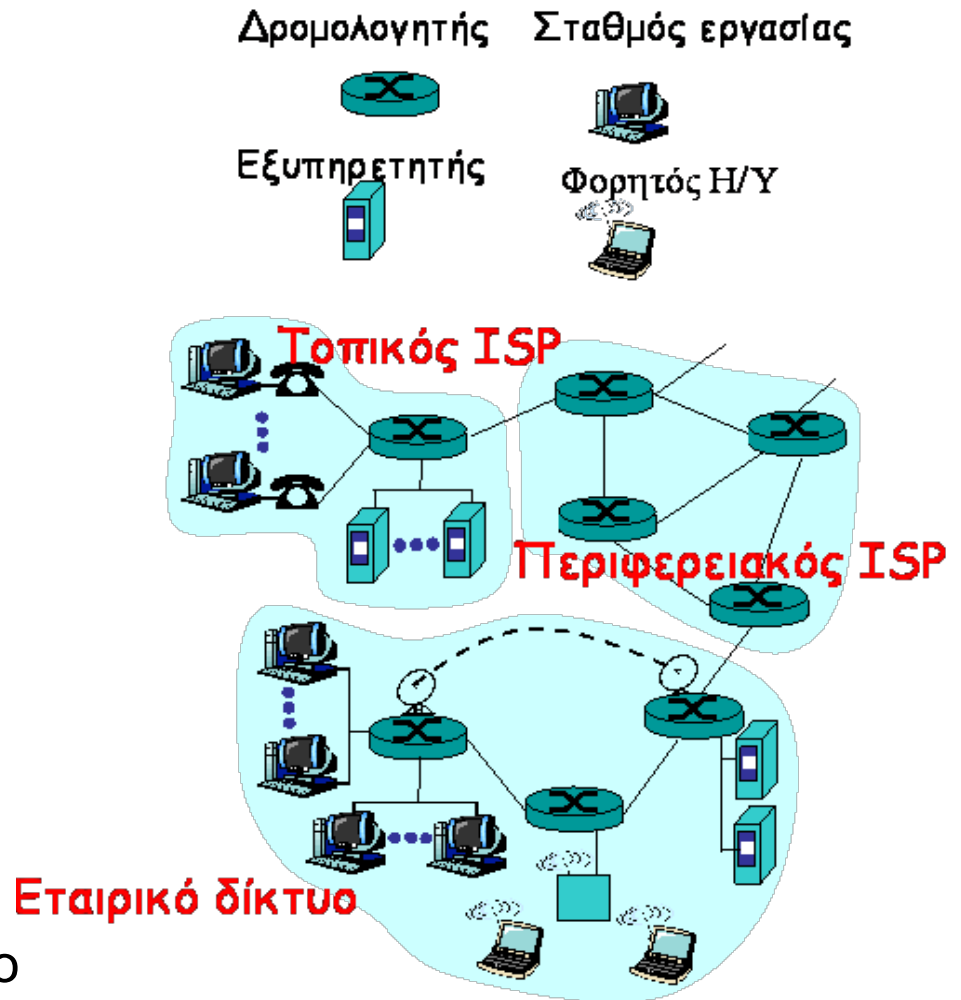


Εισαγωγικά θέματα σε δίκτυα τηλεπικοινωνιών

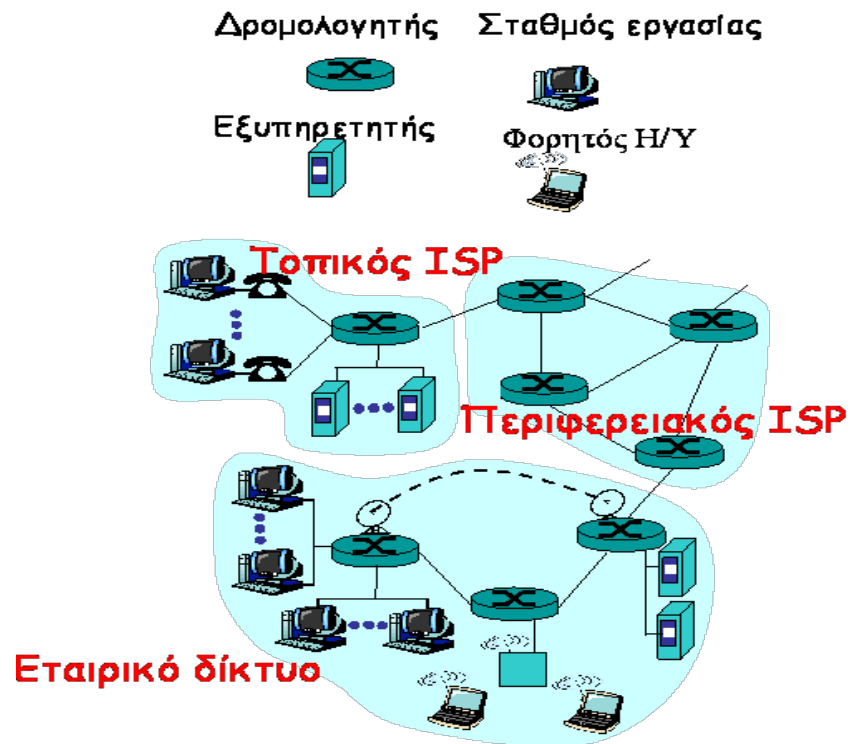
Τι είναι το Διαδίκτυο

- Εκατομμύρια συνδεδεμένες υπολογιστικές συσκευές:
τερματικά συστήματα
 - pc's, σταθμοί εργασίας, εξυπηρετητές
 - PDA's, τηλέφωνα, τoστίερες!Που τρέχουν δικτυακές εφαρμογές
- *Γραμμές επικοινωνίας*
 - Οπτικές ίνες, χαλκός, ραδιοκύματα
- *Δρομολογητές*: προωθούν πακέτα (κομμάτια πληροφορίας) μέσα στο δίκτυο



Τι είναι το Διαδίκτυο

- **πρωτόκολλα:** ελέγχουν την αποστολή και λήψη μηνυμάτων
 - Π.χ., TCP, IP, HTTP, FTP, PPP
- **Διαδίκτυο: Δίκτυο δικτύων**
 - Χαλαρά ιεραρχικό
 - δημόσιο Internet, ιδιωτικό intranet
- Πρότυπα του Internet
 - RFC: Request for comments (περισσότερα από 2000)
 - IETF: Internet Engineering Task Force (www.ietf.org)



“Cool” internet appliances



IP picture frame
<http://www.ceiva.com/>



Web-enabled toaster +
weather forecaster



World's smallest web server
<http://www-ccs.cs.umass.edu/~shri/iPic.html>

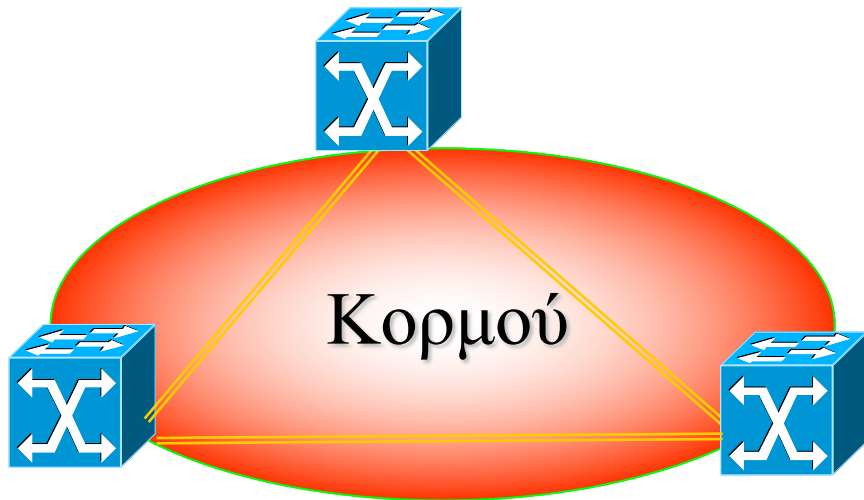


Internet phones

Τρέχουσα τάση



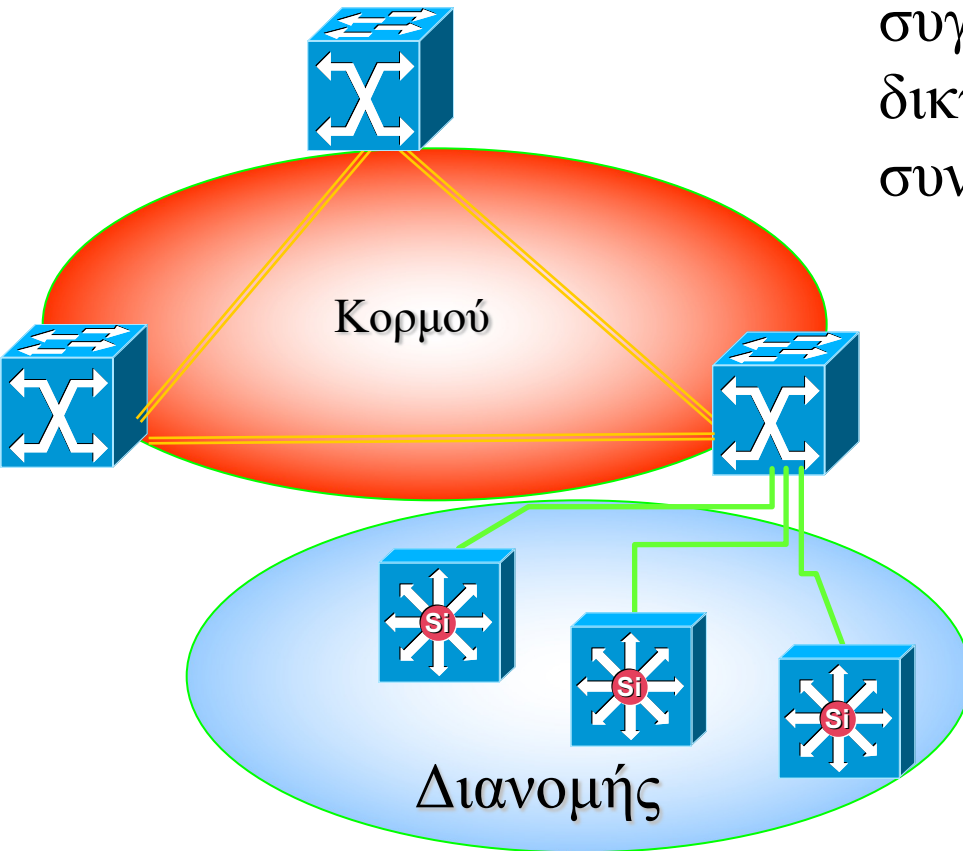
ΔΙΚΤΥΑΚΕΣ ΤΟΠΟΛΟΓΙΕΣ



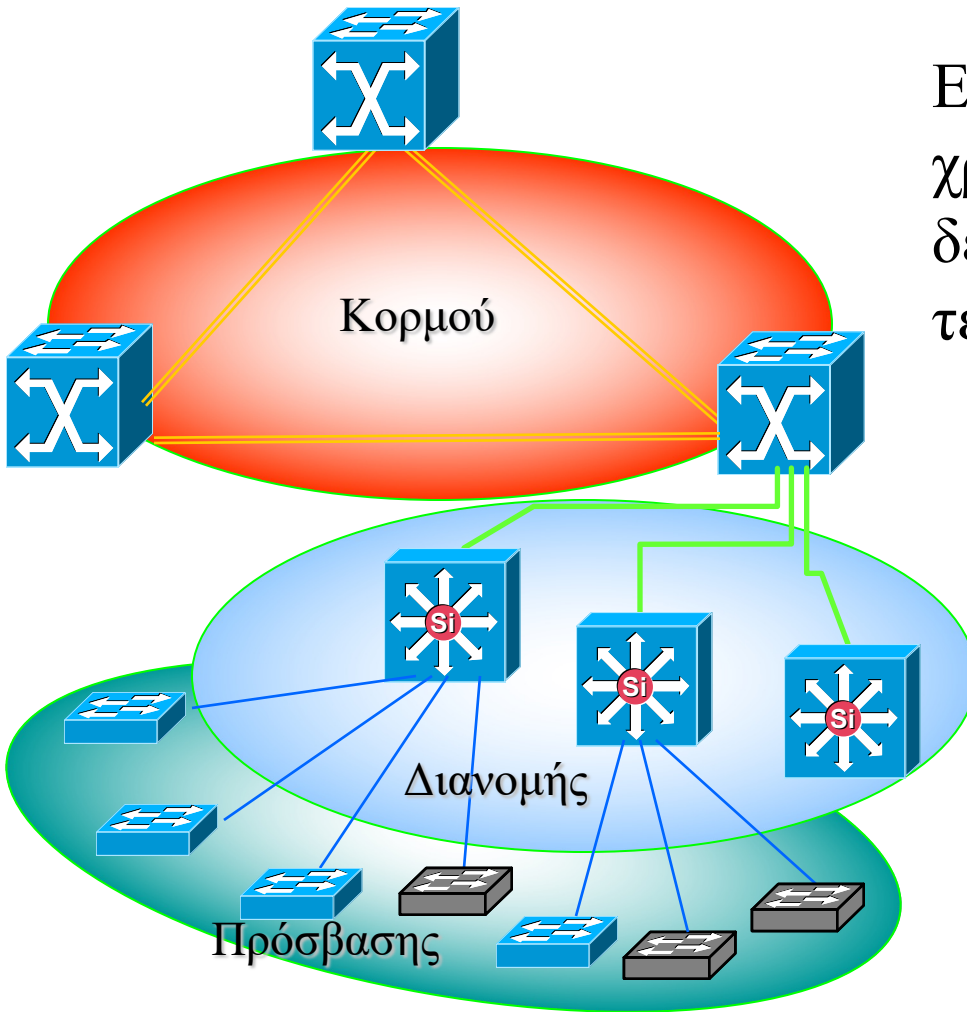
Υλοποιείται από ισχυρές δικτυακές συσκευές με απώτερο σκοπό τη μεταγωγή της κίνησης από το ένα σημείο του συστήματος στο άλλο, με χρήση τεχνολογιών απόδοσης υψηλών ρυθμών μετάδοσης (Gigabit Ethernet)

ΔΙΚΤΥΑΚΕΣ ΤΟΠΟΛΟΓΙΕΣ

Σχηματίζεται από συσκευές που συγκεντρώνουν την κατά τόπους δικτυακή κίνηση και οι οποίες συνδέονται με το δίκτυο κορμού

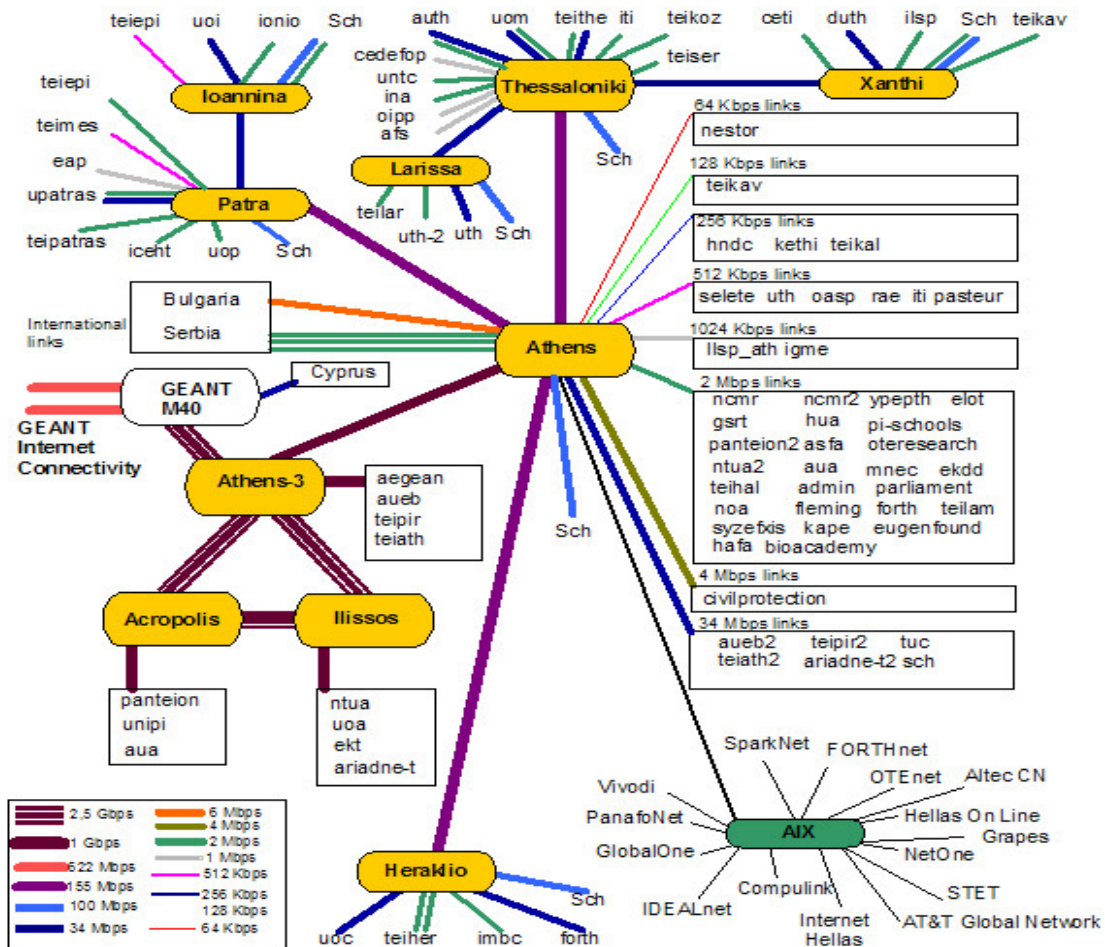


ΔΙΚΤΥΑΚΕΣ ΤΟΠΟΛΟΓΙΕΣ

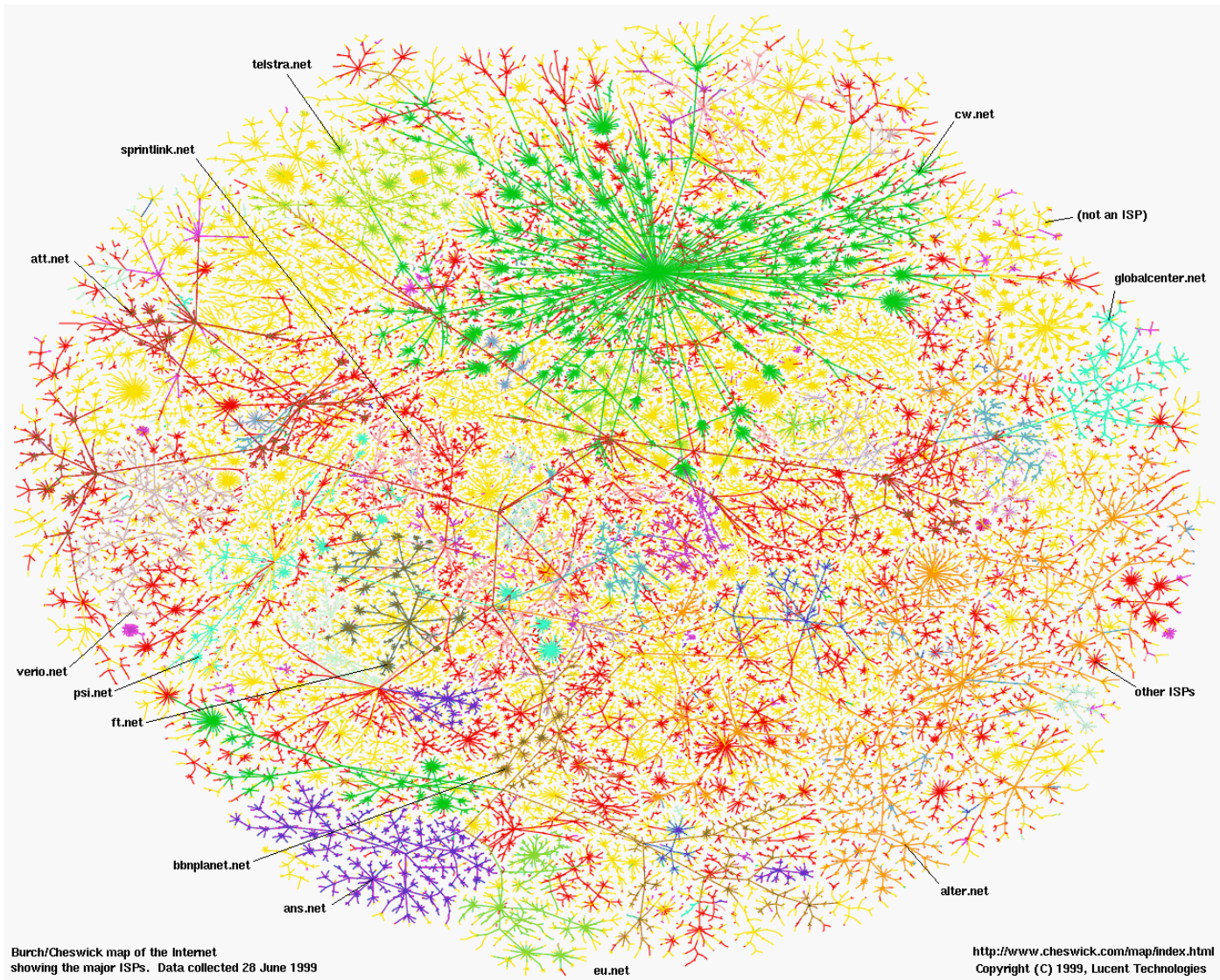


Είναι το δίκτυο εξυπηρέτησης των χρηστών και συνδέεται με το δευτερεύον δίκτυο συνήθως με τεχνολογία Ethernet

Η τοπολογία των ακαδημαϊκών και ερευνητικών δικτύων στην Ελλάδα



Χάρτης του Internet



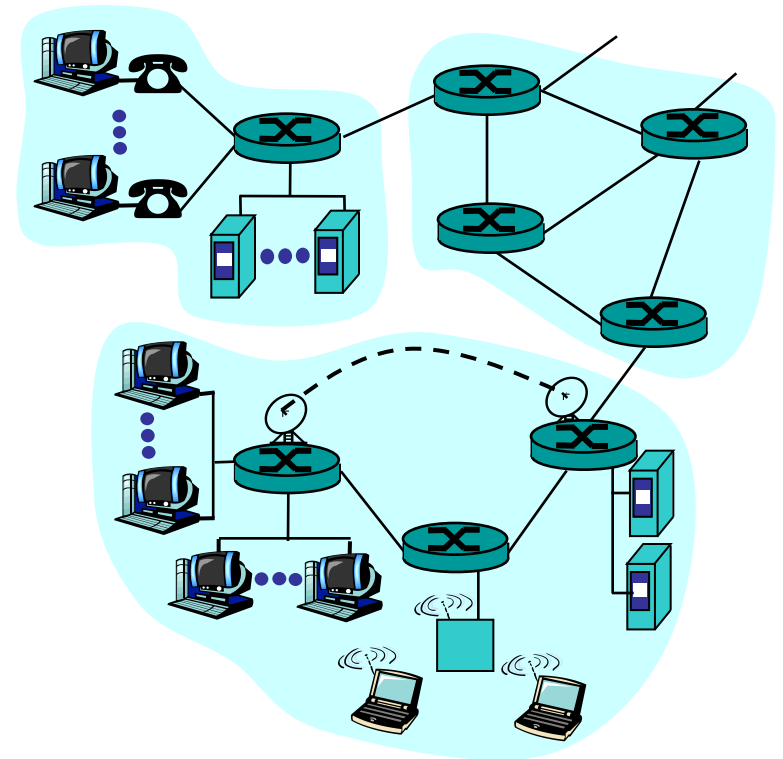
Τι είναι το διαδίκτυο; Μια οπτική γωνία από τις παρεχόμενες υπηρεσίες

• **Επικοινωνιακή υποδομή** καθιστά δυνατές τις καταναλωμένες εφαρμογές:

- WWW, email, μεταφορά αρχείων, παιχνίδια, ηλεκτρονικό εμπόριο, βάσεις δεδομένων

• **Παρεχόμενες δικτυακές υπηρεσίες**

- Υπηρεσία με σύνδεση
- Υπηρεσία χωρίς σύνδεση
- Ποιότητα υπηρεσίας?



Κάποιοι καλοί σύνδεσμοι

- 1) Internet Engineering Task Force (IETF): www.ietf.org
- 2) Internet Society: www.isoc.org
- 3) World Wide Web Consortium (W3C):
www.w3.org/Consortium
- 4) The Association for Computing Machinery (ACM):
www.acm.org
- 5) Institute of Electronic and Electrical Engineering:
www.ieee.org
- 6) IEEE Communications Society: www.comsoc.org
- 7) IEEE Computer Society: www.computer.org

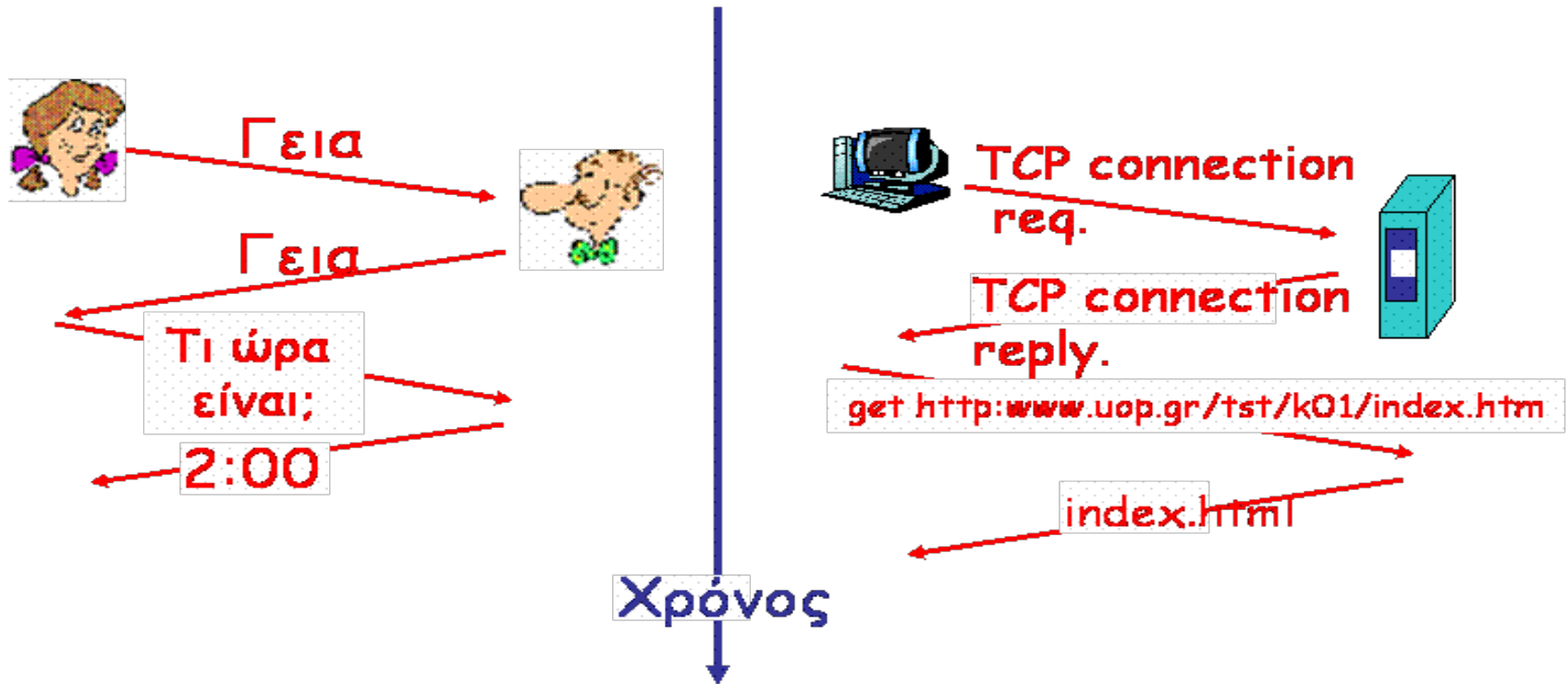
Τι είναι ένα πρωτόκολλο

- Ανθρώπινα πρωτόκολλα:
 - “Τι ώρα είναι”
 - “έχω μια ερώτηση”
 - ... συγκεκριμένα μηνύματα στέλνονται
 - ... συγκεκριμένες ενέργειες εκτελούνται
 - Όταν λαμβάνονται μηνύματα ή συμβαίνουν άλλα γεγονότα
- Δικτυακά πρωτόκολλα:
 - Μηχανές αντί για ανθρώπους
 - Κάθε επικοινωνία στο διαδίκτυο γίνεται με πρωτόκολλα

Τα πρωτόκολλα ορίζουν τη μορφή, τη σειρά των μηνυμάτων που στέλνονται ή λαμβάνονται ανάμεσα στις δικτυακές οντότητες, και τις ενέργειες που εκτελούνται με κάθε αποστολή και λήψη

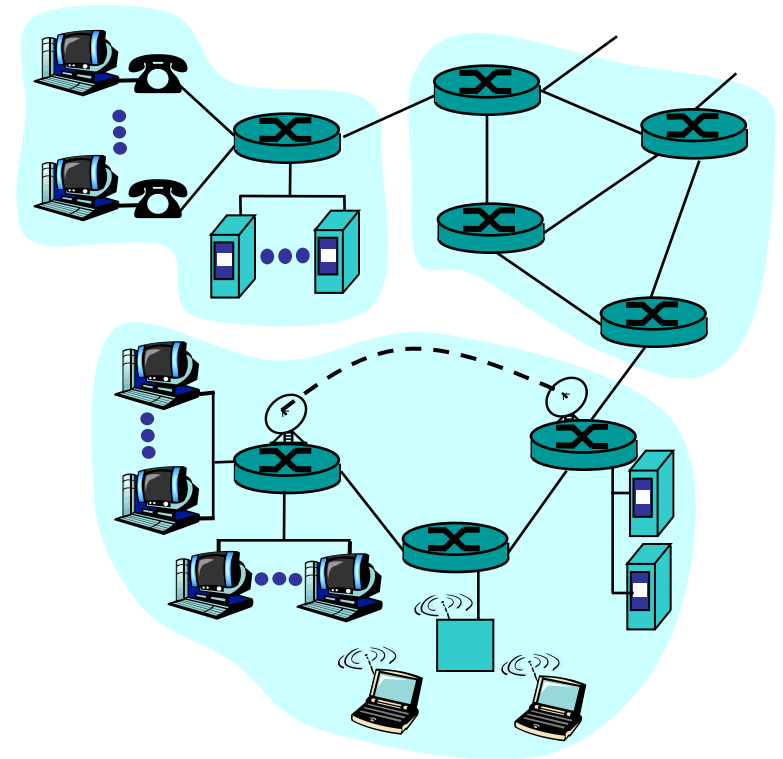
Τι είναι ένα πρωτόκολλο

- Αναλογία ανθρώπινου πρωτοκόλλου και πρωτοκόλλου υπολογιστών



Μια καλύτερη εξέταση της δικτυακής υποδομής

- Στην άκρη του δικτύου: εφαρμογές και τερματικές συσκευές
- Στο κέντρο του δικτύου:
 - δρομολογητές
 - Δίκτυα δικτύων
- Δίκτυα πρόσβασης, Φυσικά μέσα: γραμμές επικοινωνίας



Στην άκρη του δικτύου

- **Τερματικές συσκευές:**

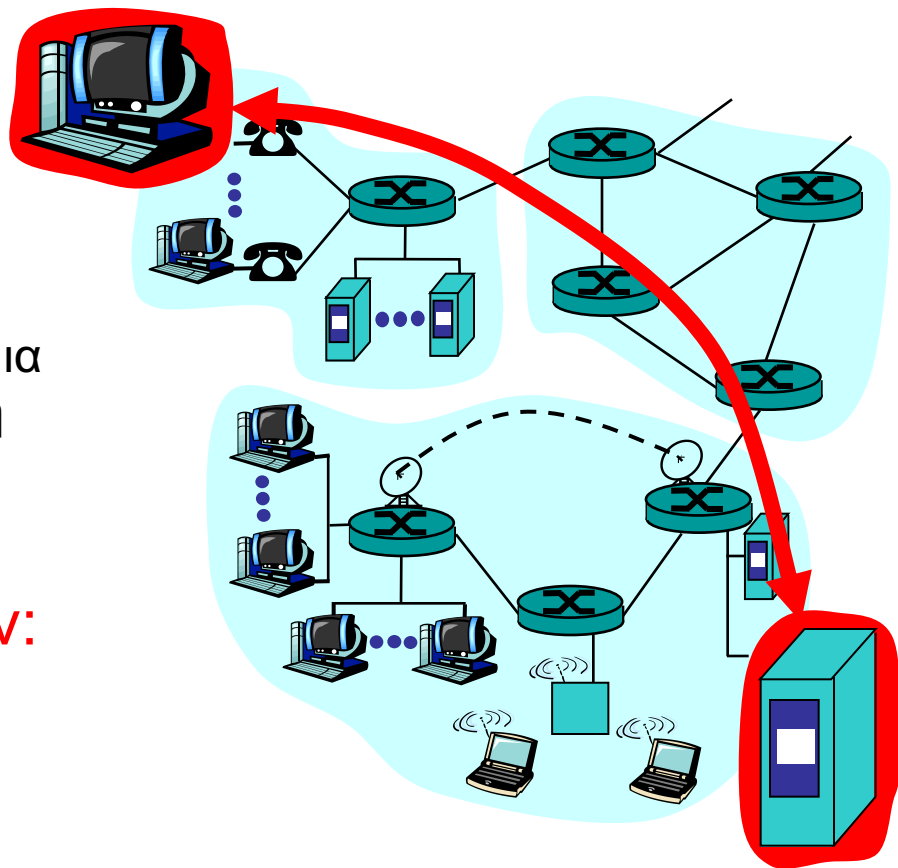
- Εκτελούν εφαρμογές
- Π.χ., WWW, email

- **Μοντέλο πελάτη εξυπηρετητή**

- Ο πελάτης αιτείται και δέχεται μια υπηρεσία από τον εξυπηρετητή
- Π.χ., WWW client (browser)/ server; email client/server

- **Μοντέλο ομότιμων κόμβων:**

- Συμμετρική αλληλεπίδραση κόμβων
- Π.χ.,: τηλεδιάσκεψη,



Στην άκρη του δικτύου: υπηρεσία με σύνδεση

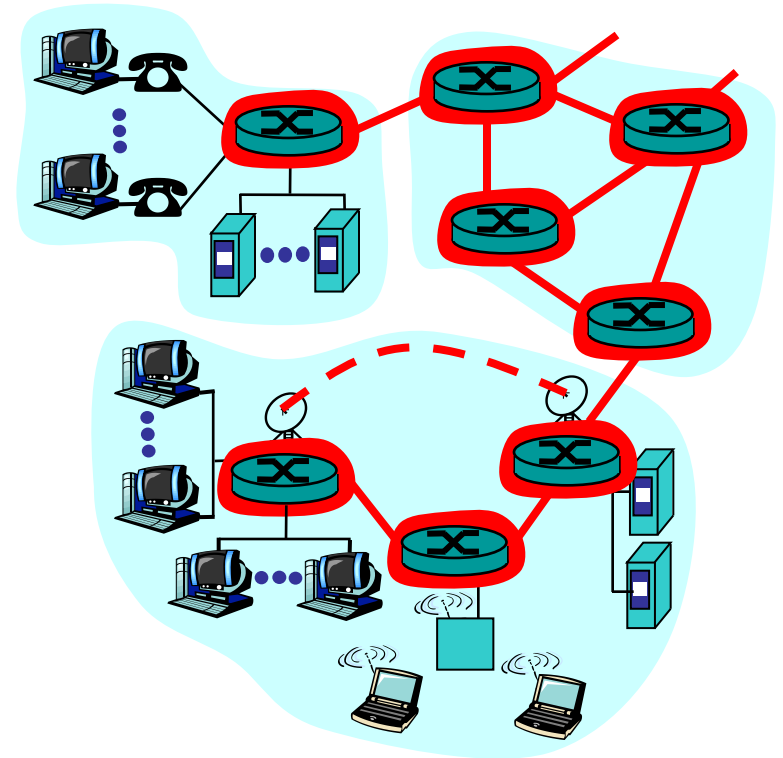
- **Στόχος:** μεταφορά δεδομένων μεταξύ τερματικών συσκευών.
- **χειραψία:** προετοιμασία για τη μεταφορά δεδομένων
 - Γεια – γεια σου σε ανθρώπινο πρωτόκολλο
 - **Δημιουργία κατάστασης** σε δύο επικοινωνούντες κόμβους
- TCP - Transmission Control Protocol
 - Η υπηρεσία με σύνδεση του Internet
- **TCP service** [RFC 793]
- *Αξιόπιστη και σε σειρά μεταφορά ροών από bytes*
 - απώλειες: επιβεβαιώσεις και επαναμεταδόσεις
- *Έλεγχος ροής:*
 - Διασφάλιση ότι ο αποστολέας δε στέλνει πιο γρήγορα από ότι μπορεί να δεχτεί ο παραλήπτης
- *Έλεγχος συμφόρησης:*
 - Οι αποστολείς μειώνουν το ρυθμό μετάδοσης όταν διαπιστώσουν συμφόρηση στο δίκτυο

Στην άκρη του δικτύου: υπηρεσία χωρίς σύνδεση

- Στόχος: μεταφορά δεδομένων μεταξύ τερματικών συσκευών.
 - Ο ίδιος όπως και πριν!
- **UDP** - User Datagram Protocol [RFC 768]: Η υπηρεσία χωρίς σύνδεση του Internet
 - Μη αξιόπιστη μεταφορά δεδομένων
 - Χωρίς έλεγχο ροής
 - Χωρίς έλεγχο συμφόρησης
- Εφαρμογές που χρησιμοποιούν το TCP:
 - HTTP (WWW), FTP (file transfer), Telnet (remote login), SMTP (email)
- Εφαρμογές που χρησιμοποιούν το UDP:
 - streaming media, teleconferencing, Internet telephony

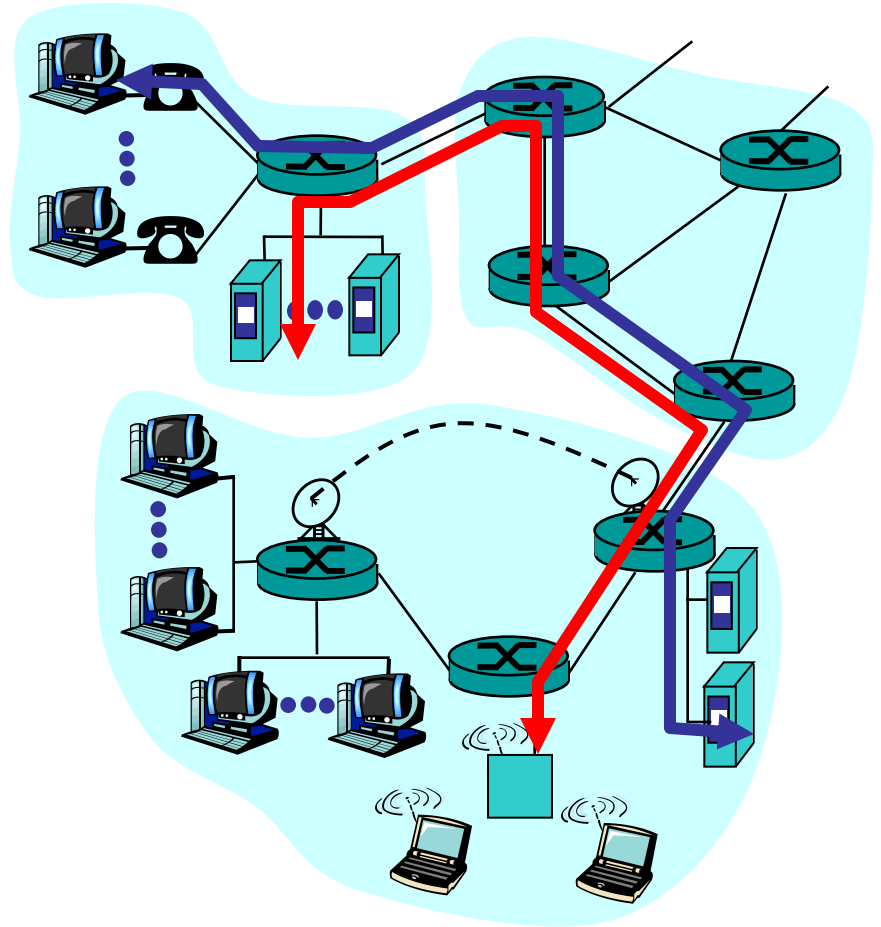
ΣΤΟ ΕΣΩΤΕΡΙΚΟ ΤΟΥ ΔΙΚΤΥΟΥ

- Πλέγμα διασυνδεδεμένων δρομολογητών
- Η βασική ερώτηση: πώς περνούν τα δεδομένα μέσα από το δίκτυο?
 - **Μεταγωγή κυκλωμάτων:** αφιερωμένο κύκλωμα σε κάθε κλήση: όπως στο τηλεφωνικό δίκτυο
 - **Μεταγωγή πακέτων:** τα δεδομένα στέλνονται σαν ξεχωριστά κομμάτια

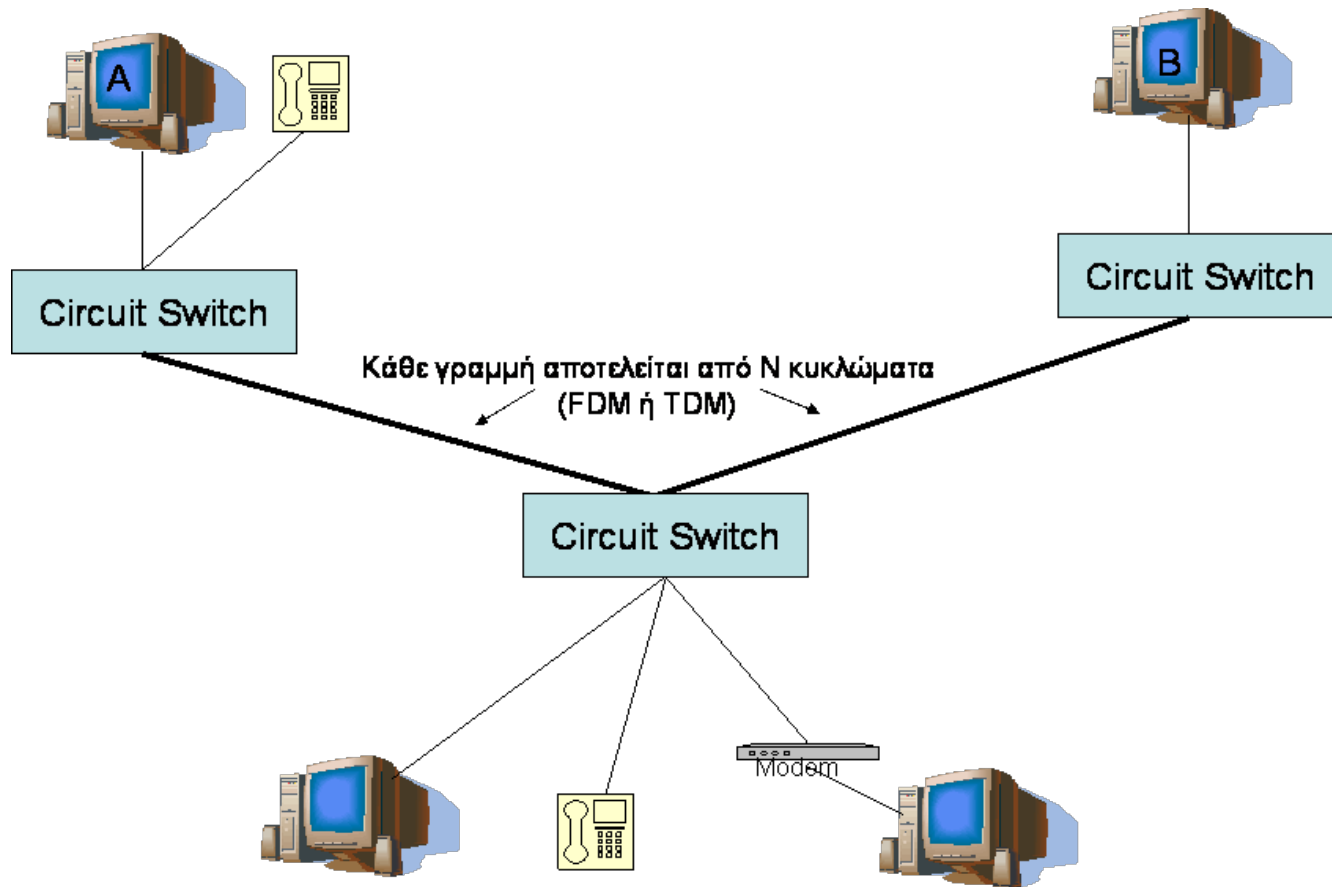


ΣΤΟ ΕΣΩΤΕΡΙΚΟ ΤΟΥ ΔΙΚΤΥΟΥ: ΜΕΤΑΓΩΓΗ ΚΥΚΛΩΜΑΤΟΣ

- Δεσμεύονται πόροι από άκρη σε άκρη για την κλήση
- Εύρος ζώνης της γραμμής, δυνατότητα μεταγωγής
- Αποκλειστικοί πόροι: δεν μοιράζονται
- Εγγυημένη απόδοση
- Απαιτείται διαδικασία εγκατάστασης κλήσης

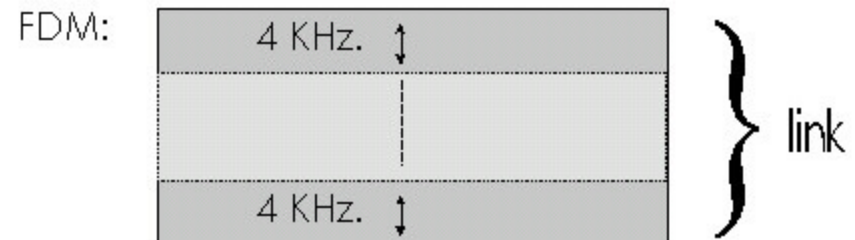


Ένα απλό δίκτυο μεταγωγής κυκλωμάτων

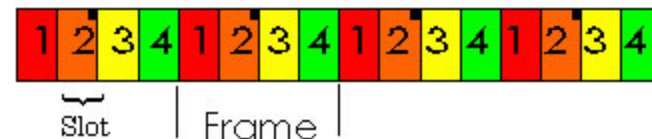


ΣΤΟ ΕΣΩΤΕΡΙΚΟ ΤΟΥ ΔΙΚΤΥΟΥ: ΜΕΤΑΓΩΓΗ ΚΥΚΛΩΜΑΤΟΣ

- Δικτυακοί πόροι (π.χ., εύρος ζώνης) **χωρίζεται σε διαφορετικά μέρη**
- Το κάθε μέρος κατανέμεται σε μια κλήση
- Οι πόροι παραμένουν αδρανείς όσο δεν χρησιμοποιούνται από τους συμμετέχοντες στην κλήση (δεν τους μοιράζονται)
- Πώς χωρίζεται το εύρος ζώνης μιας γραμμής σε διαφορετικά μέρη
 - Διαίρεση συχνότητας
 - Διαίρεση χρόνου (ρυθμός μετάδοσης: 8.000 πλαίσια/δευτ of 8 bits/χρονοθυρίδα → 64Kbps)



TDM:



All slots labelled **2** are dedicated to a specific sender-receiver pair.

ΣΤΟ ΕΣΩΤΕΡΙΚΟ ΤΟΥ ΔΙΚΤΥΟΥ: ΜΕΤΑΓΩΓΗ ΚΥΚΛΩΜΑΤΟΣ

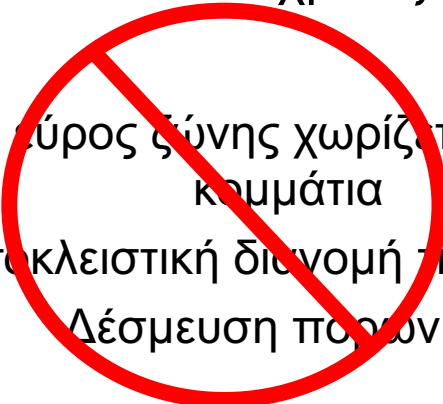
- Μεταγωγή κυκλώματος
- Αρχείο 640 Kbits
- TDM γραμμή των 24 slots (χρονοθυρίδων)
- Ρυθμός μετάδοσης της γραμμής = 1.536Mbps
- Εγκατάσταση κυκλώματος: 500 msec
- Καθυστέρηση στη μετάδοση?

$1.536 \times 10^6 / 24 = 64\text{Kbps}$ για κάθε κύκλωμα

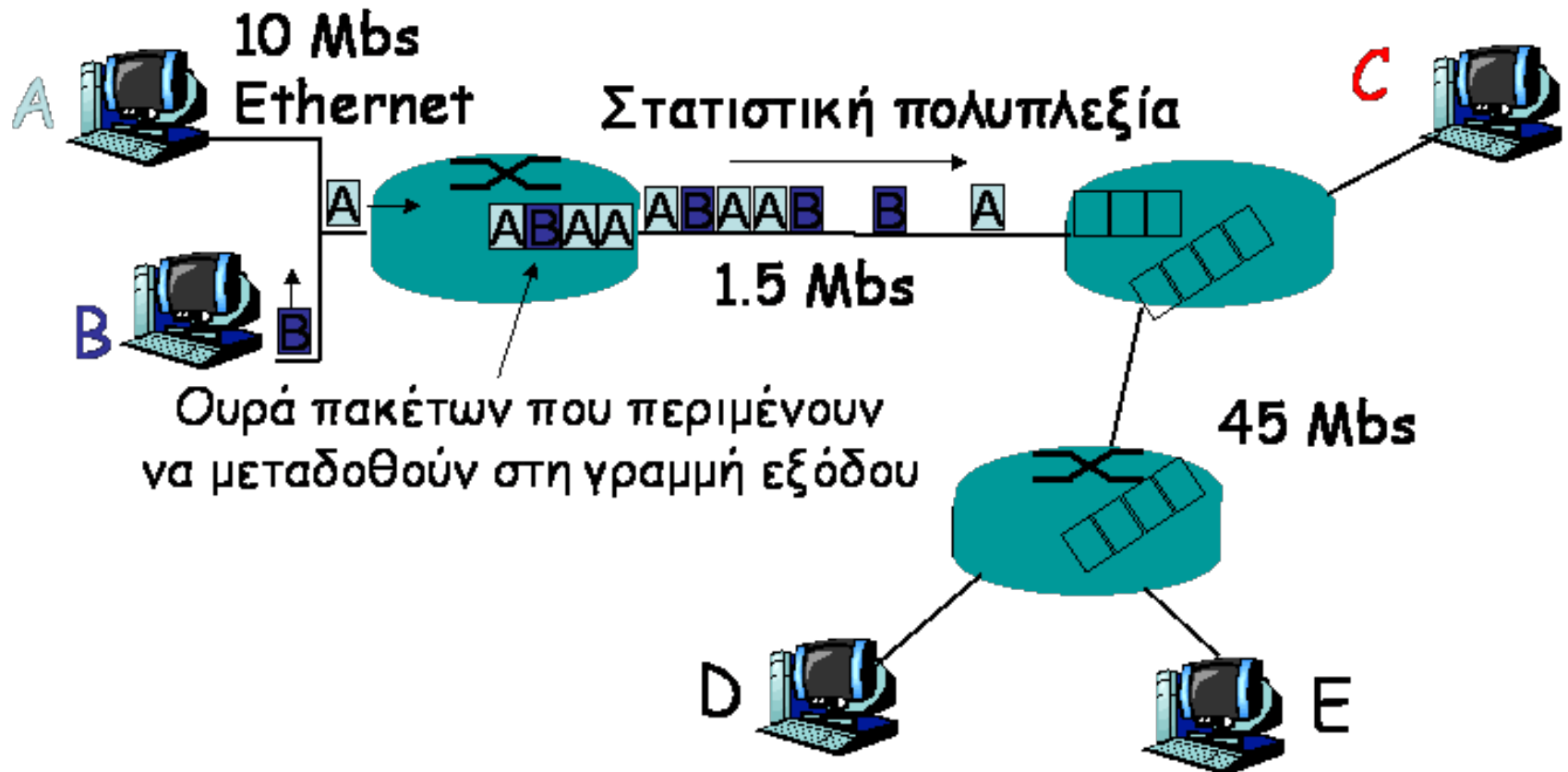
Καθυστέρηση στη μετάδοση = $640\text{Kbits} / 64\text{Kbps} = 10 \text{ sec}$

Συνολικός χρόνος = 10,5 sec

ΣΤΟ ΕΣΩΤΕΡΙΚΟ ΤΟΥ ΔΙΚΤΥΟΥ: ΜΕΤΑΓΩΓΗ ΠΑΚΕΤΩΝ

- Κάθε ροή πακέτων από άκρη σε άκρη χωρίζεται σε πακέτα
 - Τα πακέτα των διαφόρων χρηστών μοιράζονται τους πόρους του δικτύου
 - Κάθε πακέτο χρησιμοποιεί όλο το εύρος ζώνης της γραμμής
 - Οι πόροι χρησιμοποιούνται ΜΟΝΟ όταν χρειάζεται
 - Ανταγωνισμός για τους πόρους:
 - Οι αιτήσεις σε συναθροισμένους πόρους μπορούν να ξεπεράσουν τον αριθμό των διαθέσιμων
 - συμφόρηση: τα πακέτα μπαίνουν σε ουρές και περιμένουν να κάνουν χρήση της γραμμής
 - Αποθήκευση και προώθηση: τα πακέτα μετακινούνται ένα βήμα τη φορά
 - Μετάδοση πάνω από μία γραμμή
 - Περιμένουν τη σειρά τους στην επόμενη γραμμή
- Το εύρος ζώνης χωρίζεται σε κομμάτια
- Αποκλειστική διανομή πόρων
- Δέσμευση πόρων
- 

ΣΤΟ ΕΣΩΤΕΡΙΚΟ ΤΟΥ ΔΙΚΤΥΟΥ: ΜΕΤΑΓΩΓΗ ΠΑΚΕΤΩΝ



- Μεταγωγή πακέτων εναντίον μεταγωγή κυκλωμάτων: αναλογία με τα εστιατόρια!

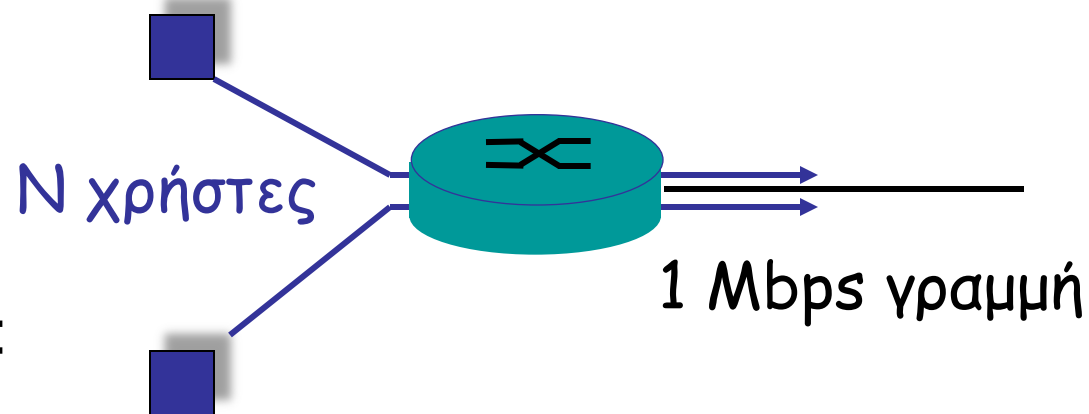
ΣΤΟ ΕΣΩΤΕΡΙΚΟ ΤΟΥ ΔΙΚΤΥΟΥ: ΜΕΤΑΓΩΓΗ ΠΑΚΕΤΩΝ

- Η μεταγωγή πακέτων επιτρέπει σε περισσότερους χρήστες να χρησιμοποιήσουν το δίκτυο
- 1 Mbit γραμμή

- Κάθε χρήστης:
 - 100Kbps όταν είναι ενεργός
 - Ενεργός το 10% του χρόνου

- Μεταγωγή κυκλώματος:
 - 10 χρήστες

- Μεταγωγή πακέτων:
 - με 35 χρήστες, πιθανότητα > 10 ενεργοί μικρότερη του .004



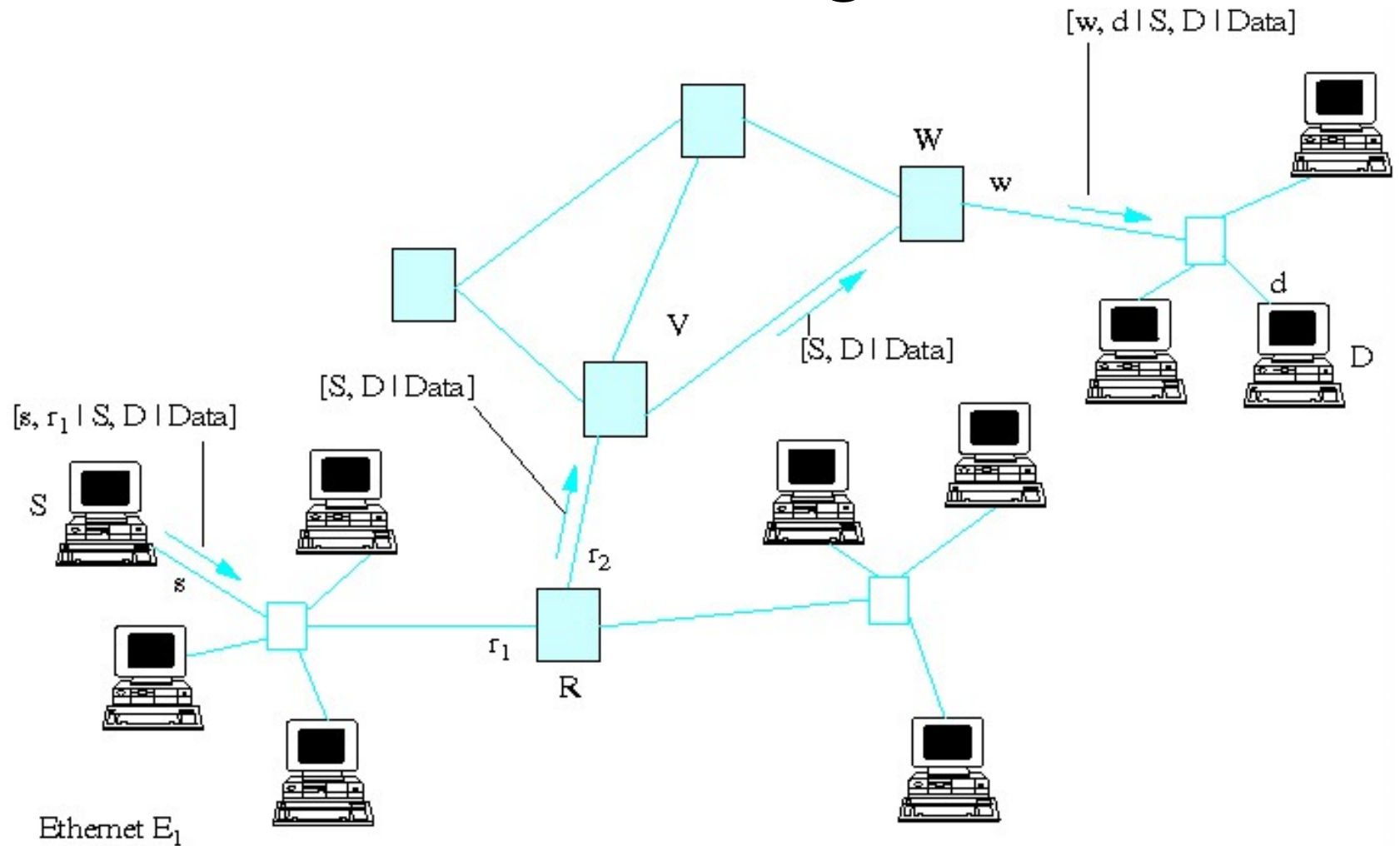
μεταγωγή πακέτων ή μεταγωγή κυκλωμάτων

- Είναι λοιπόν καλύτερη η μεταγωγή πακέτων από τη μεταγωγή κυκλωμάτων;
- Ιδανική για δεδομένα που μεταδίδονται σε ριπές
 - Μοιράζονται οι πόροι
 - Δεν απαιτείται εγκατάσταση κλήσης
- Συμφόρηση: τα πακέτα καθυστερούν ή και χάνονται
 - Χρειάζονται πρωτόκολλα για την ασφαλή μεταφορά δεδομένων και τον έλεγχο συμφόρησης
- Ε: Πώς μπορεί να παρασχεθεί συμπεριφορά όμοια με αυτή της μεταγωγής κυκλώματος όταν χρειάζεται;
 - Εγγυήσεις στο εύρος ζώνης χρειάζεται για εφαρμογές ήχου και video
 - Παραμένει ακόμα ένα άλυτο πρόβλημα για ροές πακέτων που περνάνε μέσα από δίκτυα που διαχειρίζονται διαφορετικοί πάροχοι

Μεταγωγή πακέτων: Δρομολόγηση

- **Στόχος:** να μεταφέρουμε τα πακέτα από την πηγή στον προορισμό περνώντας από διάφορους δρομολογητές
 - Θα μελετήσουμε διάφορους αλγόριθμους επιλογής μονοπατιών
- **datagram δίκτυα:**
 - *Η διεύθυνση προορισμού καθορίζει και το ποιος είναι ο επόμενος κόμβος*
 - Τα μονοπάτια μπορεί να αλλάξουν κατά τη διάρκεια μιας συνεδρίας
 - Αναλογία: οδηγώντας και ρωτώντας συνεχώς για κατευθύνσεις
- **virtual circuit δίκτυα:**
 - Κάθε πακέτο μεταφέρει μια ετικέτα (virtual circuit ID), που καθορίζει και το επόμενο βήμα
 - Το μονοπάτι είναι σταθερό κατά τη διάρκεια της κλήσης και καθορίζεται στη φάση εγκατάστασης της κλήσης
 - Οι δρομολογητές διατηρούν πληροφορία για την κατάσταση της κάθε κλήσης

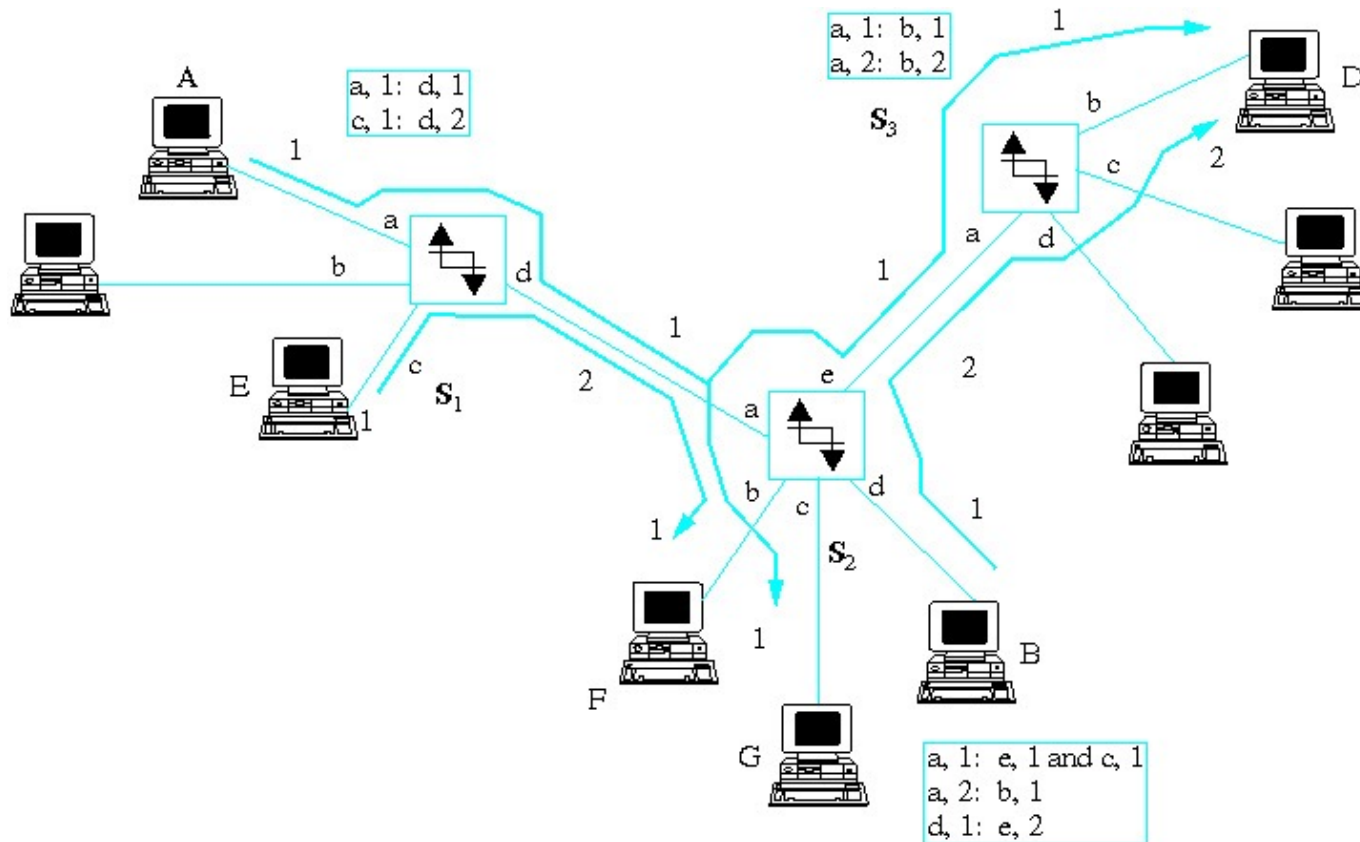
Δίκτυα datagrams



Asynchronous Transfer Mode (ATM)

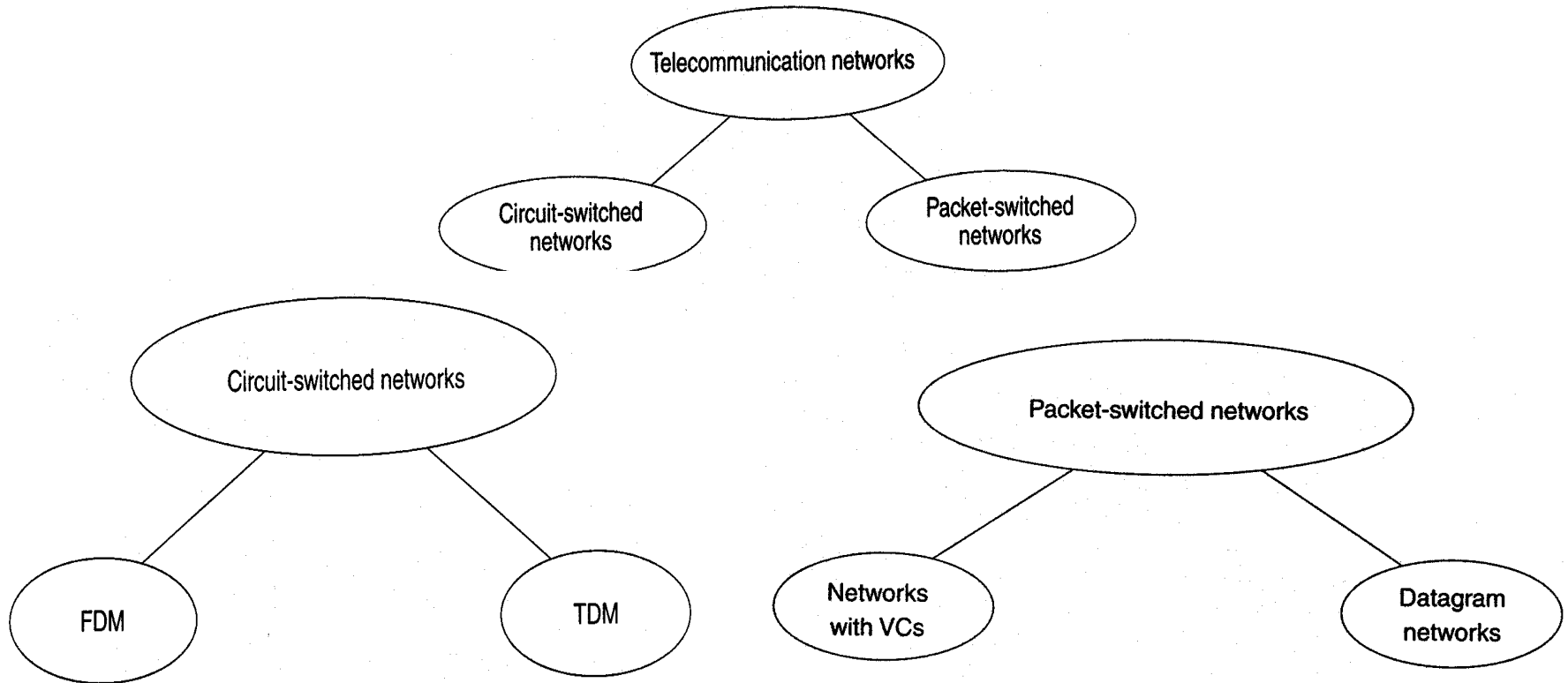
- Πακέτα των 53 bytes (cells)
- Επιλογή του νοητού κυκλώματος με βάση τους διαθέσιμους πόρους πριν την μετάδοση
- Δέσμευση πόρων και ενημέρωση των μεταγωγών/πηγής
- Απόρριψη σύνδεσης, δυνατή λόγω έλλειψης πόρων
- Απελευθέρωση πόρων στο τέλος της σύνδεσης
- Η επικεφαλίδα των cells φέρει κάποιον αριθμό σύνδεσης βάση του οποίου γίνεται η δρομολόγηση
- ATM σχεδιασμένο για υποστήριξη επιθυμητής ποιότητας ενός μεγάλου εύρους εφαρμογών

Δρομολόγηση



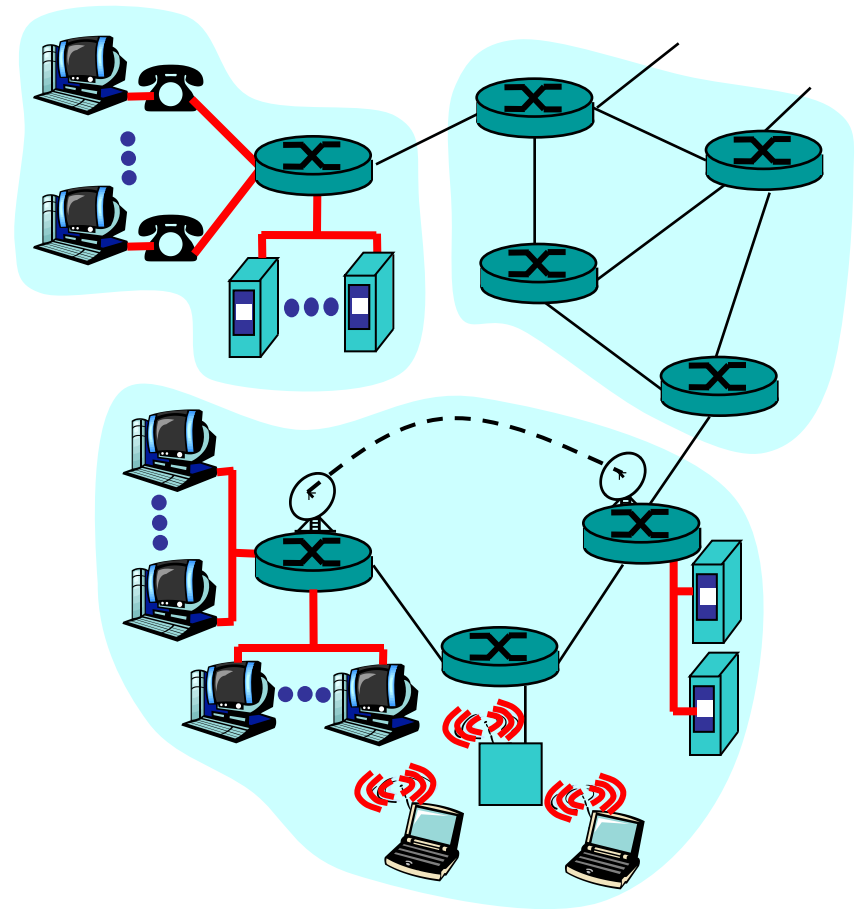
- Τρία VC : $A \rightarrow D, G$, $E \rightarrow F$, $B \rightarrow D$
- Αρίθμηση VCs με τον μικρότερο διαθέσιμο αριθμό
- (ζεύξη εισόδου, VC#) \rightarrow (ζεύξη εξόδου, VC#)
- Virtual Paths (εικονικά μονοπάτια) για περιγραφή VCs τα οποία έχουν κοινή δρομολόγηση - εξοικονόμηση μνήμης και αύξηση ταχύτητας

Ταξινόμηση δικτύων



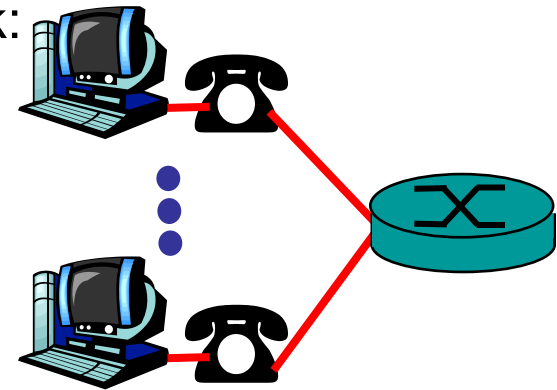
Δίκτυα πρόσβασης και φυσικά μέσα

- *Ε: Πώς συνδέονται οι τερματικές συσκευές με τους δρομολογητές;*
 - Δίκτυα πρόσβασης στα σπίτια
 - σε Ιδρύματα, εταιρείες
 - Κινητά δίκτυα πρόσβασης
- *Θυμηθείτε:*
 - Εύρος ζώνης (bits per second) των δικτύων πρόσβασης?
 - Μοιραζόμενο ή αποκλειστικό?



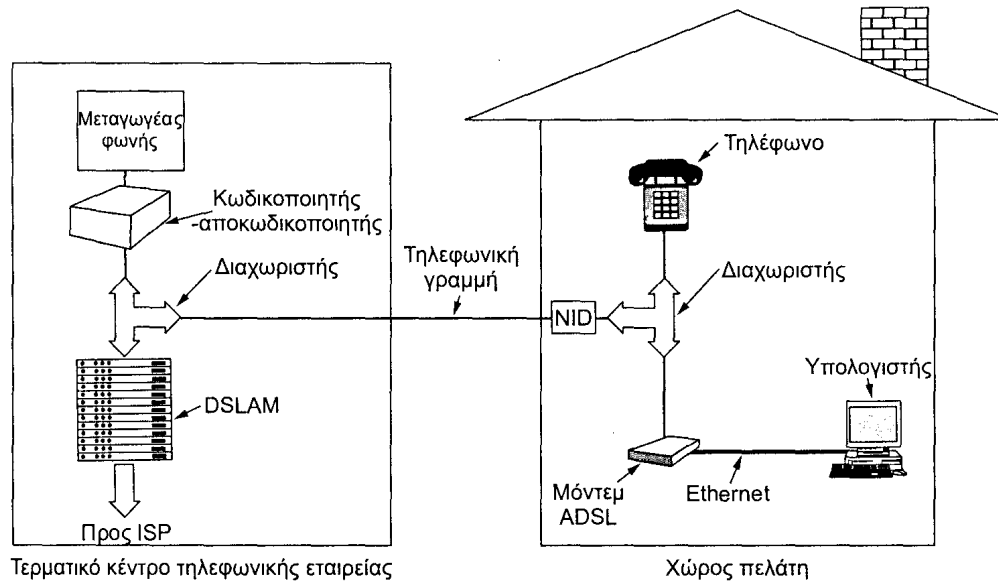
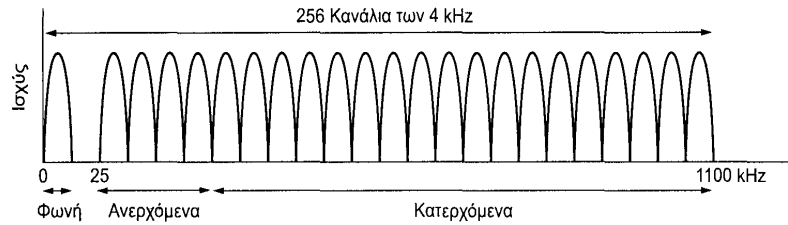
Οικιακά δίκτυα πρόσβασης: πρόσβαση σημείου προς σημείο

- **Dialup μέσω modem**
 - μέχρι 56Kbps απευθείας πρόσβαση σε δρομολογητή (ιδεατά)
- **ISDN:** integrated services digital network: 128Kbps ψηφιακή σύνδεση μέχρι το δρομολογητή
- **ADSL:** asymmetric digital subscriber line
 - μέχρι 1 Mbps home-to-router (4-50KHz)
 - μέχρι 8 Mbps router-to-home (50KHz-1MHz)
 - An ordinary POTS (0-4KHz)
 - Ανάπτυξη ADSL



xDSL (Digital Subscriber Line)

ADSL

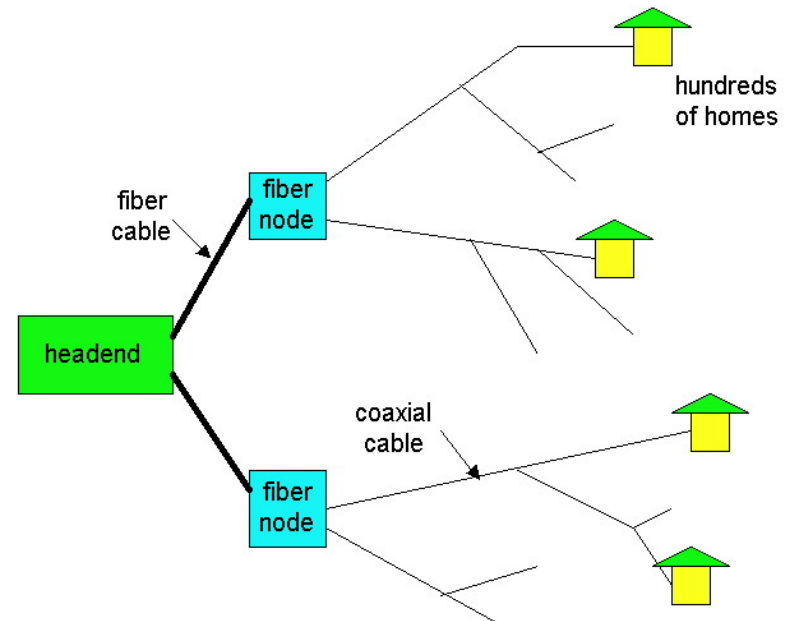


DSLAM: Digital Subscriber Line Access Multiplexer

NID: Network Interface Device

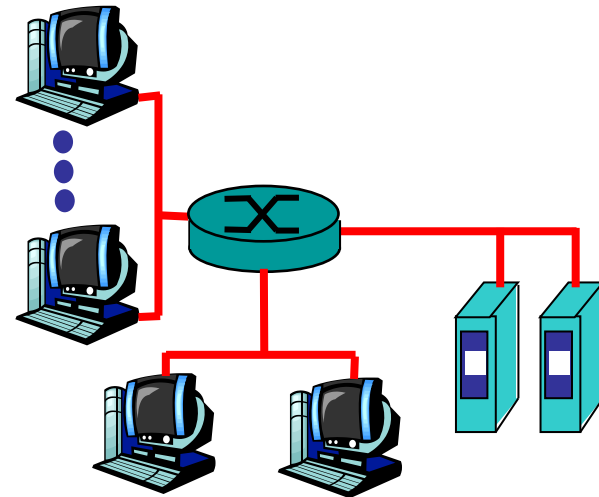
Οικιακά δίκτυα : καλωδιακά modems

- **HFC: hybrid fiber coax**
 - Ασύμμετρο: π.χ., μέχρι 10Mbps upstream, 1 Mbps downstream
- Ένα δίκτυο καλωδίων και οπτικών ινών συνδέει τα σπίτια με τον δρομολογητή του ISP
 - Μοιράζεται η πρόσβαση στο δρομολογητή μεταξύ των σπιτιών
 - Θέματα: συμφόρηση, διαστασιοποίηση
- Ανάπτυξη: διαθέσιμα από τις καλωδιακές εταιρείες



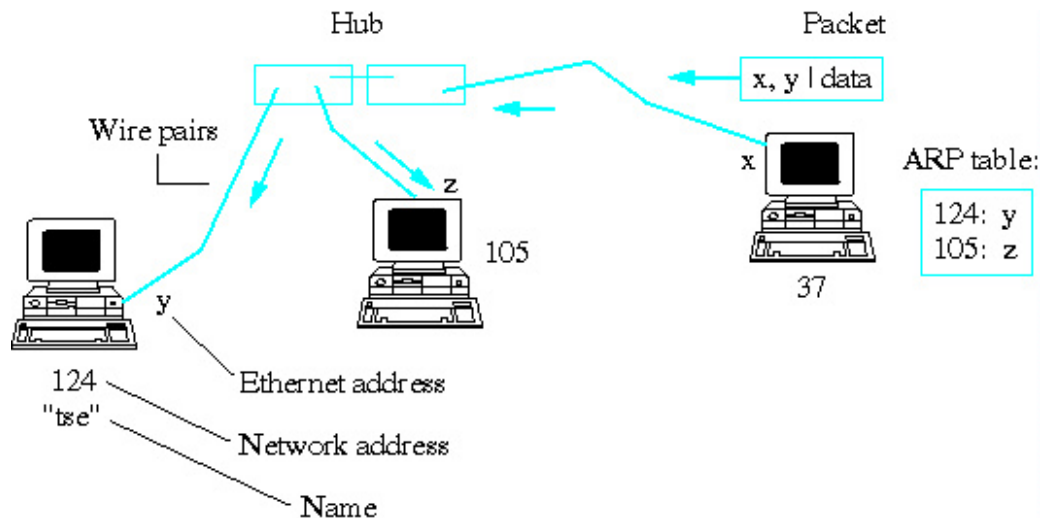
Δίκτυα πρόσβασης σε Ιδρύματα/εταιρίες: Τοπικά δίκτυα πρόσβασης

- Τα τοπικά δίκτυα (Local Area Networks- LANs) συνδέουν τις τερματικές συσκευές με ένα δρομολογητή
- **Ethernet:**
 - Μοιραζόμενο ή αποκλειστικό καλώδιο συνδέει τερματικά συστήματα μεταξύ τους και με τον δρομολογητή
 - 10 Mbs, 100Mbps, Gigabit Ethernet
- **Ανάπτυξη:** Ιδρύματα, εταιρίες, σύντομα και σε σπίτια?



Ethernet

- **Shared Ethernet: Hubs and Collisions**
(συγκρούσεις)



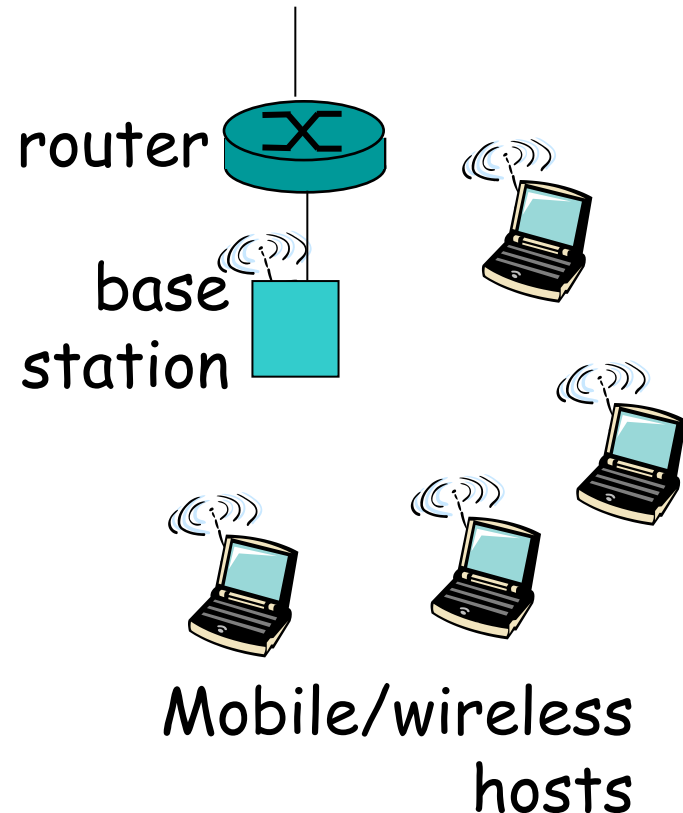
- Η/Υ συνδεδεμένοι σε ένα Hub. Πολλά συνδεδεμένα Hubs.
- Τα Hubs επαναμεταδίδουν πακέτα προς όλες τις θυρίδες (ζεύξεις).
- Κάθε σύνολο διασυνδεδεμένων Hubs ορίζει ένα πεδίο συγκρούσεως (collision domain).

Διευθύνσεις

- Όνομα computer: Ermis
- Δικτυακή διεύθυνση υπολογιστή: 124 (λογική τοποθεσία)
- Ethernet διεύθυνση υπολογιστή: y (physical, LAN, MAC)
 - Ορίζεται από adapter (card) - static
 - 6-byte / 48-bit (2^{48} δυνατές) - Δεκαεξαδική περιγραφή: F8-37-B1-1F-33-BA
 - Πρώτα 24 bits ορίζονται από IEEE / Τελευταία 24 bits από τον κατασκευαστή
- Πλεονεκτήματα των φυσικών διευθύνσεων:
 - Ανεξάρτητες από διευθύνσεις επιπέδου δικτύου (e.g., IP)
 - Δεν επηρεάζονται από αλλαγή δικτυακής διεύθυνσης λόγω μετακίνησης.
 - Περιορίζει στην διεπαφή την επεξεργασία των διερχομένων πακέτων, χωρίς να μεταφέρονται / επεξεργάζονται από τις διεργασίες επιπέδου δικτύου.

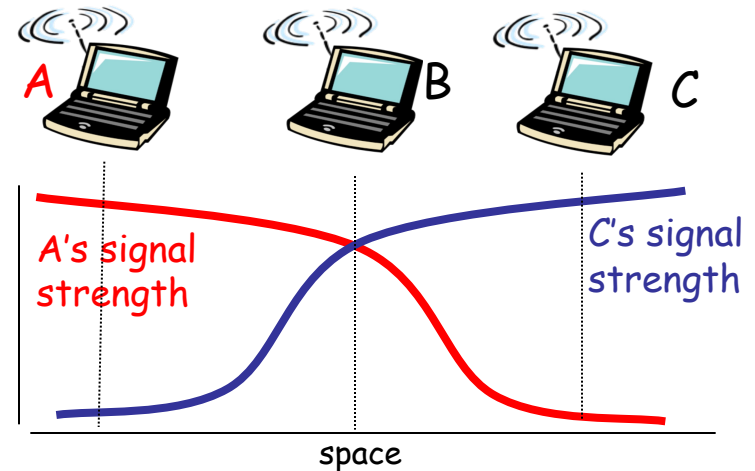
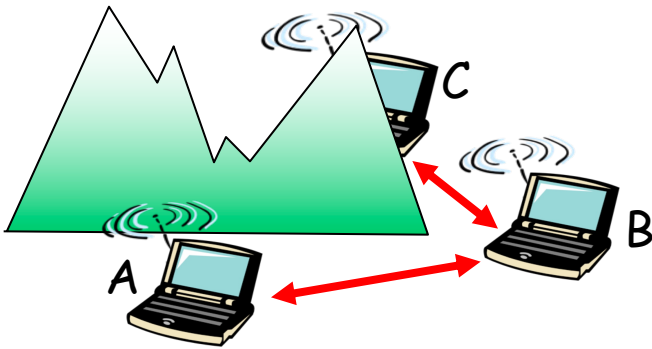
Ασύρματα δίκτυα πρόσβασης

- Μοιραζόμενο ασύρματο δίκτυο πρόσβασης, συνδέει τερματικές συσκευές με τον δρομολογητή
- **wireless LANs:**
 - Το ράδιο φάσμα αντικαθιστά τα σύρματα
 - 802.11.χ έχει ραγδαία ανάπτυξη τα τελευταία χρόνια (ταχύτητες έως και 50 Mbps)
- **wider-area wireless access**
 - LTE – 100 Mbps



Wireless network characteristics

Multiple wireless senders and receivers create additional problems (beyond multiple access):



Hidden terminal problem

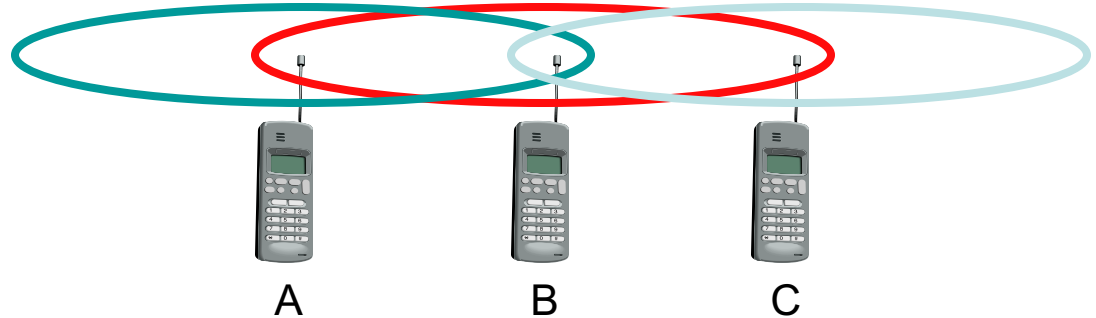
- B, A hear each other
 - B, C hear each other
 - A, C can not hear each other
- means A, C unaware of their interference at B

Signal attenuation:

- B, A hear each other
- B, C hear each other
- A, C can not hear each other interfering at B

Exposed terminals

- Exposed terminals
 - B sends to A, C wants to send to another terminal (not A or B)
 - C has to wait, CS signals a medium in use
 - but A is outside the radio range of C, therefore waiting is not necessary
 - C is “exposed” to B



Wireless Link Characteristics (1)

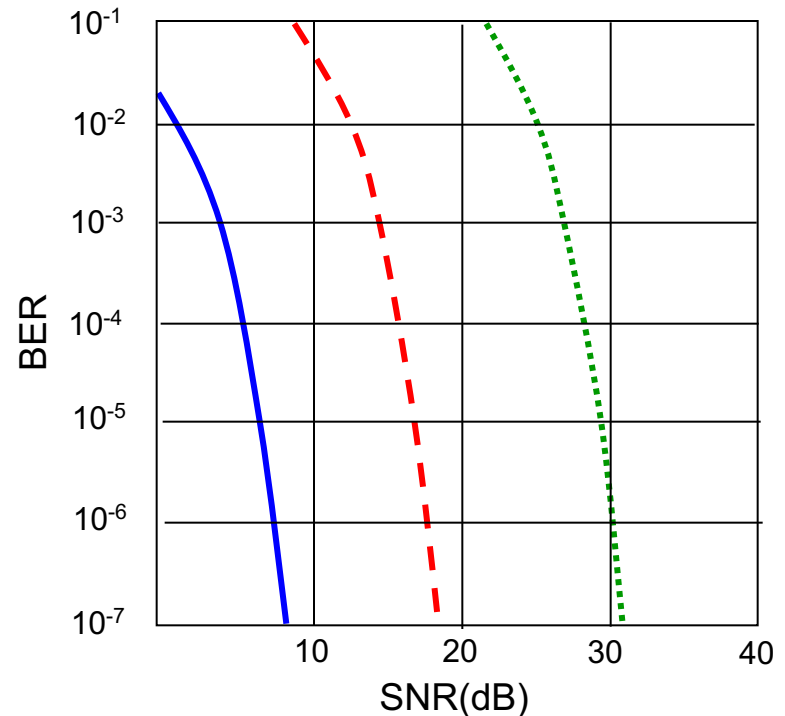
Differences from wired link

- **decreased signal strength:** radio signal attenuates as it propagates through matter (path loss)
- **interference from other sources:** standardized wireless network frequencies (e.g., 2.4 GHz) shared by other devices (e.g., phone); devices (motors) interfere as well
- **multipath propagation:** radio signal reflects off objects ground, arriving at destination at slightly different times

.... make communication across (even a point to point) wireless link much more “difficult”

Wireless Link Characteristics (2)

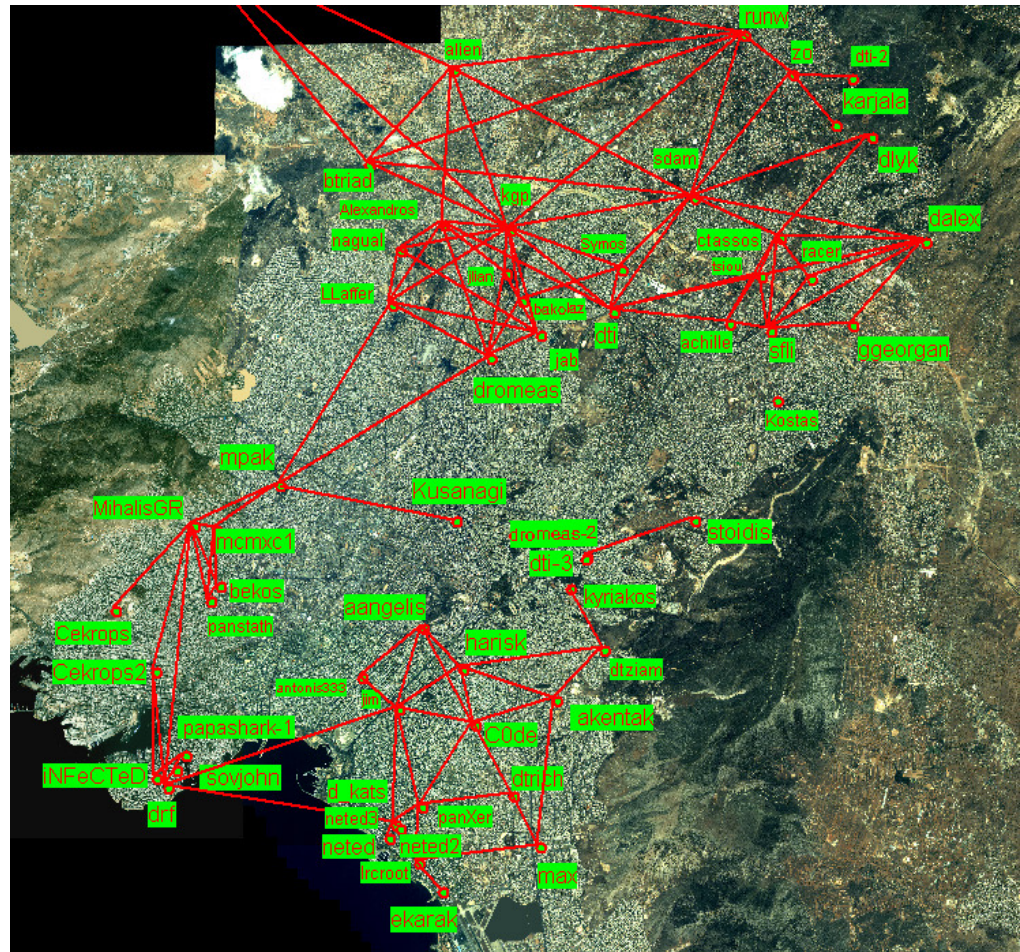
- SNR: signal-to-noise ratio
 - larger SNR – easier to extract signal from noise (a “good thing”)
- *SNR versus BER tradeoffs*
 - *given physical layer*: increase power \rightarrow increase SNR \rightarrow decrease BER
 - *given SNR*: choose physical layer that meets BER requirement, giving highest throughput
 - SNR may change with mobility: dynamically adapt physical layer (modulation technique, rate)



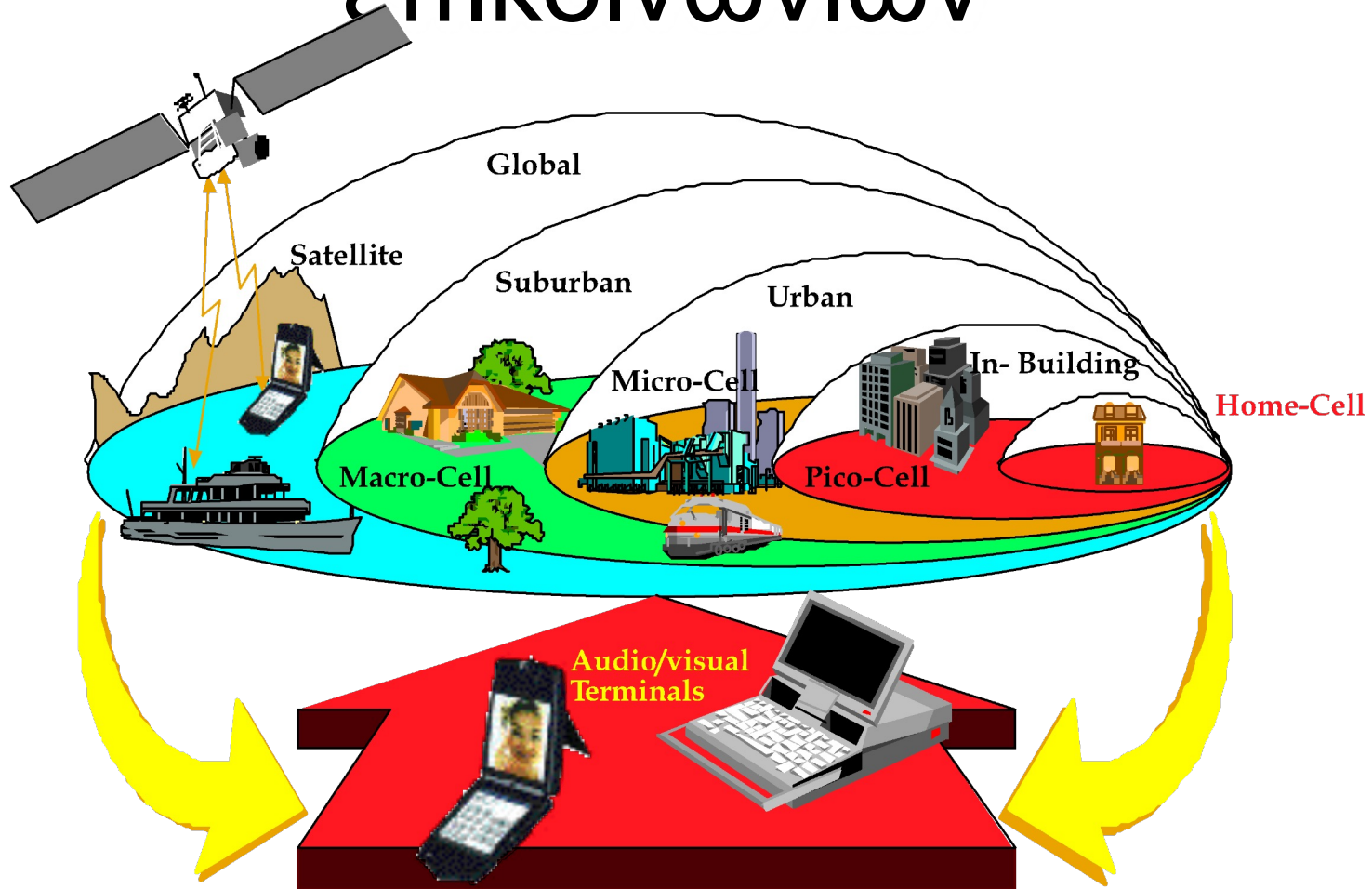
- QAM256 (8 Mbps)
- - - QAM16 (4 Mbps)
- BPSK (1 Mbps)

WLAN

- Σύλλογοι προώθησης ασύρματων δικτύων στην Ελλάδα
- <http://www.athenswireless.net/>



Κυψελωτά συστήματα ΕΠΙΚΟΙΝΩΝΙΩΝ



Inter-Network Roaming
Seamless end-to-end Service

Φυσικά μέσα

- **Φυσική γραμμή:**
μετάδοση των δεδομένων σε bits διαδίδεται πάνω από τη γραμμή
 - **Κατευθυνόμενα μέσα:**
 - Τα σήματα διαδίδονται σε στέρεα μέσα: χαλκός, ππική ίνα
 - **Μη κατευθυνόμενα μέσα:**
 - Τα σήματα διαδίδονται ελεύθερα στο χώρο: ραδιοκύματα
- Συστρεμμένο ζεύγος
 - Category 3: τυπικά τηλεφωνικά καλώδια, 10 Mbps ethernet
 - Category 5 TP: 100Mbps ethernet



Φυσικό μέσο: ομοαξονικό καλώδιο, οπτική ίνα

- **Ομοαξονικό καλώδιο:**
- Χάλκινος αγωγός (φορέας σήματος) μέσα σε προστατευτικό περίβλημα
 - baseband: ένα κανάλι στο καλώδιο
 - broadband: πολλαπλά κανάλια στο καλώδιο
- Δι-κατευθυντικό
- Εκτεταμένη χρήση σε 10Mbs Ethernet
- **Καλώδιο οπτικής ίνας:**
- Ίνα από γυαλί που μεταφέρει παλμούς φωτός
- Χαμηλή εξασθένιση
- Επίτευξη υψηλών ταχυτήτων:
 - Gigabit Ethernet
- Πολύ χαμηλός ρυθμός λαθών

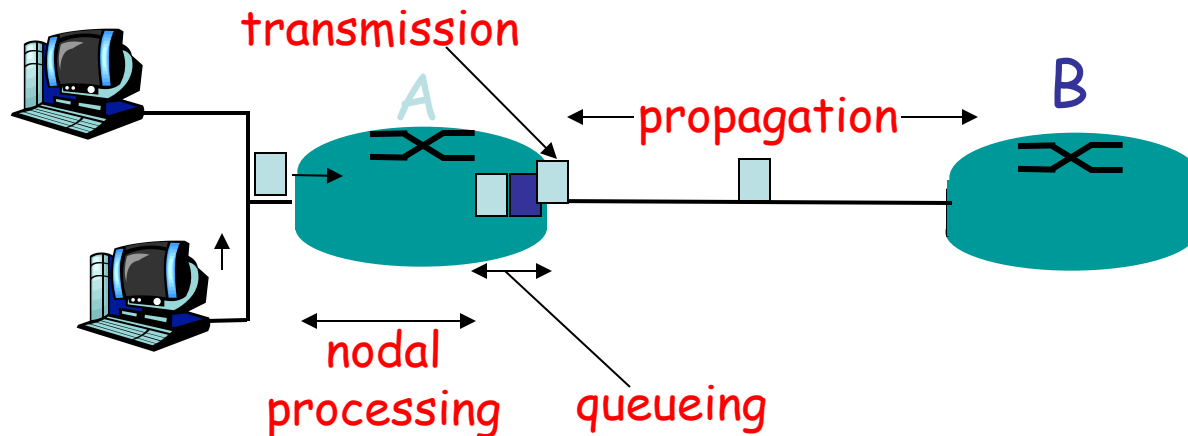


Φυσικό μέσο: ραδιοκύματα

- Το σήμα μεταφέρεται στο ηλεκτρομαγνητικό φάσμα
- Δεν υπάρχει φυσικό «σύρμα»
- Χρησιμοποιείται σε τοπικά δίκτυα αλλά και σε δίκτυα πιο ευρείας περιοχής στην περίπτωση των δικτύων κινητών επικοινωνιών
- Επιδράσεις στη μετάδοση:
 - ανάκλαση
 - Εμπόδια από αντικείμενα
 - Παρεμβολές
- **Τύποι ράδιο ζεύξεων:**
- **μικροκυματικές**
 - Π.χ., . κανάλια μέχρι τα 45 Mbps
- **LAN** (e.g., 802.11.b)
 - 10 Mbps
- **wide-area** (e.g., κυψελωτά)
 - LTE/EPC, 50 Mbps
- **δορυφορικά**
 - Κανάλια μέχρι τα 50Mbps (ή πολλαπλά μικρότερα)
 - 270 msec end-end delay
 - γεωστατικοί και LEOS

Καθυστέρηση στα δίκτυα μεταγωγής πακέτων

- Τα πακέτα υφίστανται μια καθυστέρηση στο μονοπάτι από τη μία άκρη μέχρι την άλλη
- **Τέσσερις** πηγές καθυστέρησης σε κάθε βήμα
- Κομβική καθυστέρηση επεξεργασίας:
 - Έλεγχος για λάθη στα bits
 - Καθορισμός γραμμής εξόδου
- Καθυστέρηση αναμονής
 - Χρόνος αναμονής στη γραμμή εξόδου για τη μετάδοση
 - Εξαρτάται από το επίπεδο συμφόρησης στο δρομολογητή



Καθυστέρηση στα δίκτυα μεταγωγής πακέτων

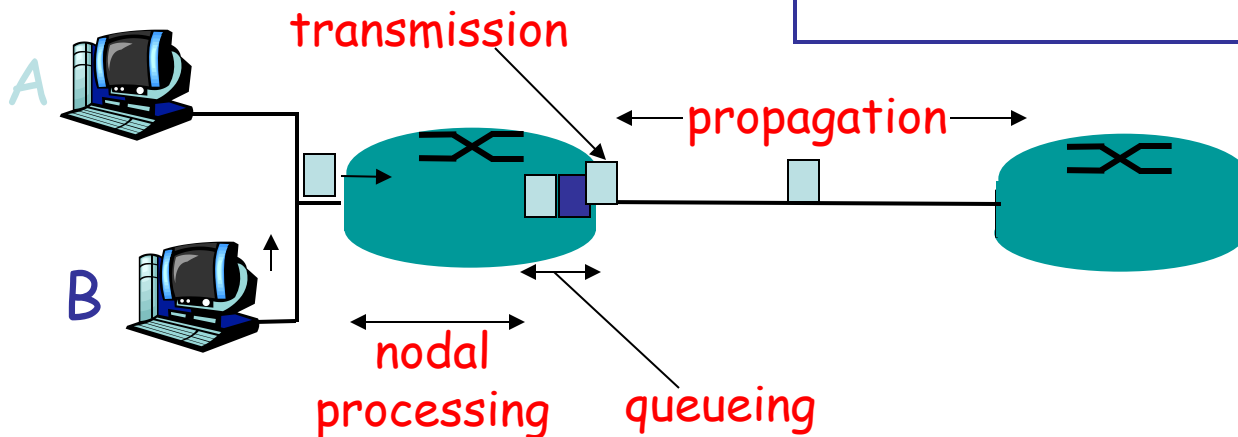
•Καθυστέρηση μετάδοσης:

- R =link bandwidth (bps)
- L =packet length (bits)
- Χρόνος αποστολής των bits στη γραμμή = L/R

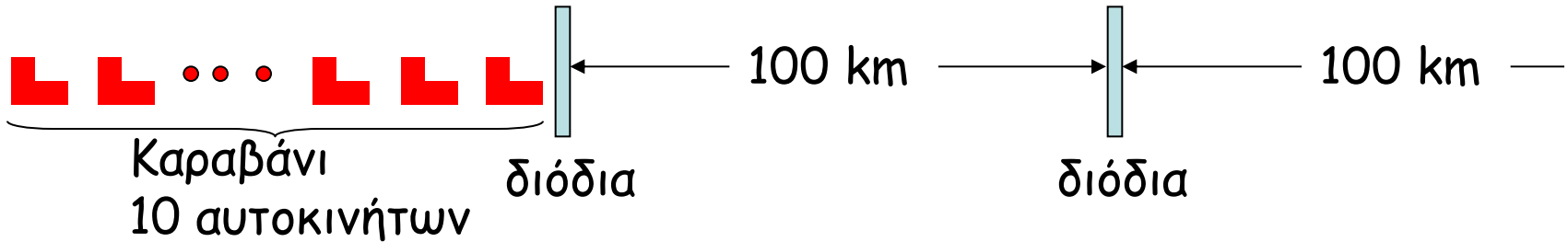
•Καθυστέρηση διάδοσης:

- d = length of physical link (m)
- s = propagation speed in medium (π.χ., 2×10^8 m/sec)
- propagation delay = d/s

Σημείωση: Τα s & R είναι πολύ διαφορετικές ποσότητες



Παράδειγμα



Αυτοκίνητα κινούνται με
100Km/h

Διόδια εξυπηρετούν ένα
αυτοκίνητο κάθε 12 sec
(transmission time)

car~bit; caravan ~ packet

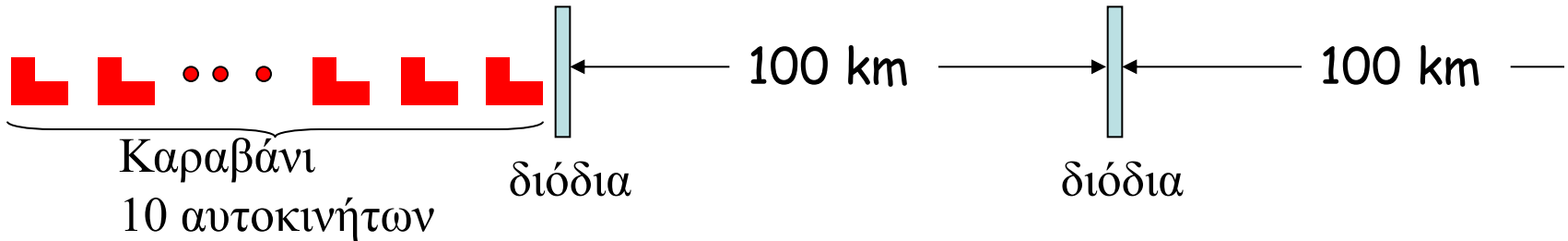
Q: Πόσος χρόνος περνάει μέχρι να μαζευτούν στα δεύτερα διόδια αν τα πρώτα αυτοκίνητα περιμένουν και το τελευταίο πριν αρχίσουν να περνάνε

Όλα τα αυτοκίνητα περνάνε από ένα σταθμό σε $= 12 \cdot 10 = 120$ sec

Χρόνος που χρειάζεται ένα αμάξι για να διασχίσει τη διαδρομή ανάμεσα στους σταθμούς:
 $100\text{km}/(100\text{km/hr}) = 1$ hr

A: 62 minutes

Παράδειγμα



Αυτοκίνητα κινούνται με
1000Km/h

Διόδια εξυπηρετούν ένα
αυτοκίνητο κάθε 1 λεπτό
(transmission time)

Q: Θα φθάσουν αυτοκίνητα
στο δεύτερο σταθμό πριν
φύγουν όλα από τον
πρώτο?

ΝΑΙ! Μετά από 7 λεπτά το πρώτο
αυτοκίνητο είναι στο δεύτερο
σταθμό και 3 αυτοκίνητα στον
πρώτο.

1ο bit ενός πακέτου φθάνει στο
δεύτερο δρομολογητή πριν όλο
το πακέτο μεταδοθεί από τον
πρώτο δρομολογητή

Κομβική καθυστέρηση

$$d_{\text{nodal}} = d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}}$$

d_{proc} = processing delay

- typically a few microseconds or less

d_{queue} = queuing delay

- depends on congestion

d_{trans} = transmission delay

- = L/R , significant for low-speed links

d_{prop} = propagation delay

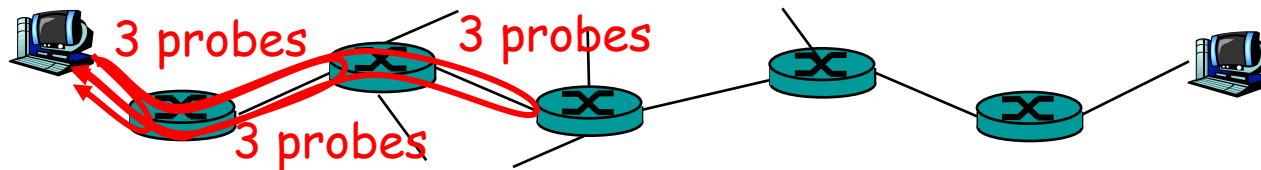
- a few microseconds to hundreds of msecs

Πραγματικές διαδρομές και καθυστερήσεις στο Διαδίκτυο

What do “real” Internet delay & loss look like?

Traceroute program: provides delay measurement from source to router along end-end Internet path towards destination. For all i :

- sends three packets that will reach router i on path towards destination
- router i will return packets to sender
- sender times interval between transmission and reply.



“Real” Internet delays and routes

traceroute: gaia.cs.umass.edu to www.eurecom.fr

Three delay measurements from
gaia.cs.umass.edu to cs-gw.cs.umass.edu

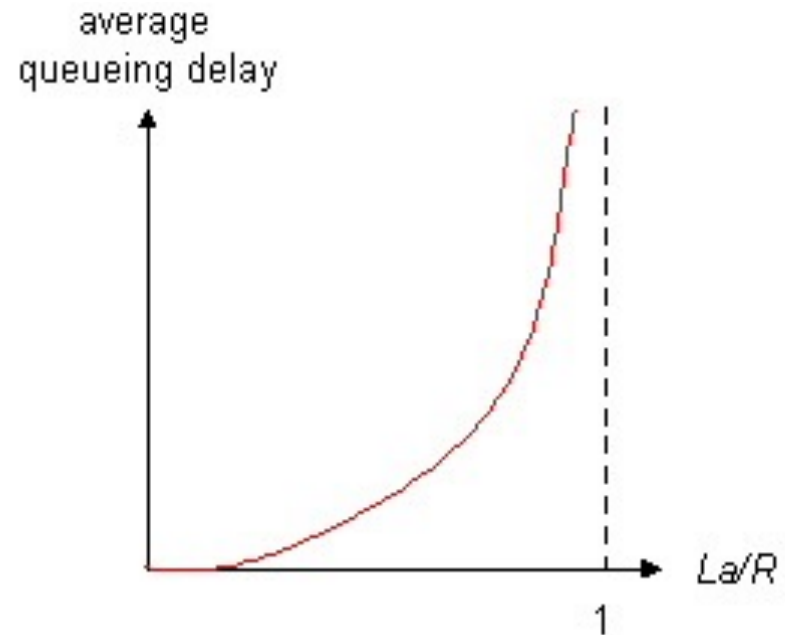
```
1 cs-gw (128.119.240.254) 1 ms 1 ms 2 ms
2 border1-rt-fa5-1-0.gw.umass.edu (128.119.3.145) 1 ms 1 ms 2 ms
3 cht-vbns.gw.umass.edu (128.119.3.130) 6 ms 5 ms 5 ms
4 jn1-at1-0-0-19.wor.vbns.net (204.147.132.129) 16 ms 11 ms 13 ms
5 jn1-so7-0-0-0.wae.vbns.net (204.147.136.136) 21 ms 18 ms 18 ms
6 abilene-vbns.abilene.ucaid.edu (198.32.11.9) 22 ms 18 ms 22 ms
7 nycm-wash.abilene.ucaid.edu (198.32.8.46) 22 ms 22 ms 22 ms
8 62.40.103.253 (62.40.103.253) 104 ms 109 ms 106 ms
9 de2-1.de1.de.geant.net (62.40.96.129) 109 ms 102 ms 104 ms
10 de.fr1.fr.geant.net (62.40.96.50) 113 ms 121 ms 114 ms
11 renater-gw.fr1.fr.geant.net (62.40.103.54) 112 ms 114 ms 112 ms
12 nio-n2.cssi.renater.fr (193.51.206.13) 111 ms 114 ms 116 ms
13 nice.cssi.renater.fr (195.220.98.102) 123 ms 125 ms 124 ms
14 r3t2-nice.cssi.renater.fr (195.220.98.110) 126 ms 126 ms 124 ms
15 eurecom-valbonne.r3t2.ft.net (193.48.50.54) 135 ms 128 ms 133 ms
16 194.214.211.25 (194.214.211.25) 126 ms 128 ms 126 ms
17 * * *
18 * * *
19 fantasia.eurecom.fr (193.55.113.142) 132 ms 128 ms 136 ms
```

trans-oceanic
link

* means no response (probe lost, router not replying)

Θεωρία ουρών

- R =link bandwidth (bps)
- L =packet length (bits)
- a =average packet arrival rate
- traffic intensity = $\lambda a/R$



Επίπεδα πρωτοκόλλων

- Τα δίκτυα είναι πολύπλοκα
- Πολλά κομμάτια:
 - Hosts (τερματικές συσκευές)
 - Routers (δρομολογητές)
 - Γραμμές διαφόρων μέσων
 - Applications (εφαρμογές)
 - Protocols (πρωτόκολλα)
 - hardware, software (υλικό, λογισμικό)

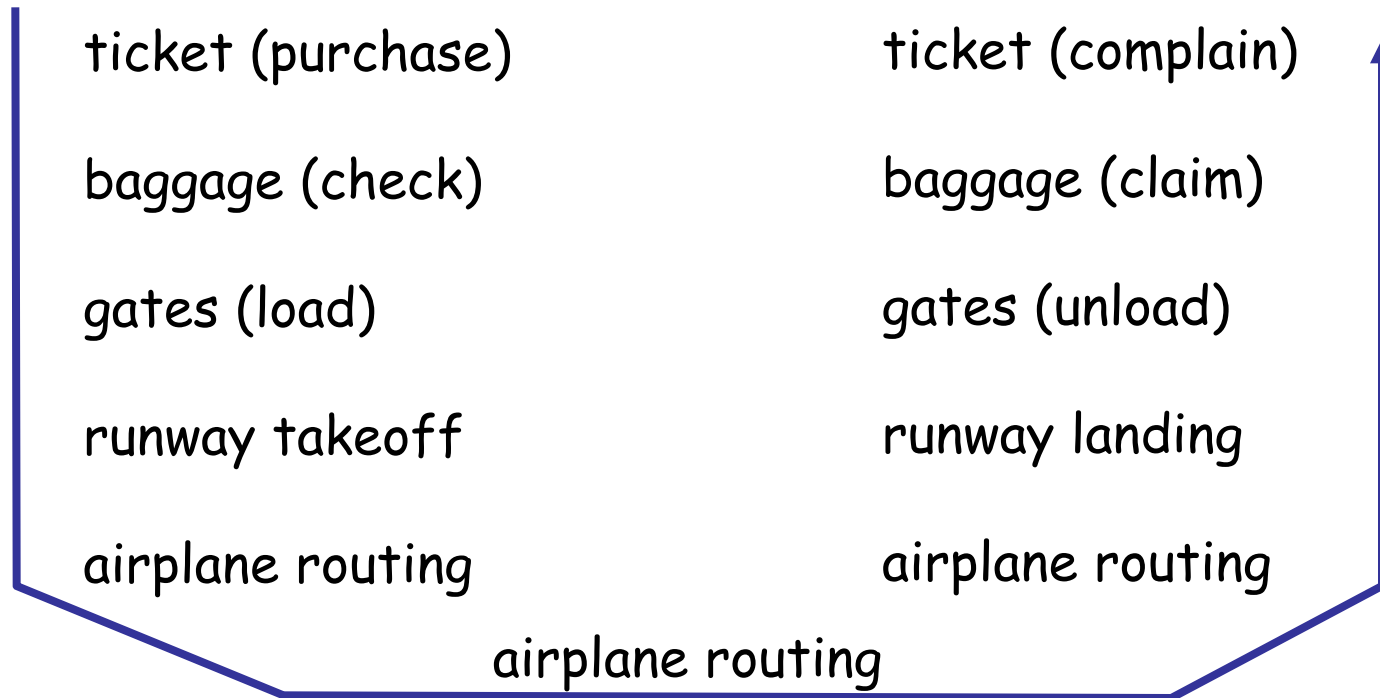
Ερώτηση:

Πώς μπορούμε να οργανώσουμε τη δομή των δικτύων

Ή έστω

τη συζήτηση μας για τα δίκτυα

Οργάνωση ενός αεροπορικού ταξιδιού



- Μια σειρά βημάτων

Οργάνωση ενός αεροπορικού ταξιδιού

ticket (purchase)	ticket (complain)
baggage (check)	baggage (claim)
gates (load)	gates (unload)
runway takeoff	runway landing
airplane routing	airplane routing
airplane routing	

- **Επίπεδα:** Κάθε επίπεδο υλοποιεί μια υπηρεσία
 - Εκτελώντας εσωτερικές λειτουργίες στο ίδιο επίπεδο
 - Βασίζεται σε υπηρεσίες των πιο κάτω επιπέδων

Οργάνωση ενός αεροπορικού ταξιδιού: Υπηρεσίες

Counter-to-counter delivery of person+bags

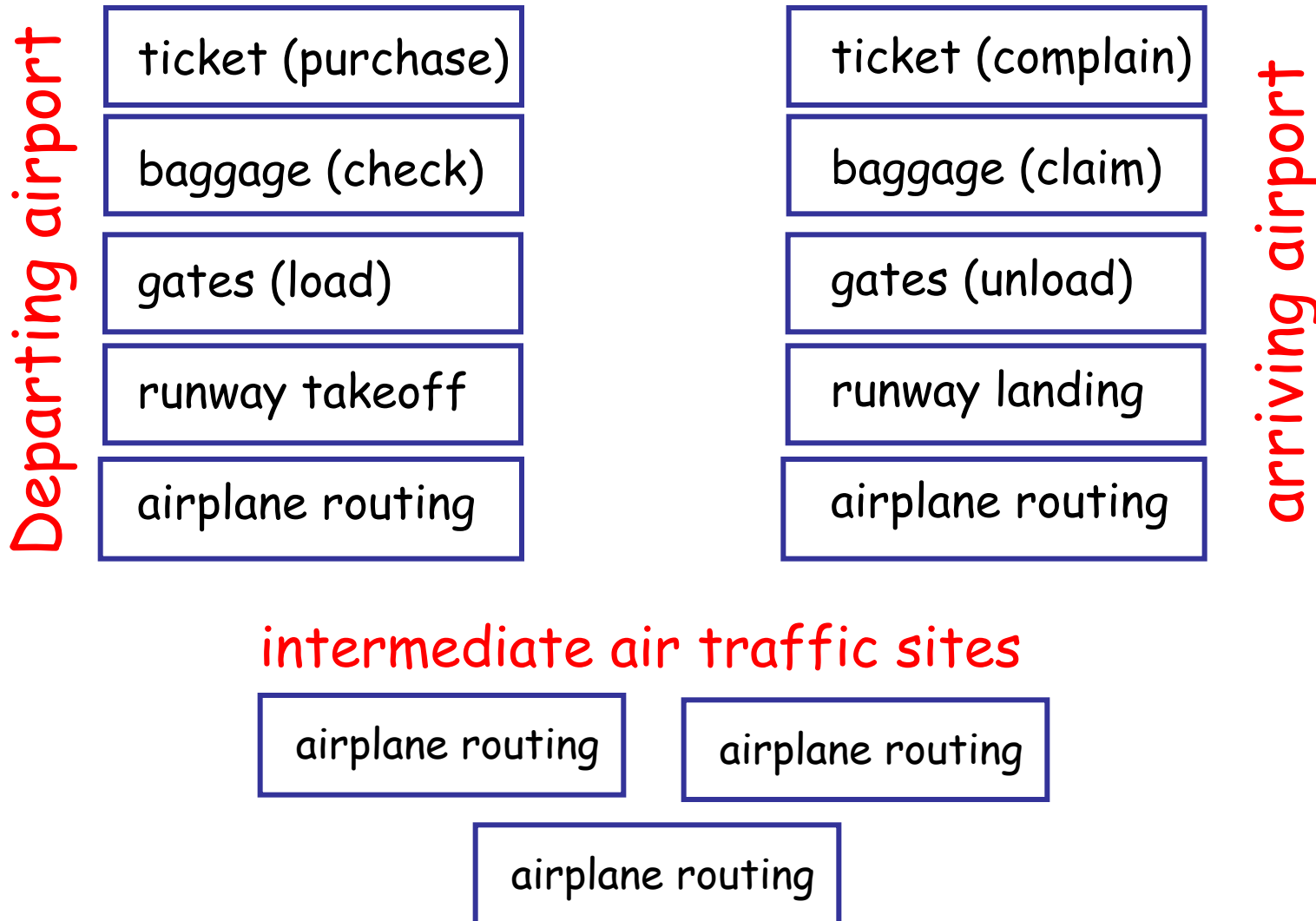
baggage-claim-to-baggage-claim delivery

people transfer: loading gate to arrival gate

runway-to-runway delivery of plane

airplane routing from source to destination

Κατανεμημένη υλοποίηση της λειτουργίας σε επίπεδα

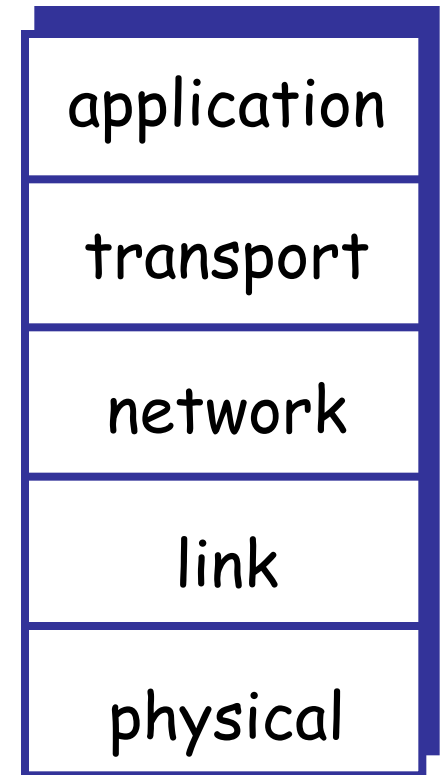


Γιατί χρησιμοποιούμε επίπεδα?

- Όταν μελετάμε πολύπλοκα συστήματα:
- Σαφής δομή επιτρέπει τον προσδιορισμό και τις σχέσεις μεταξύ των κομματιών του συστήματος
 - **μοντέλο αναφοράς** σε επίπεδα κατάλληλο για συζήτηση
- Τμηματοποίηση διευκολύνει τη συντήρηση (maintenance) και την αναβάθμιση (upgrade) του συστήματος
 - Αλλαγή στην υλοποίηση μιας υπηρεσίας ενός επιπέδου είναι μια διάφανη διαδικασία σε σχέση με το υπόλοιπο σύστημα
 - Π.χ., η αλλαγή του τρόπου ελέγχου στις θύρες δεν επηρεάζει το υπόλοιπο σύστημα

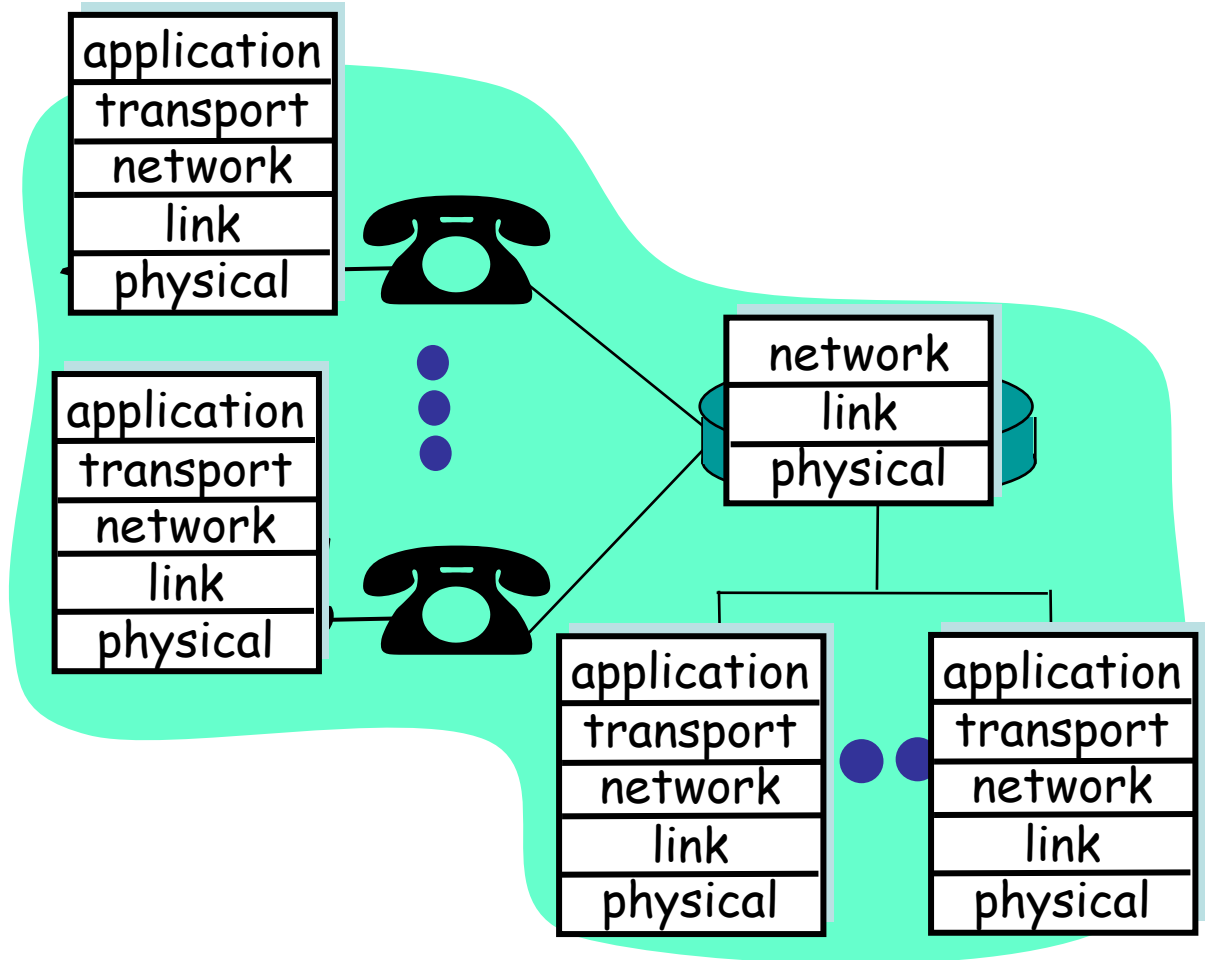
Η στοίβα πρωτοκόλλου στο διαδίκτυο

- **application:** υποστήριξη δικτυακών υπηρεσιών
 - ftp, smtp, http
- **transport:** μεταφορά δεδομένων από τερματική συσκευή σε τερματική συσκευή
 - tcp, udp
- **network:** δρομολόγηση datagrams από την πηγή στον προορισμό
 - ip, routing protocols
- **link:** μεταφορά δεδομένων μεταξύ γειτονικών δικτυακών στοιχείων
 - ppp, ethernet
- **physical:** μετάδοση bits πάνω από το φυσικό μέσο



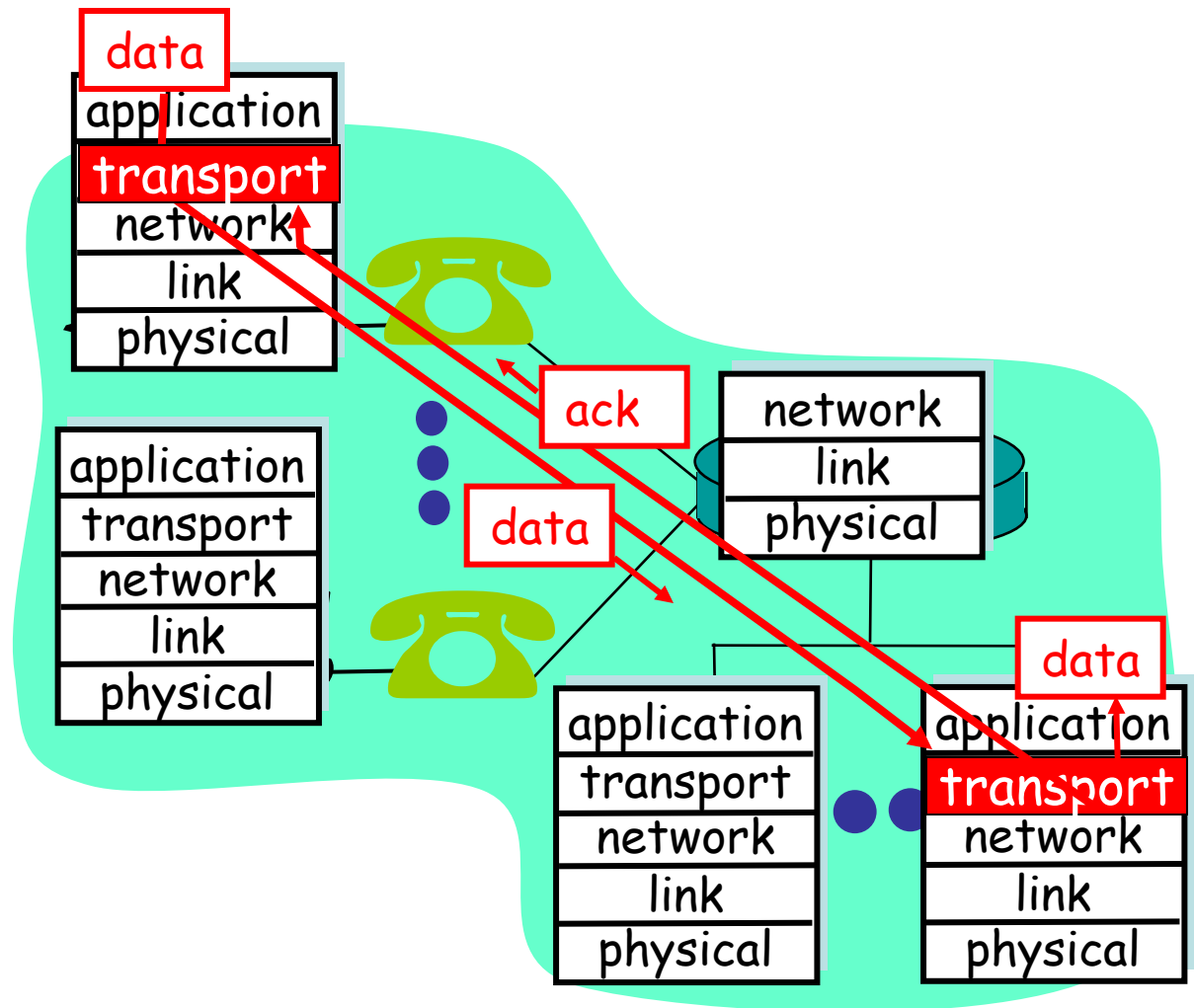
Επίπεδα: Λογική επικοινωνία

- Σε κάθε επίπεδο:
- Κατανεμημένες οντότητες υλοποιούν συγκεκριμένες λειτουργίες εκτελώντας συγκεκριμένες ενέργειες και ανταλλάσσοντας μηνύματα με ομότιμες οντότητες

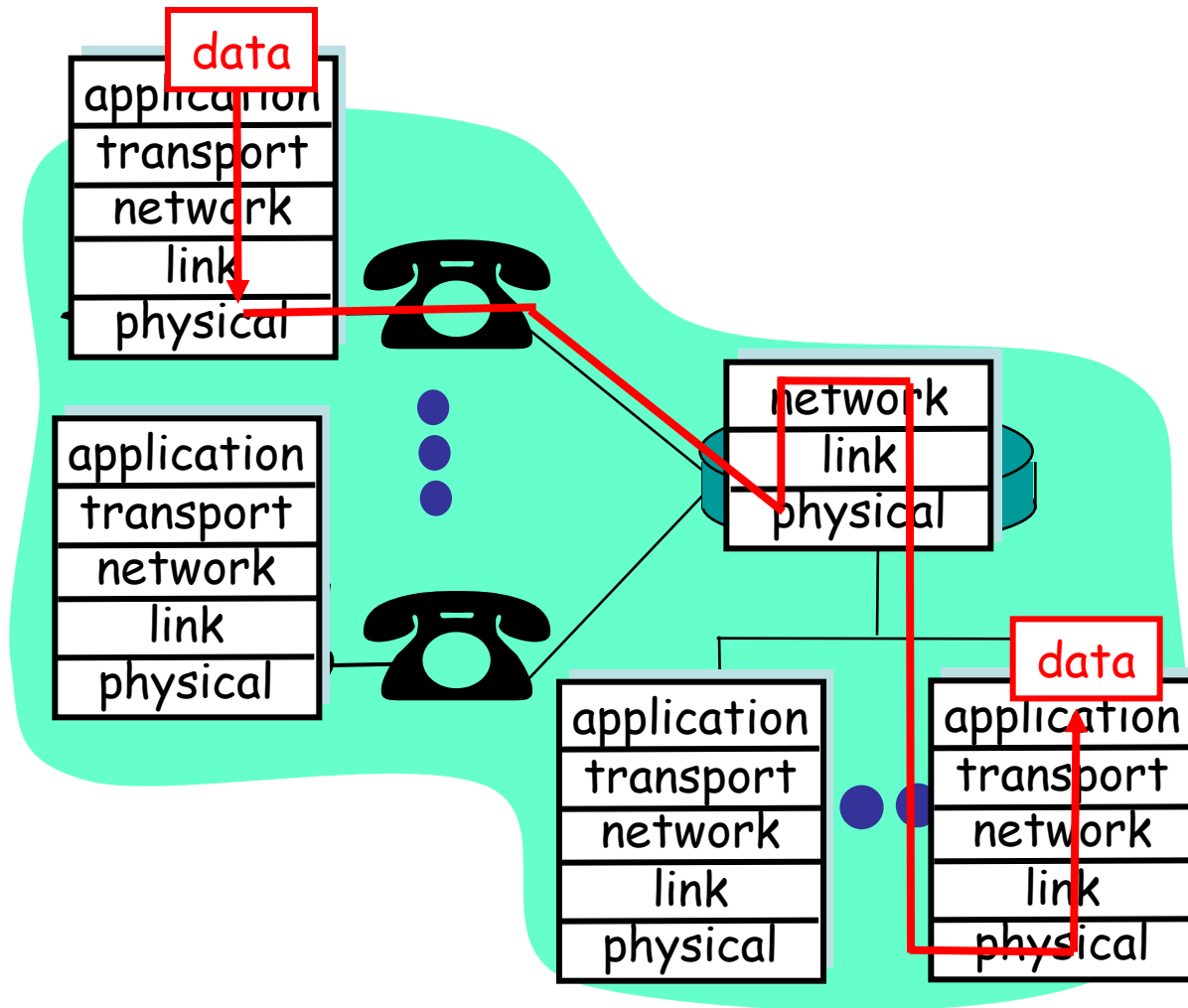


Επίπεδα: Λογική επικοινωνία

- Π.χ.,: **transport**
- Παίρνουμε τα δεδομένα από την εφαρμογή προσθέτουμε τη διεύθυνση, προσθέτουμε πληροφορία ελέγχου ορθότητας και φτιάχνουμε ένα πακέτο
- Στέλνουμε το πακέτο
- Περιμένουμε για ack
- αναλογία: ταχυδρομική υπηρεσία

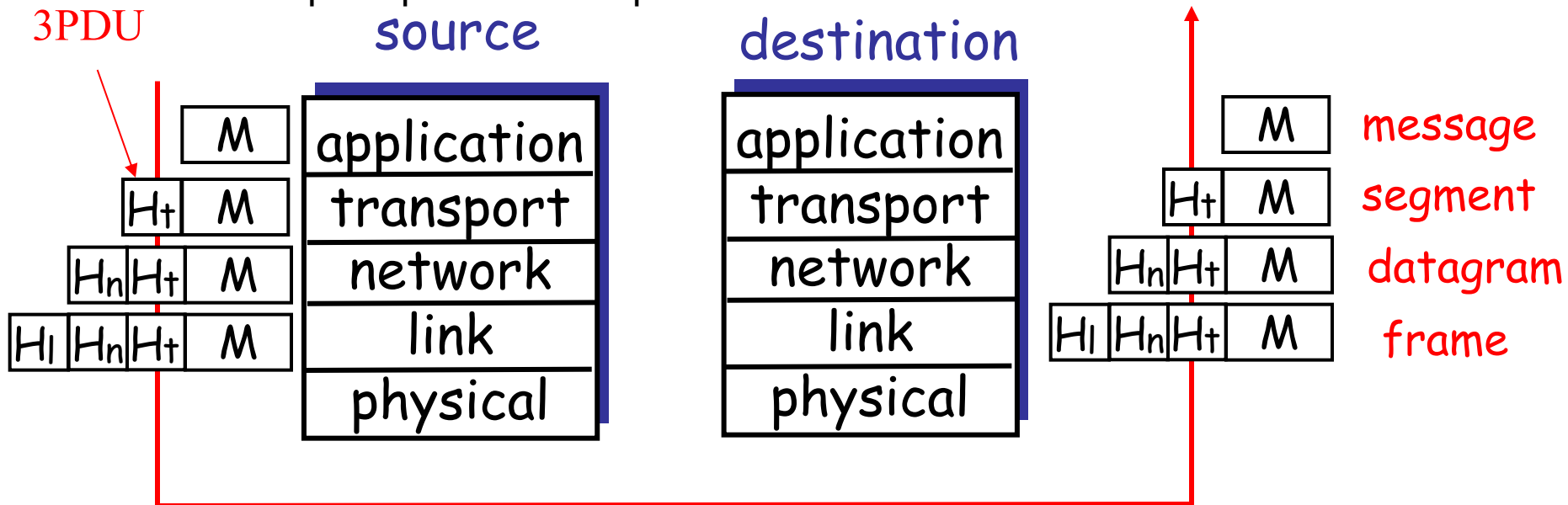


Επίπεδα : Φυσική επικοινωνία



Επίπεδα πρωτοκόλλων και δεδομένα

- Κάθε επίπεδο λαμβάνει δεδομένα από το πιο πάνω επίπεδο
- Προσθέτει επικεφαλίδα με πληροφορίες και δημιουργεί μια νέα μονάδα δεδομένων
- Στέλνει τη νέα μονάδα δεδομένων στο πιο κάτω επίπεδο

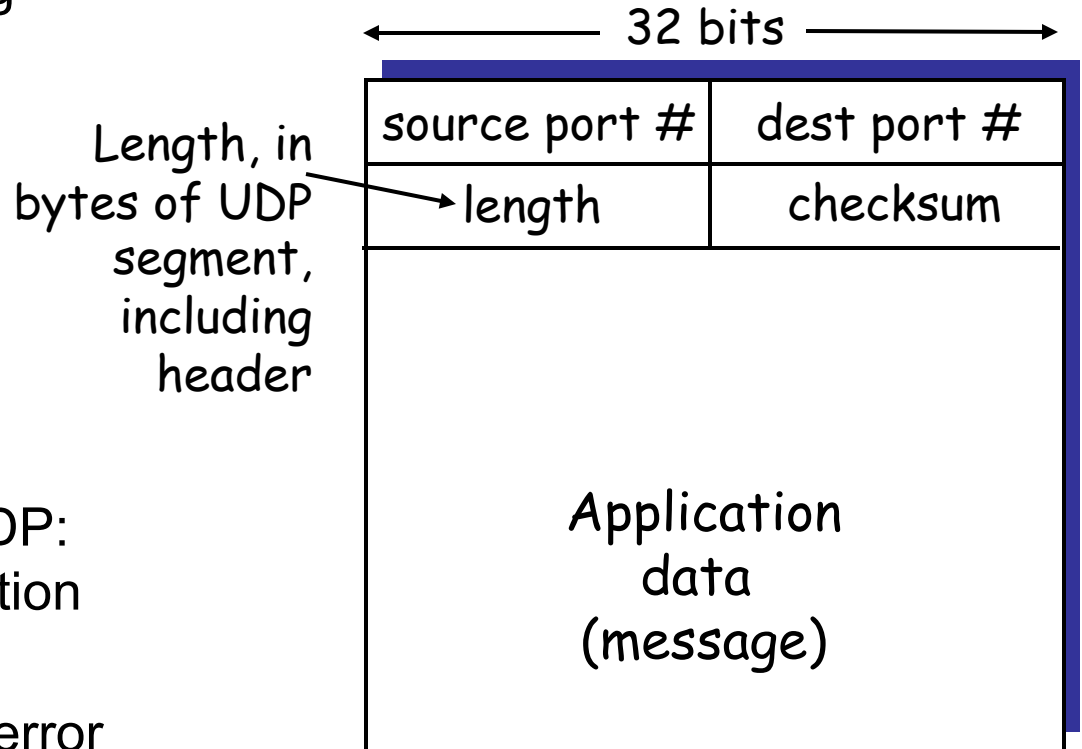


Λειτουργίες επιπέδων

- Σε ένα δίκτυο υπολογιστών κάθε επίπεδο μπορεί να εκτελεί μία ή περισσότερες εργασίες:
 - Error control: μετατρέπει το λογικό κανάλι μεταξύ των ομότιμων επιπέδων πιο αξιόπιστο
 - Flow control: αποφεύγει να κατακλύσει μια πιο αργή ομότιμη οντότητα με πακέτα
 - Segmentation and reassembly: Χωρισμός των μεγάλων πακέτων σε μικρότερα και επανασύσταση τους
 - Multiplexing: Επιτρέπει πολλές συνεδρίες (sessions) να μοιράζονται μια μοναδική χαμηλότερου επιπέδου σύνδεση
 - Connection setup: Διαδικασία χειραψίας με μία ομότιμη οντότητα
-
- **Πιθανά θέματα που μπορεί να προκύψουν από την αρχιτεκτονική επιπέδων;**

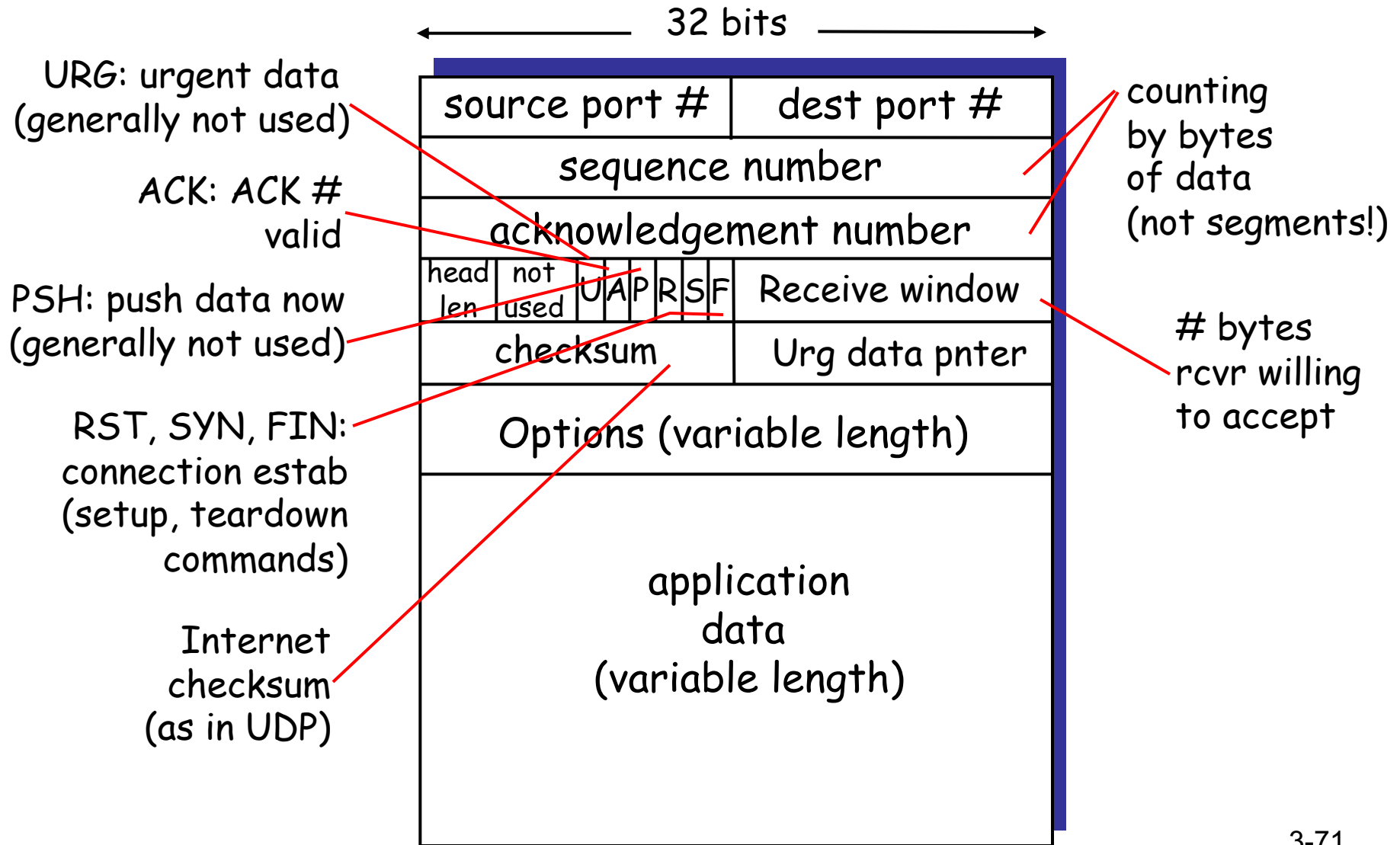
UDP: more

- often used for streaming multimedia apps
 - loss tolerant
 - rate sensitive
- other UDP uses
 - DNS
 - SNMP
- reliable transfer over UDP: add reliability at application layer
 - application-specific error recovery!

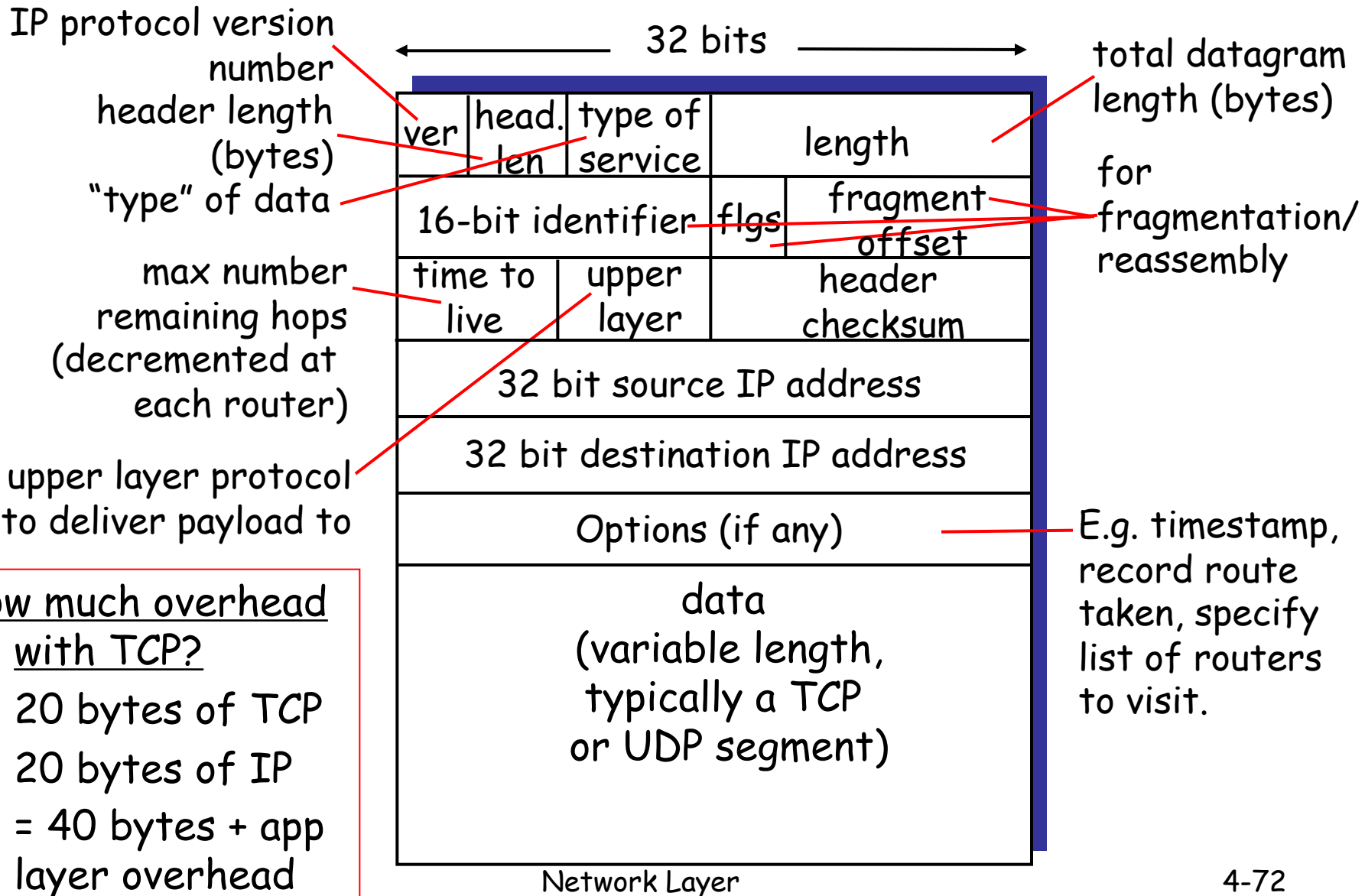


UDP segment format

TCP segment structure



IP datagram format

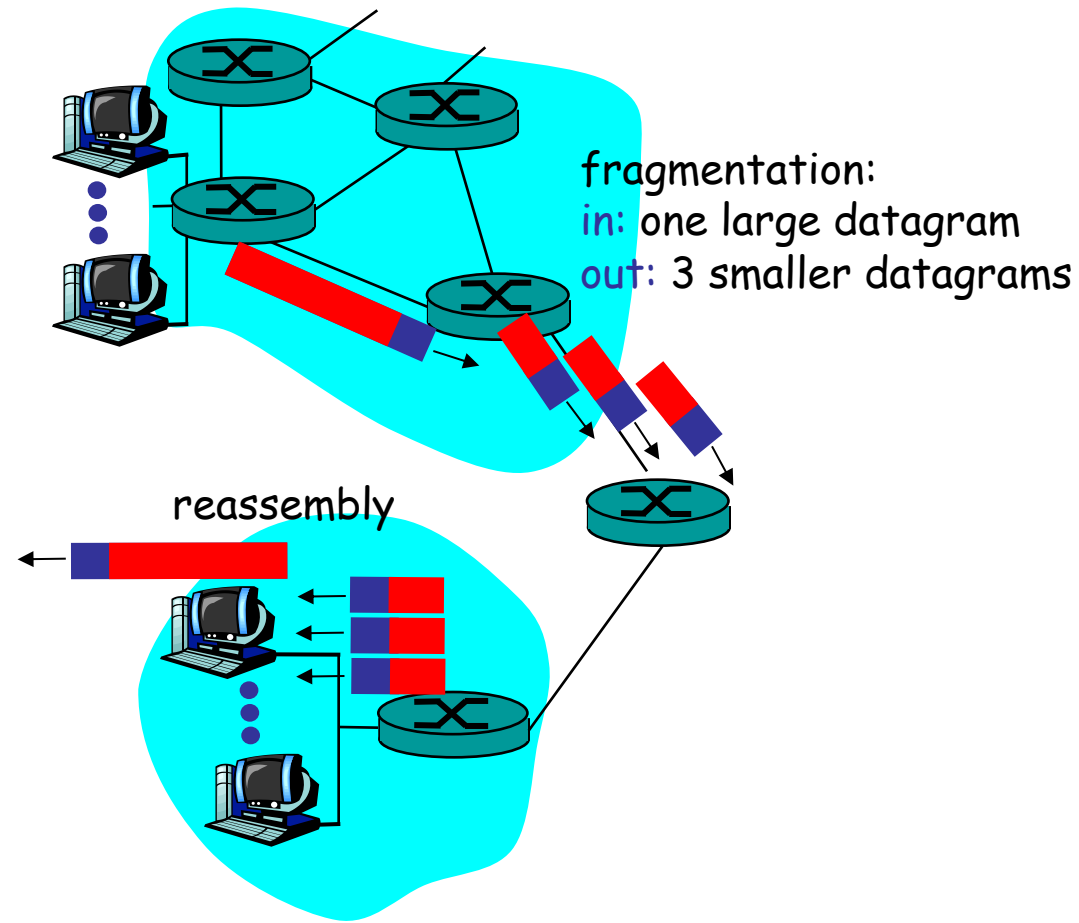


how much overhead with TCP?

- ❑ 20 bytes of TCP
- ❑ 20 bytes of IP
- ❑ = 40 bytes + app layer overhead

IP Fragmentation & Reassembly

- network links have MTU (max.transfer size) - largest possible link-level frame.
 - different link types, different MTUs
- large IP datagram divided (“fragmented”) within net
 - one datagram becomes several datagrams
 - “reassembled” only at final destination
 - IP header bits used to identify, order related fragments



IP Fragmentation and Reassembly

Example

- ❑ 4000 byte datagram
- ❑ MTU = 1500 bytes

	length	ID	fragflag	offset	
	=4000	=x	=0	=0	

One large datagram becomes several smaller datagrams

1480 bytes in data field

offset = $1480/8$

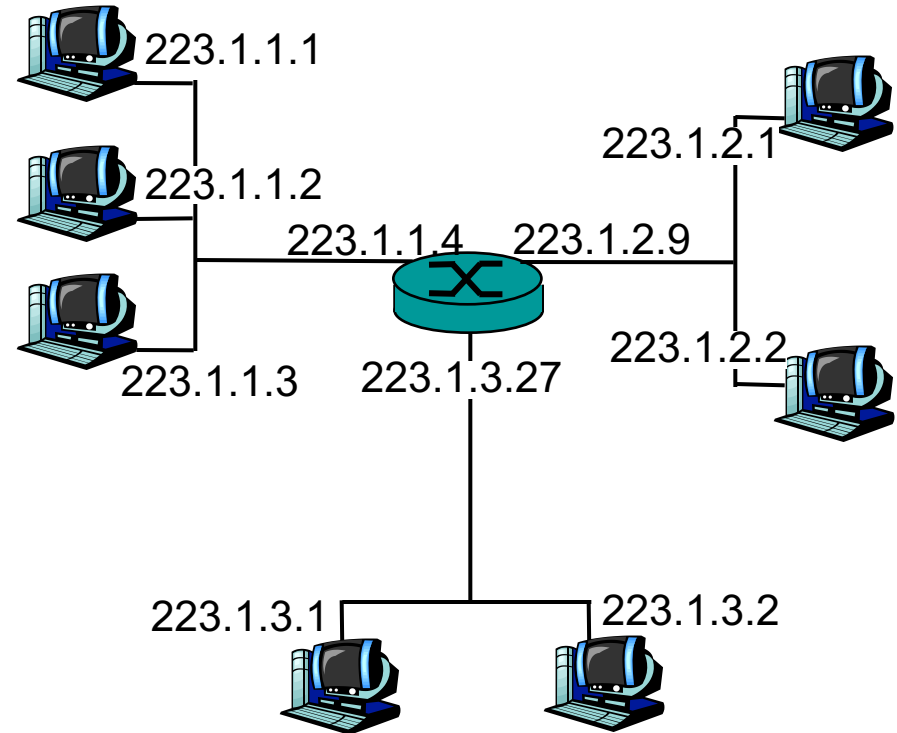
	length	ID	fragflag	offset	
	=1500	=x	=1	=0	

	length	ID	fragflag	offset	
	=1500	=x	=1	=185	

	length	ID	fragflag	offset	
	=1040	=x	=0	=370	

IP Addressing: introduction

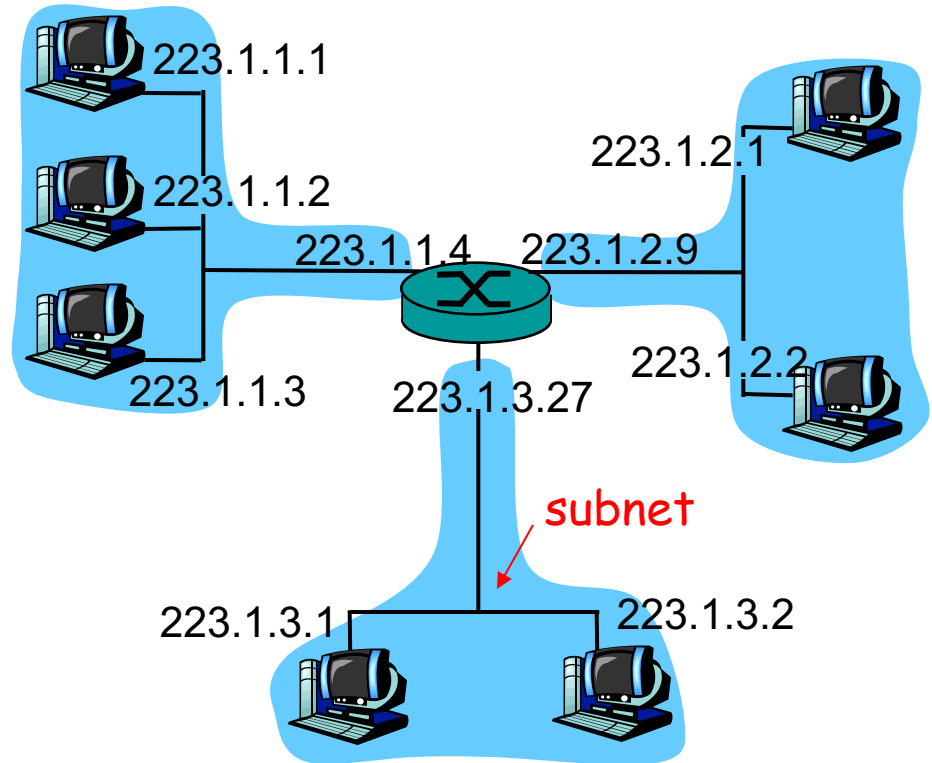
- **IP address:** 32-bit identifier for host, router *interface*
- ***interface:*** connection between host/router and physical link
 - router's typically have multiple interfaces
 - host typically has one interface
 - IP addresses associated with each interface



223.1.1.1 = $\underbrace{11011111}_{223} \underbrace{00000001}_1 \underbrace{00000001}_1 \underbrace{00000001}_1$

Subnets

- IP address:
 - subnet part (high order bits)
 - host part (low order bits)
- *What's a subnet ?*
 - device interfaces with same subnet part of IP address
 - can physically reach each other without intervening router

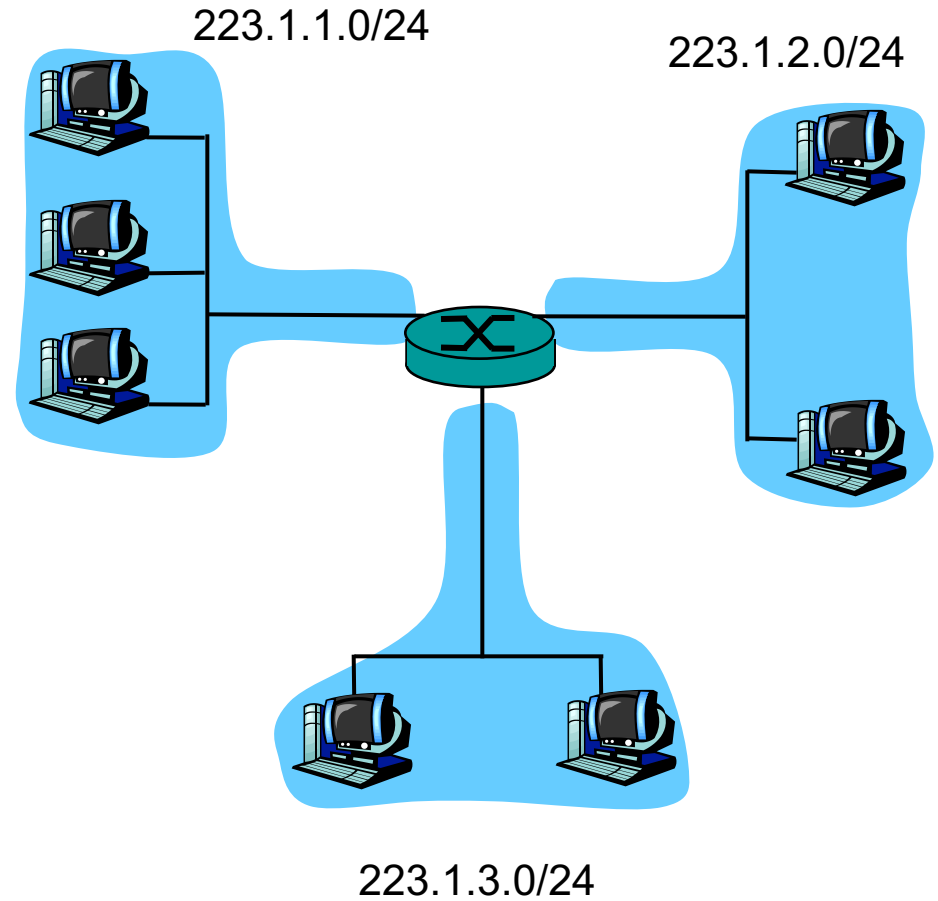


network consisting of 3 subnets

Subnets

Recipe

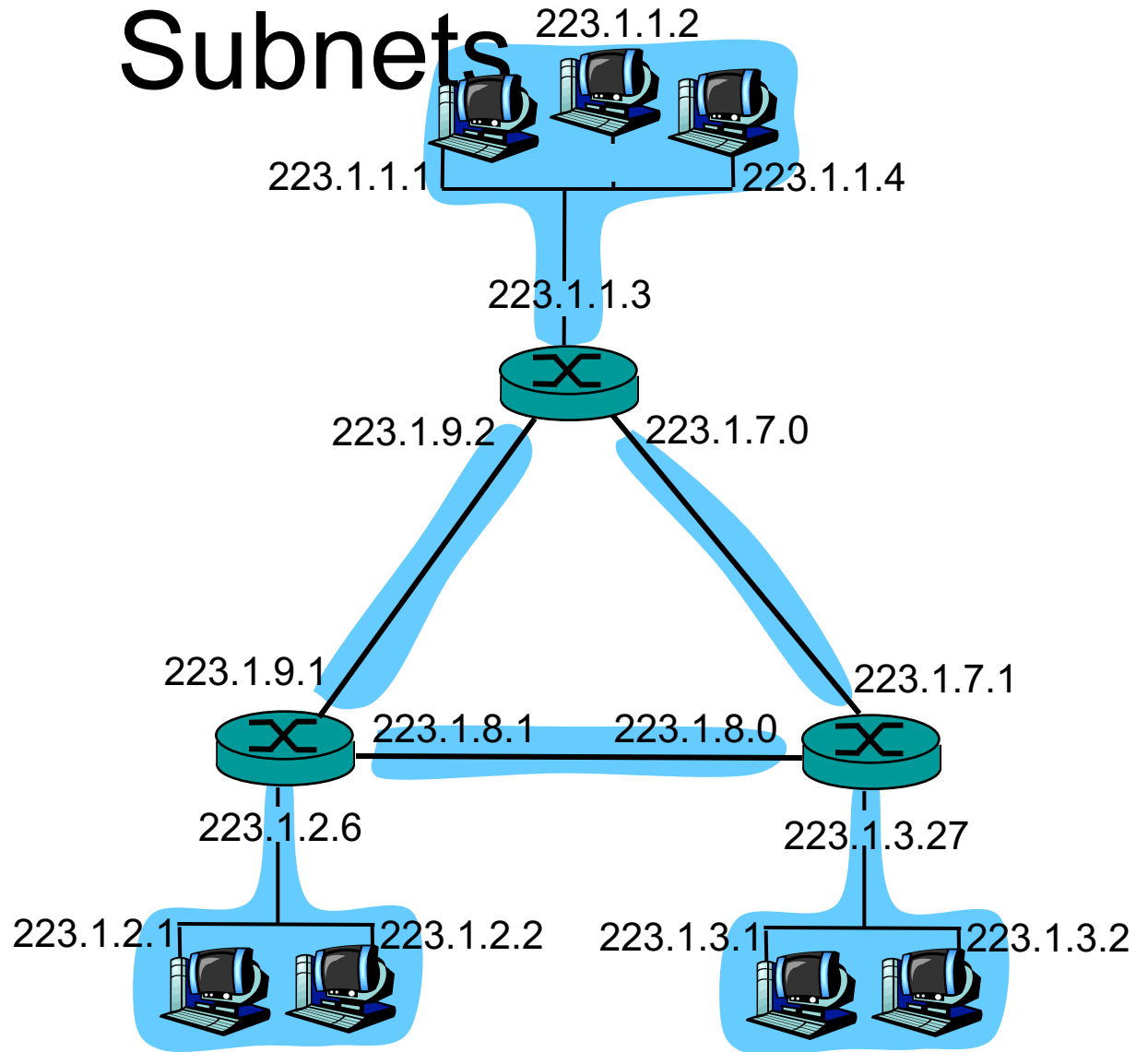
- To determine the subnets, detach each interface from its host or router, creating islands of isolated networks. Each isolated network is called a **subnet**.



Subnet mask: /24

Subnets

How many?



Getting a datagram from source to dest.

- IP datagram:

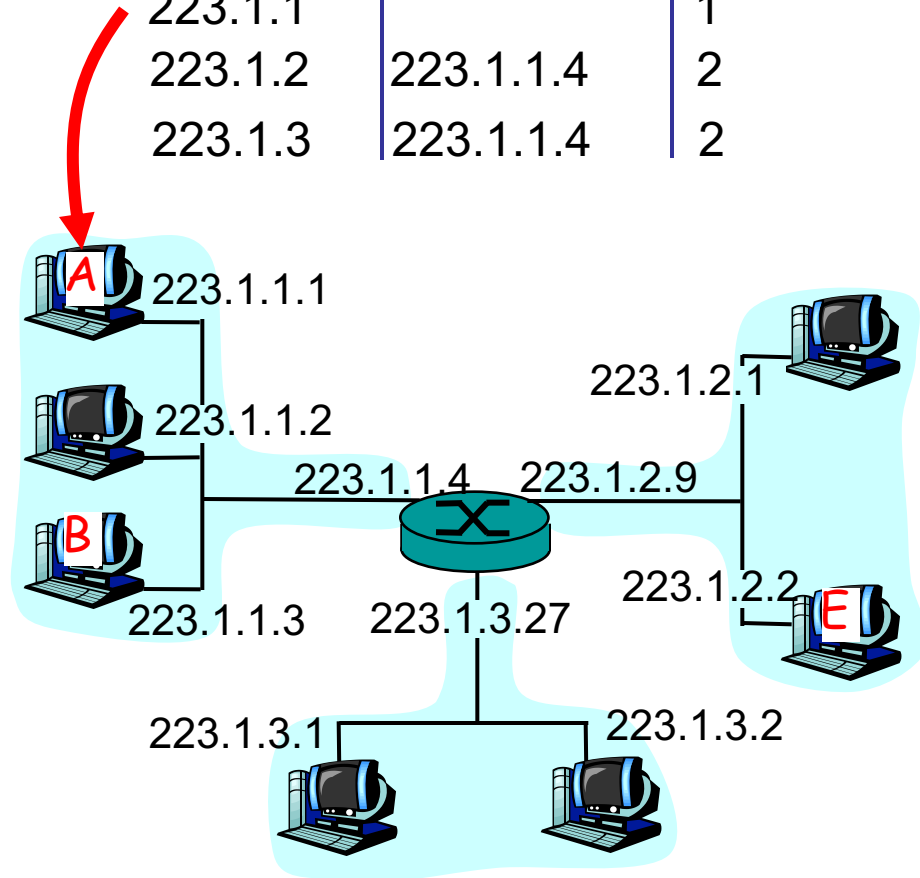
misc	source	dest	data
fields	IP addr	IP addr	

datagram remains unchanged,
as it travels source to
destination

addr fields of interest here

routing table in A

Dest. Net.	next router	Nhops
223.1.1		1
223.1.2	223.1.1.4	2
223.1.3	223.1.1.4	2

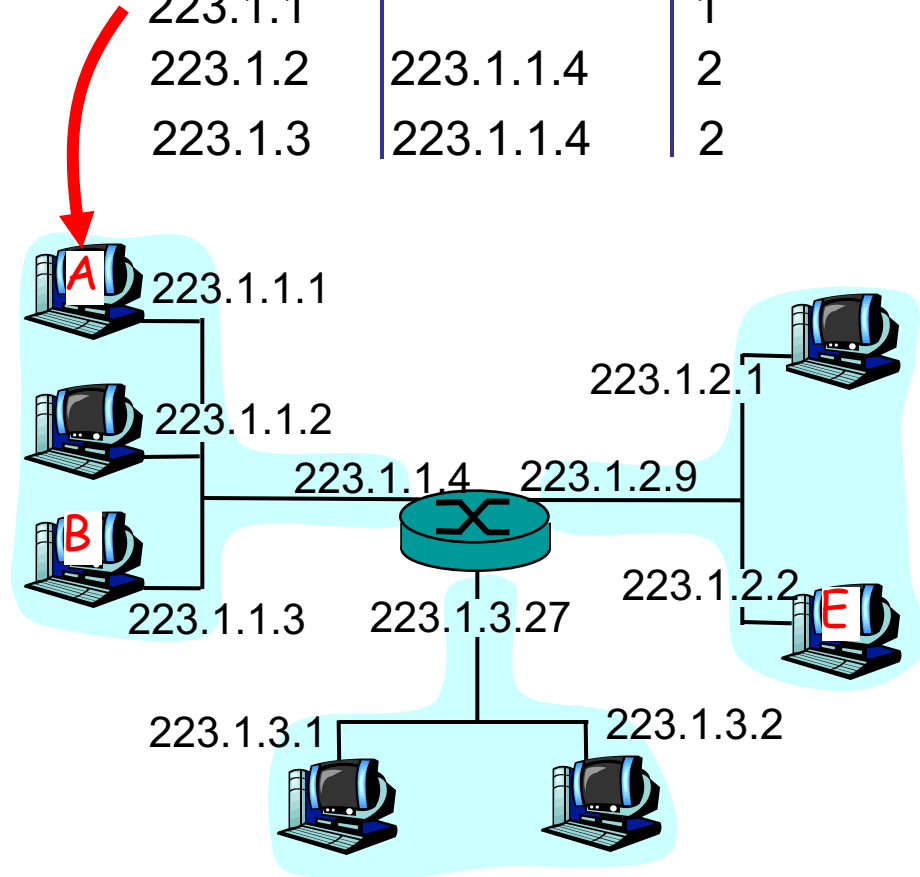


Getting a datagram from source to dest.

misc fields	223.1.1.1	223.1.1.3	data
-------------	-----------	-----------	------

- Starting at A, given IP datagram addressed to B:
- look up net. address of B
- find B is on same net. as A
- link layer will send datagram directly to B inside link-layer frame
- B and A are directly connected

Dest. Net.	next router	Nhops
223.1.1		1
223.1.2	223.1.1.4	2
223.1.3	223.1.1.4	2



Getting a datagram from source to dest.

misc fields	223.1.1.1	223.1.2.3	data
-------------	-----------	-----------	------

Starting at A, dest. E:
 look up network address of E
 E on *different* network

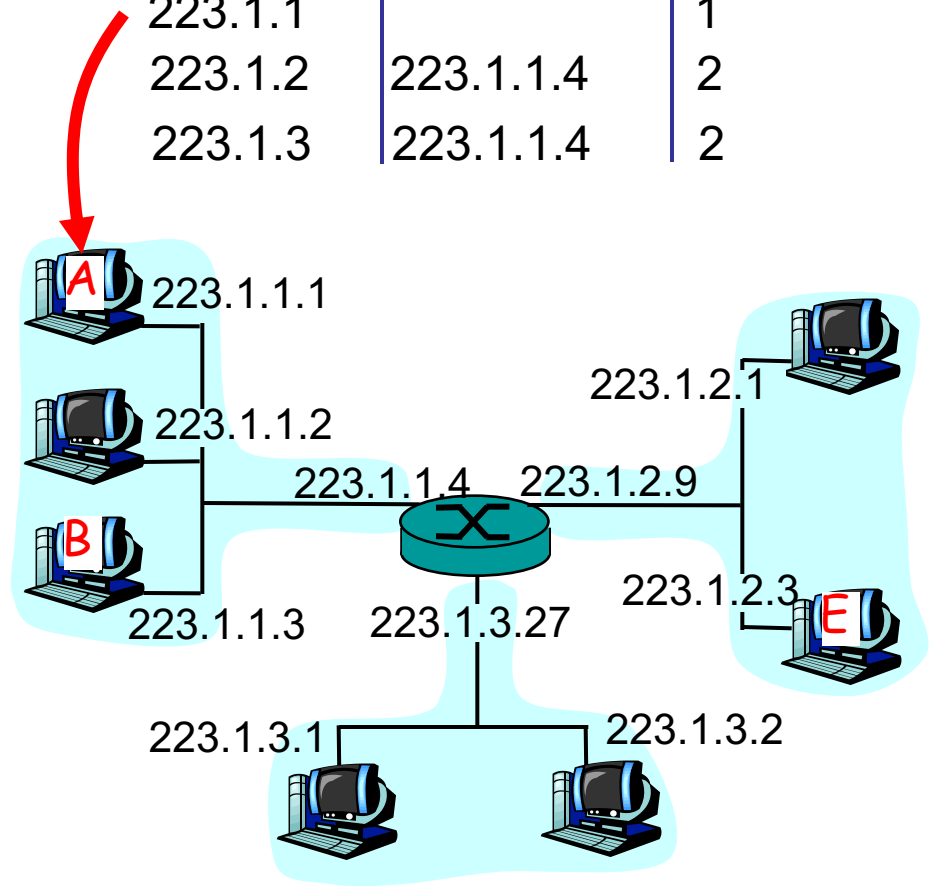
- A, E not directly attached

routing table: next hop router to E
 is 223.1.1.4

link layer sends datagram to router
 223.1.1.4 inside link-layer frame

datagram arrives at 223.1.1.4
 continued.....

Dest. Net.	next router	Nhops
223.1.1		1
223.1.2	223.1.1.4	2
223.1.3	223.1.1.4	2



Getting a datagram from source to dest.

misc fields	223.1.1.1	223.1.2.3	data
-------------	-----------	-----------	------

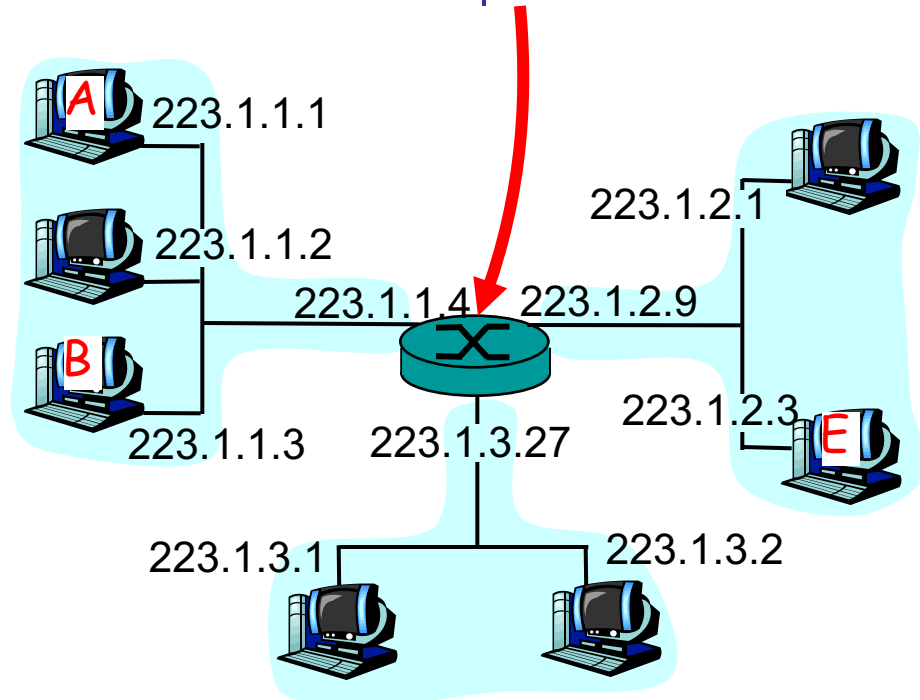
Arriving at 223.1.4, destined for 223.1.2.3

look up network address of E
E on same network as router's interface 223.1.2.9

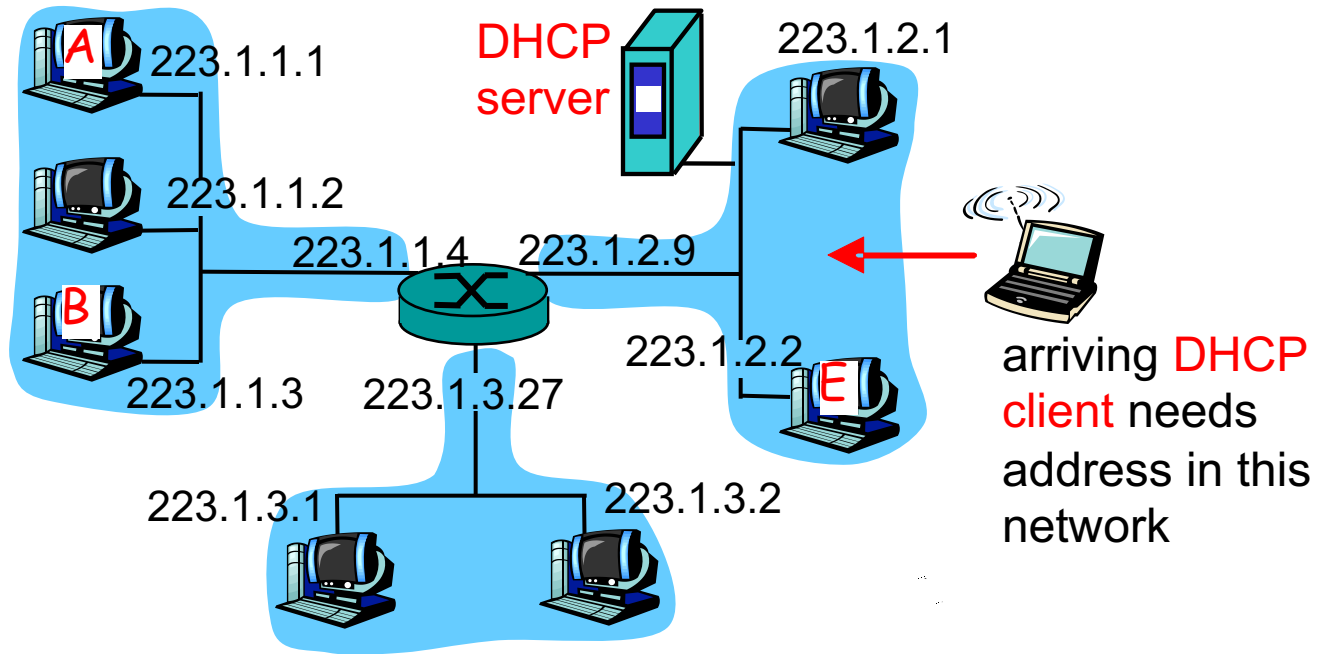
➤ router, E directly attached link layer sends datagram to 223.1.2.3 inside link-layer frame via interface 223.1.2.9

datagram arrives at 223.1.2.3!!!
(hooray!)

Dest. network	next router	Nhops	interface
223.1.1	-	1	223.1.1.4
223.1.2	-	1	223.1.2.9
223.1.3	-	1	223.1.3.27



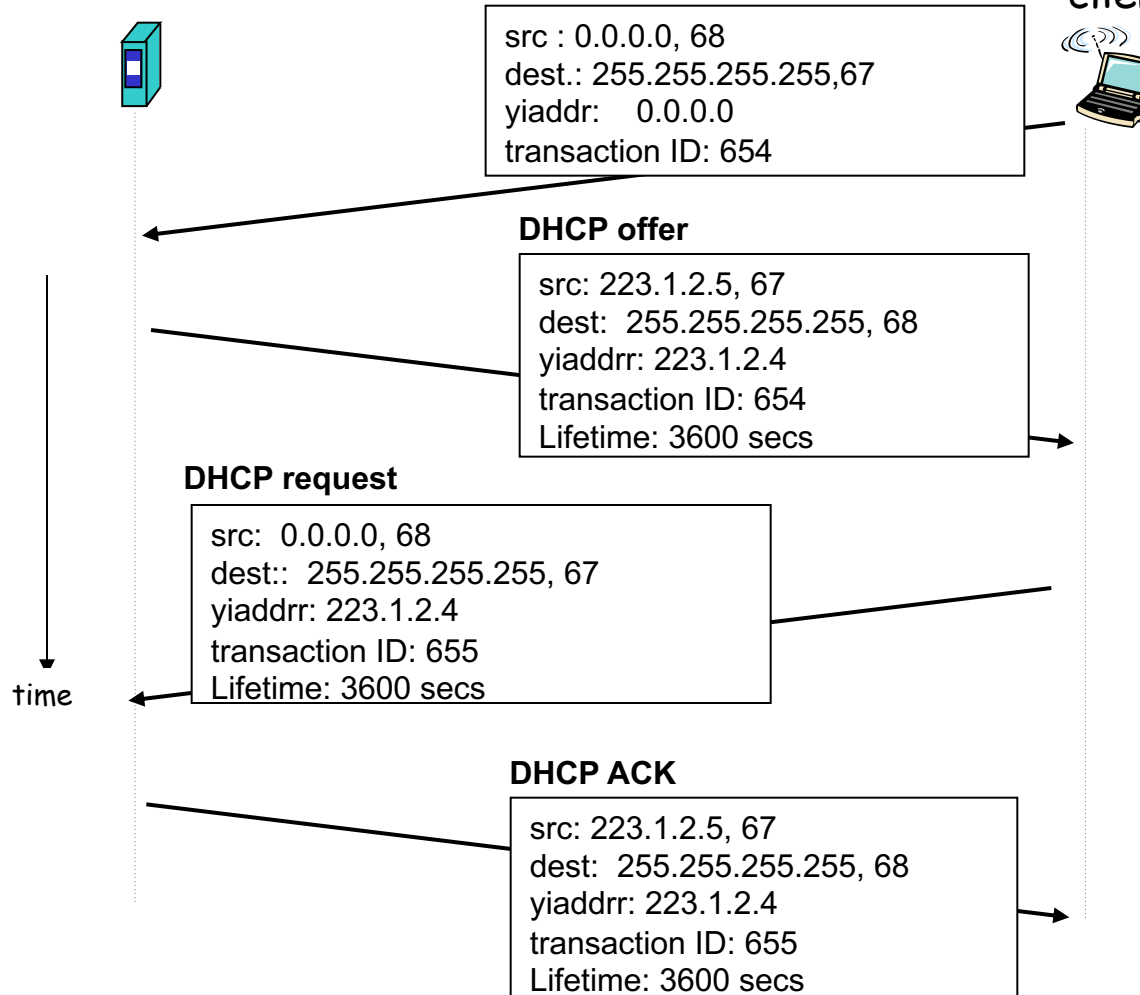
DHCP client-server scenario



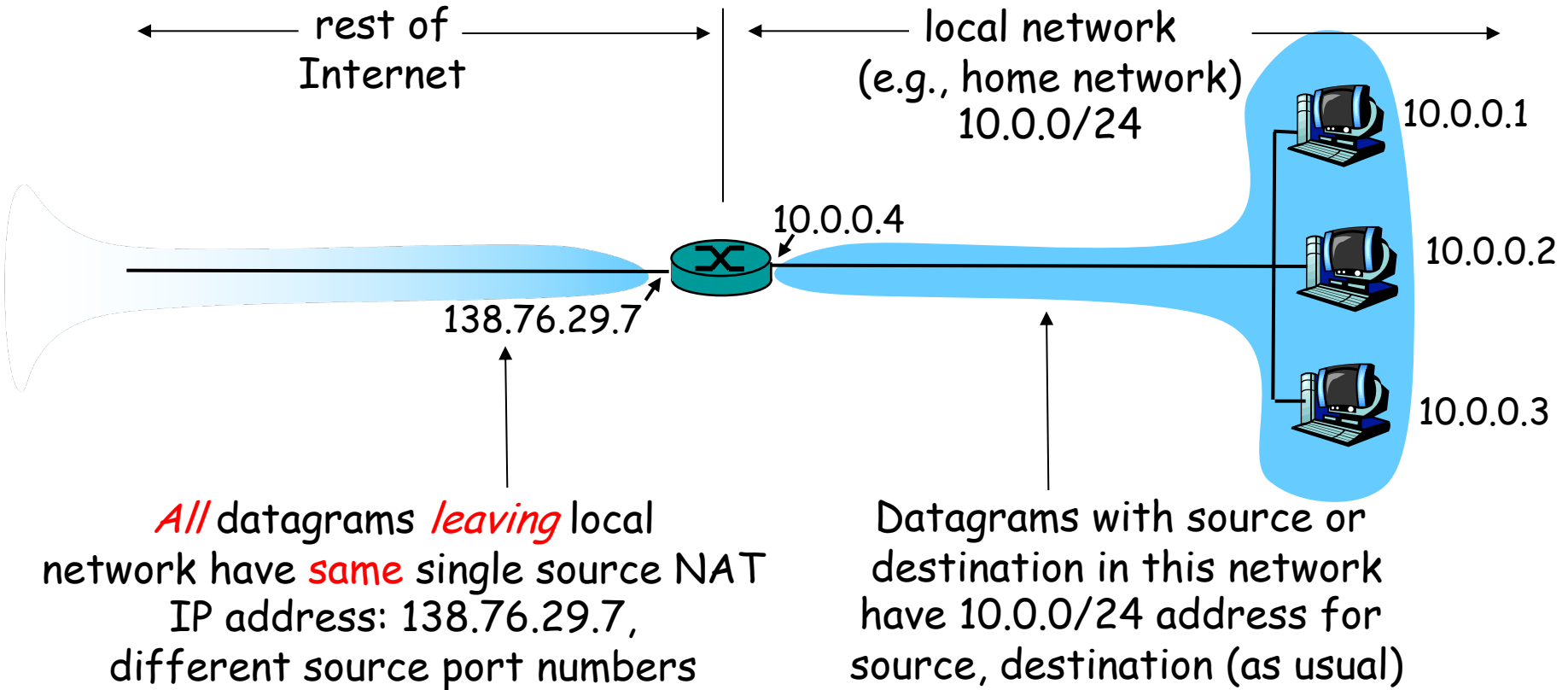
DHCP client-server scenario

DHCP server: 223.1.2.5

arriving client



NAT: Network Address Translation



NAT: Network Address Translation

NAT translation table	
WAN side addr	LAN side addr
138.76.29.7, 5001	10.0.0.1, 3345
.....

1: host 10.0.0.1 sends datagram to 128.119.40.186, 80

S: 10.0.0.1, 3345
D: 128.119.40.186, 80

10.0.0.1

10.0.0.2

10.0.0.3



10.0.0.4

S: 138.76.29.7, 5001
D: 128.119.40.186, 80

2

138.76.29.7

S: 128.119.40.186, 80
D: 138.76.29.7, 5001

3

S: 128.119.40.186, 80
D: 10.0.0.1, 3345

4

4: NAT router changes datagram dest addr from 138.76.29.7, 5001 to 10.0.0.1, 3345

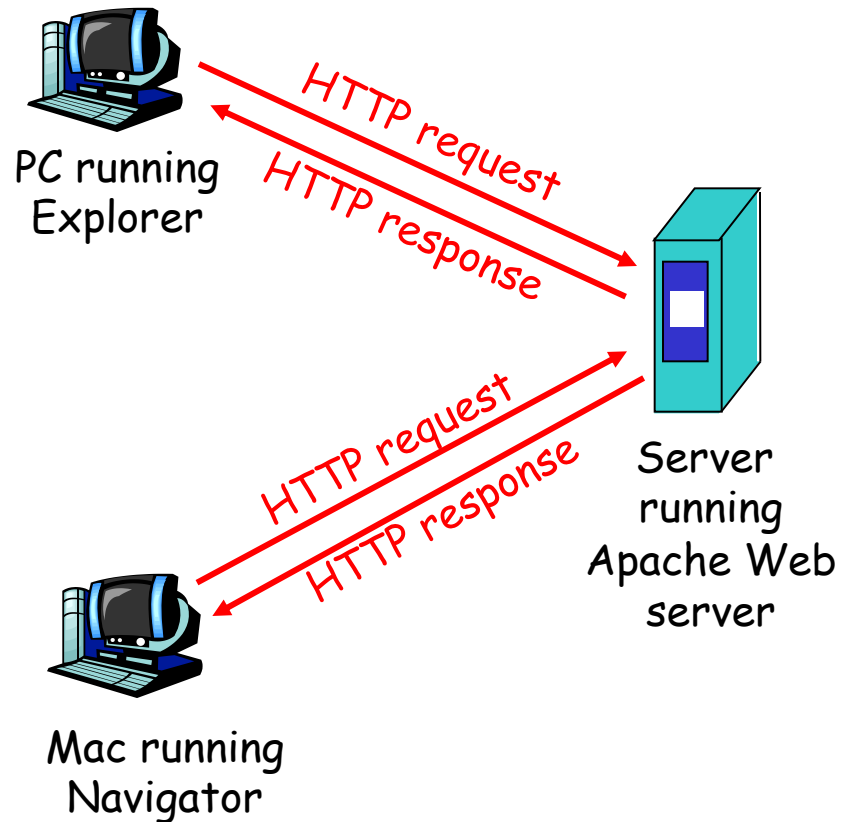
3: Reply arrives dest. address: 138.76.29.7, 5001

2: NAT router changes datagram source addr from 10.0.0.1, 3345 to 138.76.29.7, 5001, updates table

HTTP overview

HTTP: hypertext transfer protocol

- Web's application layer protocol
- client/server model
 - *client*: browser that requests, receives, “displays” Web objects
 - *server*: Web server sends objects in response to requests
- HTTP 1.0: RFC 1945
- HTTP 1.1: RFC 2068



HTTP overview (continued)

Uses TCP:

- client initiates TCP connection (creates socket) to server, port 80
- server accepts TCP connection from client
- HTTP messages (application-layer protocol messages) exchanged between browser (HTTP client) and Web server (HTTP server)
- TCP connection closed

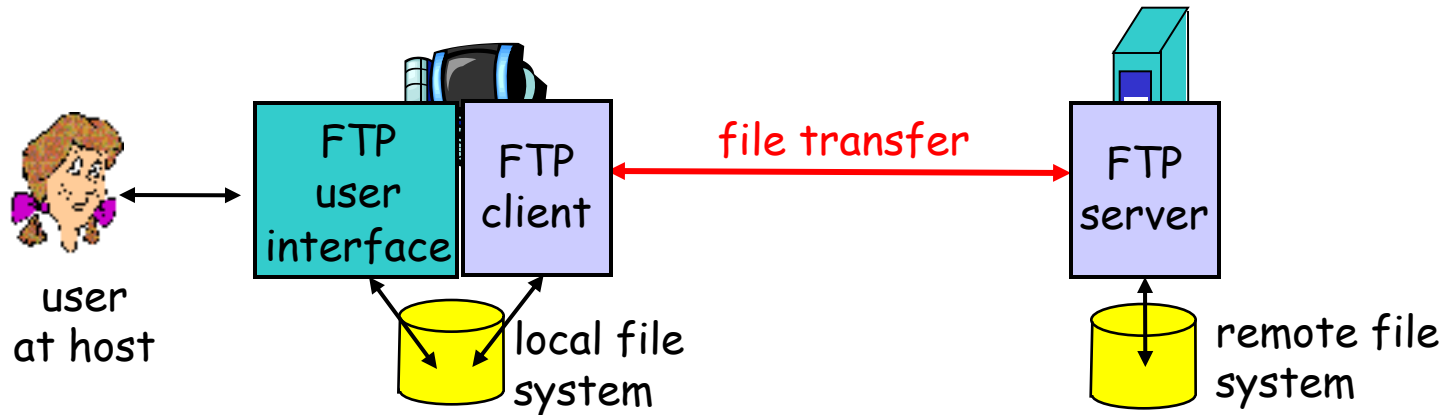
HTTP is “stateless”

- server maintains no information about past client requests

Protocols that maintain “state” are complex! aside

- ❑ past history (state) must be maintained
- ❑ if server/client crashes, their views of “state” may be inconsistent, must be reconciled

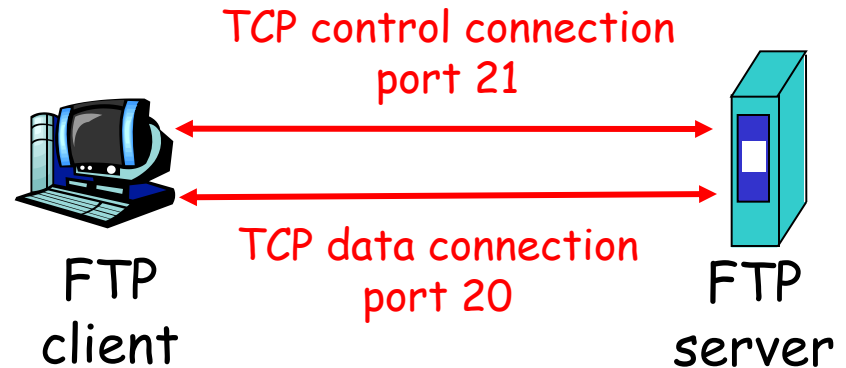
FTP: the file transfer protocol



- transfer file to/from remote host
- client/server model
 - *client*: side that initiates transfer (either to/from remote)
 - *server*: remote host
- ftp: RFC 959
- ftp server: port 21

FTP: separate control, data connections

- FTP client contacts FTP server at port 21, specifying TCP as transport protocol
- Client obtains authorization over control connection
- Client browses remote directory by sending commands over control connection.
- When server receives a command for a file transfer, the server opens a TCP data connection to client
- After transferring one file, server closes connection.



- Server opens a second TCP data connection to transfer another file.
- Control connection: "out of band"
- FTP server maintains "state": current directory, earlier authentication

FTP commands, responses

Sample commands:

- sent as ASCII text over control channel
- **USER *username***
- **PASS *password***
- **LIST** return list of file in current directory
- **RETR *filename*** retrieves (gets) file
- **STOR *filename*** stores (puts) file onto remote host

Sample return codes

- status code and phrase (as in HTTP)
- **331 Username OK, password required**
- **125 data connection already open; transfer starting**
- **425 Can't open data connection**
- **452 Error writing file**

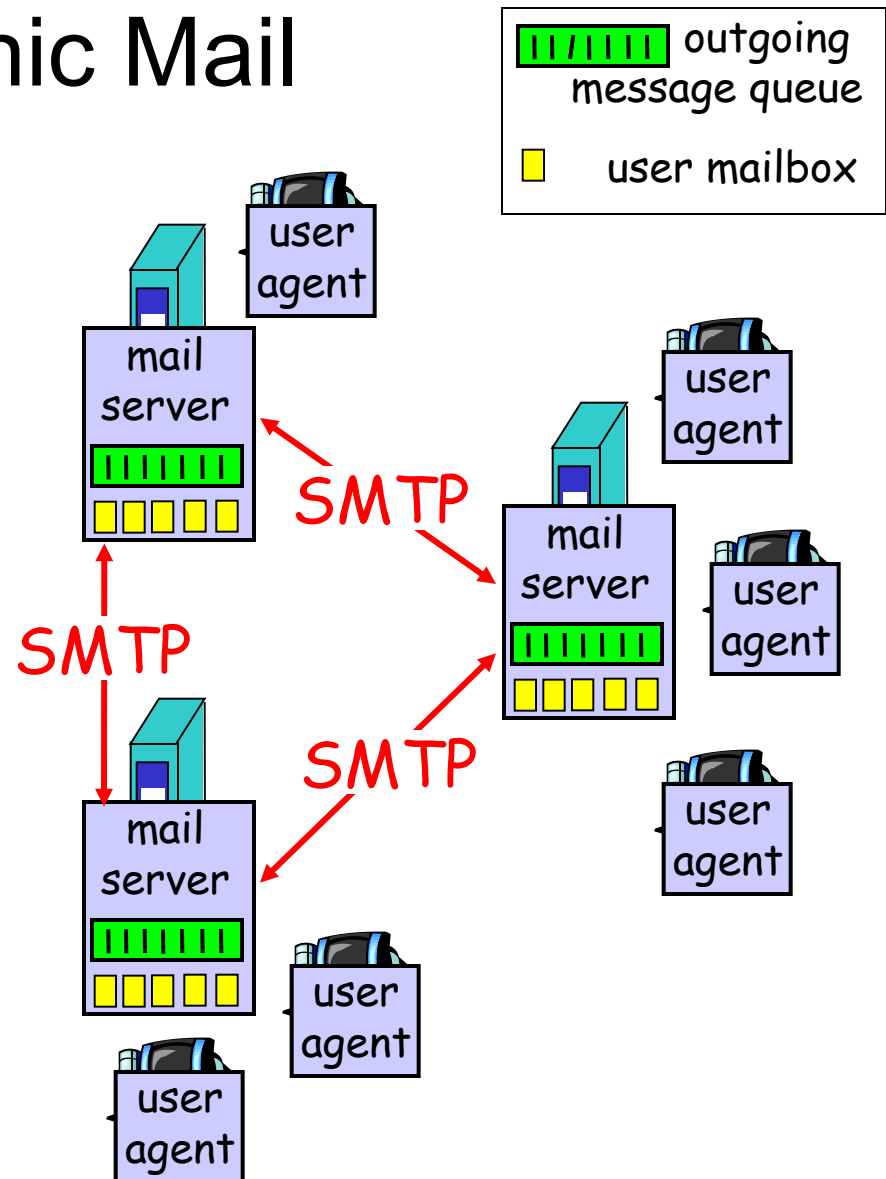
Electronic Mail

Three major components:

- user agents
- mail servers
- simple mail transfer protocol: SMTP

User Agent

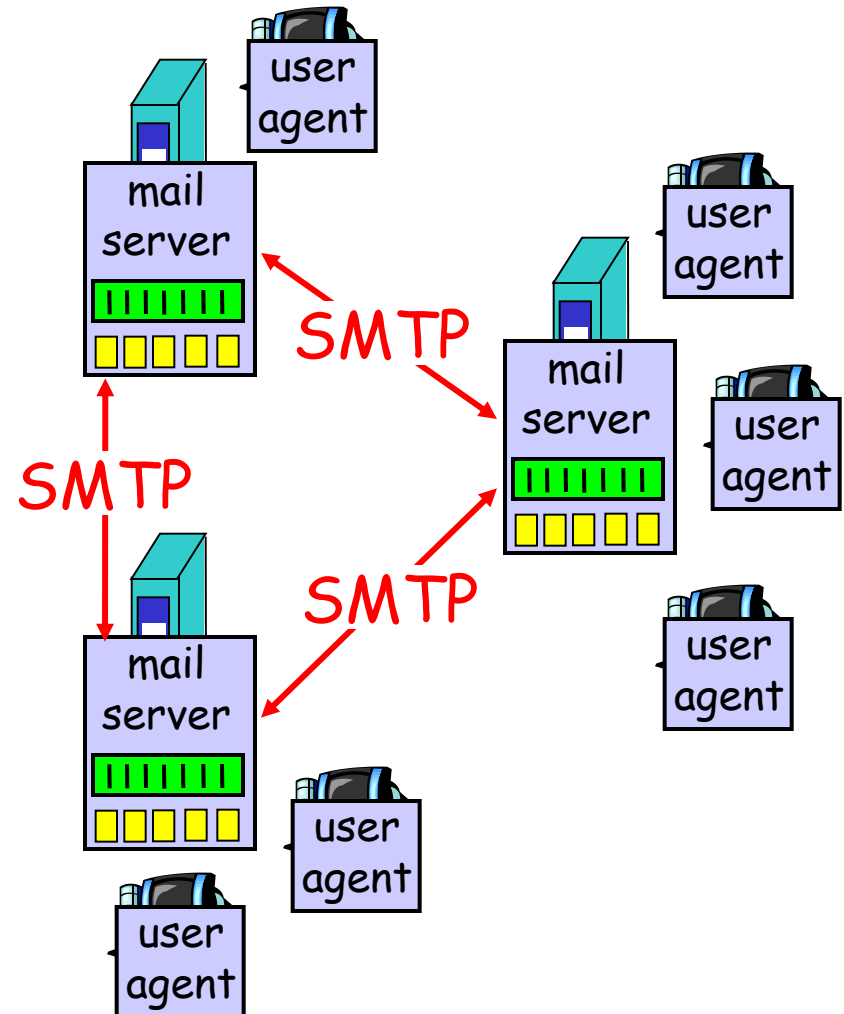
- a.k.a. “mail reader”
- composing, editing, reading mail messages
- e.g., Thunderbird, Outlook, elm, Netscape Messenger
- outgoing, incoming messages stored on server



Electronic Mail: mail servers

Mail Servers

- **mailbox** contains incoming messages for user
- **message queue** of outgoing (to be sent) mail messages
- **SMTP protocol** between mail servers to send email messages
 - client: sending mail server
 - “server”: receiving mail server

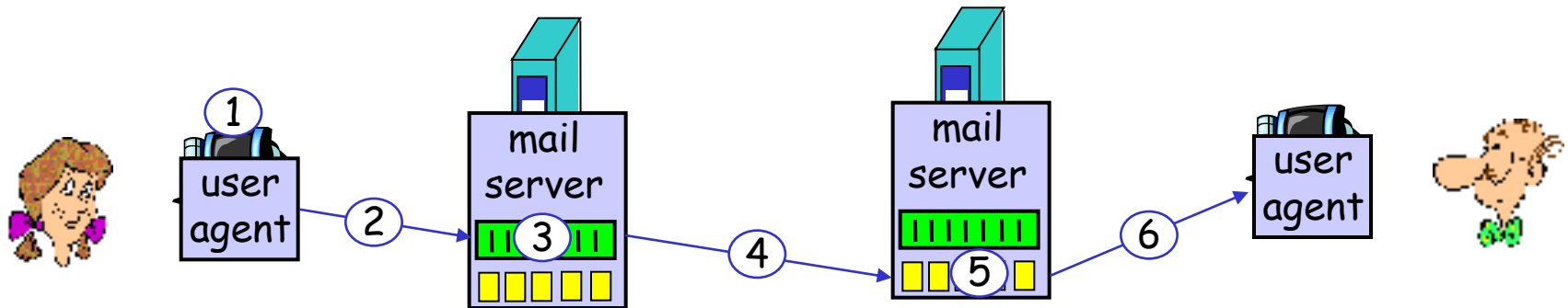


Electronic Mail: SMTP [RFC 2821]

- uses TCP to reliably transfer email message from client to server, port 25
- direct transfer: sending server to receiving server
- three phases of transfer
 - handshaking (greeting)
 - transfer of messages
 - closure
- command/response interaction
 - **commands**: ASCII text
 - **response**: status code and phrase
- messages must be in 7-bit ASCII

Scenario: Alice sends message to Bob

- 1) Alice uses UA to compose message and “to”
`bob@someschool.edu`
- 2) Alice’s UA sends message to her mail server; message placed in message queue
- 3) Client side of SMTP opens TCP connection with Bob’s mail server
- 4) SMTP client sends Alice’s message over the TCP connection
- 5) Bob’s mail server places the message in Bob’s mailbox
- 6) Bob invokes his user agent to read message



Sample SMTP interaction

```
S: 220 hamburger.edu
C: HELO crepes.fr
S: 250 Hello crepes.fr, pleased to meet you
C: MAIL FROM: <alice@crepes.fr>
S: 250 alice@crepes.fr... Sender ok
C: RCPT TO: <bob@hamburger.edu>
S: 250 bob@hamburger.edu ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: Do you like ketchup?
C: How about pickles?
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 hamburger.edu closing connection
```


Try SMTP interaction for yourself:

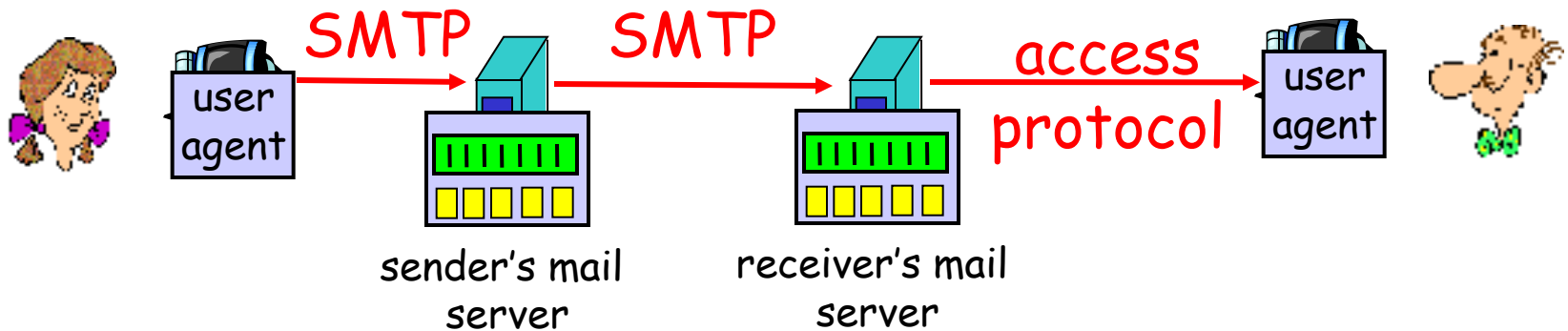
- `telnet servername 25`
- see 220 reply from server
- enter HELO, MAIL FROM, RCPT TO, DATA, QUIT commands

above lets you send email without using email client (reader)

Sample SMTP interaction

```
S: 220 hamburger.edu
C: HELO crepes.fr
S: 250 Hello crepes.fr, pleased to meet you
C: MAIL FROM: <alice@crepes.fr>
S: 250 alice@crepes.fr... Sender ok
C: RCPT TO: <bob@hamburger.edu>
S: 250 bob@hamburger.edu ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: Do you like ketchup?
C: How about pickles?
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 hamburger.edu closing connection
```

Mail access protocols



- SMTP: delivery/storage to receiver's server
- Mail access protocol: retrieval from server
 - POP: Post Office Protocol [RFC 1939]
 - authorization (agent <-->server) and download
 - IMAP: Internet Mail Access Protocol [RFC 1730]
 - more features (more complex)
 - manipulation of stored msgs on server
 - HTTP: Hotmail , Yahoo! Mail, etc.

POP3 protocol

authorization phase

- client commands:
 - **user**: declare username
 - **pass**: password
- server responses
 - **+OK**
 - **-ERR**

```
S: +OK POP3 server ready
C: user bob
S: +OK
C: pass hungry
S: +OK user successfully logged on
```

transaction phase, client:

- **list**: list message numbers
- **retr**: retrieve message by number
- **dele**: delete
- **quit**

```
C: list
S: 1 498
S: 2 912
S: .
C: retr 1
S: <message 1 contents>
S: .
C: dele 1
C: retr 2
S: <message 1 contents>
S: .
C: dele 2
C: quit
S: +OK POP3 server signing off
```

DNS: Domain Name System

People: many identifiers:

- name, passport #

Internet hosts, routers:

- IP address (32 bit) - used for addressing datagrams
- “name”, e.g.,
ww.yahoo.com - used by humans

Q: map between IP addresses and name ?

Domain Name System:

- *distributed database* implemented in hierarchy of many *name servers*
- *application-layer protocol* host, routers, name servers to communicate to *resolve* names (address/name translation)
 - note: core Internet function, implemented as application-layer protocol
 - complexity at network’s “edge”