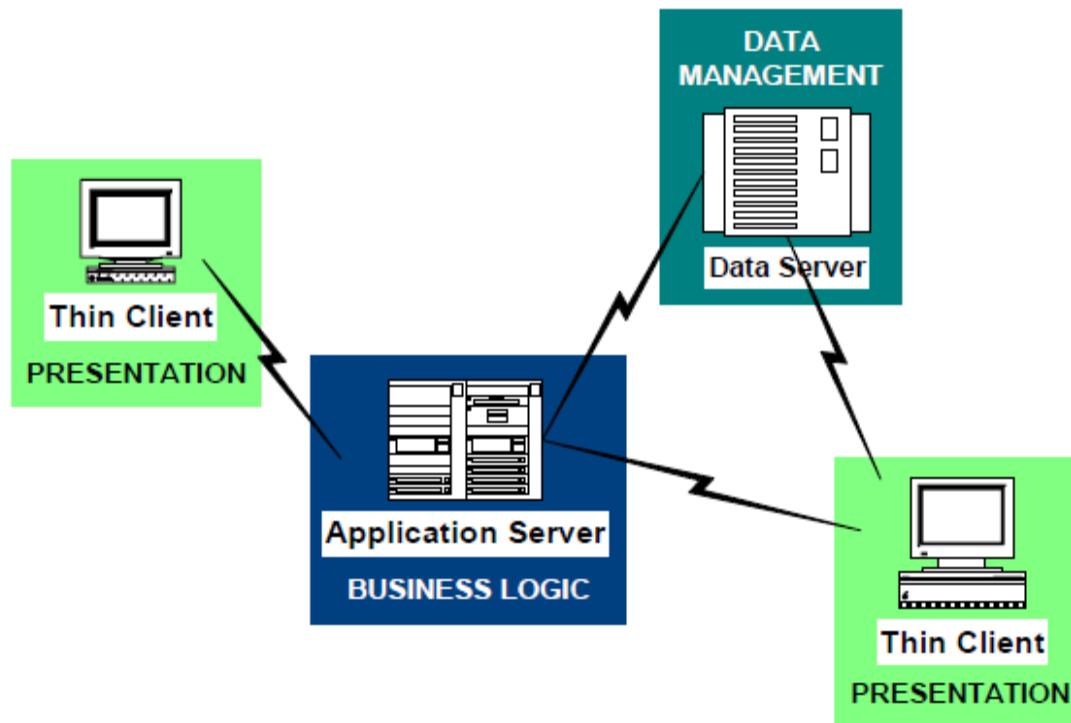
Διμερής διάταξη με FAT clients





Διατάξεις λογισμικού

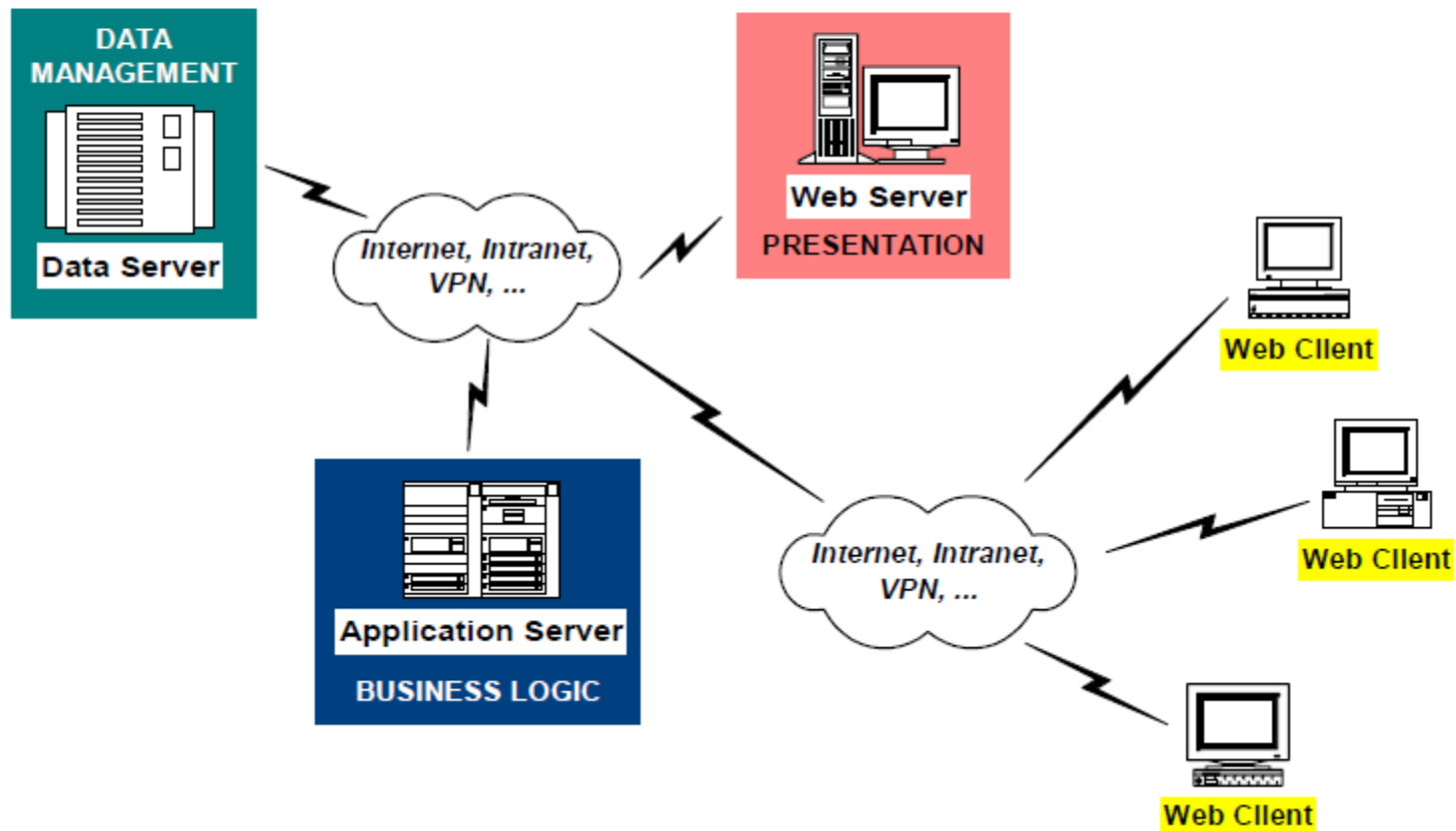
- Διάταξη πελάτη – εξυπηρετητή (2)
- Πολυμερής διάταξη με THIN clients





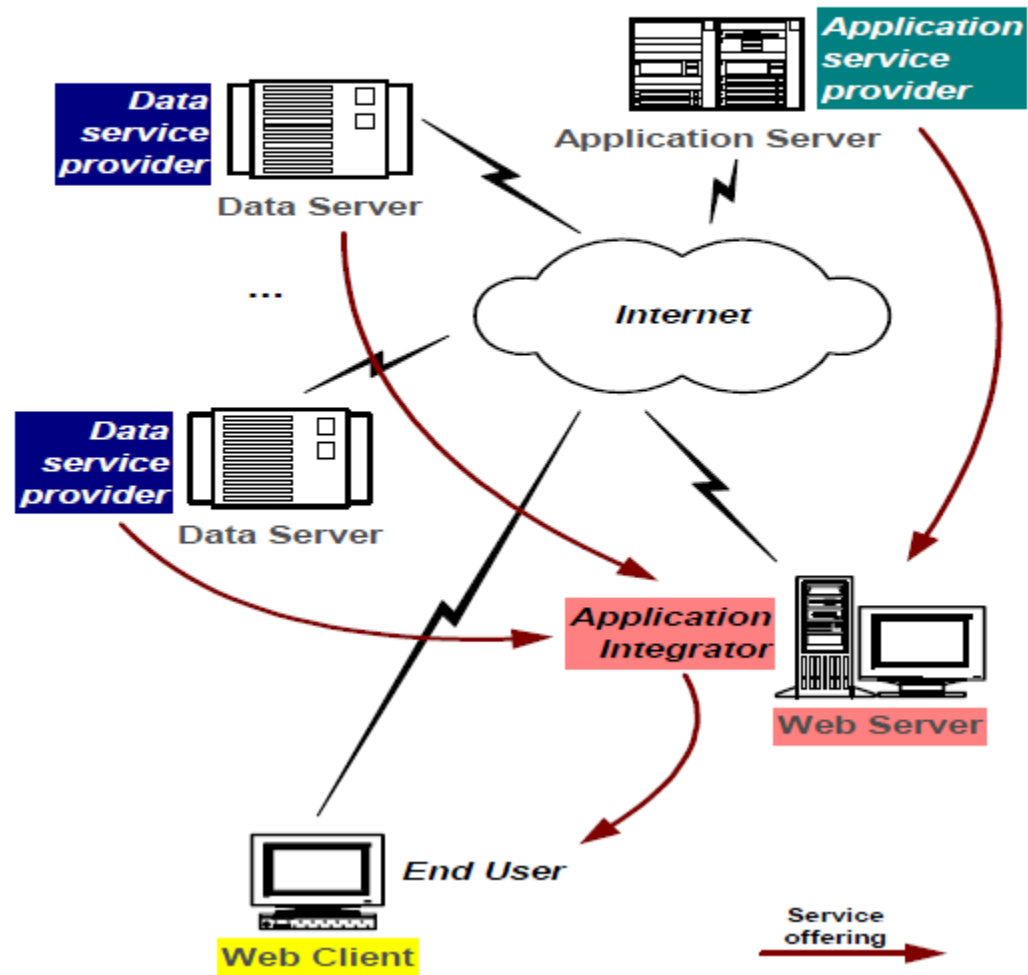
Διατάξεις λογισμικού

- Πολυμερής κατανομημένη διάταξη με web clients





Διατάξεις λογισμικού



Cloud Architectures



Ο πόροι διατίθενται ως υπηρεσίες
(Computing is offered rather as a service than as a product)

- **Η Υποδομή ως Υπηρεσία (IaaS)**

Compute + storage + networking

- Amazon EC2, Rackspace, Microsoft Azure, Google Compute Engine

- **Η Πλατφόρμα ως Υπηρεσία (PaaS)**

Deploy & run web applications without setting up the infrastructure

- Google App Engine, AWS Elastic Beanstalk, Microsoft Azure

- **Το Λογισμικό ως Υπηρεσία (SaaS)**

Remotely hosted software

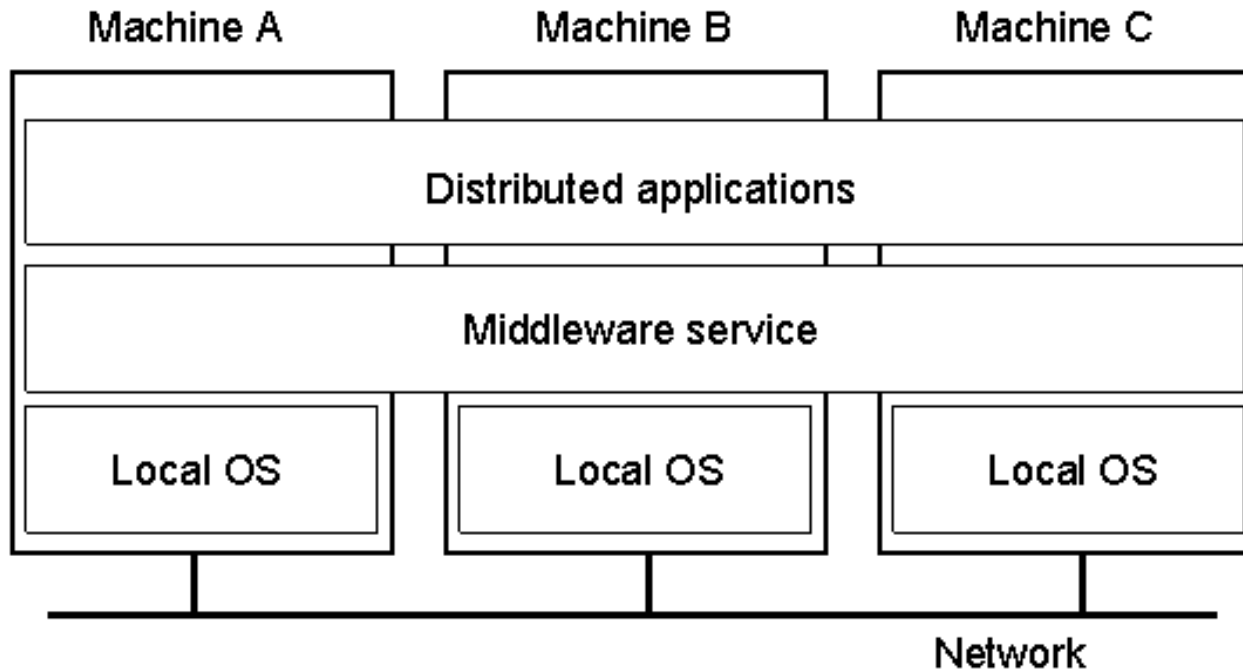
- Salesforce.com, Google Apps, Microsoft Office 365

Middleware – Ενδιάμεσο Λογισμικό



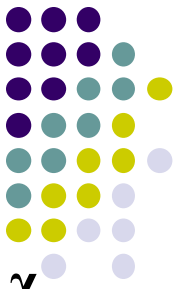
- Προκειμένου να υποστηριχτούν διαφορετικά είδη υπολογιστών και δικτύων και παράλληλα να προσφέρουν μια ομοιόμορφη εικόνα στους τελικούς χρήστες, τα ΚΣ συνήθως αποτελούνται από ένα επίπεδο λογισμικού, το **middleware**
- Το λογισμικό αυτό τοποθετείται λογικά μεταξύ ενός υψηλότερου επιπέδου που αποτελείται από τους χρήστες και τις εφαρμογές και ενός χαμηλότερου επιπέδου που αποτελείται από λειτουργικά συστήματα

Middleware



Παράδειγμα ΚΣ οργανωμένου ως middleware, το επίπεδο του οποίου καλύπτει πολλά διαφορετικά υπολογιστικά συστήματα

Παραδείγματα Σχεδιασμού



- Δίκτυο σταθμών εργασίας, π.χ. με κοινό σύστημα αρχείων (π.χ. NFS based)
- Σύστημα workflow, π.χ. αυτόματη διεκπεραίωση παραγγελιών, αποθήκης (ERPs)
- Εφαρμογές World Wide Web, π.χ. URL Browser (άγνωστο πού βρίσκονται τα έγγραφα) ή Email



Στόχοι Κατανεμημένων Συστημάτων

- Να συνδέει εύκολα τους χρήστες με τους πόρους (συνδεσιμότητα)
- Να κρύβει το γεγονός ότι οι πόροι είναι κατανεμημένοι σε δίκτυο (transparency)
- Να υποστηρίζει ανοικτές αρχιτεκτονικές (openness)
- Να είναι επεκτάσιμο (scalability)
 - στο μέγεθος
 - γεωγραφικά
 - διαχειριστικά
 - ...

Συνδεσιμότητα



Σκοπός

- εύκολη προσπέλαση απομακρυσμένων πόρων
- να επιτρέπει τον διαμοιρασμό αυτών των πόρων με πολλούς χρήστες με ένα ελεγχόμενο τρόπο
- Διευκόλυνση της συνεργασίας (computer-supported cooperative work/games), π.χ. groupware, e-commerce

Η αυξανόμενη συνδεσιμότητα μπορεί να οδηγήσει

- σε ανεπιθύμητη επικοινωνία (π.χ. Spam)
- προβλήματα ασφάλειας και πρόσβασης σε προσωπικά δεδομένα

Διαφάνεια



Ένα κατανεμημένο σύστημα που μπορεί να εμφανίζεται στους χρήστες και τις εφαρμογές σαν ένα ενιαίο υπολογιστικό σύστημα χαρακτηρίζεται ως διαφανές (transparent)

Σκοπός:

- Απόκρυψη του γεγονότος ότι οι διεργασίες και οι πόροι του είναι από φυσική άποψη κατανεμημένοι σε πολλούς υπολογιστές.

Η διαφάνεια δεν είναι πάντα εφικτή ή εύκολα υλοποιήσιμη. Εξαρτάται από την αρχιτεκτονική του ΚΣ και την απόδοση του συστήματος, π.χ.

- αντίγραφα βάσεων δεδομένων,
- ταχύτητα μετάδοσης σήματος,

Μορφές Διαφάνειας



Μορφή Διαφάνειας	Περιγραφή
Προσπέλασης (Access)	Απόκρυψη των διαφορών στην αναπαράσταση των δεδομένων και στον τρόπο πρόσβασης ενός κόμβου (π.χ. διαφορετικές συμβάσεις ονομασίας αρχείων σε Λ.Σ.)
Θέσης (Location)	Απόκρυψη της τοποθεσίας του κόμβου (απόδοση μόνο λογικών ονομάτων στους κόμβους)
Μετανάστευσης (Migration)	Απόκρυψη του γεγονότος ότι ένας κόμβος μπορεί να αλλάξει τοποθεσία
Μετακόμισης (Relocation)	Απόκρυψη του γεγονότος ότι ένας κόμβος μπορεί να αλλάξει τοποθεσία ενώ χρησιμοποιείται
Αναπαραγωγής (Replication)	Απόκρυψη του γεγονότος ότι υπάρχουν πολλά αντίγραφα ενός πόρου
Αστοχιών (Failure)	Απόκρυψη της αποτυχίας και ανάκτησης ενός κόμβου
Διατήρησης (Persistence)	Απόκρυψη της θέσης του λογισμικού (στη μνήμη ή στο δίσκο)

Ανοικτή Αρχιτεκτονική



Αποτελεί σημαντικό παράγοντα:

- **Διαλειτουργικότητας (interoperability):** Χαρακτηρίζει το κατά πόσον δύο υλοποιήσεις συστημάτων ή στοιχείων που προέρχονται από διαφορετικούς κατασκευαστές μπορούν να συνυπάρχουν και να συνεργάζονται βασιζόμενες απλώς η μία στις υπηρεσίες της άλλης, όπως αυτές καθορίζονται από ένα κοινό πρότυπο.
- **Φορητότητας (portability):** Χαρακτηρίζει το κατά πόσον μια εφαρμογή που έχει αναπτυχθεί για ένα κατανεμημένο σύστημα Α μπορεί να εκτελεστεί χωρίς τροποποιήσεις σε ένα διαφορετικό κατανεμημένο σύστημα Β, το οποίο υλοποιεί τις ίδιες διασυνδέσεις με το Α.
- **Ευελιξίας (flexibility):**
 - δυνατότητα ένα σύστημα να διαμορφωθεί από διαφορετικά στοιχεία υλοποιημένα από διαφορετικούς κατασκευαστές
 - Το σύστημα είναι μία συλλογή επιμέρους στοιχείων τα οποία μπορούν εύκολα να προσαρμοστούν ή να αντικατασταθούν



Επεκτασιμότητα (Scalability)

- Η επεκτασιμότητα αποτελεί έναν από τους σημαντικότερους σχεδιαστικούς στόχους για τους κατασκευαστές κατανεμημένων συστημάτων
- Η επεκτασιμότητα μετριέται σε τρεις διαστάσεις:
 - το μέγεθος: μπορούμε να προσθέσουμε στο σύστημα περισσότερους χρήστες και πόρους
 - γεωγραφική επεκτασιμότητα: οι χρήστες και οι πόροι είναι δυνατόν να βρίσκονται σε μεγάλες αποστάσεις μεταξύ τους
 - διαχειριστική επεκτασιμότητα: η διαχείριση του συστήματος μπορεί να παραμένει εύκολη ακόμα και αν αυτό εκτείνεται σε πολλούς διαχειριστικά ανεξάρτητους οργανισμούς.

Επεκτασιμότητα - Μέγεθος



Προβλήματα

- **Κεντρικός server για όλους τους χρήστες (centralized server)**
 - Ανασχετικός παράγοντας καθώς ο αριθμός των χρηστών αυξάνει
 - Υπερφόρτωση επικοινωνίας
 - Όμως υπάρχουν περιπτώσεις όπου η χρήση πολλαπλών servers δεν είναι δυνατή (π.χ. λόγω ασφάλειας)
- **Συγκεντρωτικά data (centralized data)**
 - Υπερφόρτωση επικοινωνίας, π.χ. online index (central DNS)
- **Συγκεντρωτικοί Αλγόριθμοι (centralized algorithms)**
 - Κεντρικός αλγόριθμος δρομολόγησης με βάση το σύνολο των μηνυμάτων που πρέπει να σταλούν



- **Αποκεντρωμένοι αλγόριθμοι**
 - **Κανένα μηχάνημα δεν έχει πλήρη πληροφόρηση για τη συνολική κατάσταση του συστήματος**
 - **Κάθε μηχανή παίρνει αποφάσεις στηριζόμενη σε πληροφορίες που είναι διαθέσιμες τοπικά**
 - **Δυσλειτουργία μίας μηχανής δεν επηρεάζει τον αλγόριθμο**
 - **Δεν υπάρχει παραδοχή για καθολικό ρολόϊ**

Επεκτασιμότητα - Γεωγραφία



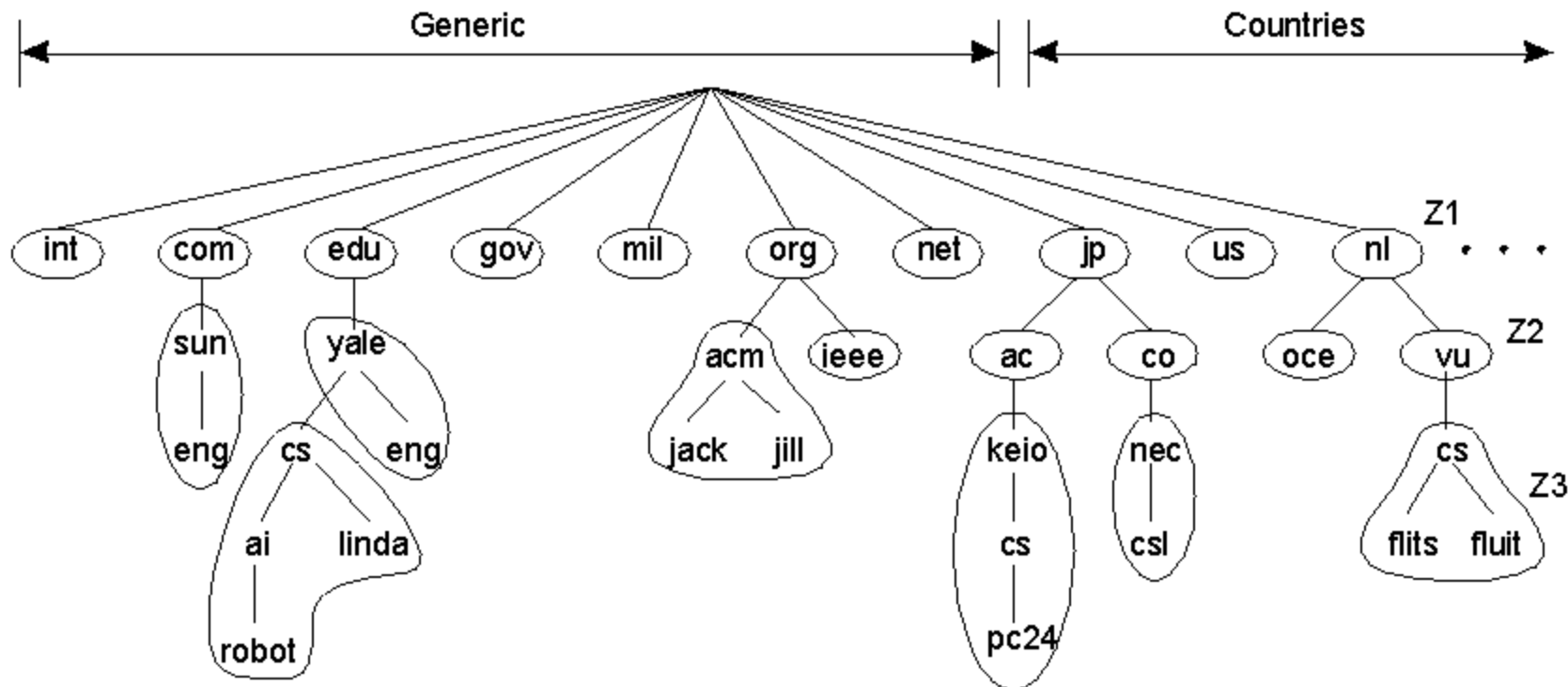
- **Σύγχρονη επικοινωνία: Ο client που ζητά μία υπηρεσία μπλοκάρεται μέχρι να πάρει απάντηση**
 - Η προσέγγιση δουλεύει καλά σε LANs (επικοινωνία απαιτεί μερικές εκατοντάδες μικροδευτερόλεπτα) όχι όμως σε WANs (επικοινωνία απαιτεί εκατοντάδες χιλιοστά δευτερολέπτου)
 - Οι διαδραστικές εφαρμογές έχουν χρονικές απαιτήσεις
- Σε WANs η επικοινωνία είναι point-to-point και δεν είναι τόσο αξιόπιστη
 - π.χ. εντοπισμός μίας δικτυακής υπηρεσίας (service location)
- Συγκεντρωτικές υπηρεσίες
 - π.χ. Mail server για ολόκληρη τη χώρα

Επεκτασιμότητα - Γεωγραφία



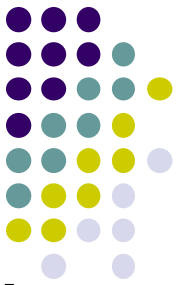
- **Κατανομή**
 - Διαίρεσε ένα στοιχείο σε μικρότερα κομμάτια και μοίρασε σε όλο το σύστημα, π.χ. DNS – Domain Name Service
 - Για την αντιστοίχιση ονομάτων σε διευθύνσεις δεν χρησιμοποιείται ένας μόνο server
 - Ιεραρχικά δομημένο σαν δέντρο
 - Χωρισμός σε (μη επικαλυπτόμενες) ζώνες– τομείς αρμοδιότητας (domains)
 - Τα ονόματα σε κάθε ζώνη εξυπηρετούνται από ένα server π.χ., nl.vu.cs.flits
- Ένα ΚΣ που επεκτείνεται γεωγραφικά συνήθως παρουσιάζει προβλήματα απόδοσης

Επεκτασιμότητα – Γεωγραφία - DNS



Παράδειγμα διαχωρισμού του χώρου ονομάτων DNS σε ζώνες

Επεκτασιμότητα - Διαχείριση

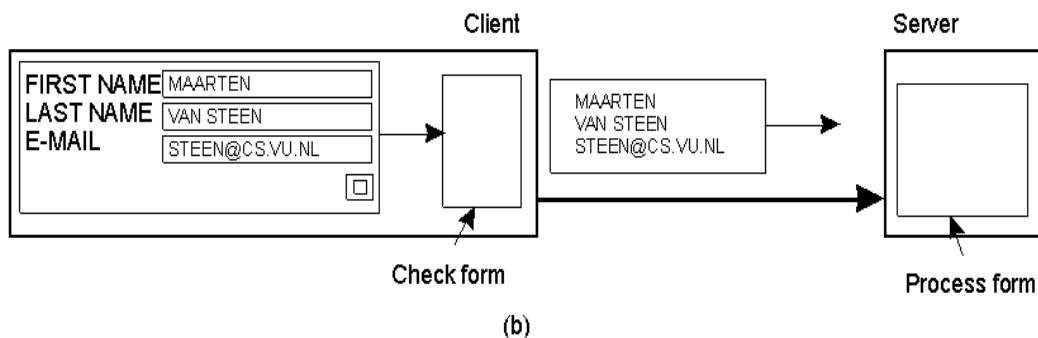
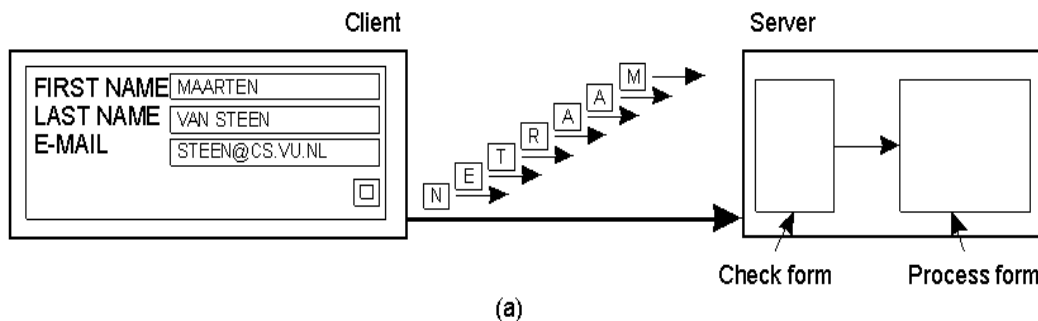


- Σε ένα ΚΣ έχουμε πολλούς και ανεξάρτητους διοικητικά τομείς
- Υπάρχουν αντιθέσεις όσον αφορά στις πολιτικές που κάθε τομέας εφαρμόζει σε θέματα χρήσης πόρων, πληρωμών, διαχείρισης, ασφάλειας, π.χ. στον ίδιο τομέα οι χρήστες εμπιστεύονται τον administrator και τους υπόλοιπους χρήστες
- Η εμπιστοσύνη όμως δεν είναι η ίδια και για άλλους τομείς
- Ένας τομέας πρέπει να προστατεύεται από επιθέσεις από ένα άλλο τομέα
 - read only file systems
 - Χρήση ακριβών πόρων

Τεχνικές Επίτευξης Επεκτασιμότητας



- Απόκρυψη καθυστερήσεων στην επικοινωνία
 - Κάνε κάτι άλλο καθώς περιμένεις απάντηση από τον server, π.χ. συμπλήρωση φόρμας βάσης δεδομένων. Ασύγχρονη επικοινωνία: δεν εφαρμόζεται στις διαδραστικές εφαρμογές
 - Μείωση του επικοινωνιακού φόρτου. Κάποιοι υπολογισμοί αντί να γίνονται στον server γίνονται στον client



Διαφορά μεταξύ του ποιος ελέγχει τις φόρμες καθώς αυτές συμπληρώνονται

a) ο server ή

b) ο client

Τεχνικές Επίτευξης Επεκτασιμότητας



- **Caching**
 - Ειδική μορφή αντιγράφου
 - Κατασκευή αντιγράφων όσο αφορά data/services κοντά στον client
 - Η απόφαση παίρνεται από τον client και όχι από τον server
 - Σημαντικό πρόβλημα η επικαιροποίηση των αντιγράφων, π.χ. web page vs. stock quote



ΠΕΡΙΕΧΟΜΕΝΑ ΜΑΘΗΜΑΤΟΣ (1)

- Υπολογιστικά Συστήματα πολλών CPUs
- Λειτουργικά Συστήματα Υπολογιστικών Συστημάτων
- Ενδιάμεσο Λογισμικό (Middleware)
- Οργάνωση Κατανεμημένων Συστημάτων
 - Το Μοντέλο Πελάτη-Διακομιστή
 - Ομότιμα Συστήματα (P2P Systems)



ΠΕΡΙΕΧΟΜΕΝΑ ΜΑΘΗΜΑΤΟΣ (2)

Τα πρωτόκολλα επικοινωνίας του ενδιαμέσου λογισμικού υποστηρίζουν υπηρεσίες επικοινωνίας υψηλού επιπέδου:

- **κλήση απομακρυσμένων διαδικασιών (RPC)**
- **κλήση απομακρυσμένων αντικειμένων (RMI)**

Μοντέλα Ενδιάμεσου Λογισμικού



- Ενδιάμεσο λογισμικό που βασίζεται στα **ΚΣ αρχείων (NFS, HDFS κλπ)**: Η επικοινωνία ανάγεται στην προσπέλαση των διαμοιραζόμενων αρχείων.
- Ενδιάμεσο λογισμικό που βασίζεται στις **Κλήσεις Απομακρυσμένων Διαδικασιών (RPC)**: Μια διεργασία μπορεί να καλεί διαδικασία που υλοποιείται σε απομακρυσμένη μηχανή. Η καλούσα διεργασία δεν αντιλαμβάνεται ότι πραγματοποιήθηκε επικοινωνία μέσω δικτύου.
- Ενδιάμεσο λογισμικό που βασίζεται στα **Κατανεμημένα Αντικείμενα (RMI, CORBA κ.α.)**: Κάθε αντικείμενο τοποθετείται σε μία μηχανή και οι μέθοδοι που υλοποιεί γίνονται διαθέσιμοι σε άλλες μηχανές. Όταν μια διεργασία καλεί μια μέθοδο, στέλνεται μήνυμα στο αντικείμενο το οποίο εκτελεί τη μέθοδο και στέλνει πίσω το αποτέλεσμα.

ΠΕΡΙΕΧΟΜΕΝΑ ΜΑΘΗΜΑΤΟΣ (3)



- **Πρότυπο Κατανεμημένου Υπολογισμού**
- **Λογικά Ρολόγια**
- **Χρονοσφραγίδες Lamport**
- **Διανυσματικές χρονοσφραγίδες**
- **Διασφάλιση αιτιοκρατικής παράδοσης μηνυμάτων**



ΠΕΡΙΕΧΟΜΕΝΑ ΜΑΘΗΜΑΤΟΣ (4)

Πρόβλημα Εκλογής Αρχηγού: Επιλογή μιας μόνο διεργασίας αρχηγού/συντονιστή προκειμένου να εκτελέσει ένα συγκεκριμένο καθήκον (εκτέλεση συγκεντρωτικού αλγορίθμου, επαναφορά από αδιέξοδο)

- Δεν μπορεί αυθαίρετα μια διεργασία να αυτοανακηρυχθεί αρχηγός. Θα πρέπει να προηγηθεί η εκτέλεση ενός αλγορίθμου εκλογής.
- Αλγόριθμοι Εκλογής Αρχηγού

ΠΕΡΙΕΧΟΜΕΝΑ ΜΑΘΗΜΑΤΟΣ (5)



Το πρόβλημα του αμοιβαίου αποκλεισμού:

- Πρόσβαση **πολλών** διεργασιών σε **κοινό** πόρο (π.χ., κοινή μεταβλητή, κοινή δομή δεδομένων, κοινό αρχείο)
- Οι διεργασίες πρέπει να αποκτήσουν **αποκλειστική πρόσβαση**

Αλγόριθμοι Αμοιβαίου Αποκλεισμού

ΠΕΡΙΕΧΟΜΕΝΑ ΜΑΘΗΜΑΤΟΣ (6)



- **Ανοχή σε βλάβες**
- **Μοντέλα Αστοχιών**
- **Συγκάλυψη αστοχιών μέσω υπερεπάρκειας**
- **Συγκάλυψη Αστοχιών και Αναπαραγωγή**

ΠΕΡΙΕΧΟΜΕΝΑ ΜΑΘΗΜΑΤΟΣ (7)



Συμφωνία σε κατανεμημένα συστήματα με σφάλματα

Πρόβλημα των Βυζαντινών στρατηγών οι οποίοι πρέπει να συντονίσουν την επίθεση τους έναντι ενός κοινού εχθρού. Κάποιοι από τους στρατηγούς μπορεί να είναι προδότες.

Consensus & Blockchain technology



ΠΕΡΙΕΧΟΜΕΝΑ ΜΑΘΗΜΑΤΟΣ (8)

Κατανεμημένη δέσμευση: αφορά την εκτέλεση μιας λειτουργίας είτε από όλες τις εργασίες μιας ομάδας είτε από καμία.

Μία **απλή μέθοδος** στην κατανεμημένη δέσμευση: η χρήση συντονιστή ο οποίος ειδοποιεί όλες τις διεργασίες να εκτελέσουν (τοπικά) ή όχι την εν λόγω λειτουργία

- Πρωτόκολλο Δέσμευσης δύο φάσεων (2PC)
- Πρωτόκολλο Δέσμευσης Τριών Φάσεων (3PC)



ΠΕΡΙΕΧΟΜΕΝΑ ΜΑΘΗΜΑΤΟΣ (9)

Υπολογισμός σε Συστοιχίες Υπολογιστών (cluster computing):

HTC vs. HPC (High Throughput Computing) vs. High Performance Computing

Λογισμικό συστήματος, απαιτήσεις δικτυακής επικοινωνίας, ανάθεση-δρομολόγηση εργασιών, τεχνικές και εργαλεία προγραμματισμού κ.α.

- High availability
- Fault tolerance
- Parallel processing
- Load balancing



ΠΕΡΙΕΧΟΜΕΝΑ ΜΑΘΗΜΑΤΟΣ (10)

Κατανεμημένα συστήματα αρχείων:

**Υπηρεσίες ονομάτων και καταλόγου,
namenodes & datanodes, replication, fault
tolerance, HDFS/Hadoop**

Υπολογιστική Νέφους

**Ορισμοί, βασικά χαρακτηριστικά, μοντέλα
διανομής, μοντέλα υπηρεσιών, το μοντέλο του
MapReduce κ.α.**