



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
Εθνικόν και Καποδιστριακόν
Πανεπιστήμιον Αθηνών

Δίκτυα Επικοινωνιών Ι

Ενότητα 2: Επίπεδο Εφαρμογής

Διδάσκων: Λάζαρος Μεράκος

Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Εθνικό & Καποδιστριακό Πανεπιστήμιο Αθηνών

Δίκτυα Επικοινωνιών Ι

Τμήμα Πληροφορικής και Τηλεπικοινωνιών



Εθνικό & Καποδιστριακό
Πανεπιστήμιο Αθηνών

Θεματικές Ενότητες (ΘΕ) μαθήματος:

ΘΕ1: Εισαγωγή
(Κεφ. 1 του βιβλίου)

ΘΕ2: Επίπεδο Εφαρμογής
(Κεφ. 2 του βιβλίου)

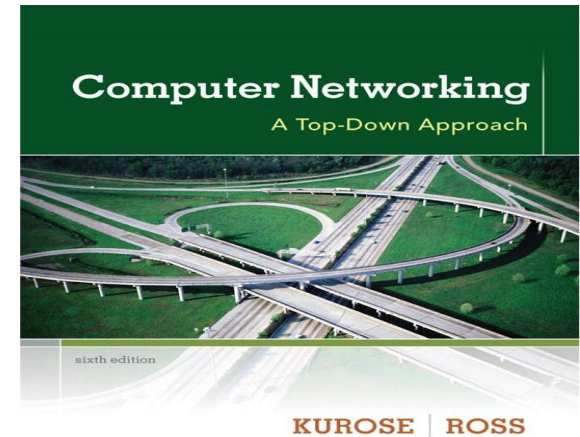
ΘΕ3: Επίπεδο Μεταφοράς
(Κεφ. 3 του βιβλίου)

ΘΕ4: Επίπεδο Δικτύου
(Κεφ. 4 του βιβλίου)

**ΘΕ5: Επίπεδο Ζεύξης: Ζεύξεις, Δίκτυα
Πρόσβασης, Δίκτυα Τοπικής Περιοχής**
(Κεφ. 5 του βιβλίου)

Συνιστώμενο Βιβλίο:
**Computer Networking: A Top-Down
Approach, by Kurose & Ross,
Addison-Wesley**

Ελληνική Μετάφραση:
Εκδόσεις : **Μ. Γκιούρδας**



Οι περισσότερες από τις διαφάνειες αυτής της ενότητας αποτελούν προσαρμογή και απόδοση στα ελληνικά των διαφανειών που συνοδεύουν το βιβλίο *Computer Networking: A Top-Down Approach*, J.F. Kurose and K.W. Ross, 6/E, Addison-Wesley.

All material copyright 1996-2012
J.F. Kurose and K.W. Ross, All Rights Reserved

Προσαρμογή και επιμέλεια της απόδοσης των πρωτότυπων διαφανειών στα ελληνικά :
Λάζαρος Μεράκος

Κεφάλαιο 2: Επίπεδο εφαρμογής

- 2.1 Αρχές δικτυακών εφαρμογών
- 2.2 Web και HTTP
- 2.3 Ηλεκτρονικό ταχυδρομείο
 - SMTP, POP3, IMAP
- 2.4 DNS
- 2.5 Εφαρμογές P2P

Κεφάλαιο 2: Επίπεδο εφαρμογής

Οι στόχοι μας:

- Εννοιολογικά, σχεδιαστικά θέματα πρωτοκόλλων δικτυακών εφαρμογών
 - Μοντέλα υπηρεσιών επιπέδου μεταφοράς
 - Μοντέλο πελάτη εξυπηρετητή (client-server)
 - Μοντέλο ομότιμων (peer-to-peer)
- Μαθαίνουμε για τα πρωτόκολλα εξετάζοντας δημοφιλή πρωτόκολλα επιπέδου εφαρμογής
 - HTTP
 - SMTP / POP3 / IMAP
 - DNS

Μερικές δικτυακές εφαρμογές

- ❑ Ηλεκτρονικό ταχυδρομείο (e-mail)
- ❑ Ιστός (web)
- ❑ Μηνύματα κειμένου (instant messaging)
- ❑ Απομακρυσμένη σύνδεση
- ❑ Κοινή χρήση αρχείων μεταξύ ομότιμων (P2P file sharing)
- ❑ Δικτυακά παιχνίδια πολλών χρηστών (multi-user network games)
- ❑ Ροή αποθηκευμένου βίντεο (YouTube, Hulu, Netflix)
- ❑ IP τηλεφωνία (voice over IP, π.χ. Skype)
- ❑ Τηλεσυνδιάσκεψη πραγματικού χρόνου
- ❑ Κοινωνική δικτύωση
- ❑ Αναζήτηση
- ❑ ...

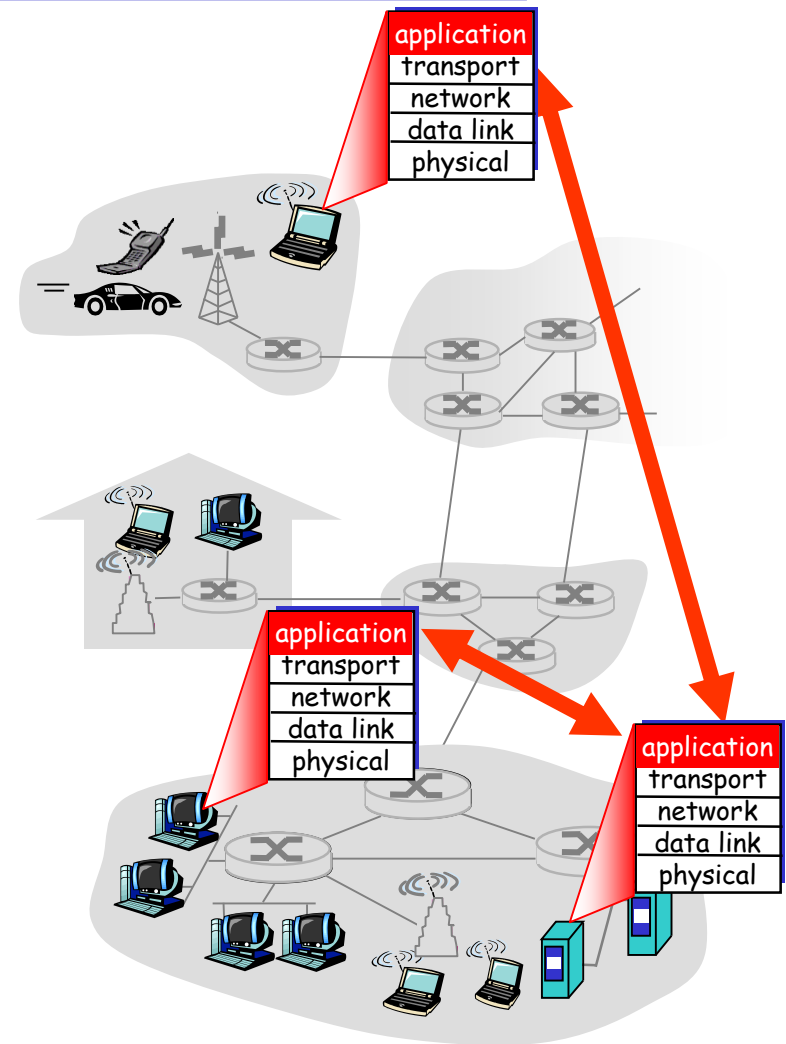
Δημιουργώντας μια δικτυακή εφαρμογή

Ανάπτυξη προγραμμάτων που

- τρέχουν σε (διαφορετικά) τερματικά συστήματα
- επικοινωνούν πάνω από το δίκτυο
- π.χ., το λογισμικό του εξυπηρετή web επικοινωνεί με το λογισμικό του browser

Δεν υπάρχει ανάγκη να γραφτεί λογισμικό για συσκευές του πυρήνα του δικτύου

- Οι συσκευές του πυρήνα του δικτύου δεν τρέχουν εφαρμογές χρήστη
- Το ότι οι εφαρμογές είναι στα τερματικά συστήματα επιτρέπει την ταχεία ανάπτυξη, διάδοσή τους



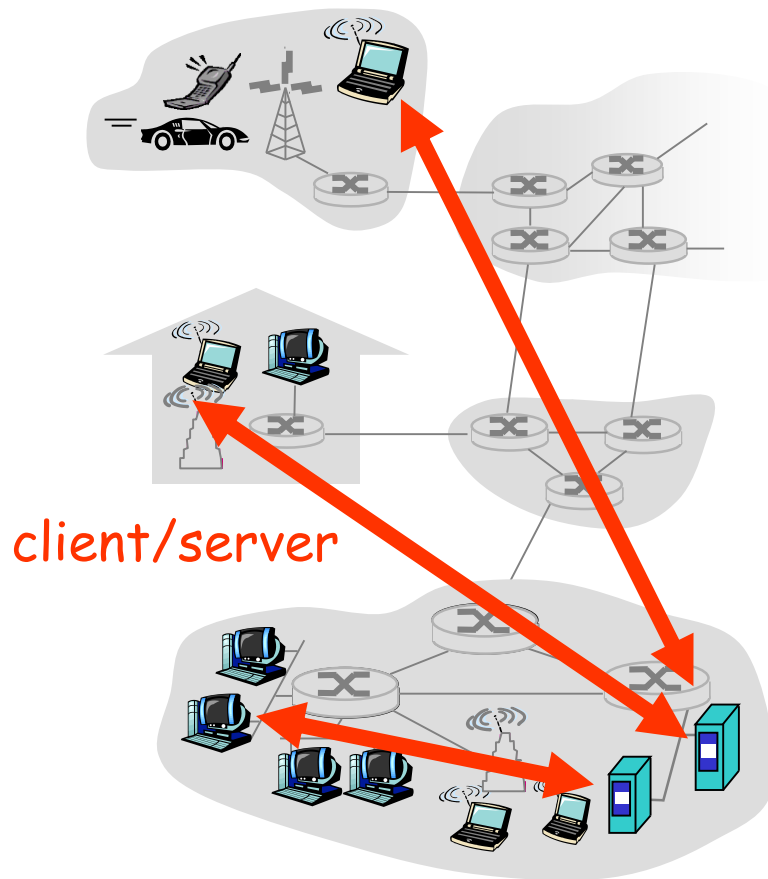
Κεφάλαιο 2: Επίπεδο εφαρμογής

- 2.1 Αρχές δικτυακών εφαρμογών
- 2.2 Web και HTTP
- 2.3 Ηλεκτρονικό ταχυδρομείο
 - SMTP, POP3, IMAP
- 2.4 DNS
- 2.5 Εφαρμογές P2P

Αρχιτεκτονικές εφαρμογών

- Πελάτη-εξυπηρέτη (Client-server)
- Μεταξύ ομότιμων (Peer-to-peer (P2P))

Αρχιτεκτονική πελάτη-εξυπηρέτη



Εξυπηρέτης (server):

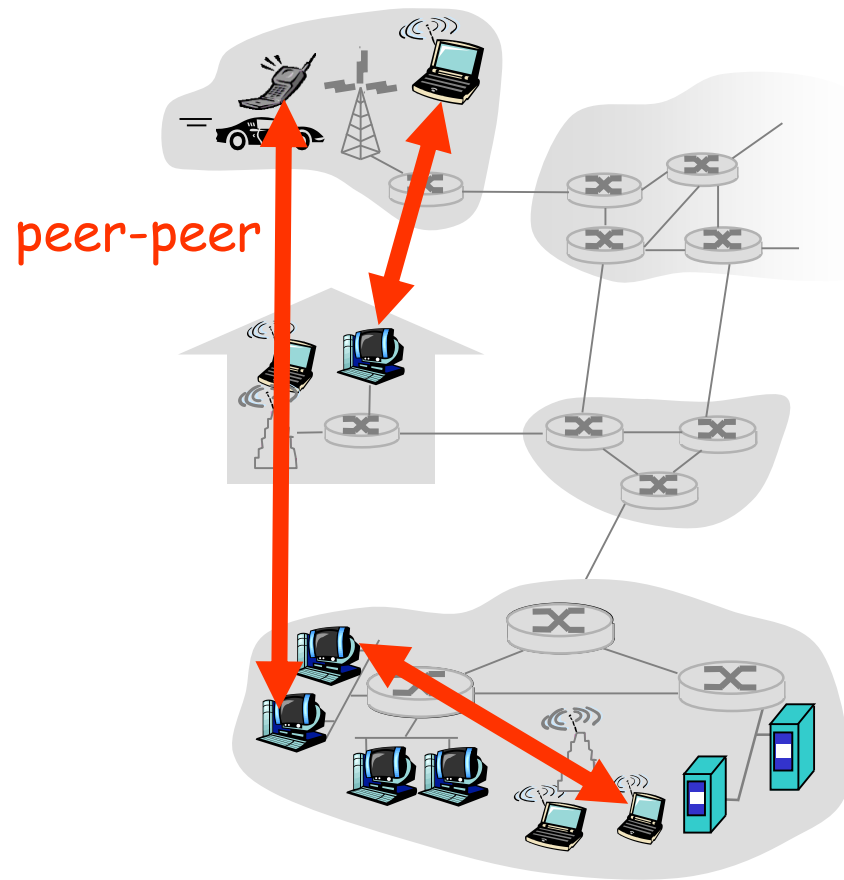
- Διαρκώς ενεργός υπολογιστής
- Μόνιμη διεύθυνση IP
- Κέντρα δεδομένων (data centers) για κλιμάκωση

Πελάτες (clients):

- Επικοινωνούν με τον εξυπηρέτη
- Ενδεχομένως να έχουν διακοπτόμενη σύνδεση
- Ενδεχομένως έχουν δυναμική διεύθυνση IP
- Δεν επικοινωνούν απευθείας μεταξύ τους

Αρχιτεκτονική μεταξύ ομότιμων

- Δεν υπάρχει διαρκώς ενεργός εξυπηρετής
- Τυχαία τερματικά συστήματα επικοινωνούν απευθείας μεταξύ τους
- Ομότιμοι ζητούν υπηρεσίες από άλλους ομότιμους, παρέχουν υπηρεσίες με τη σειρά τους σε άλλους ομότιμους
 - Αυτοεπεκτασιμότητα - νέοι ομότιμοι φέρνουν ικανότητα παροχής και ζήτησης νέων υπηρεσιών
- Διακοπτόμενη σύνδεση των ομότιμων και αλλαγή των διευθύνσεων IP
 - Πολύπλοκη διαχείριση



Διεργασίες που επικοινωνούν

Διεργασία (process): πρόγραμμα που τρέχει σε ένα υπολογιστή

- Στον **ίδιο υπολογιστή**, δύο διεργασίες επικοινωνούν χρησιμοποιώντας **διαδιεργασιακή επικοινωνία** (ορίζεται από το λειτουργικό).
- Οι διεργασίες σε **διαφορετικούς υπολογιστές** επικοινωνούν ανταλλάσσοντας **μηνύματα**

clients, servers

Διεργασία πελάτη: διεργασία που εκκινεί την επικοινωνία

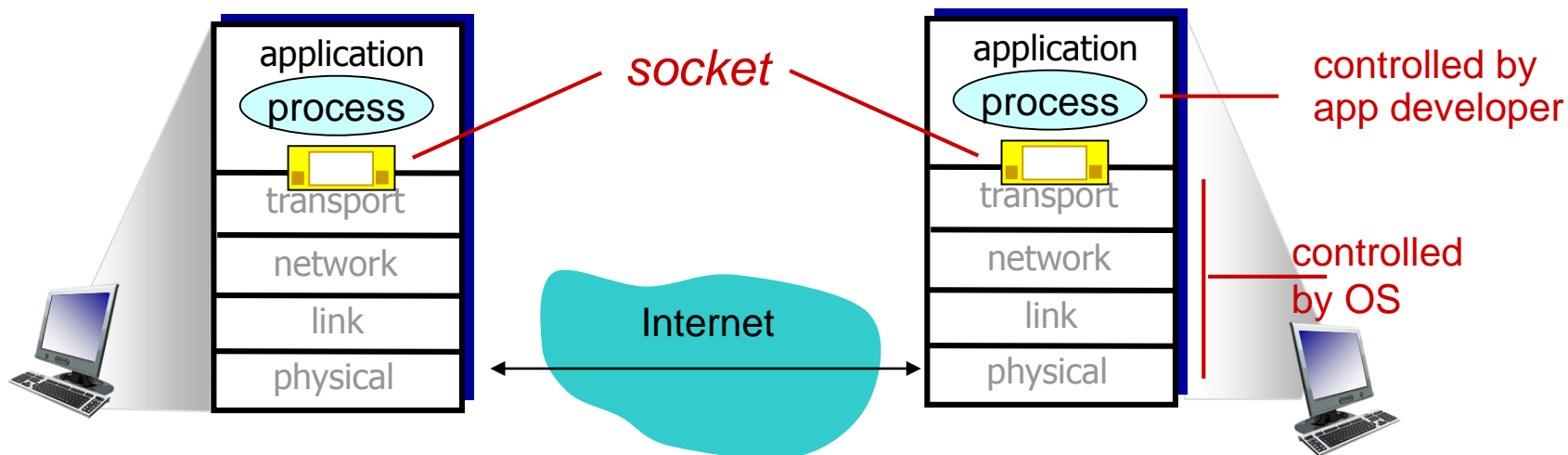
Διεργασία εξυπηρέτη: διεργασία που αναμένει να επικοινωνήσουν μαζί της

- Σημείωση: εφαρμογές με αρχιτεκτονικές P2P έχουν διεργασίες πελάτες και διεργασίες εξυπηρέτες

Sockets

Socket / API (Application Programmer's Interface) : διεπαφή ανάμεσα στην εφαρμογή και το δίκτυο (επίπεδο μεταφοράς) της διεργασίας

- Η διεργασία στέλνει/λαμβάνει μηνύματα στο/από το socket της
- socket ανάλογο "πόρτας"
 - Η διεργασία αποστολέας στέλνει μήνυμα "έξω από την πόρτα"
 - Η διεργασία αποστολέας βασίζεται στην υποδομή μεταφοράς στην άλλη πλευρά της πόρτας για να παραδώσει το μήνυμα στο socket της διεργασίας παραλήπτη



Διευθυνσιοδότηση διεργασιών

- Για να λαμβάνει μία διεργασία μηνύματα πρέπει να έχει ένα αναγνωριστικό (*identifier*)
- Κάθε υπολογιστής έχει μία μοναδική 32-bit διεύθυνση IP
- E: Αρκεί η IP διεύθυνση του υπολογιστή που τρέχει η διεργασία για την αναγνώριση της διεργασίας:
 - A: Όχι, πολλές διεργασίες μπορεί να τρέχουν στον ίδιο υπολογιστή
- Το **αναγνωριστικό** περιλαμβάνει τόσο τη **διεύθυνση IP** όσο και τον **αριθμό θύρας** που σχετίζεται με τη διεργασία στον υπολογιστή
- Παραδείγματα αριθμών θύρας:
 - Εξυπηρετής HTTP: 80
 - Εξυπηρετής Mail: 25
- Για να σταλεί μήνυμα HTTP στον εξυπηρετή web `gaia.cs.umass.edu`:
 - **Διεύθυνση IP**: 128.119.245.12
 - **Αριθμός θύρας**: 80
- Περισσότερα σε λίγο...

Ένα πρωτόκολλο επιπέδου εφαρμογής ορίζει

- ❑ Τα είδη των μηνυμάτων που ανταλλάσσονται,
 - π.χ. αίτηση, απόκριση
- ❑ Σύνταξη μηνύματος:
 - τι πεδία στα μηνύματα & πώς αυτά διαχωρίζονται
- ❑ Σημασιολογία των πεδίων
 - σημασία της πληροφορίας των πεδίων
- ❑ Κανόνες για το πότε και πώς οι διεργασίες στέλνουν και απαντούν σε μηνύματα

Ανοιχτά πρωτόκολλα:

- ❑ ορίζονται στα RFCs
- ❑ επιτρέπουν τη διαλειτουργικότητα
- ❑ π.χ. HTTP, SMTP

Ιδιοταγή πρωτόκολλα:

- ❑ π.χ. Skype

Τι υπηρεσίες μεταφοράς απαιτούν οι εφαρμογές;

Ακεραιότητα δεδομένων

- ❑ κάποιες εφαρμογές (π.χ. ήχου), ανέχονται κάποιες απώλειες
- ❑ άλλες εφαρμογές (π.χ. μεταφορά αρχείου, διαδικτυακές συναλλαγές) απαιτούν 100% αξιόπιστη μεταφορά δεδομένων

Χρονισμός

- ❑ κάποιες εφαρμογές (π.χ. τηλεφωνία Διαδικτύου, διαδραστικά παιχνίδια) απαιτούν χαμηλή καθυστέρηση για να είναι «αποτελεσματικές»

Ρυθμαπόδοση (Throughput)

- ❑ Κάποιες εφαρμογές (π.χ., multimedia) απαιτούν κάποια ελάχιστη ρυθμαπόδοση για να είναι «αποτελεσματικές»
- ❑ Άλλες εφαρμογές («ελαστικές») «βολεύονται» με όση ρυθμαπόδοση πάρουν

Ασφάλεια

- ❑ Κρυπτογράφηση, ακεραιότητα δεδομένων...

Απαιτήσεις υπηρεσιών μεταφοράς: κοινές εφαρμογές

Εφαρμογή	Απώλεια δεδομένων	Ρυθμαπόδοση	Ευαισθησία ως προς το χρόνο
Μεταφορά αρχείου	όχι απώλειες	ελαστική	όχι
e-mail	όχι απώλειες	ελαστική	όχι
Έγγραφα Web	όχι απώλειες	ελαστική	όχι
Ήχος/ βίντεο πραγματικού χρόνου	ανοχή στις απώλειες	ήχος: 5kbps-1Mbps, βίντεο:10kbps-5Mbps	ναι, 100δες msec
Αποθηκευμένος ήχος/βίντεο	ανοχή στην απώλειες	ίδια με παραπάνω	ναι, λίγα secs
Διαδραστικά παιχνίδια	ανοχή στην απώλειες	ως λίγα kbps	ναι, 100δες msec
Μηνύματα κειμένου	όχι απώλειες	ελαστική	ναι και όχι

Υπηρεσίες των πρωτοκόλλων μεταφοράς

Υπηρεσία του TCP:

- ❑ **συνδεδεισμενός (connection-oriented):** απαιτείται αρχικοποίηση μεταξύ διεργασιών πελάτη και εξυπηρέτη
- ❑ **αξιόπιστη μεταφορά δεδομένων** μεταξύ διεργασίας αποστολής και λήψης
- ❑ **έλεγχος ροής:** ο αποστολέας δεν υπερφορτώνει τον παραλήπτη
- ❑ **έλεγχος συμφόρησης:** επιβράδυνση αποστολέα όταν το δίκτυο είναι υπερφορτωμένο
- ❑ **δεν παρέχει:** συγχρονισμό, εγγυήσεις ως προς ελάχιστη ρυθμαπόδοση, ασφάλεια

Υπηρεσία του UDP:

- ❑ **αναξιόπιστη μεταφορά δεδομένων** μεταξύ διεργασίας αποστολής και λήψης
- ❑ **δεν παρέχει:** αρχικοποίηση σύνδεσης, αξιοπιστία, έλεγχο ροής, έλεγχο συμφόρησης, εγγύηση ως προς τον χρόνο ή τη ρυθμαπόδοση

Ε: Γιατί υπάρχει το UDP;

Εφαρμογές Διαδικτύου: πρωτόκολλο εφαρμογής, μεταφοράς

Εφαρμογή	Πρωτόκολλο εφαρμογής	Υποκείμενο Πρωτόκολλο μεταφοράς
Ηλεκτρονικό ταχυδρομείο e-mail	SMTP [RFC 2821]	TCP
Απομακρυσμένη προσπέλαση τερματικού	Telnet [RFC 854]	TCP
Web	HTTP [RFC 2616]	TCP
Μεταφορά αρχείων	FTP [RFC 959]	TCP
Πολυμέσα συνεχούς ροής	HTTP (π.χ. YouTube), RTP[RFC 1889]	TCP ή UDP
Τηλεφωνία Διαδικτύου	SIP, RTP, ιδιοταγές (π.χ. Skype)	TCP ή UDP

Κεφάλαιο 2: Επίπεδο εφαρμογής

- 2.1 Αρχές δικτυακών εφαρμογών
- 2.2 Web και HTTP
- 2.3 Ηλεκτρονικό ταχυδρομείο
 - SMTP, POP3, IMAP
- 2.4 DNS
- 2.5 Εφαρμογές P2P

Web και HTTP

- ❑ Μία ιστοσελίδα αποτελείται από αντικείμενα (objects)
- ❑ Το αντικείμενο μπορεί να είναι αρχείο HTML, εικόνα JPEG, Java applet, αρχείο ήχου, ...
- ❑ Η σελίδα Web αποτελείται από ένα βασικό αρχείο HTML που περιλαμβάνει διάφορα αντικείμενα στα οποία γίνεται αναφορά
- ❑ Κάθε αντικείμενο διευθυνσιοδοτείται με ένα URL

`www.someschool.edu/someDept/pic.gif`

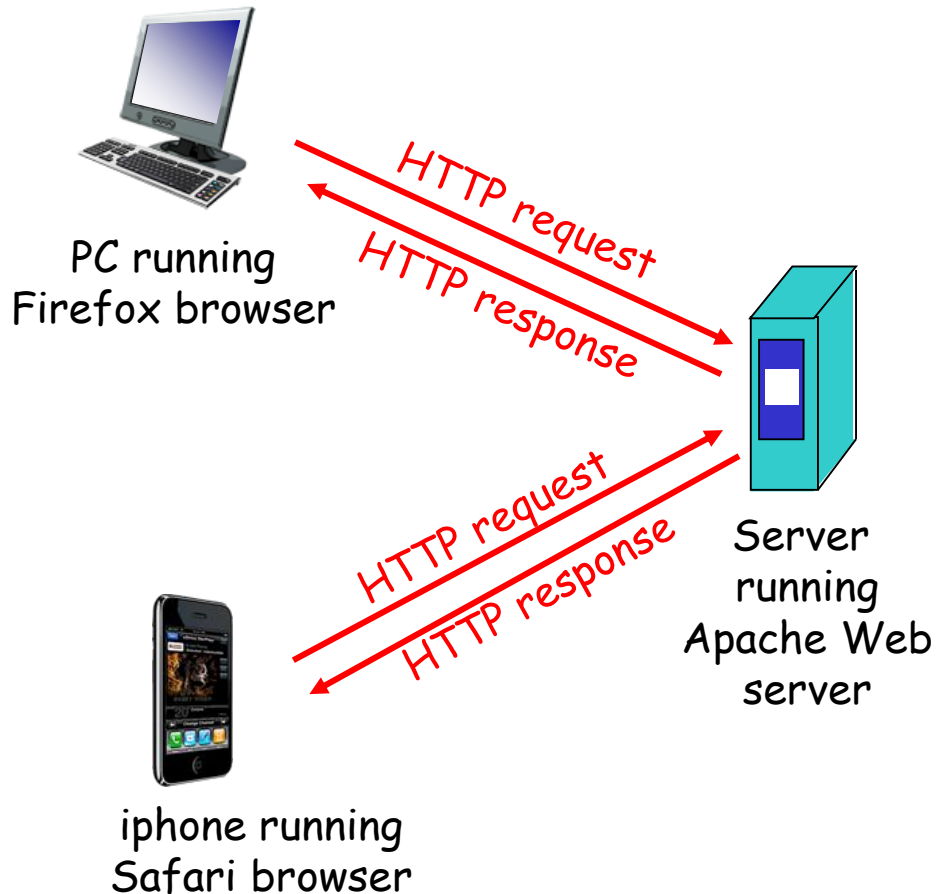
όνομα υπολογιστή

όνομα διαδρομής

Επισκόπηση του HTTP

HTTP: HyperText Transfer Protocol (Πρωτόκολλο Μεταφοράς Υπερκειμένου)

- Πρωτόκολλο επιπέδου εφαρμογής του Web
- Μοντέλο πελάτη/εξυπηρετή
 - **πελάτης:** browser που ζητά, λαμβάνει, (χρησιμοποιώντας το HTTP πρωτόκολλο) και "παρουσιάζει" αντικείμενα του Web
 - **εξυπηρετής:** Web server στέλνει (χρησιμοποιώντας το HTTP πρωτόκολλο) αντικείμενα ως απόκριση σε αιτήσεις



Επισκόπηση του HTTP (συνέχεια)

Χρησιμοποιεί TCP:

- ❑ ο πελάτης ξεκινά σύνδεση TCP με τον εξυπηρέτη (δημιουργεί socket), θύρα 80
- ❑ ο εξυπηρέτης αποδέχεται τη σύνδεση TCP από τον πελάτη
- ❑ ανταλλάσσονται μηνύματα HTTP (μηνύματα πρωτοκόλλου επιπέδου εφαρμογής) μεταξύ browser (πελάτης HTTP) και Web server (εξυπηρέτης HTTP)
- ❑ κλείνει η σύνδεση TCP

Το HTTP είναι ακαταστατικό (stateless)

- ❑ ο εξυπηρέτης δε διατηρεί πληροφορία σχετικά με προηγούμενες αιτήσεις του πελάτη

παρενθετικά

Τα πρωτόκολλα που διατηρούν στο περιθώριο την κατάσταση (state) είναι πολύπλοκα

- ❑ πρέπει να διατηρείται προηγούμενη ιστορία (κατάσταση)
- ❑ αν ο εξυπηρέτης / πελάτης καταρρεύσει, οι απόψεις τους για την κατάσταση μπορεί να μην είναι συνεπείς, και να πρέπει να συγκεραστούν

Συνδέσεις HTTP

Μη παραμένον HTTP (Non-persistent HTTP)

- ❑ Στέλνεται το πολύ ένα αντικείμενο πάνω από μία σύνδεση TCP
 - η σύνδεση μετά κλείνει
- ❑ η λήψη πολλαπλών αντικειμένων απαιτεί πολλαπλές συνδέσεις

Παραμένον HTTP (Persistent HTTP)

- ❑ Πολλαπλά αντικείμενα μπορεί να σταλούν πάνω από την ίδια σύνδεση TCP μεταξύ πελάτη και εξυπηρετή

Μη παραμένον HTTP

Υποθέστε ότι ο χρήστης εισάγει το URL

`www.someSchool.edu/someDepartment/home.index`

(περιέχει κείμενο,
αναφορές σε 10
εικόνες jpeg)

1a. Ο πελάτης HTTP ξεκινά σύνδεση TCP προς τη διεργασία του εξυπηρέτη HTTP στο `www.someSchool.edu` στη θύρα 80

1b. Ο εξυπηρέτης HTTP στον υπολογιστή `www.someSchool.edu` περιμένει σύνδεση TCP στη θύρα 80, "αποδέχεται" τη σύνδεση, ειδοποιεί τον πελάτη

2. Ο HTTP client στέλνει HTTP μήνυμα αίτησης (*request message*) (που περιέχει το URL) στη socket της σύνδεσης TCP. Το μήνυμα υποδεικνύει ότι ο client θέλει το αντικείμενο `someDepartment/home.index`

3. Ο εξυπηρέτης HTTP λαμβάνει το μήνυμα αίτησης, σχηματίζει μήνυμα απόκρισης (*response message*) που περιέχει το αντικείμενο που ζητήθηκε και στέλνει το μήνυμα στη socket του

χρόνος



Μη παραμένον HTTP (συνέχεια)

4. Ο εξυπηρέτης HTTP κλείνει τη σύνδεση TCP

5. Ο πελάτης HTTP λαμβάνει το μήνυμα απόκρισης που περιέχει το html αρχείο, απεικονίζει την html. Επεξεργαζόμενος το html αρχείο, βρίσκει 10 αναφερόμενα αντικείμενα jpeg

6. Τα βήματα 1-5 επαναλαμβάνονται για καθένα από τα 10 αντικείμενα jpeg

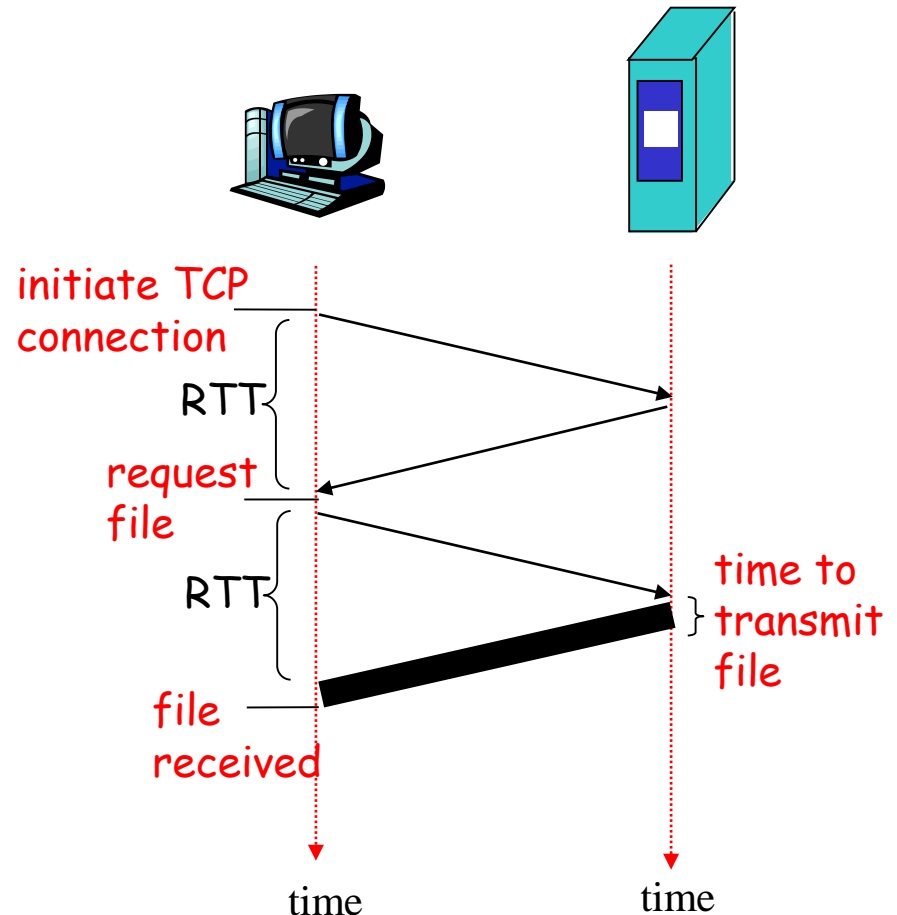
χρόνος

Μη παραμένον HTTP: χρόνος απόκρισης

RTT (round trip time): ο χρόνος που απαιτείται για ένα μικρό πακέτο να ταξιδέψει από τον πελάτη στον εξυπηρέτη και να επιστρέψει πίσω

Χρόνος απόκρισης:

- ένα RTT για την έναρξη της σύνδεσης TCP
- ένα RTT για την αίτηση HTTP και επιστροφή των πρώτων bytes της απόκρισης HTTP
- χρόνος μετάδοσης αρχείου
- χρόνος απόκρισης = $2RTT +$ χρόνος μετάδοσης αρχείου



Παραμένον HTTP

Θέματα με το μη παραμένον

HTTP:

- ❑ απαιτεί 2 RTT ανά αντικείμενο
- ❑ το λειτουργικό επιβαρύνεται για κάθε σύνδεση TCP
- ❑ οι browsers ανοίγουν συχνά **παράλληλες συνδέσεις TCP** για να φέρουν αντικείμενα που αναφέρονται

Παραμένον HTTP

- ❑ ο εξυπηρέτης αφήνει τη σύνδεση ανοιχτή αφού στείλει την απόκριση
- ❑ τα επόμενα HTTP μηνύματα μεταξύ του ίδιου πελάτη / εξυπηρέτη στέλνονται μέσω της ανοιχτής σύνδεσης
- ❑ ο πελάτης στέλνει αιτήσεις μόλις συναντήσει αναφερόμενο αντικείμενο
- ❑ ένα RTT για όλα τα αναφερόμενα αντικείμενα (**pipelining**)

Μήνυμα αίτησης HTTP

- ❑ δύο είδη HTTP μηνυμάτων : *αίτηση (request)*, *απόκριση (response)*
- ❑ μήνυμα HTTP αίτησης :
 - ASCII (μορφή αναγνώσιμη από τον άνθρωπο)

γραμμή αίτησης (request line)
(εντολές GET, POST, HEAD)

γραμμές κεφαλίδας
(header lines)

```
GET /somedir/page.html HTTP/1.1
Host: www.someschool.edu
User-agent: Mozilla/4.0
Connection: close
Accept-language: fr
```

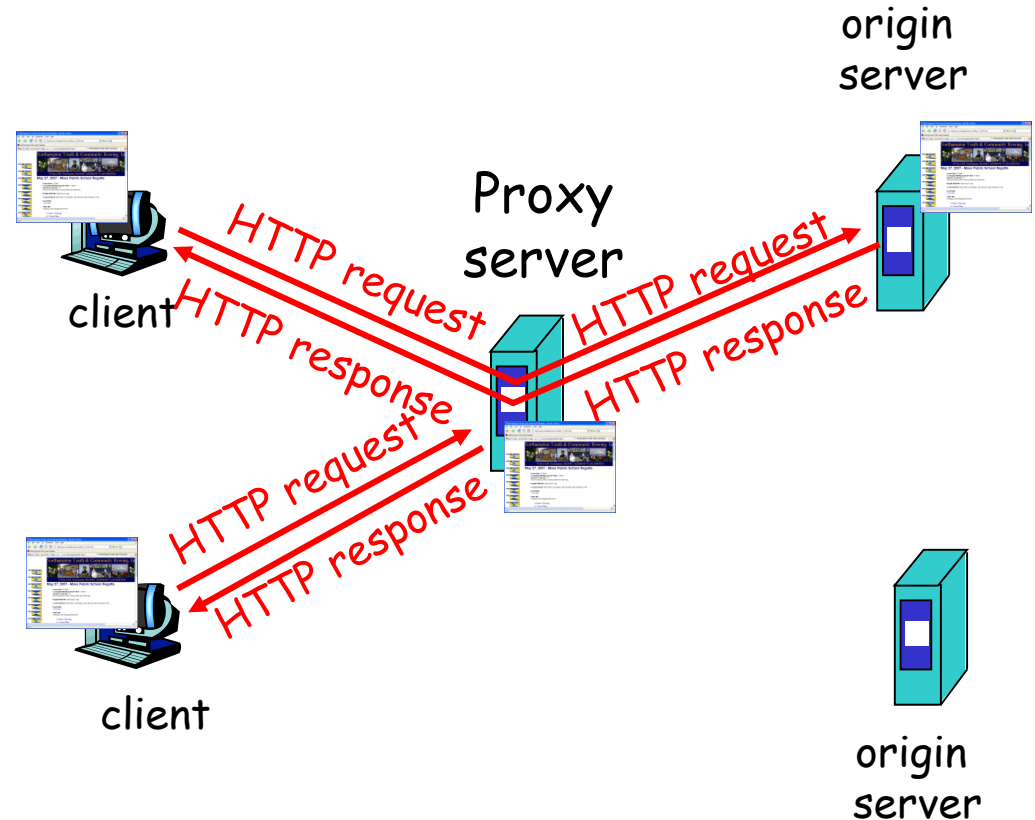
(extra carriage return, line feed)

υποδεικνύει το τέλος του μηνύματος

Web caches (proxy server)

Σκοπός: να εξυπηρετούνται κάποιες αιτήσεις πελατών χωρίς εμπλοκή του εξυπηρετή προέλευσης

- Ο χρήστης παραμετροποιεί τον browser: πρόσβαση στο Web μέσω της cache
- Ο browser στέλνει όλες τις αιτήσεις HTTP στην cache
 - Το αντικείμενο στην cache: η cache επιστρέφει το αντικείμενο
 - Διαφορετικά η cache ζητά το αντικείμενο από τον εξυπηρετή προέλευσης, μετά επιστρέφει το αντικείμενο στον πελάτη



Περισσότερα για το Web caching

- Η cache δρα τόσο ως πελάτης όσο και ως εξυπηρετής
 - εξυπηρετής για τον αρχικό αιτούντα πελάτη
 - πελάτης για τον εξυπηρετή προέλευσης
- Τυπικά η cache εγκαθίσταται από τον ISP (πανεπιστήμιο, εταιρία, περιφερειακός ISP)

Γιατί Web caching :

- Μείωση του χρόνου απόκρισης των αιτήσεων πελατών
- Μείωση της κίνησης στη ζεύξη πρόσβασης ενός οργανισμού
- Στο Διαδίκτυο, επιτρέπει σε «φτωχούς» πάροχους περιεχομένου να παραδίδουν αποτελεσματικά το περιεχόμενό τους (το ίδιο επιτυγχάνει και η P2P κοινή χρήση αρχείων)

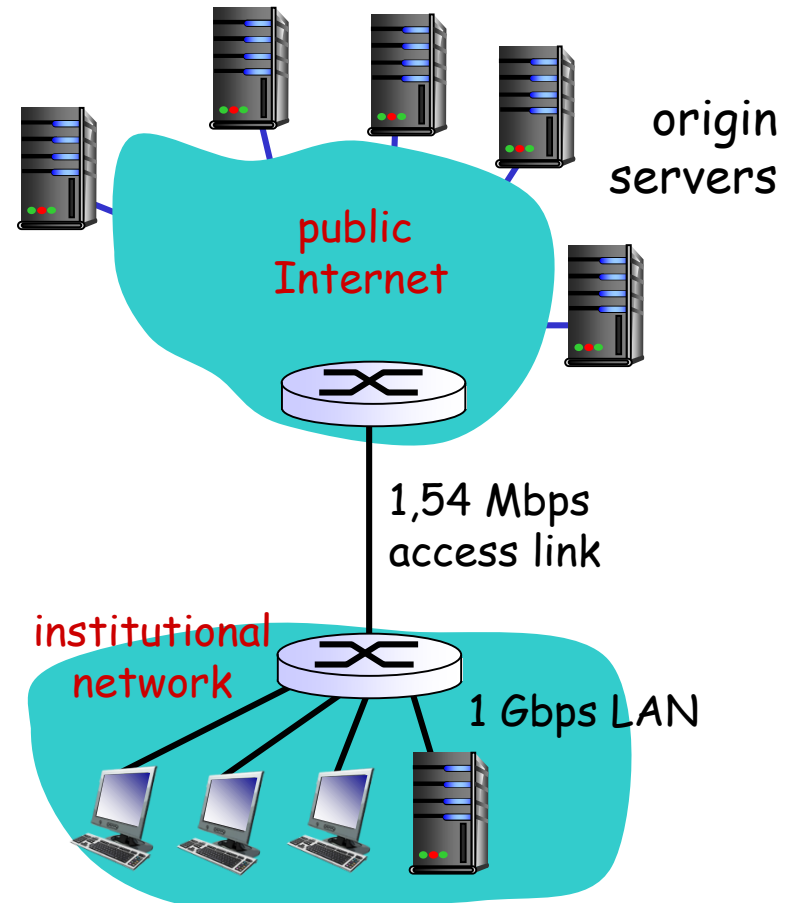
Παράδειγμα Caching

Υποθέσεις

- Μέσο μέγεθος αντικειμένου = 100 Kbits
- Μέσος ρυθμός αιτήσεων από τους browsers προς τους εξυπηρέτες προέλευσης = 15/sec
- Μέσος ρυθμός δεδομένων προς τους browsers: 1,50 Mbps
- RTT από το δρομολογητή του ιδρύματος προς κάθε εξυπηρέτη προέλευσης και πίσω στο δρομολογητή = 2 sec
- Ρυθμός ζεύξης πρόσβασης: 1,54 Mbps

Συνέπειες

- Αξιοποίηση του LAN = 0,15%
- Αξιοποίηση της ζεύξης πρόσβασης = 99%
- Συνολική καθυστέρηση = καθυστέρηση Διαδικτύου + καθυστέρηση πρόσβασης + καθυστέρηση LAN
= 2 sec + minutes + microseconds



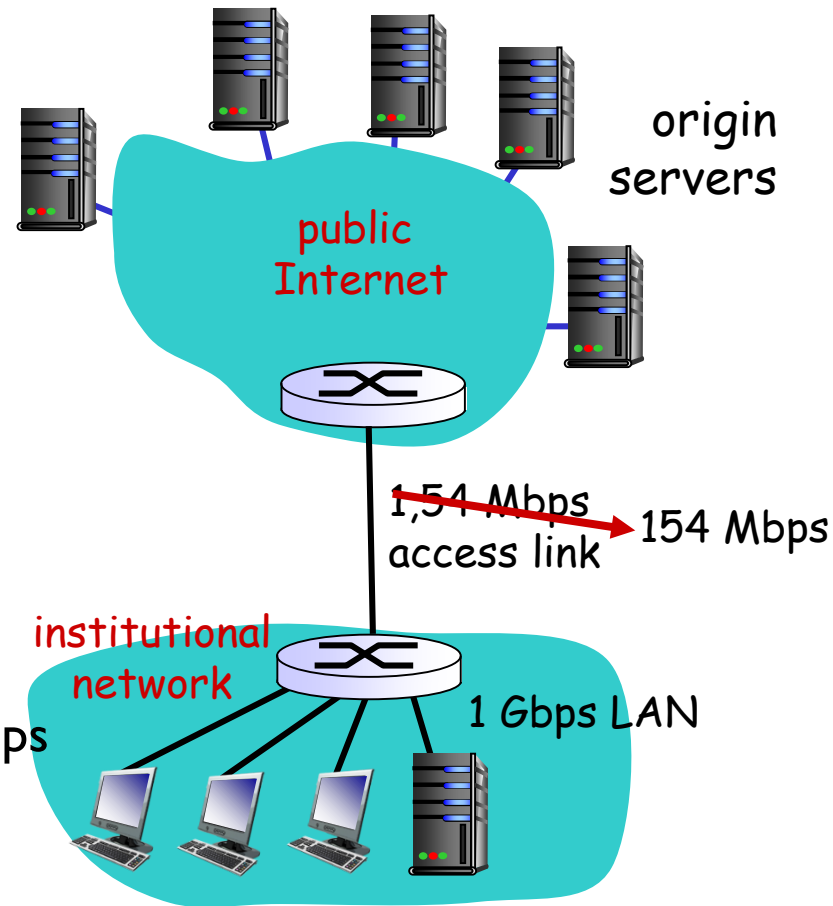
Παράδειγμα Caching: ταχύτερη ζεύξη πρόσβασης

Υποθέσεις

- ❑ Μέσο μέγεθος αντικειμένου = 100 Kbits
- ❑ Μέσος ρυθμός αιτήσεων από τους browsers προς τους εξυπηρετές προέλευσης = 15/sec
- ❑ Μέσος ρυθμός δεδομένων προς τους browsers: 1,50 Mbps
- ❑ RTT από το δρομολογητή του ιδρύματος προς κάθε εξυπηρετή προέλευσης και πίσω στο δρομολογητή = 2 sec
- ❑ Ρυθμός ζεύξης πρόσβασης: 1,54 Mbps

Συνέπειες

- ❑ Αξιοποίηση του LAN = 0,15%
 - ❑ Αξιοποίηση της ζεύξης πρόσβασης = 99%
 - ❑ Συνολική καθυστέρηση = καθυστέρηση Διαδικτύου + καθυστέρηση πρόσβασης + καθυστέρηση LAN
- = 2 sec + minutes + μsecs



Κόστος: αυξημένη ταχύτητα ζεύξης πρόσβασης (ακριβό!)

msecs

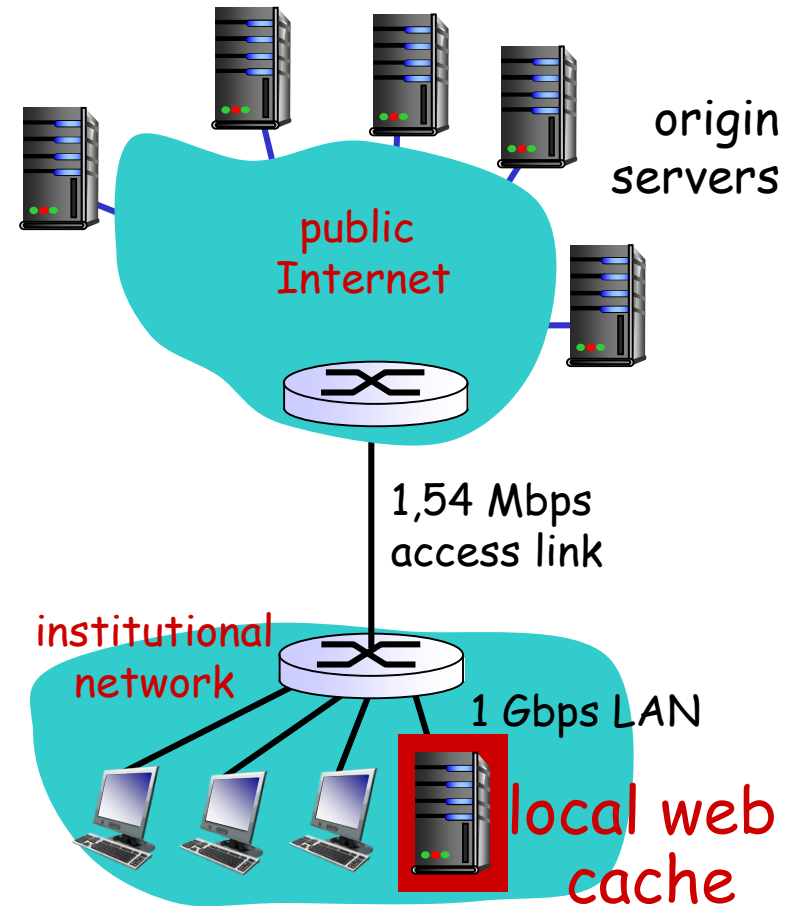
Παράδειγμα Caching: εγκατάσταση τοπικής cache

Υποθέσεις

- ❑ Μέσο μέγεθος αντικειμένου = 100 Kbits
- ❑ Μέσος ρυθμός αιτήσεων από τους browsers προς τους εξυπηρέτες προέλευσης = 15/sec
- ❑ Μέσος ρυθμός δεδομένων προς τους browsers: 1,50 Mbps
- ❑ RTT από το δρομολογητή του ιδρύματος προς κάθε εξυπηρέτη προέλευσης και πίσω στο δρομολογητή = 2 sec
- ❑ Ρυθμός ζεύξης πρόσβασης: 1,54 Mbps

Συνέπειες

- ❑ Αξιοποίηση του LAN = 0,15%
- ❑ Αξιοποίηση της ζεύξης πρόσβασης = ;
- ❑ Συνολική καθυστέρηση = ;



πώς υπολογίζεται η αξιοποίηση της ζεύξης και η καθυστέρηση;

Κόστος: web cache (φθινό!)

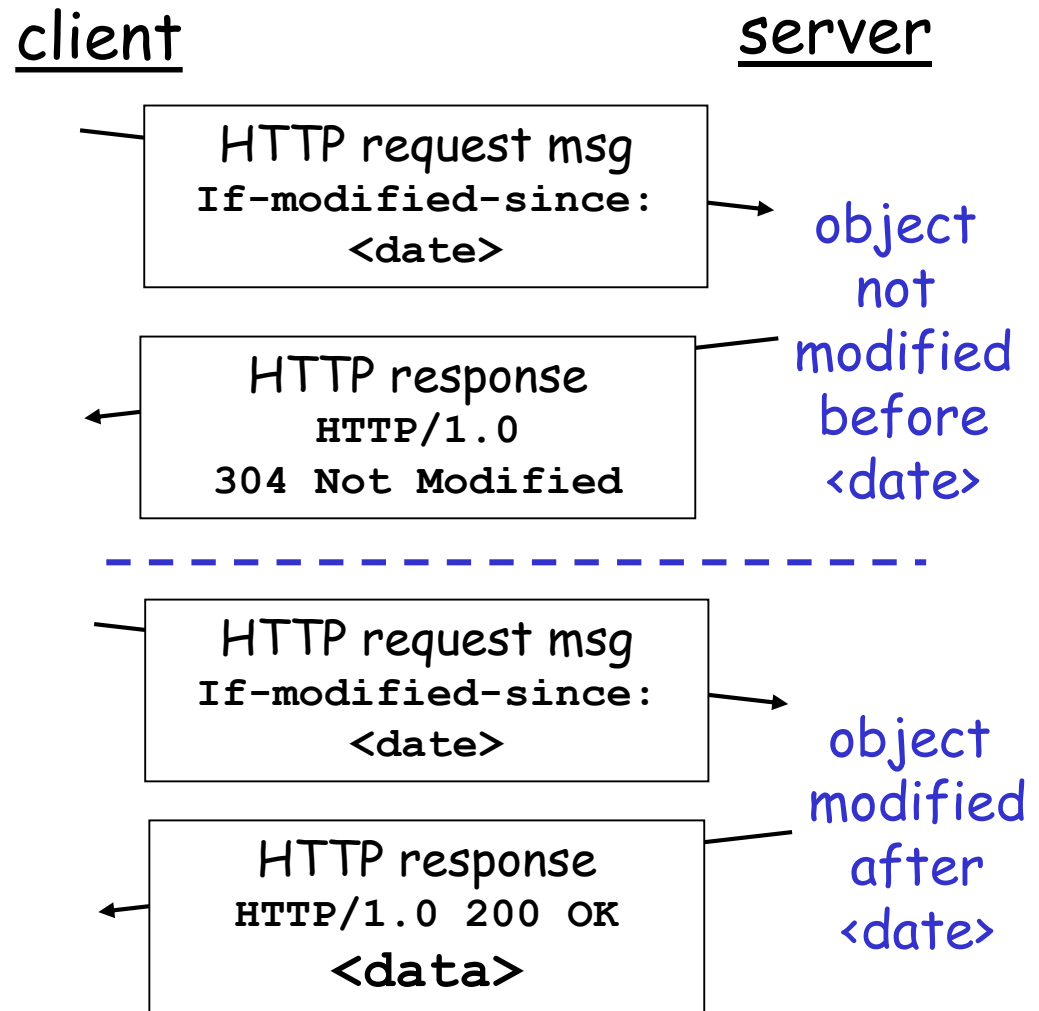
Παράδειγμα Caching: εγκατάσταση τοπικής cache (συνέχεια)

Υπολογισμός αξιοποίησης της ζεύξης πρόσβασης και καθυστέρησης με cache:

- ❑ έστω ότι το ποσοστό επιτυχίας της cache είναι 0.4
 - 40% των αιτήσεων εξυπηρετούνται στην cache
 - 60% των αιτήσεων εξυπηρετούνται στην προέλευση
- ❑ αξιοποίηση ζεύξης πρόσβασης:
 - 60% των αιτήσεων χρησιμοποιούν τη ζεύξη πρόσβασης
- ❑ ρυθμός δεδομένων προς browsers μέσω της ζεύξης = $0,6 * 1,50 \text{ Mbps} = 0,9 \text{ Mbps}$
 - αξιοποίηση = $0,9 / 1,54 = 0,58$
- ❑ συνολική καθυστέρηση
 - = $0,6 * (\text{καθυστέρηση από εξυπηρέτες προέλευσης}) + 0,4 * (\text{καθυστέρηση από εξυπηρέτηση από cache})$
 - = $0,6 (2,01) + 0,4 (\sim \text{msecs})$
 - = $\sim 1,2 \text{ secs}$
 - λιγότερο σε σχέση με τη ζεύξη 154 Mbps (και φθηνότερο!)

Το Get υπό συνθήκη (Conditional GET)

- ❑ **Σκοπός:** να μην στέλνεται το αντικείμενο αν η cache έχει ενήμερη (up-to-date) έκδοση
 - χωρίς καθυστέρηση μετάδοσης του αντικειμένου
 - χαμηλότερη χρήση ζεύξης
- ❑ **cache:** καθορίζει την ημερομηνία του αποθηκευμένου αντιγράφου στην αίτηση HTTP
`If-modified-since:`
`<date>`
- ❑ **εξυπηρέτης:** η απόκριση δεν περιέχει αντικείμενο αν το αντίγραφο στην cache δεν έχει τροποποιηθεί:
`HTTP/1.0 304 Not Modified`



Κεφάλαιο 2: Επίπεδο εφαρμογής

- 2.1 Αρχές δικτυακών εφαρμογών
- 2.2 Web και HTTP
- 2.3 Ηλεκτρονικό ταχυδρομείο
 - SMTP, POP3, IMAP
- 2.4 DNS
- 2.5 Εφαρμογές P2P

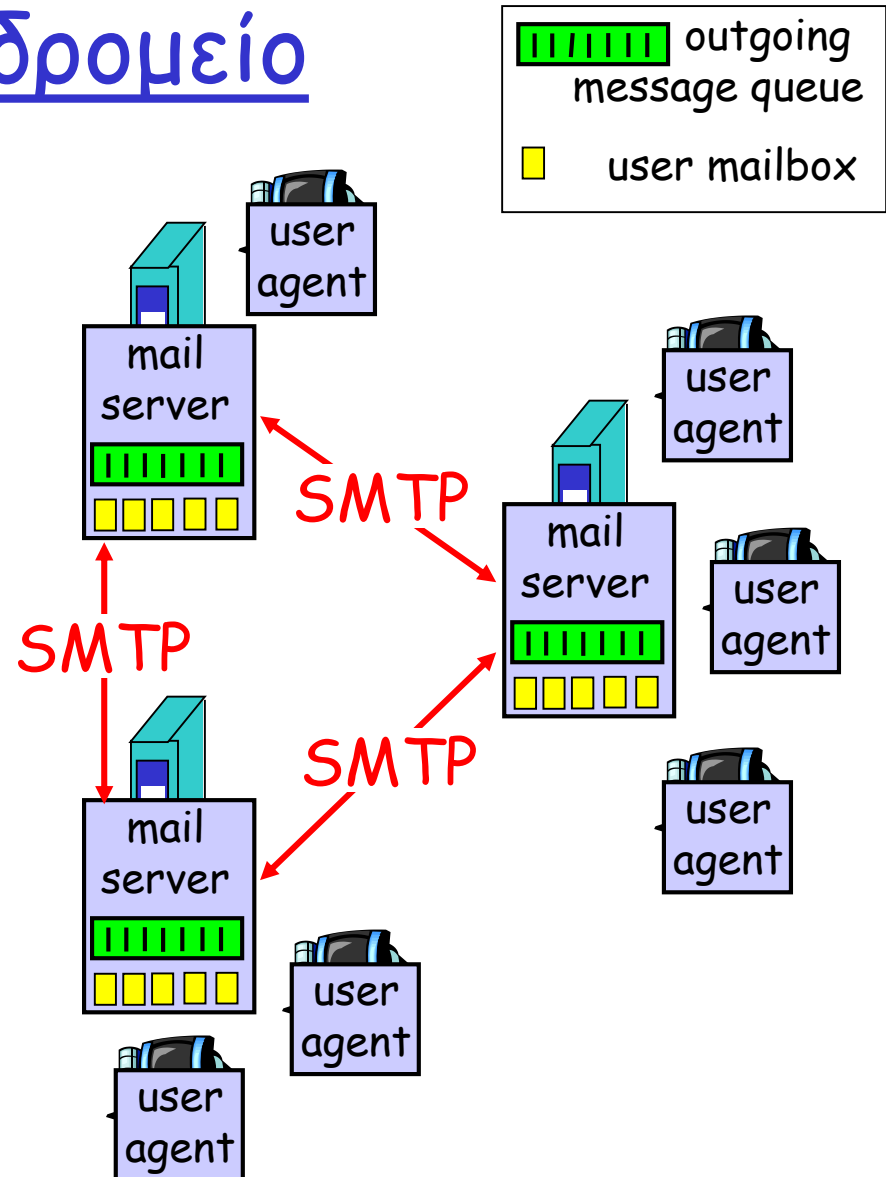
Ηλεκτρονικό Ταχυδρομείο

Τρία κύρια συστατικά μέρη:

- Πράκτορες χρήστη (user agents)
- Εξυπηρετές ταχυδρομείου (mail servers)
- simple mail transfer protocol: SMTP

Πράκτορας Χρήστη (User Agent)

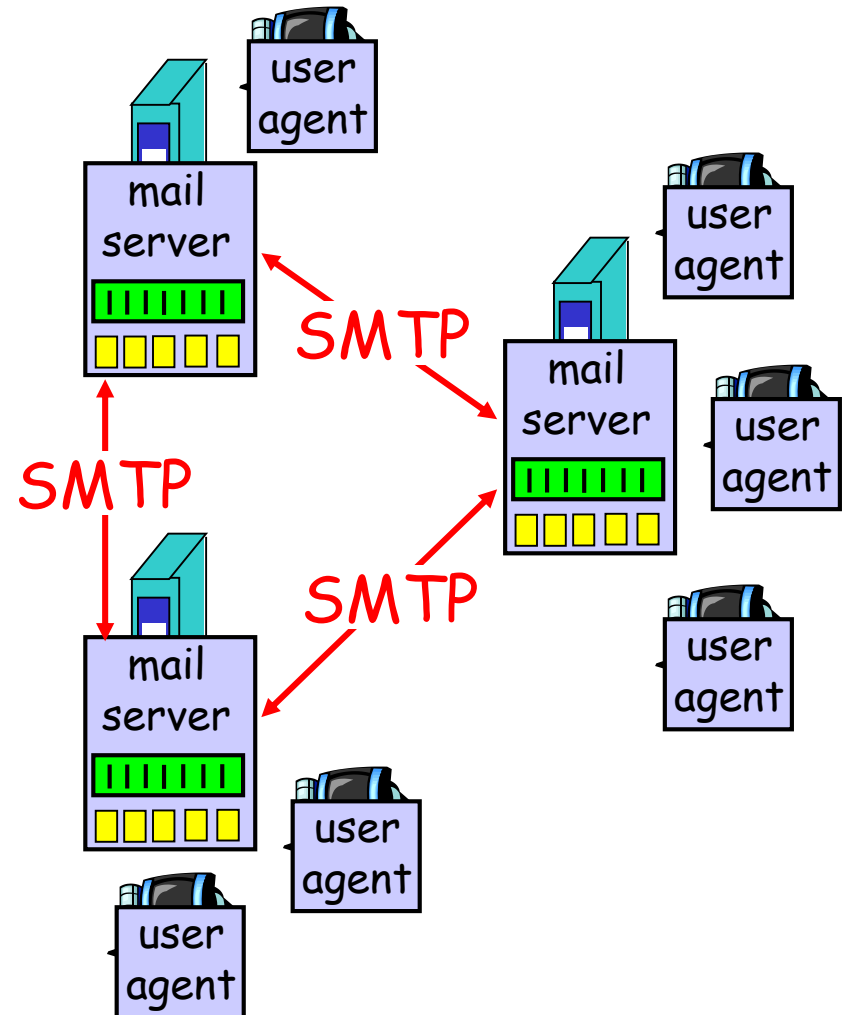
- Δηλ. "mail reader" (αναγνώστης ταχυδρομείου)
- Σύνθεση, επεξεργασία, ανάγνωση μηνυμάτων αλληλογραφίας
- π.χ., Outlook, Mozilla Thunderbird, iPhone mail client
- Εξερχόμενα, εισερχόμενα μηνύματα αποθηκευμένα στον εξυπηρετή



Ηλεκτρονικό ταχυδρομείο: εξυπηρέτες ταχυδρομείου (mail servers)

Εξυπηρέτες Ταχυδρομείου (Mail Servers)

- Η ταχυδρομική θυρίδα (mailbox) περιέχει τα εισερχόμενα μηνύματα του χρήστη
- Ουρά εξερχομένων μηνυμάτων αλληλογραφίας που πρέπει να σταλούν
- Πρωτόκολλο SMTP μεταξύ εξυπηρετών ταχυδρομείου για την αποστολή email
 - ❖ πελάτης : εξυπηρέτης ταχυδρομείου αποστολέας
 - ❖ «εξυπηρέτης»: εξυπηρέτης ταχυδρομείου παραλήπτης

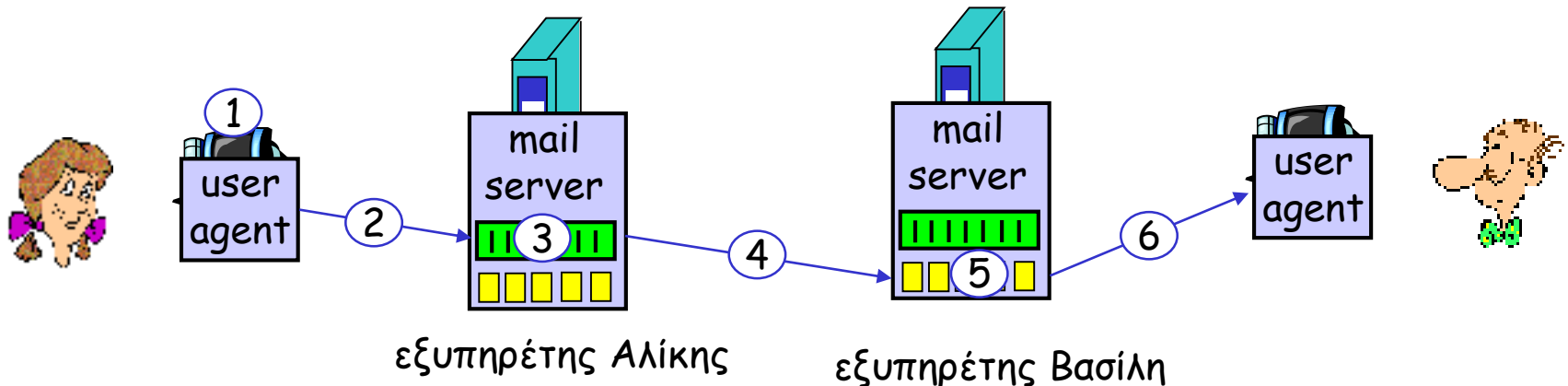


Ηλεκτρονικό Ταχυδρομείο: SMTP [RFC 2821]

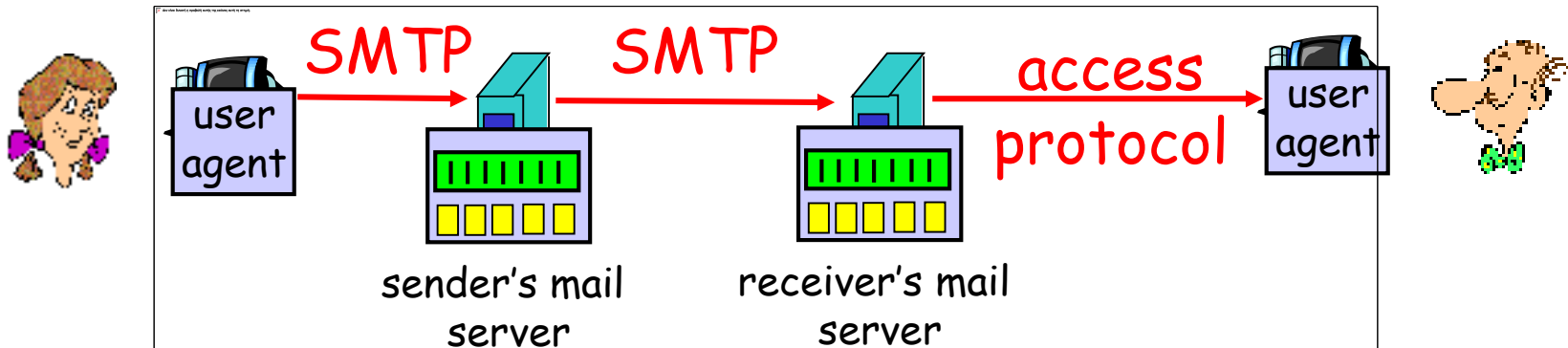
- ❑ **Χρησιμοποιεί TCP** για την αξιόπιστη μεταφορά μηνυμάτων από τον πελάτη στον εξυπηρέτη, **θύρα 25**
- ❑ Απευθείας μεταφορά: από τον εξυπηρέτη αποστολέα προς τον εξυπηρέτη παραλήπτη
- ❑ Τρεις φάσεις της μεταφοράς
 - ❖ χειραψία (χαιρετισμός)
 - ❖ μεταφορά μηνυμάτων
 - ❖ τερματισμός
- ❑ Αλληλεπίδραση εντολής/απόκρισης (όπως το HTTP, FTP)
 - **εντολές:** κείμενο ASCII
 - **απόκριση:** κωδικός κατάστασης και φράση
- ❑ Τα μηνύματα πρέπει να είναι σε 7-bit ASCII

Σενάριο: Η Αλίκη στέλνει μήνυμα στον Βασίλη

- 1) Η Αλίκη χρησιμοποιεί πράκτορα χρήστη (user agent-UA) για τη σύνθεση του μηνύματος "προς" `vassilis@di.uoa.gr`
- 2) Ο πράκτορας της Αλίκης στέλνει το μήνυμα στον εξυπηρέτη ταχυδρομείου της, το μήνυμα τοποθετείται στην ουρά μηνυμάτων
- 3) Η πλευρά του πελάτη του SMTP ανοίγει TCP σύνδεση με τον εξυπηρέτη ταχυδρομείου του Βασίλη
- 4) Ο πελάτης SMTP στέλνει το μήνυμα της Αλίκης πάνω από τη σύνδεση TCP
- 5) Ο εξυπηρέτης ταχυδρομείου του Βασίλη τοποθετεί το μήνυμα στην ταχυδρομική θυρίδα του Βασίλη
- 6) Ο Βασίλη χρησιμοποιεί το δικό του πράκτορα για να το διαβάσει



Πρωτόκολλα προσπέλασης ταχυδρομείου



- **SMTP**: παράδοση/αποθήκευση στον εξυπηρέτη του παραλήπτη
- Πρωτόκολλο προσπέλασης ταχυδρομείου (mail access protocol): ανάκτηση από τον εξυπηρέτη
 - **POP**: Post Office Protocol [RFC 1939]: εξουσιοδότηση και «κατέβασμα» ("download")
 - ❖ **IMAP**: Internet Mail Access Protocol [RFC 1730]: περισσότερες δυνατότητες, συμπεριλαμβανομένης της διαχείρισης αποθηκευμένων μηνυμάτων στον εξυπηρέτη
 - **HTTP**: Gmail, Hotmail, Yahoo! Mail, κτλ.

Κεφάλαιο 2: Επίπεδο εφαρμογής

- 2.1 Αρχές δικτυακών εφαρμογών
- 2.2 Web και HTTP
- 2.3 Ηλεκτρονικό ταχυδρομείο
 - SMTP, POP3, IMAP
- 2.4 DNS
- 2.5 Εφαρμογές P2P

DNS: Domain Name System (Σύστημα Ονομασίας Τομέων)

Άνθρωποι: πολλοί τρόποι αναγνώρισης:

- αριθμός κοινωνικής ασφάλισης, όνομα, αρ. διαβατηρίου

Υπολογιστές, δρομολογητές Διαδικτύου:

- διεύθυνση IP (32 bit) - χρησιμοποιείται για διευθυνσιοδότηση datagrams
- «όνομα», π.χ., www.yahoo.com - χρησιμοποιείται από τους ανθρώπους

Ε: Αντιστοιχία ανάμεσα στη διεύθυνση IP και το όνομα και αντίστροφα;

Σύστημα Ονομασίας Τομέων (Domain Name System):

- *Κατανεμημένη βάση δεδομένων* που υλοποιείται σε ιεραρχία πολλών εξυπηρετών ονομάτων (name servers)
- *Πρωτόκολλο επιπέδου εφαρμογής* για να επικοινωνούν υπολογιστές, δρομολογητές, εξυπηρετές ονομάτων για **επίλυση (resolution)** ονομάτων (μετάφραση διεύθυνσης/ονόματος)
 - Σημείωση: βασική λειτουργία του Διαδικτύου που υλοποιείται ως πρωτόκολλο επιπέδου εφαρμογής
 - Πολυπλοκότητα στο "άκρο" του δικτύου

DNS

Υπηρεσίες DNS

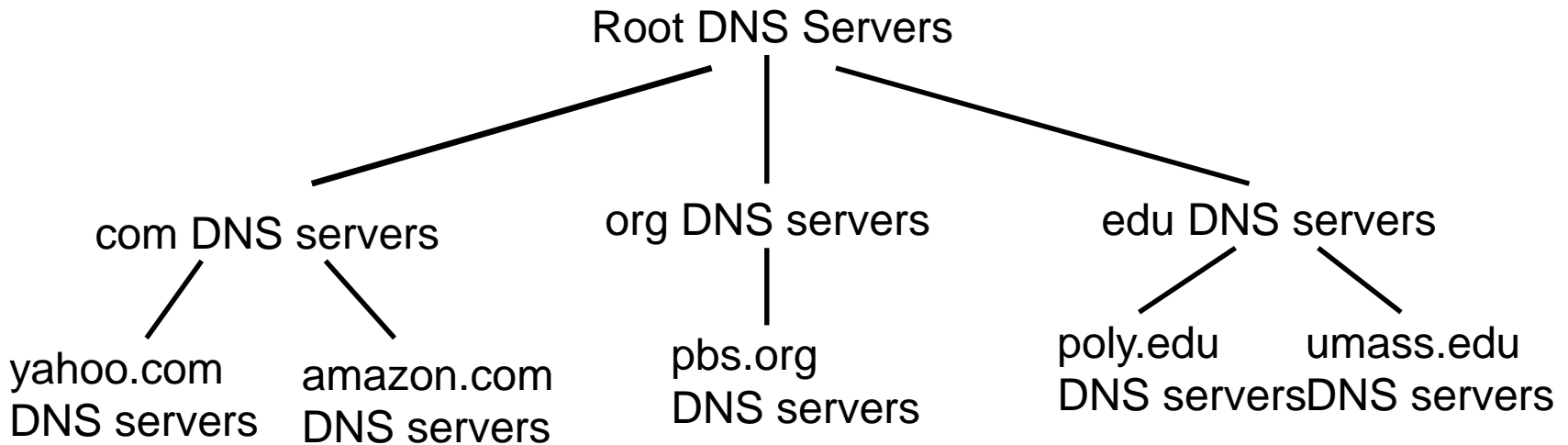
- ❑ Μετάφραση ονόματος (hostname) σε διεύθυνση IP
- ❑ Ψευδώνυμα υπολογιστών (host aliasing) -ευκολομνημόνευτα
 - Ψευδώνυμα → Κανονικά ονόματα
- ❑ Ψευδώνυμα εξυπηρετών ταχυδρομείου (mail server aliasing)
- ❑ Κατανομή φορτίου
 - Εξυπηρετές-αντίγραφα Web: πολλές διευθύνσεις IP αντιστοιχούν σε ένα όνομα

Γιατί όχι κεντριοποιημένο;

- ❑ μοναδικό σημείο αποτυχίας
- ❑ όγκος κίνησης
- ❑ απομακρυσμένη κεντριοποιημένη βάση δεδομένων
- ❑ συντήρηση

A: *δεν κλιμακώνει!*

Κατανεμημένη, Ιεραρχική Βάση Δεδομένων

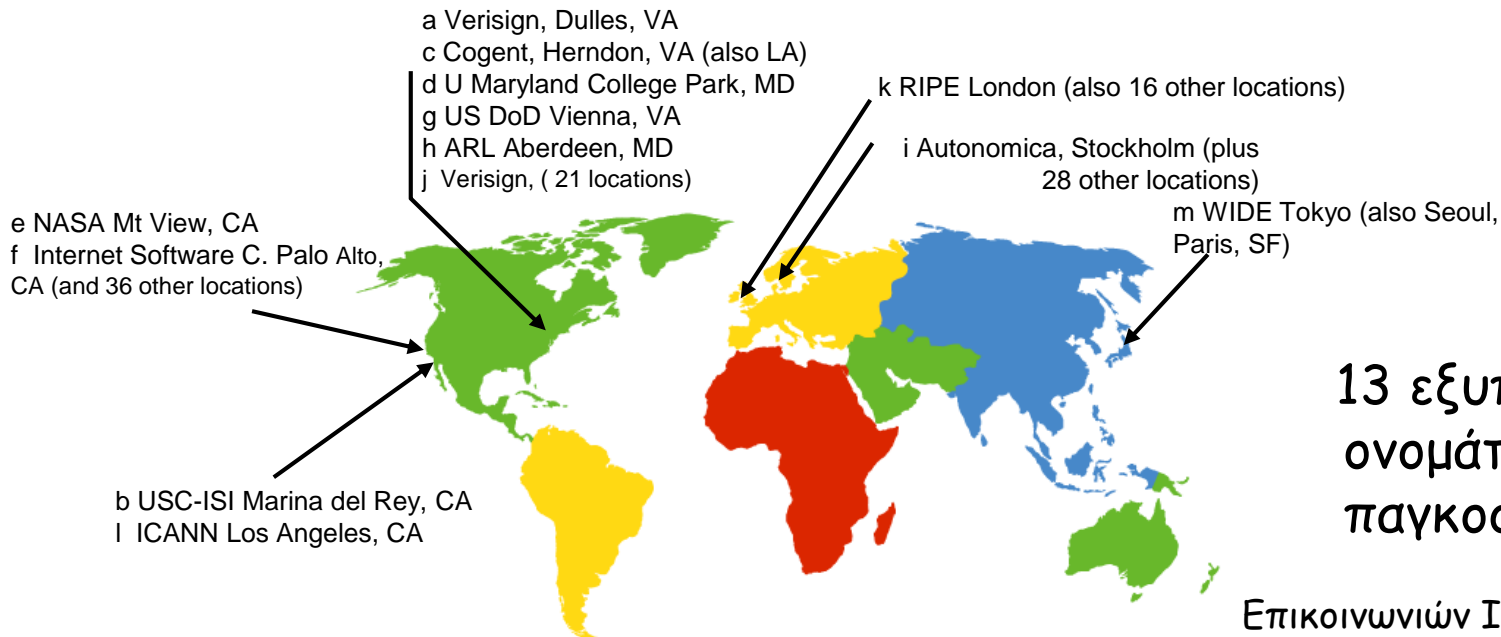


Ο πελάτης θέλει τη διεύθυνση IP για το www.amazon.com, 1^η προσέγγιση:

- ❑ Ο πελάτης ρωτά έναν εξυπηρέτη ρίζας (root server) για να βρει τον εξυπηρέτη DNS com (com DNS server)
- ❑ Ο πελάτης ρωτά τον εξυπηρέτη DNS com για να πάρει τον εξυπηρέτη DNS amazon.com (amazon.com DNS server)
- ❑ Ο πελάτης ρωτά τον εξυπηρέτη DNS amazon.com (amazon.com DNS server) για να πάρει τη διεύθυνση IP του www.amazon.com

DNS: Εξυπηρέτες ονομάτων ρίζας (Root name servers)

- ❑ έρχεται σε επαφή ο τοπικός εξυπηρέτης ονομάτων που δεν μπορεί να μεταφράσει το όνομα
- ❑ εξυπηρέτης ονομάτων ρίζας:
 - έρχεται σε επαφή με τον αυθεντικό (authoritative) εξυπηρέτη ονομάτων, αν η αντιστοιχία του ονόματος δεν είναι γνωστή
 - παίρνει την αντιστοιχία & την επιστρέφει στον τοπικό εξυπηρέτη ονομάτων
 - Ή απλά παραπέμπει σε άλλον εξυπηρέτη



13 εξυπηρέτες
ονομάτων ρίζας
παγκοσμίως

TLD και αυθεντικοί εξυπηρέτες

- Εξυπηρέτες τομέων ανώτατου επιπέδου (Top-level domain (TLD) servers):
 - ❖ Υπεύθυνοι για com, org, net, edu, aero, jobs, museums και όλους τους ανώτατου επιπέδου τομείς χωρών, π.χ. uk, fr, ca, jp
 - ❖ Η Network Solutions διατηρεί εξυπηρέτες για τους com TLD
 - ❖ Η Educause για τους edu TLD
- **Αυθεντικοί εξυπηρέτες DNS (Authoritative DNS servers):**
 - ❖ Οι εξυπηρέτες DNS του οργανισμού που παρέχουν αυθεντικές αντιστοιχίσεις ονομάτων υπολογιστών σε διευθύνσεις IP για τους εξυπηρέτες του οργανισμού (π.χ., Web, mail)
 - ❖ Μπορεί να διατηρείται από τον οργανισμό ή τον πάροχο υπηρεσιών

Τοπικός Εξυπηρέτης Ονομάτων (Local DNS Name Server)

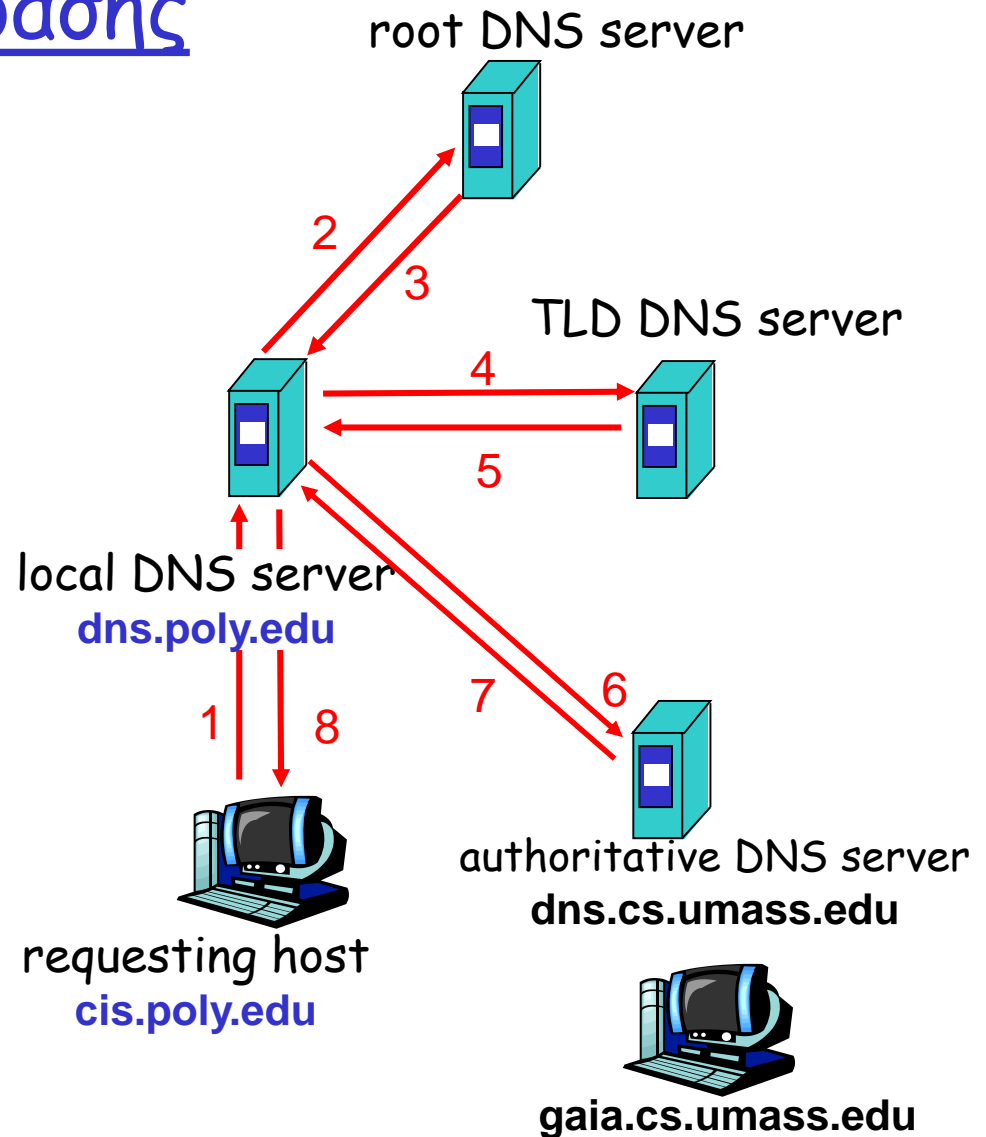
- ❑ δεν ανήκει αυστηρά στην ιεραρχία
- ❑ κάθε ISP (περιφερειακός ISP, εταιρία, πανεπιστήμιο) έχει έναν
 - καλείται επίσης "προεπιλεγμένος εξυπηρέτης ονομάτων"
- ❑ όταν ένας υπολογιστής πραγματοποιεί ένα ερώτημα DNS, το ερώτημα στέλνεται στον τοπικό εξυπηρέτη DNS
 - έχει τοπική cache των πρόσφατων ζευγαριών μετάφρασης όνομα-σε-διεύθυνση (μπορεί να μην είναι ενημερωμένη!)
 - Λειτουργεί ως proxy, προωθεί το ερώτημα στην ιεραρχία

Παράδειγμα μετάφρασης ονόματος DNS

- Ο υπολογιστής στο cis.poly.edu θέλει τη διεύθυνση IP του gaia.cs.umass.edu

Επαναληπτικό ερώτημα:

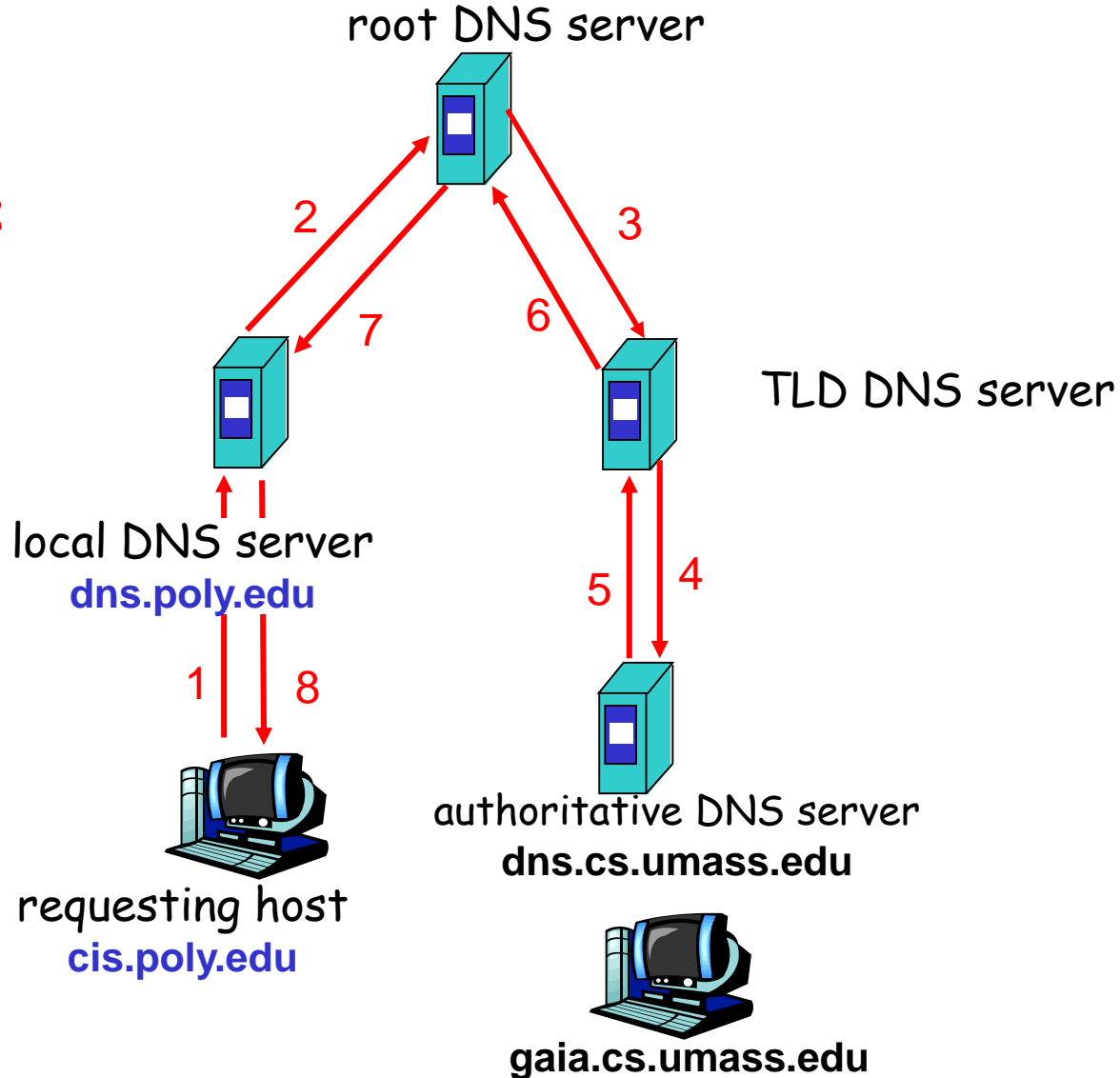
- Ο εξυπηρετής που ρωτήθηκε απαντά με το όνομα του εξυπηρετή που πρέπει να ερωτηθεί
- "Δεν ξέρω το όνομα αλλά ρώτα αυτόν τον εξυπηρετή"



Παράδειγμα μετάφρασης ονόματος DNS

Αναδρομικό ερώτημα:

- ❑ Εναποθέτει το βάρος της μετάφρασης ονόματος στον εξυπηρέτη που ρωτήθηκε
- ❑ Βαρύ φορτίο στα ανώτερα επίπεδα της ιεραρχίας



DNS: προσωρινή αποθήκευση (caching) και ενημέρωση εγγραφών

- Όταν ο (οποιοσδήποτε) εξυπηρέτης ονομάτων μάθει μια αντιστοιχία, την αποθηκεύει προσωρινά
 - Τα περιεχόμενα της προσωρινής μνήμης (cache) λήγουν (εξαλείφονται) μετά από κάποιο χρόνο (TTL)
 - Οι εξυπηρέτες TLD τυπικά αποθηκεύονται προσωρινά σε τοπικούς εξυπηρέτες ονομάτων
 - Έτσι οι εξυπηρέτες ρίζας δε δέχονται συχνά επισκέψεις
- Οι προσωρινά αποθηκευμένες εγγραφές μπορεί **να μην είναι ενημερωμένες** (μετάφραση ονόματος-σε-διεύθυνση βέλτιστης προσπάθειας!)
 - αν ο name host αλλάξει IP διεύθυνση, μπορεί να μην γίνει γνωστό στο Διαδίκτυο μέχρι να λήξουν όλα τα TTLs
- Μηχανισμοί ενημέρωσης/ειδοποίησης υπό σχεδιασμό από IETF
 - RFC 2136

Εγγραφές DNS

DNS: κατακεμημένη βάση δεδ. που αποθηκεύει εγγραφές πόρων
[resource records (RR)]

RR format: (name, value, type, ttl)

□ Type=A

- name είναι το όνομα του υπολογιστή
- value είναι η διεύθυνση IP

□ Type=NS

- name είναι τομέας (domain) (π.χ., foo.com)
- value είναι το όνομα υπολογιστή (hostname) του αυθεντικού εξυπηρέτη ονομάτων για αυτόν τον τομέα

□ Type=CNAME

- name είναι ψευδώνυμο για κάποιο κανονικό (πραγματικό) όνομα
- www.ibm.com είναι στην πραγματικότητα servereast.backup2.ibm.com
- value είναι το κανονικό όνομα

□ Type=MX

- value είναι το όνομα του εξυπηρέτη mail που σχετίζεται με το name

Εισαγωγή εγγραφών στο DNS

- Παράδειγμα: νέα startup εταιρία "Network Utopia"
- Εγγράφει το όνομα networkutopia.com στον *DNS registrar* (π.χ., Network Solutions)
 - Παρέχει ονόματα, διευθύνσεις IP του αυθεντικού εξυπηρέτη ονομάτων (κύριου και δευτερεύοντα)
 - Ο registrar εισάγει δύο RRs στον εξυπηρέτη TLD com:

(networkutopia.com, dns1.networkutopia.com, NS)

(dns1.networkutopia.com, 212.212.212.1, A)

- Δημιουργεί εγγραφή αυθεντικού εξυπηρέτη Type A για το `www.networkutopia.com`, εγγραφή Type MX για το `networkutopia.com`

Επιθέσεις στο DNS

DDoS επιθέσεις

- ❑ Βομβαρδισμός των εξυπηρετών ρίζας με κίνηση
 - Ανεπιτυχής έως τώρα
 - Φιλτράρισμα κίνησης
 - Οι τοπικοί DNS εξυπηρέτες αποθηκεύουν προσωρινά τις IP των TLD εξυπηρετών, επιτρέποντας την παράκαμψη των εξυπηρετών ρίζας
- ❑ Βομβαρδισμός TLD εξυπηρετών
 - Πιθανώς πιο επικίνδυνο

Ανακατεύθυνση επιθέσεων

- ❑ Man-in-middle
 - Παρεμποδίζει ερωτήματα
- ❑ DNS poisoning
 - Εικονικές αποστολές στηρίζονται στον DNS εξυπηρέτη, ο οποίος αποθηκεύει προσωρινά

Εκμετάλλευση DNS για DDoS

- ❑ Στέλνει ερωτήματα με πλαστή διεύθυνση πηγής: στοχεύει IP
- ❑ Απαιτεί ενίσχυση

Κεφάλαιο 2: Επίπεδο εφαρμογής

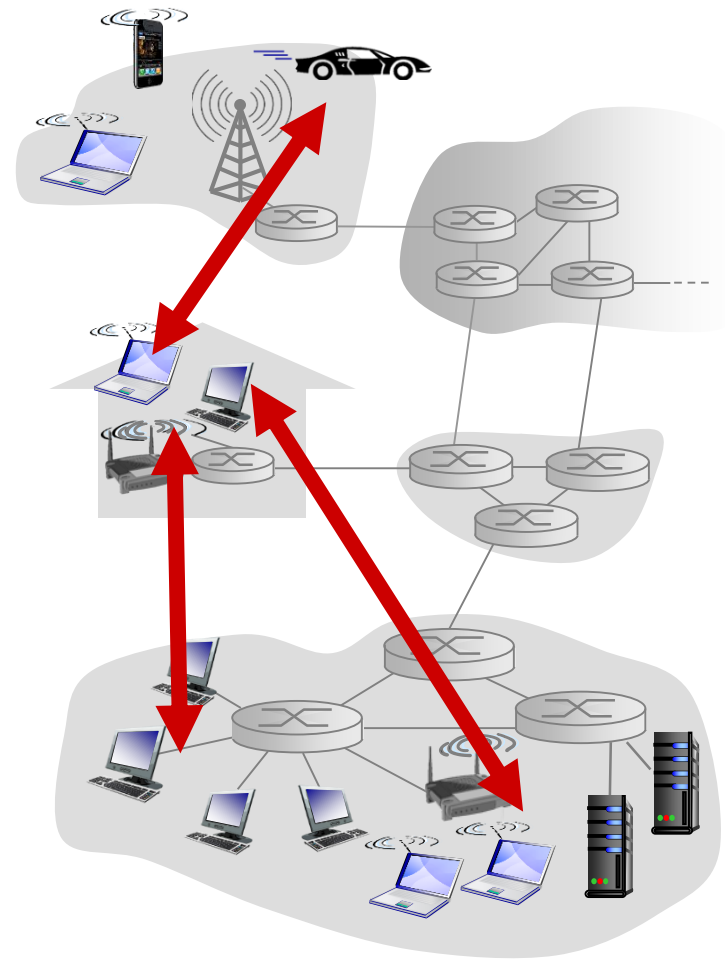
- 2.1 Αρχές δικτυακών εφαρμογών
- 2.2 Web και HTTP
- 2.3 Ηλεκτρονικό ταχυδρομείο
 - SMTP, POP3, IMAP
- 2.4 DNS
- 2.5 Εφαρμογές P2P

Αμιγώς P2P αρχιτεκτονική

- ❑ Δεν υπάρχει διαρκώς ενεργός εξυπηρέτης
- ❑ Τυχαία τερματικά συστήματα επικοινωνούν απευθείας
- ❑ Οι ομότιμοι είναι συνδεδεμένοι διακοπτόμενα και αλλάζουν διευθύνσεις IP

Παραδείγματα:

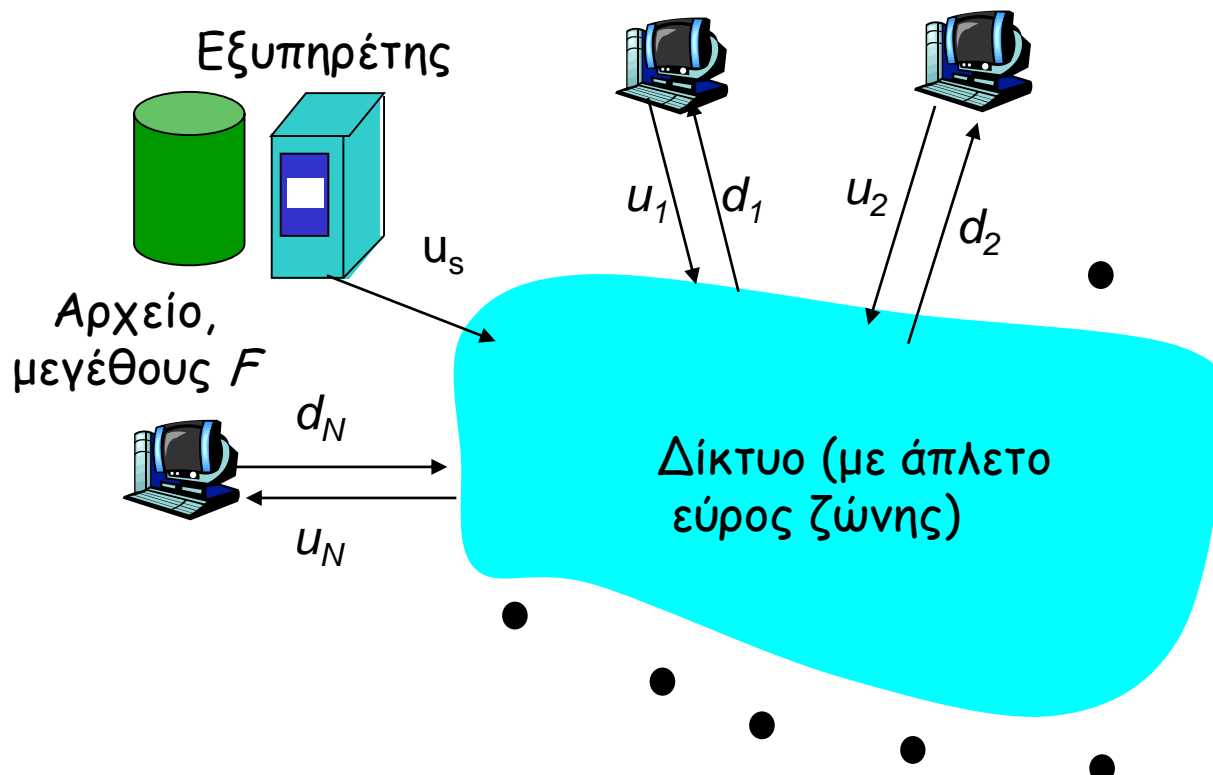
- ❖ Διανομή αρχείου (BitTorrent)
- ❖ Μετάδοση(streaming) (KanKan)
- ❖ VoIP(Skype)



Διανομή αρχείου: Server-Client vs P2P

Ερώτηση: Πόσος χρόνος απαιτείται για τη διανομή αρχείου (μεγέθους F) από έναν εξυπηρέτη σε N ομότιμους;

- η upload/download χωρητικότητα του ομότιμου είναι περιορισμένος πόρος



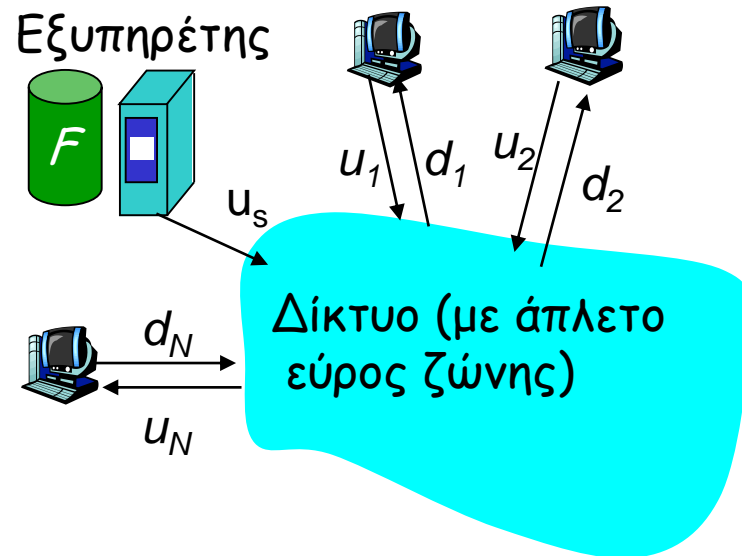
u_s : χωρητικότητα upload εξυπηρέτη

u_i : χωρητικότητα upload ομότιμου i

d_i : χωρητικότητα download ομότιμου i

Χρόνος διανομής αρχείου: εξυπηρέτης-πελάτης (server-client)

- Ο εξυπηρέτης στέλνει (upload) σειριακά N αντίγραφα:
 - Χρόνος αποστολής ενός αντίγραφου: F/u_s
 - Χρόνος αποστολής N αντιγράφων: NF/u_s
- Κάθε πελάτης πρέπει να κατεβάσει (download) το αντίγραφο
 - d_{\min} = ελάχιστος ρυθμός download πελάτη
 - ελάχιστος χρόνος download πελάτη: F/d_{\min}



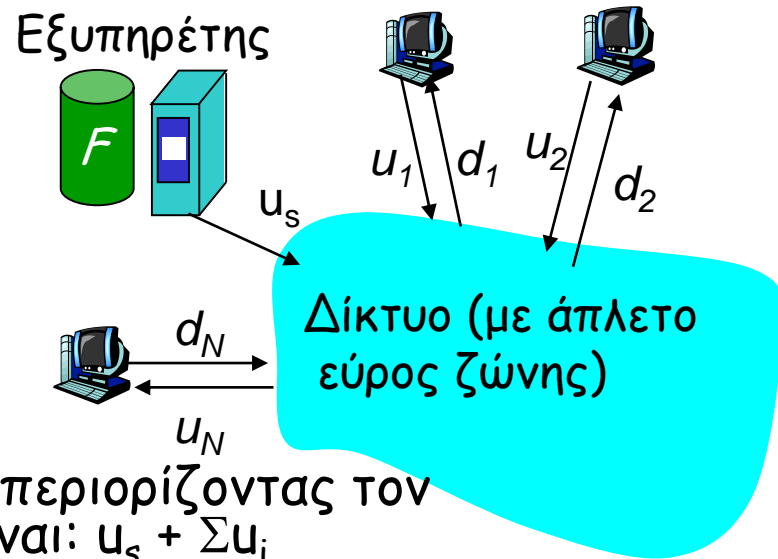
Χρόνος για να διανεμηθεί το F σε N πελάτες χρησιμοποιώντας την προσέγγιση client/server

$$D_{c-s} \geq \max \{ NF/u_s, F/d_{\min} \}$$

Αυξάνει γραμμικά με το N

Χρόνος διανομής αρχείου: P2P

- Ο εξυπηρέτης πρέπει να ανεβάσει τουλάχιστον ένα αντίγραφο: χρόνος F/u_s
- Κάθε πελάτης πρέπει να κατεβάσει το αντίγραφο: ελάχιστος χρόνος κατεβάσματος για τον πελάτη F/d_i
- Οι πελάτες πρέπει να κατεβάσουν NF bits (αθροιστικά)
 - Μέγιστος ρυθμός ανεβάσματος (περιορίζοντας τον μέγιστο ρυθμό κατεβάσματος) είναι: $u_s + \sum u_i$



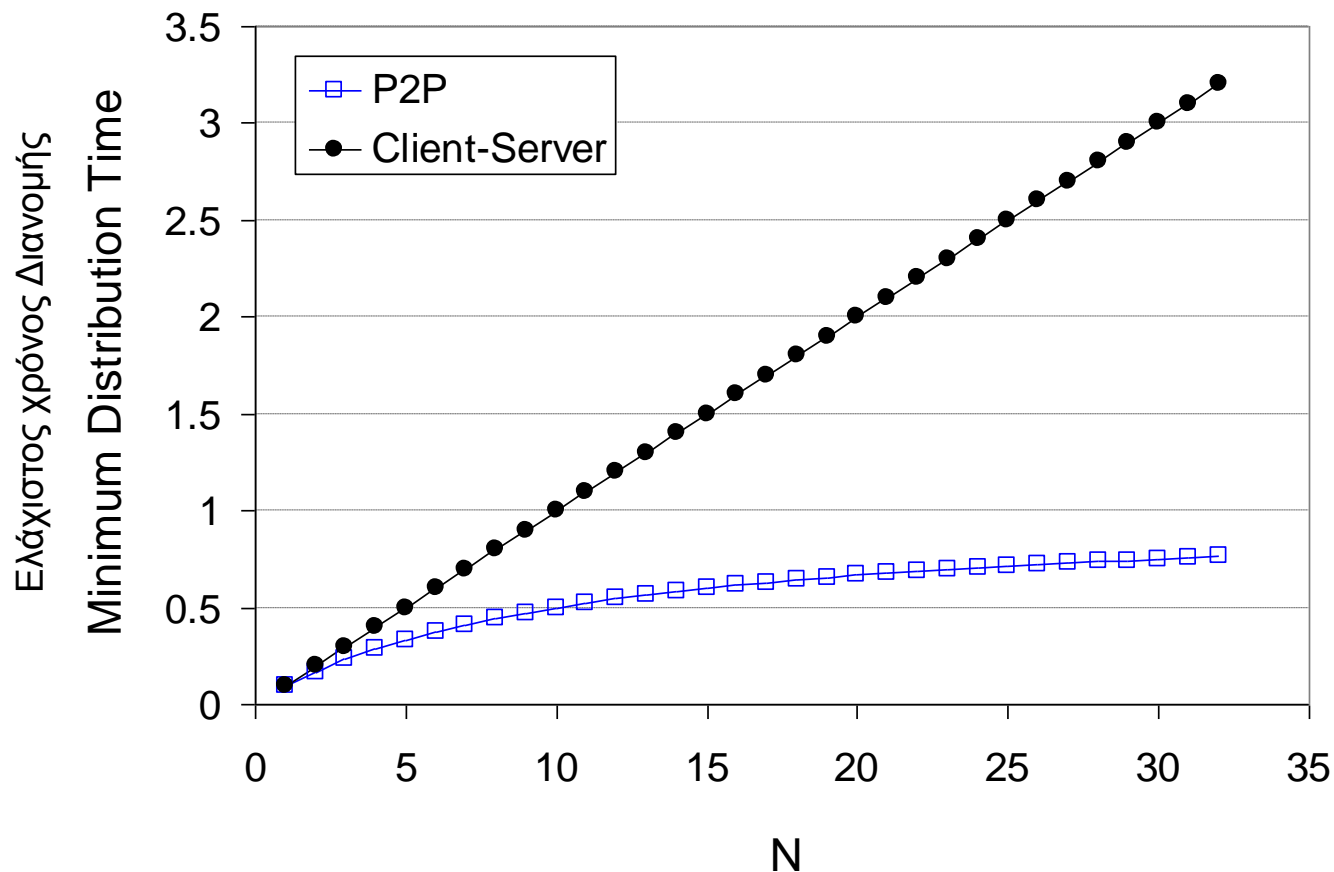
*χρόνος διανομής
του F σε N
πελάτες με την
P2P προσέγγιση*

$$D_{P2P} > \max\{F/u_s, F/d_{min}, NF/(u_s + \sum u_i)\}$$

αυξάνει γραμμικά με το N
ομοίως και αυτό, καθώς κάθε ομότιμος φέρει ικανότητα παροχής υπηρεσιών

Server-client vs. P2P: Παράδειγμα

Ρυθμός upload πελάτη = u , $F/u = 1$ ώρα, $u_s = 10u$, $d_{\min} \geq u_s$



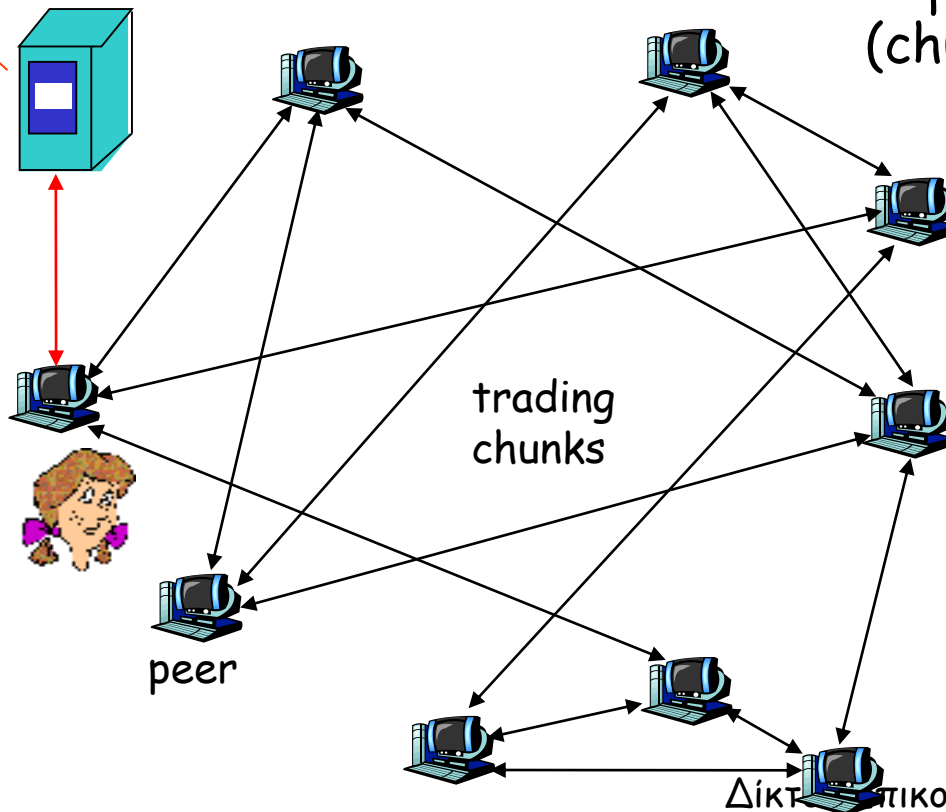
P2P διανομή αρχείου: BitTorrent

- ❑ το αρχείο διαιρείται σε τμήματα (chunks) των 256Kb
- ❑ οι ομότιμοι στο torrent στέλνουν/λαμβάνουν τμήματα ενός αρχείου

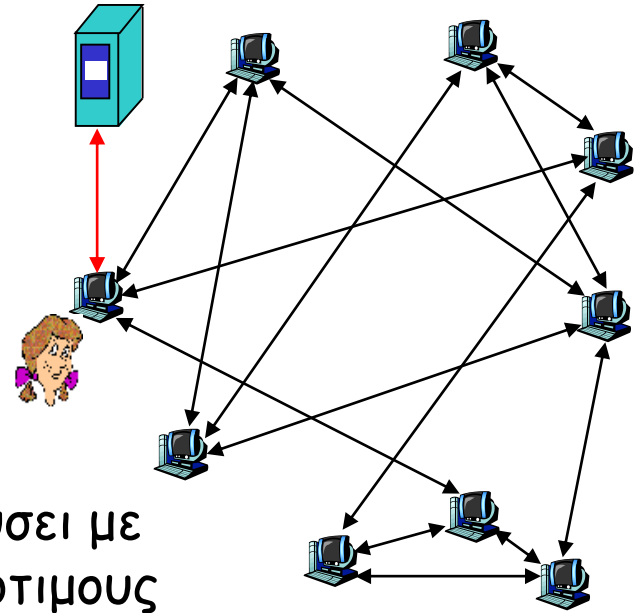
tracker: παρακολουθεί τους ομότιμους που μετέχουν στο torrent

torrent: ομάδα ομότιμων που ανταλλάσσουν τμήματα (chunks) ενός αρχείου

Η Αλίκη μπαίνει...
βρίσκει τη λίστα των ομότιμων από τον tracker
...και ξεκινά την ανταλλαγή τμημάτων του αρχείου με τους ομότιμους στο torrent



BitTorrent



- Ομότιμος που συνδέεται στο torrent:
 - Δεν έχει chunks, αλλά θα τα συσσωρεύσει με την πάροδο του χρόνου από άλλους ομότιμους
 - Εγγράφεται στον tracker για να πάρει τη λίστα των ομότιμων, συνδέεται σε υποσύνολο των ομότιμων («γείτονες» ("neighbors"))
- Ενώ κατεβάζει, ο ομότιμος ανεβάζει chunks σε άλλους ομότιμους
- Ο ομότιμος μπορεί να αλλάξει τους ομότιμους με τους οποίους ανταλλάσσει chunks
- Ομότιμοι ενδέχεται να έρχονται και να φεύγουν
- Όταν ο ομότιμος έχει ολόκληρο το αρχείο, ενδέχεται (εγωϊστικά) να φύγει ή (αλτρουϊστικά) να παραμείνει στο torrent

BitTorrent: αίτηση, αποστολή chunks αρχείων

Ζητώντας Chunks

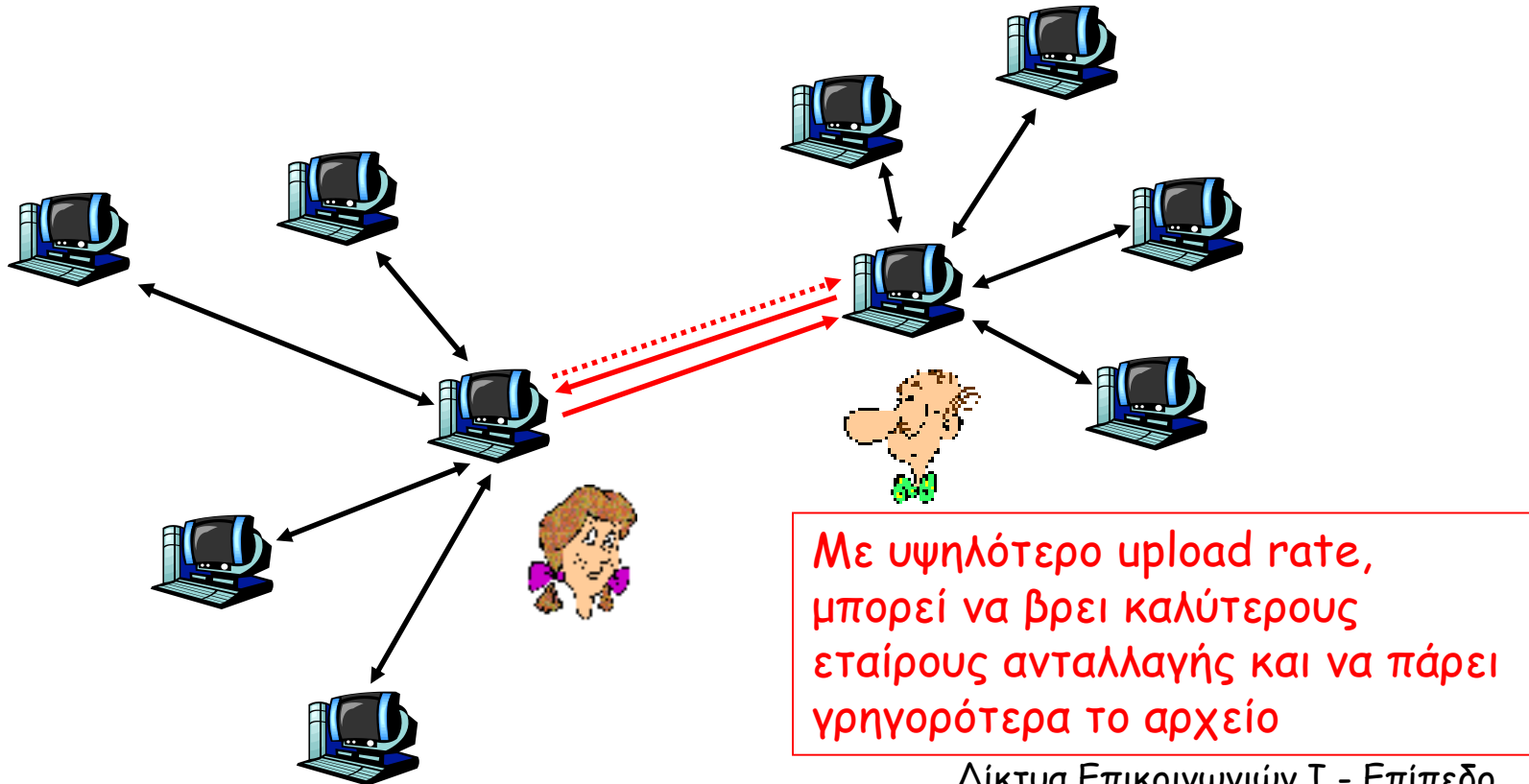
- Κάθε χρονική στιγμή, διαφορετικοί ομότιμοι έχουν διαφορετικό υποσύνολο chunks του αρχείου
- Περιοδικά, ένας ομότιμος (Αλίκη) ρωτά κάθε ομότιμο σχετικά με τη λίστα των chunks που έχει
- Η Αλίκη στέλνει αίτηση για τα chunks που της λείπουν, πρώτα το σπανιότερο

Στέλνοντας Chunks: tit-for-tat

- Η Αλίκη στέλνει chunks στους τέσσερις ομότιμους που στέλνουν στον υψηλότερο ρυθμό
 - Άλλοι ομότιμοι αγνοούνται από την Αλίκη (δε λαμβάνουν chunks από αυτή)
 - Επαναξιολογεί τους top 4 κάθε 10 secs
- κάθε 30 secs: διαλέγει τυχαία άλλον ομότιμο, ξεκινά να στέλνει chunks
 - Ο νέος ομότιμος που επιλέχτηκε ενδέχεται να συμμετάσχει στους top 4
 - Σταματάει να αγνοεί (optimistically unchoke) αυτόν τον ομότιμο

BitTorrent: Tit-for-tat

- (1) Η Αλίκη "optimistically unchokes" τον Βασίλη
- (2) Η Αλίκη γίνεται ένας από τους top-four προμηθευτές του Βασίλη, ο Βασίλης ανταποδίδει
- (3) Ο Βασίλης γίνεται ένας από τους top-four προμηθευτές της Αλίκης



Κατανεμημένος Πίνακας Κατακερματισμού (Distributed Hash Table - DHT)

- ❑ DHT: μια **κατανεμημένη P2P βάση δεδομένων**
- ❑ Η βάση έχει ζεύγη (**key, value**), παραδείγματα:
 - key: αριθμός ασφάλισης, value: ονοματεπώνυμο
 - key: τίτλος ταινίας, value: IP διεύθυνση
- ❑ Κατάνειμε τα ζεύγη (key, value) στα εκατομμύρια των ομότιμων
- ❑ Ένας ομότιμος **ρωτάει** τη DHT με το κλειδί
 - Η DHT επιστρέφει τα values που ταιριάζουν στο κλειδί
- ❑ Οι ομότιμοι μπορούν να εισάγουν ζεύγη (key, value)

Ε: πώς ανατίθενται κλειδιά στους ομότιμους;

- κεντρικό θέμα:
 - ανάθεση ζευγών (key, value) στους ομότιμους
- βασική ιδέα:
 - μετατροπή κάθε κλειδιού σε ακέραιο
 - ανάθεση έναν ακέραιο σε κάθε ομότιμο
 - βάλε το ζεύγος (key, value) στον ομότιμο που είναι **πιο κοντά** στο κλειδί

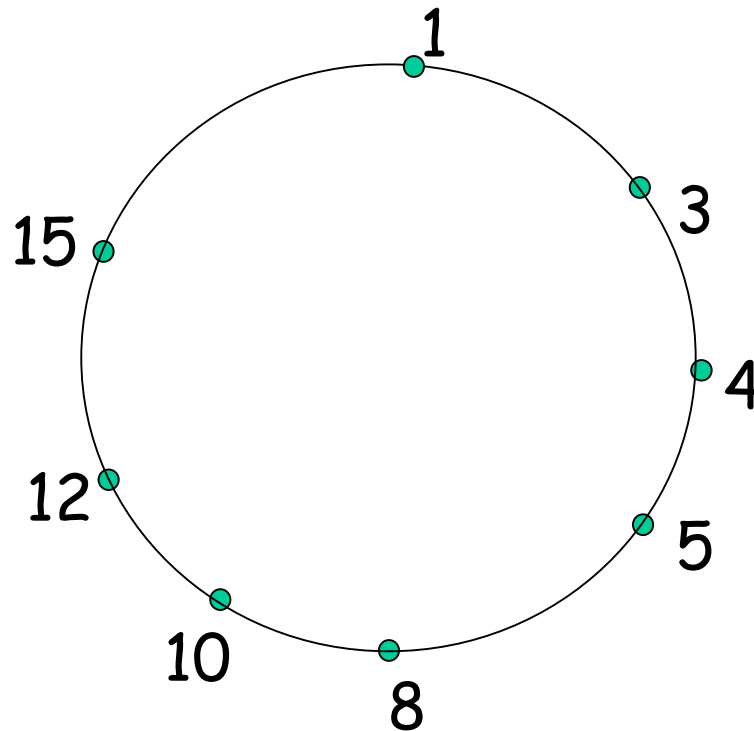
DHT αναγνωριστικά

- ανάθεσε ακέραιο αναγνωριστικό σε κάθε ομότιμο σε εύρος $[0, 2^n - 1]$ για δοσμένο n
 - κάθε αναγνωριστικό αναπαρίσταται από n bits
- κάθε κλειδί πρέπει να είναι ένας ακέραιος στο ίδιο εύρος
- για να πάρεις τον ακέραιο κλειδί, κατακερμάτισε το αυθεντικό κλειδί
 - π.χ., $key = hash(\text{"Led Zeppelin IV"})$
 - γι' αυτό αναφέρεται ως **κατανεμημένος πίνακας "κατακερματισμού"**

Ανάθεση κλειδιών στους ομότιμους

- Κανόνας: ανάθεσε το κλειδί στον ομότιμο με το **κοντινότερο ID**
- Σύμβαση: κοντινότερος είναι ο **άμεσος διάδοχος** του κλειδιού
- π.χ., $n = 4$, ομότιμοι: 1,3,4,5,8,10,12,14
 - $key = 13$, τότε ομότιμος διάδοχος = 14
 - $key = 15$, τότε ομότιμος διάδοχος = 1

Κυκλικό DHT (1)

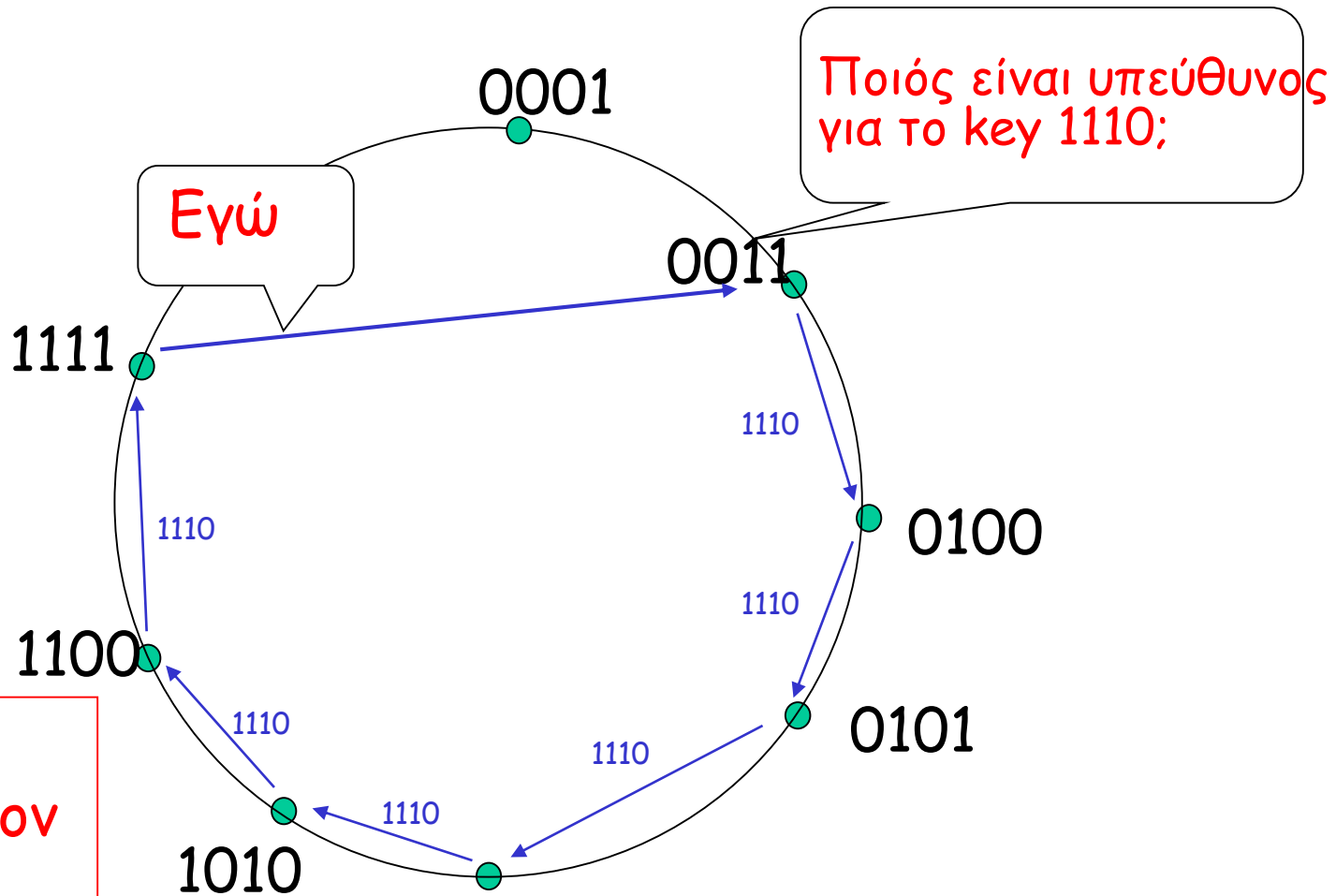


- ❑ κάθε ομότιμος γνωρίζει *μόνο* τον άμεσο διάδοχο και προκάτοχο
- ❑ δίκτυο επικάλυψης ("overlay network")

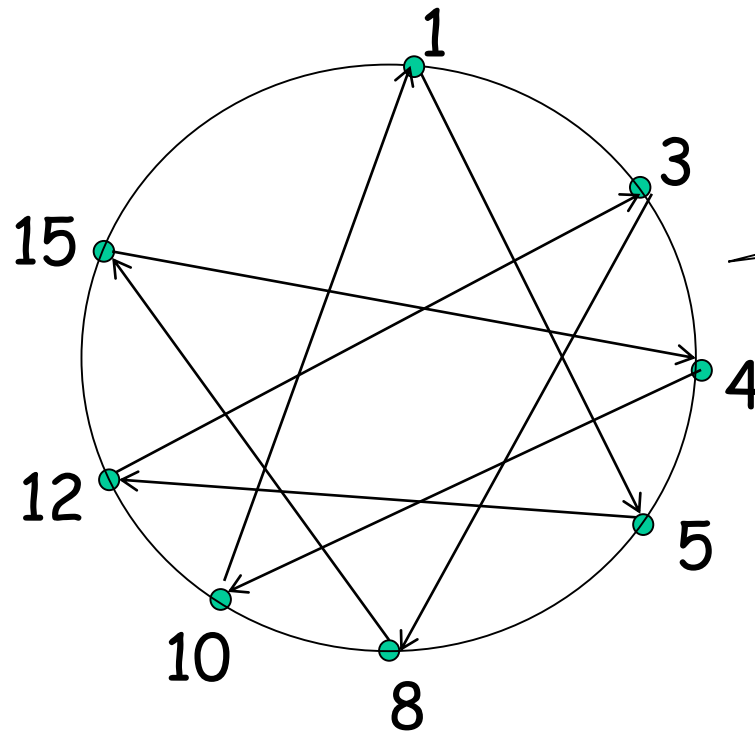
Κυκλικό DHT (2)

$O(N)$
μηνύματα κατά
μέσο όρο για
την επίλυση
του
ερωτήματος,
όταν
υπάρχουν N
ομότιμοι

Όρισε
κοντινότερο τον
κοντινότερο
διάδοχο



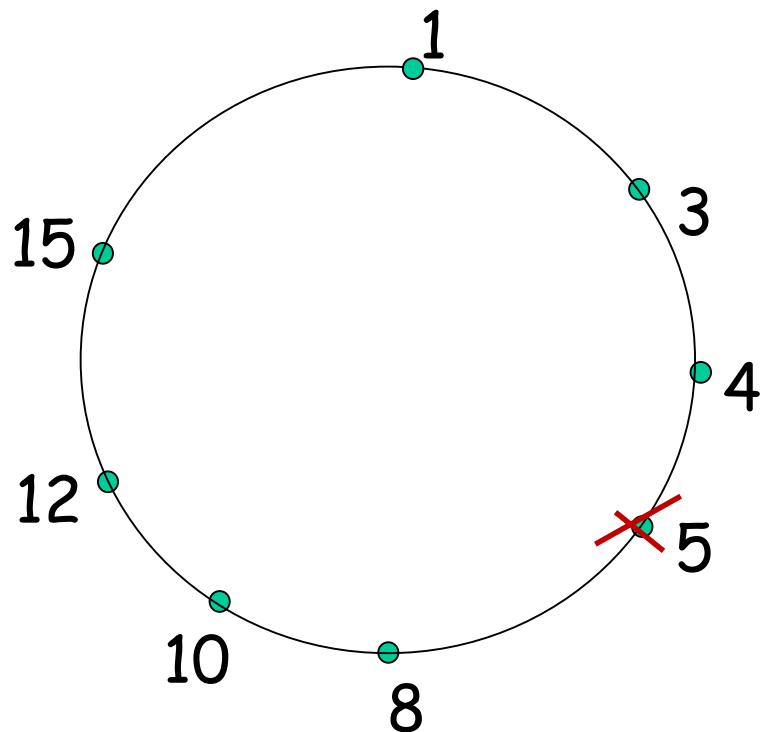
Κυκλικό DHT με συντομεύσεις



Ποιός είναι
υπεύθυνος για το
key 1110;

- ❑ κάθε ομότιμος παρακολουθεί τις IP διευθύνσεις του προκατόχου, διαδόχου και των συντομεύσεων
- ❑ μειώνεται από 6 σε 2 μηνύματα
- ❑ πιθανό να σχεδιαστούν συντομεύσεις ώστε $O(\log N)$ γείτονες, $O(\log N)$ μηνύματα στο ερώτημα

Σύνολο ομότιμων (peer churn)



Διαχείριση του peer churn:

- ❑ ομότιμοι μπορεί να "πηγαινοέρχονται"
- ❑ κάθε ομότιμος στέλνει περιοδικά rings στους 2 διαδόχους του για να ελέγξει αν είναι "ζωντανοί"
- ❑ αν ο άμεσος διάδοχος φύγει, επίλεξε τον επόμενο διάδοχο ως νέο άμεσο διάδοχο

παράδειγμα: ομότιμος 5 φεύγει αιφνίδια

- ❑ Ο ομότιμος 4 ανιχνεύει την αποχώρηση του ομότιμου 5. Κάνει τον 8 άμεσο διάδοχό του. Ρωτάει τον 8 ποιος είναι ο άμεσος διάδοχός του. Κάνει τον άμεσο διάδοχο του 8, δικό του δευτερεύοντα διάδοχο
- ❑ Αν ο ομότιμος 13 θέλει να συμμετάσχει;

Κεφάλαιο 2: Σύνοψη

Η επισκόπησή μας των δικτυακών εφαρμογών τώρα ολοκληρώθηκε!

- Αρχιτεκτονικές εφαρμογής
 - Πελάτης-εξυπηρετής
 - P2P
- Απαιτήσεις υπηρεσίας μιας εφαρμογής:
 - αξιοπιστία, εύρος ζώνης, καθυστέρηση
- Μοντέλο υπηρεσίας μεταφοράς Διαδικτύου
 - συνδεδειστροφής (connection-oriented), αξιόπιστη: TCP
 - μη αξιόπιστη, δεδομενογράμματα (datagrams): UDP
- συγκεκριμένα πρωτόκολλα:
 - HTTP
 - SMTP, POP, IMAP
 - DNS
 - P2P: BitTorrent, DHT

Σύνοψη

Το πιο σημαντικό: μάθαμε για τα πρωτόκολλα!

- Τυπική ανταλλαγή μηνυμάτων αιτήματος/απόκρισης:
 - Ο πελάτης ζητά πληροφορία ή υπηρεσία
 - Ο εξυπηρετής αποκρίνεται με δεδομένα, κωδικό κατάστασης
- Μορφές μηνυμάτων:
 - Κεφαλίδες: πεδία που δίνουν πληροφορία σχετικά με τα δεδομένα
 - Δεδομένα : πληροφορία που επικοινωνείται

Σημαντικά θέματα:

- Μηνύματα ελέγχου - Μηνύματα δεδομένων
 - Ενδοζωνική, εξωζωνική (in-band, out-of-band)
- Κεντριοποιημένο - κατανεμημένο (centralized vs. decentralized)
- ακαταστατικό - με κατάσταση (stateless vs. stateful)
- αξιόπιστη - μη αξιόπιστη μεταφορά μηνύματος
- "πολυπλοκότητα στα άκρα του δικτύου"

Τέλος Ενότητας



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Άδεια Χρήσης

Σημείωμα Αναφοράς

Copyright Εθνικών και Καποδιστριακών Πανεπιστημίων Αθηνών,
Μεράκος Λάζαρος 2015. «Δίκτυα Επικοινωνιών Ι. Ενότητα 2:
Επίπεδο Εφαρμογής». Έκδοση: 1.01. Αθήνα 2015.

Διαθέσιμο από τη δικτυακή διεύθυνση:

<http://opencourses.uoa.gr/courses/DI19>

Χρηματοδότηση

- ❑ Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- ❑ Το έργο «**Ανοικτά Μαθήματα στο Πανεπιστήμιο Αθηνών**» 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.



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ