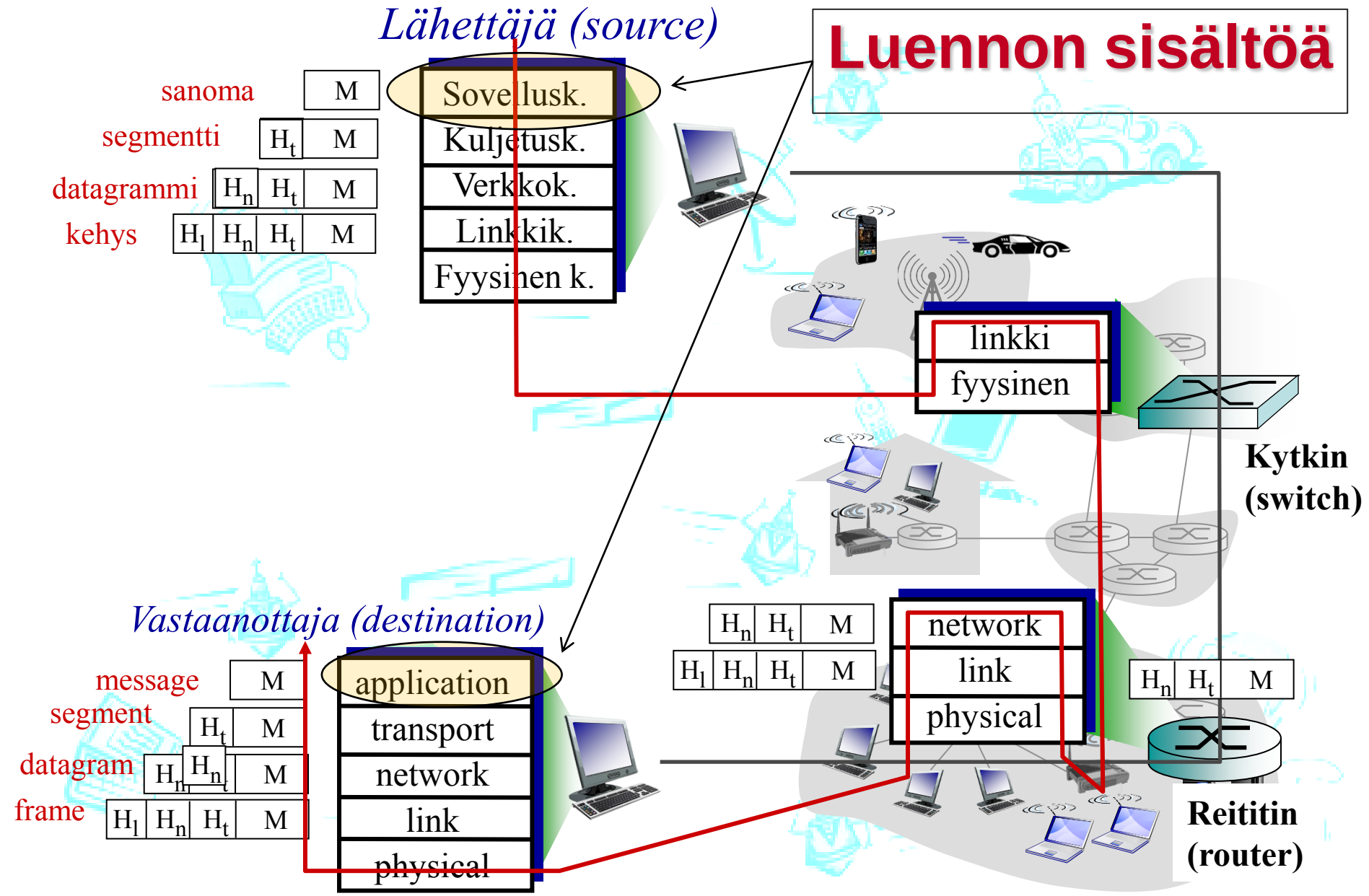
Tietoliikenteen perusteet

Luento 4: Sovelluskerros
nimipalvelu (DNS), tiedostonsiirto,
sähköposti, vertaisverkot (P2P)

Syksy 2014, Tiina Niklander

Luennon sisältöä



Sisältöä

- Verkkosovellusten periaatteet
- World Wide Web ja HTTP
- Tiedostonsiirto ja FTP
- Sähköposti ja SMTP, IMAP, POP3
- Nimipalvelu ja DNS
- Vertaistoimijat (peer-to-peer)
- Pistoke ja sen käyttö

Oppimistavoitteet:

- Osaa selittää asiakaspalvelija–malliin perustuvien verkkosovellusten toimintaperiaatteet
-Tuntee sovellusprotokollien syntaksia ja semantiikkaa
- Osaa selittää www:n ja sähköpostin toimintaideat
- Osaa kuvata nimipalvelun toiminnan

INTERNETIN NIMIPALVELU: DNS

Domain Name System (DNS)

- Hakemistopalvelu ja sovelluskerroksen protokolla
  - Isäntäkoneet ja nimipalvelimet käyttävät
  - Käyttää UDP-kuljetuspalvelua DNS-sanomien kuljettamiseen
- Hajautettu, hierarkinen tietokanta (hakemisto)
  - Toteutettu useiden replikoitujen nimipalvelimien yhteistyönä
  - skaalautuvuus, kuormantasaus, ylläpito, vikasietoisuus, ..
  - Jos oma nimipalvelija ei tunne, se kysyy muilta.

- Nimien muuttaminen IP-osoitteiksi (ja päinvastoin)
  - POSIX: gethostbyname

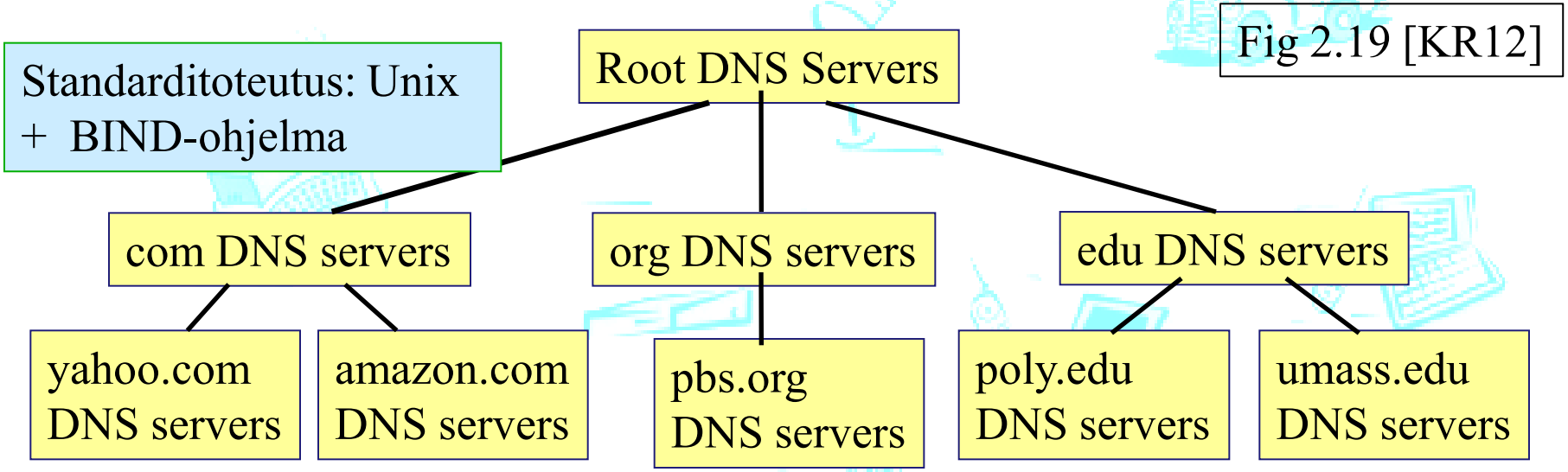
```
gethostbyname (hydra.cs.helsinki.fi)
218.214.4.29
```

  - Kone = hydra =29, verkko= cs.helsinki.fi = 218.214.4.0
- Sallii aliasnimet, palvelijan replikoinnin/toisintamisen
  - Esim. WWW.cs.helsinki.fi ja cs.helsinki.fi ovat aliasnimiä
  - Esim. www-palvelijaan voi liittyä useita IP-osoitteita, rotaatio

DNS historiaa

- Ennen 1983
  - Jokaisessa verkon koneessa HOSTS.TXT tiedosto, jossa verkkotunniste ja sitä vastaava IP-osoite
  - Tiedot haettiin SRI-yrityksen yhdeltä koneelta
  - HOSTS.TXT edelleen käytössä (staattisia asetuksia)
- 1983
  - DNS käyttöönotto
  - Berkeley BIND toteutus
- Myöhempiä laajennuksia: päivitys, replikointi, kansainväliset merkistöt, tietoturva

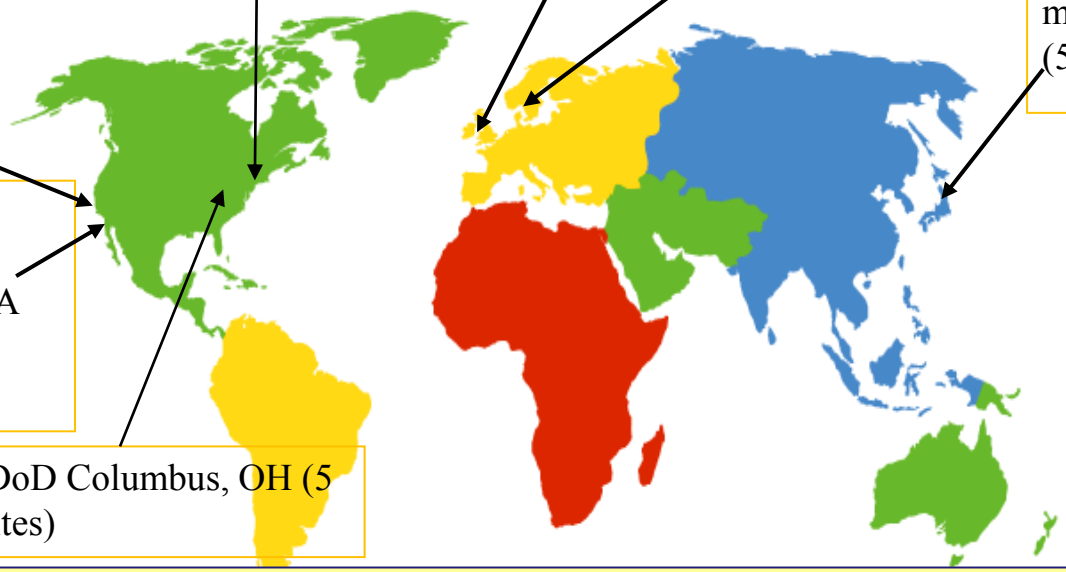
Hajautettu, hierarkinen tietokanta



- 13 juuritason nimipalvelijaa: Replikoituja, kaikilla samat tiedot
  - Internet Assigned Numbers Authority (IANA)
  - Internet Corporation for Assigned Names and Numbers (ICANN)
- Ylätason palvelimet maa- ja yleistunnuksille (n. 265 kpl)
  - .., fi, fr, uk, … edu, net, com, org, .. .中国 .. (Viestintävirasto myöntää fi)
- Autorisoidut aluepalvelimet (domain) (2-taso)
  - Isoilla yliopistoilla ja firmoilla omansa, pienet käyttävät jonkun muun ylläpitämää aluepalvelinta

Juuripalvelimet (2012)

Fig 2.20 [KR12]



c. Cogent, Herndon, VA (5 other sites)
d. U Maryland College Park, MD
h. ARL Aberdeen, MD
j. Verisign, Dulles VA (69 other sites)

k. RIPE London (17 other sites)

i. Netnod, Stockholm (37 other sites)

e. NASA Mt View, CA
f. Internet Software C. Palo Alto, CA (and 48 other sites)

m. WIDE Tokyo (5 other sites)

a. Verisign, Los Angeles CA (5 other sites)
b. USC-ISI Marina del Rey, CA
l. ICANN Los Angeles, CA (41 other sites)

g. US DoD Columbus, OH (5 other sites)

13 root name "servers" worldwide

Juuripalvelimet tietävät, mikä ylätason palvelin on vastuussa maa- ja yleistunnuksesta.
Ylätason palvelimet tuntevat omat aluepalvelimensa.
Aluepalvelimet tuntevat juuripalvelijan.
Koneen oma paikallinen nimipalvelija on koneen asetustiedoissa.

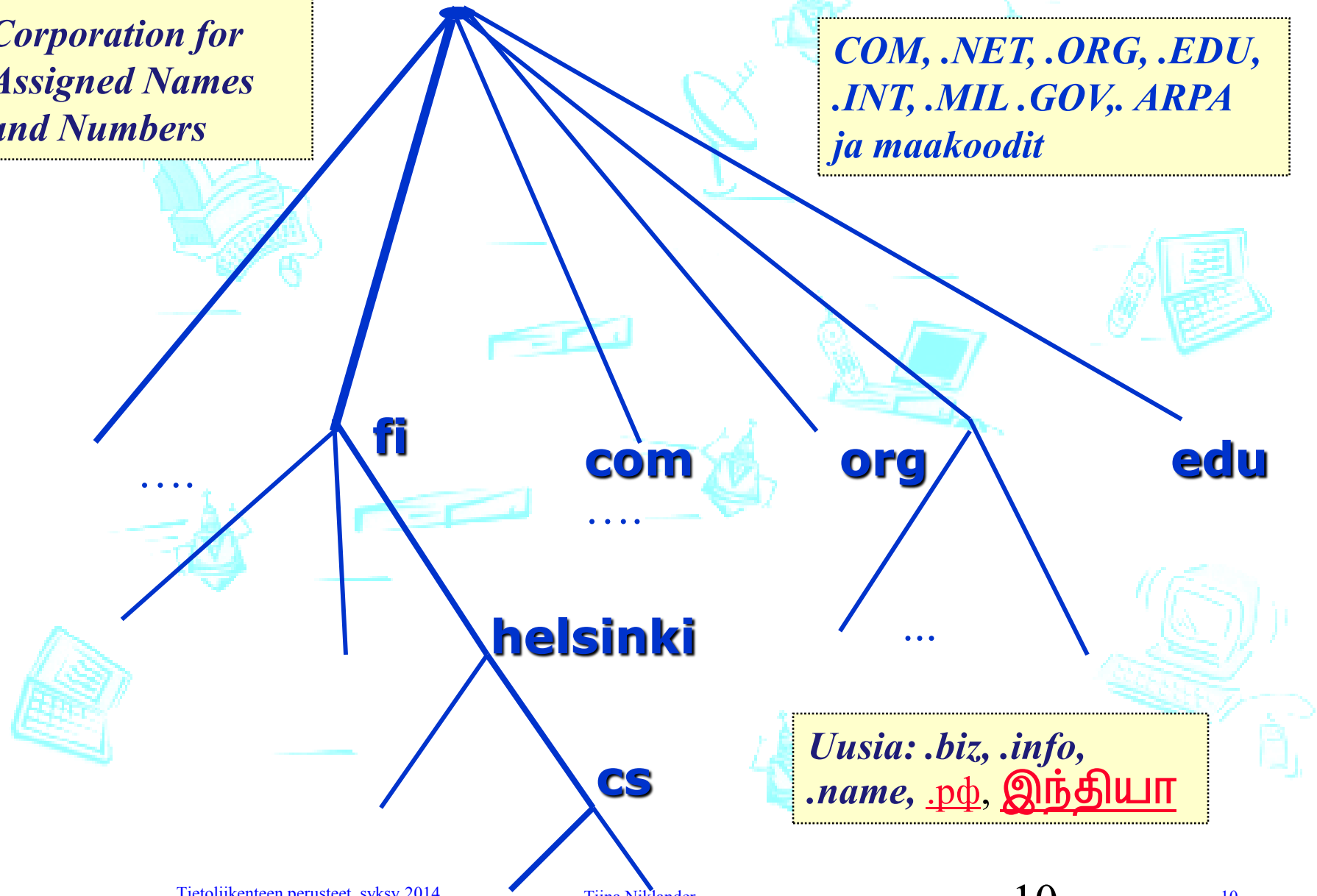
DNS-nimiavaruuden vyöhykejako

- DNS-nimiavaruus jaettu vyöhykkeisiin (zone)
  - kukin vyöhyke kattaa osan nimipuusta
  - vyöhykkeellä on yksi siitä vastaava nimipalvelija (primary) ja yksi tai useita apunimipalvelijoita (secondary)
- Vyöhykejako on hallinnollinen
  - tarpeen mukaan nimipalvelijoita vastaamaan omasta alueestaan

Domain -nimiavaruus

ICANN
The Internet Corporation for Assigned Names and Numbers

COM, .NET, .ORG, .EDU, .INT, .MIL .GOV,. ARPA ja maakoodit



Uusia: .biz, .info, .name, .ϕϕ, இந்தியா

Paikallinen nimipalvelija (local DNS name server)

- Ei ole osa varsinaisten auktorisoitujen nimipalvelijoiden muodostamaa hierarkiaa
- Jokaisella palvelutarjoajalla (verkko-operaattori, yritys, yliopisto) on oma paikallinen, oletus nimipalvelija

- Kun isäntäkoneen sovellus tekee nimipalvelukyselyn, lähettää kone sen paikalliselle nimipalvelijalle,
  - Jolla on omassa välimuistissaan viimeisimpien kyselyjen nimi-osoite muunnoksia (osa voi olla vanhentuneita!)
  - Joka toimii kuin välimuisti (eli proxy), ja lähettää kyselyn edelleen auktorisoidulle nimipalvelijalle

IP-nimen selvittäminen

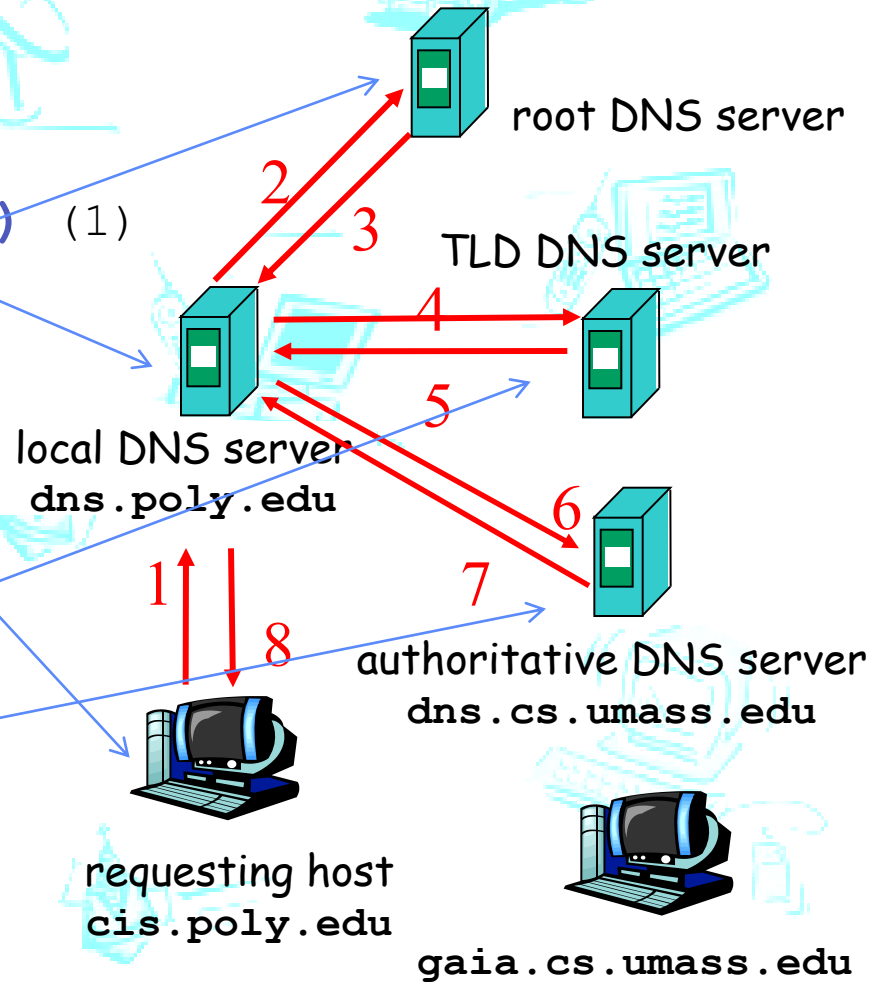
- **Sovellusohjelma** kutsuu kirjastorutiinia parametrina nimi merkkijonona
  - esim Unix:ssa gethostbyname()
- **Kirjastorutiini** lähettää UDP-datasähkeen paikalliselle DNS-palvelimelle
- **Paikallinen nimipalvelin** etsii nimeä vastaavan IP-osoitteen ja palauttaa sen kirjastorutiinille
  - etsinnässä tarvitaan usein monien palvelimien yhteistyötä
  - Iteratiivinen kysely / rekursiivinen kysely
  - Välimuistin käyttö

Iteratiivinen kysely: “kerro keneltä pitää kysyä?”

Mikä on gaia.cs.umass.edu:n IP-numero?

Fig 2.21 [KR12]

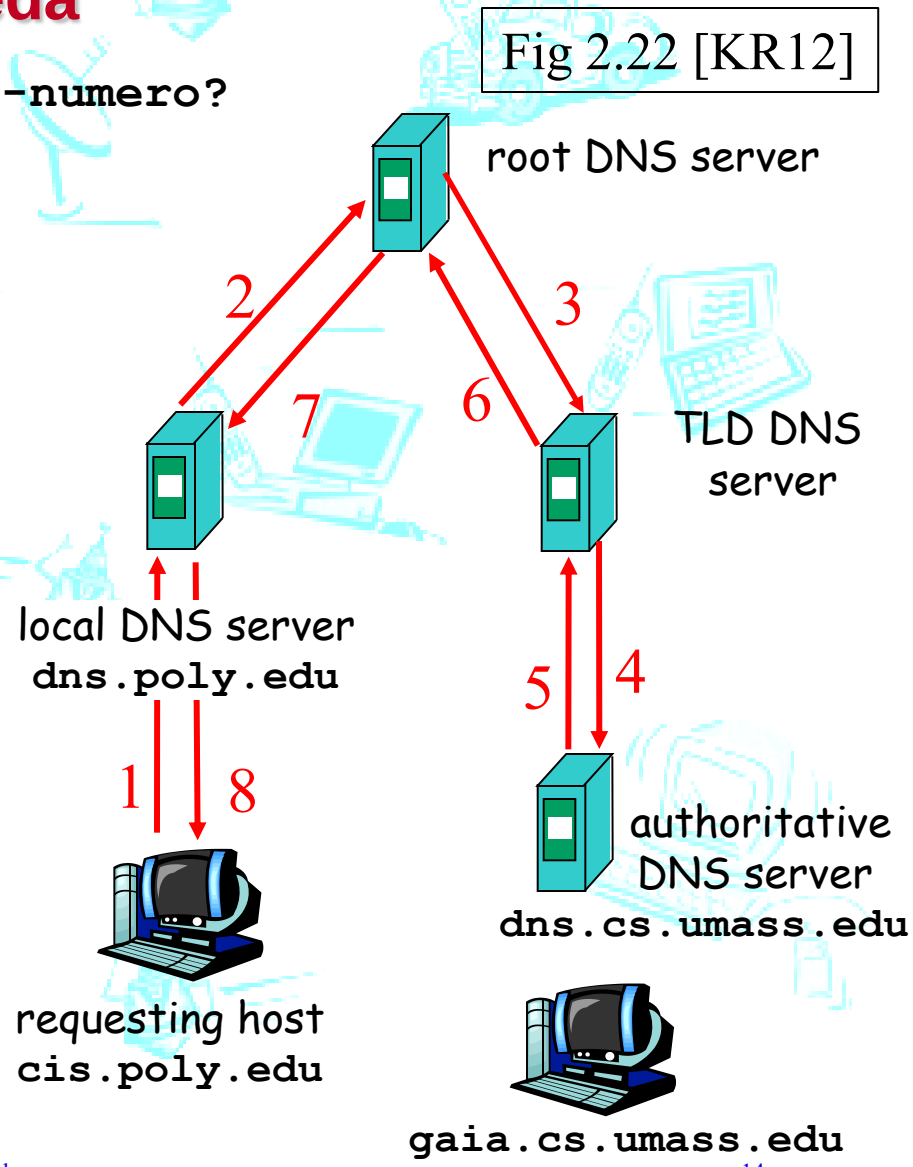
- Isäntäkone
  - Kysy omalta aluepalvelijalta
- Paikallinen nimipalvelija **(poly)** (1)
  - Ratkaise isäntäkoneen puolesta
- Juuripalvelin (3)
  - Kerro, mistä löytyy ylätason palvelin **edu**-tunnuksille
- Ylätason palvelin **(edu)** (4,5)
  - Kerro, mistä löytyy aluepalvelija **umass.edu**-tunnuksille
- Aluepalvelija (6,7)
  - Tuntee **cs**-verkon koneet.
  - Kerro koneen IP-osoite



Rekursiivinen kysely:
"kysy muilta, jos et itse tiedä"

Mikä on gaia.cs.umass.edu:n IP-numero?

- Ratkaise kysyjän puolesta koko ongelma
  - vastaa jos tiedät
  - kysy edelleen jos et tiedä
- Juuripalvelimen suorituskyky, kun paljon kyselyitä?
- Iteratiivinen on tavallisempi malli, mutta kumpikin mahdollinen



DNS-välimuisti (DNS caching)

- Suorituskyvyn parantamiseksi nimipalvelijat varastoivat välimuistiinsa näkemiään DNS-resurssitietueita.
- Ei tarvitse aina hakea uudestaan
  - Kuormittaa vähemmän ylemmän tason nimipalvelimia
  - Nopeuttaa tavallisimpia kyselyjä: löytyy läheltä
- Tiedon oikeellisuus
  - Tietueelle määrätty elinaika (TTL, time to live) kertoo voimassaoloajan (yleensä muutama päivä)
  - Kun umpeutuu, tieto poistetaan.
  - Yleensä muutokset paikallisia:
    - koneen lisäys, koneen poisto, joskus uusi verkko

DNS- resurssitietue
(resource records, RR)

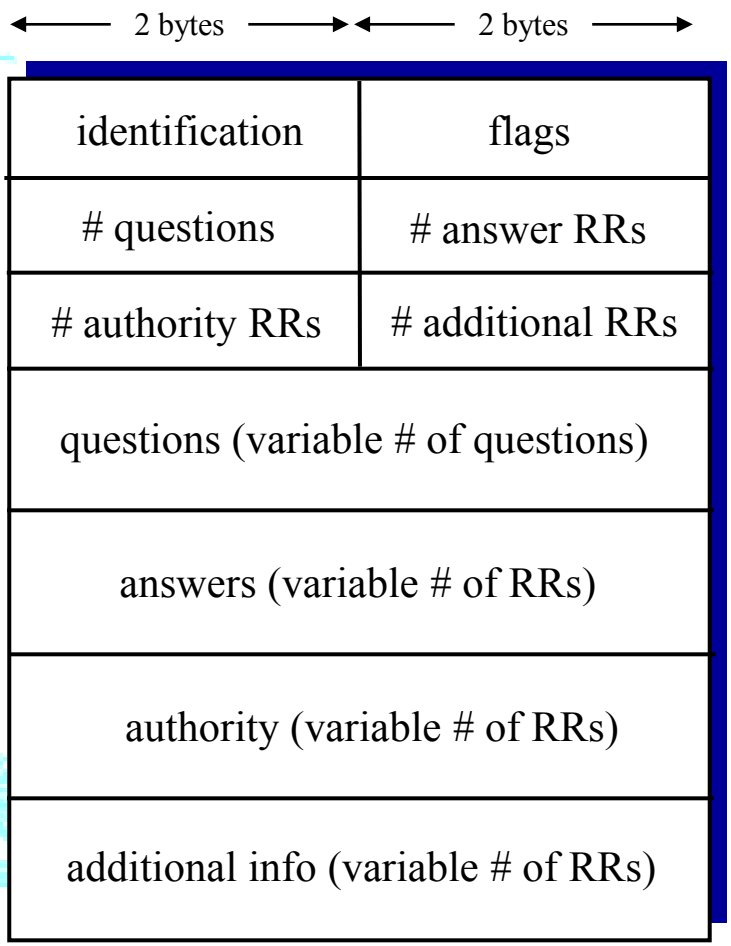
- Resurssitietueen kentät ovat **(nimi, arvo, tyyppi, elinaika)**
- Tyyppi määrää nimen ja arvon merkityksen:
  - **Tyyppi = A** (host address)
    nimi = koneen nimi, arvo = IP-osoite (Ipv4)
    esim: (relay1.bar.foo.com, 145.37.3.126, A, TTL)
  - **Tyyppi = NS** (name server)
    nimi = aluenimi (domain), arvo = autorisoidun palvelimen nimi
    esim: (foo.com, ds.foo.com, NS, TTL)
  - **Tyyppi = CNAME** (canonical name)
    nimi = koneen aliasnimi, arvo= kanoninen, oikea konenimi
    esim: (foo.com, relay1.bar.foo.com, CNAME, TTL)
  - **Tyyppi = MX** (mail exchange)
    nimi = koneen aliasnimi, arvo = postipalvelimen kanoninen nimi
    esim: (foo.com, mail.bar.com, MX,TTL)

Ipv6: AAAA

DNS-sanoma

- Kysely ja vastaus käyttävät samaa formaattia
- Sanoman otsake (header)
- Identification-kenttä
  - Kyselyn tunniste (16-bittinen numero), vastauksessa sama numero => kysely ja vastaus helposti yhdistettävissä toisiinsa.
- Lipukkeet (flags)
  - Pyyntö vai vastaus
  - Käytä rekursiivista kyselyä
  - Rekursiivinen kysely mahdollista
  - Vastaus tulee suoraan autorisoidulta palvelijalta

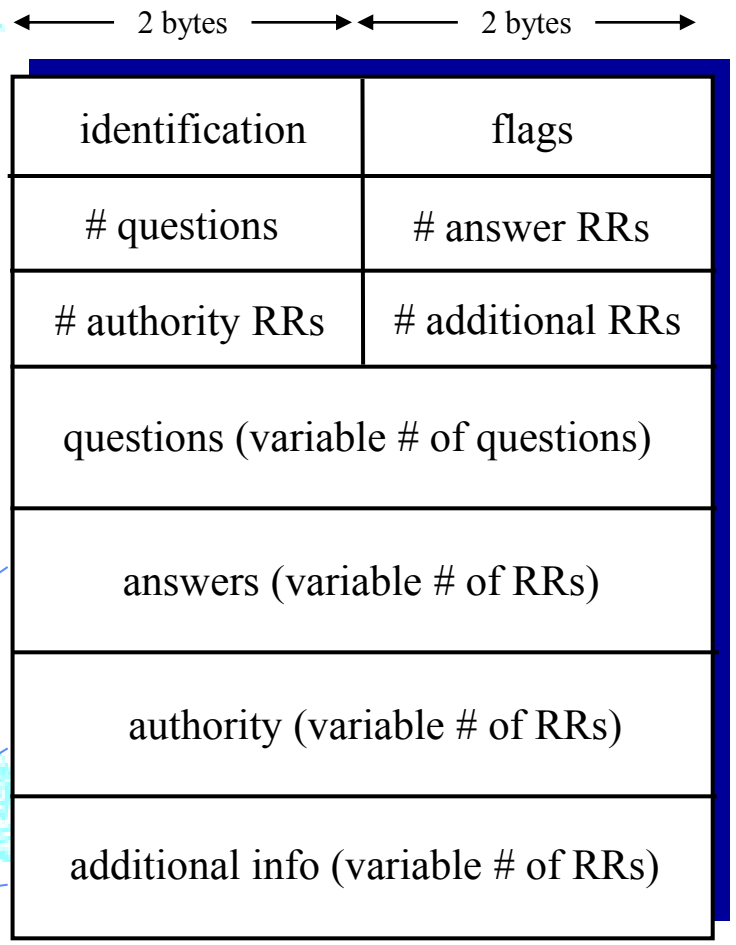
Fig 2.23 [KR12]



DNS-sanoma

- Kyselystä voi generoitua vastaus, jossa on useita resurssitietueita
  - Esim. Palvelijafarmien kuormantasaaminen: vastauksessa on useita IP-osoitteita (rotaatio)

Fig 2.23 [KR12]



Kyselyalueella etsittävän nimi ja tyyppi

Vastausalueella (useita) resurssitietueita, jotka liittyvät kysyttyyn nimeen

Tietueita muihin autorisoituihin palvelijoihin

Ylim. hyödyllisiä resurssitietueita

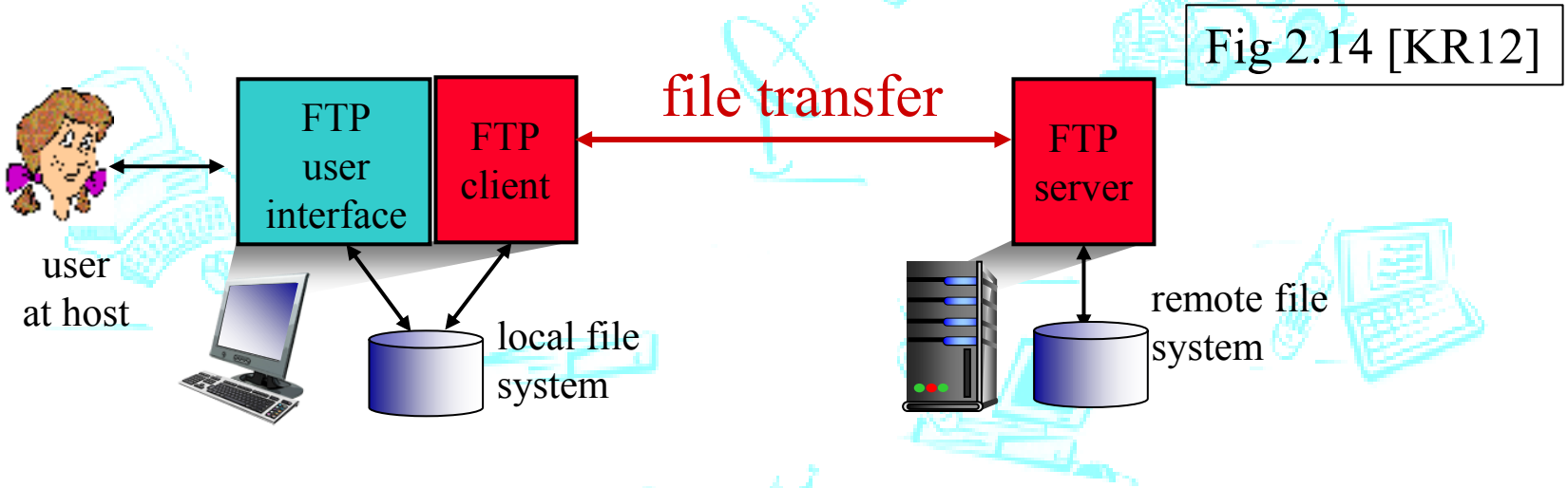
Hyökkäyksiä nimipalvelua vastaan tai käyttäen

- DDoS-hyökkäys (Distributed Denial of Service) osoitekyselyjä juuripalvelimille
  - Ei onnistu enää. (v. 2002 yritys ICMP-datagrammeilla)
  - Paikalliset DNS-välimuistit tallettavat ylätason palvelimien osoitteet, eikä kyselyjä lähetetä enää juuripalvelimelle
- Hyökkäys ylätason (TLD) nimipalvelijoita vastaan voisi olla vaarallisempi

- Uudelleenohjaus (Redirect)
  - Man-in-middle: ohjaa kyselyt toisaalle
  - DNS myrkytys (poisoning): syötä väärää tietoa DNS-palvelimen välimuistiin
- DNS:n käyttö DDoS-hyökkäyksessä tiettyä palvelinta vastaan
  - Houkuttele DNS-palvelimet kysymään tieto ko. palvelimelta
  - Järkevä vain jos oma viesti DNS-palvelimelle pienempi kuin palvelimen viesti kohteelle (ns. vahvistaminen, amplification)

TIEDOSTONSIIRTO, FTP

FTP file transfer protocol (RFC 959)



- Tiedostojen kopioiminen koneelta koneelle
  - Asiakas voi selata etäkoneen hakemistoissa FTP–sanomilla, voi noutaa tai tallettaa haluamansa tiedoston (download/upload)
- Aktiivimoodi: Asiakas vastaanottaa palvelimen pyynnön
  - Ei toimi palomuurien ja NAT-laitteiden kanssa
- Passiivimoodi (PASV-komento): Asiakas ottaa yhteyttä palvelimelta juuri pyytämäänsä IP-osoitteeseen ja porttiin, palvelin lähettää tai vastaanottaa tiedoston

FTP: eri yhteydet hallinnalle ja tiedostojen siirrolle

FTP-palvelin kuuntelee porttia 21

yhteys kontrollitiedon välitystä varten

Asiakas kuuntelee porttia 20

palvelija avaa tiedoston siirtoa varten

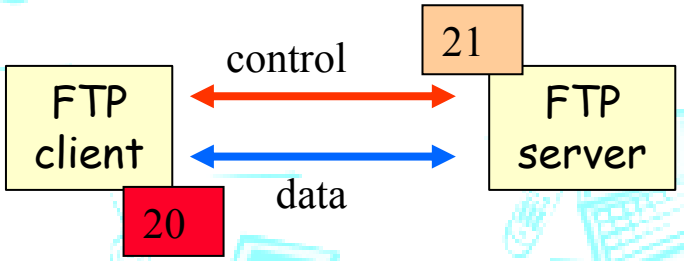
FTP-palvelin **ylläpitää tilatietoa**

mm. työhakemiston polku, autentikointi

FTP asiakas ottaa yhteyttä palvelimen porttiin 21

- käyttöoikeuksien tarkistus
- hakemistojen selailu ja kaikki muutkin asiakkaan pyynnöt tällä yhteydellä

Katso fig 2.15 [KR12]



2 TCP-yhteyttä

Aktiivimoodissa:

Kun palvelin saa tiedostonlataus-pyynnön, se avaa toisen rinnakkaisen yhteyden asiakkaaseen (portti 20) tiedoston siirtoa varten.

Siirron jälkeen palvelin sulkee yhteyden. Uudelle tiedostolle avataan taas uusi yhteys.

FTP-pyyntöjä ja -vastauksia

Kaikki sanomat ASCII-muodossa, binääritila tiedostoille

Asiakkaan pyyntöjä:

USER *username*

PASS *password*

LIST tiedostoluettelo

RETR filename

nouda tiedosto

STOR filename stores

talleta tiedosto

Palvelimen vastauksia:

331 Username OK, password required

125 data connection already open; transfer starting

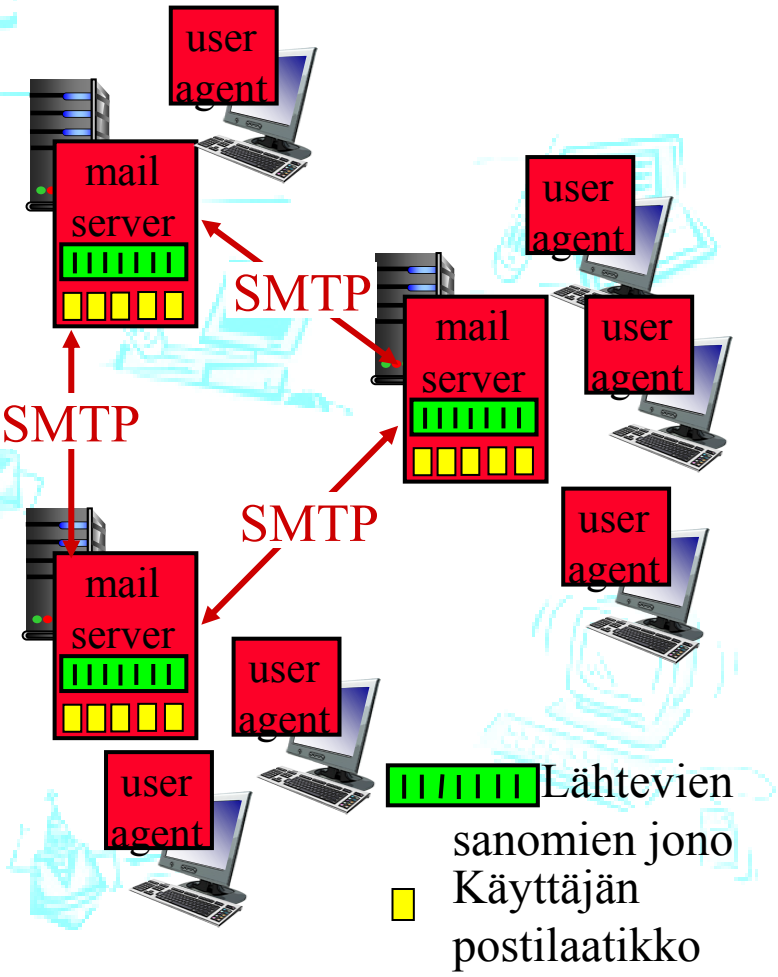
425 Can't open data connection

452 Error writing file

SÄHKÖPOSTI: SMTP, IMAP, POP3

Sähköpostin komponentit

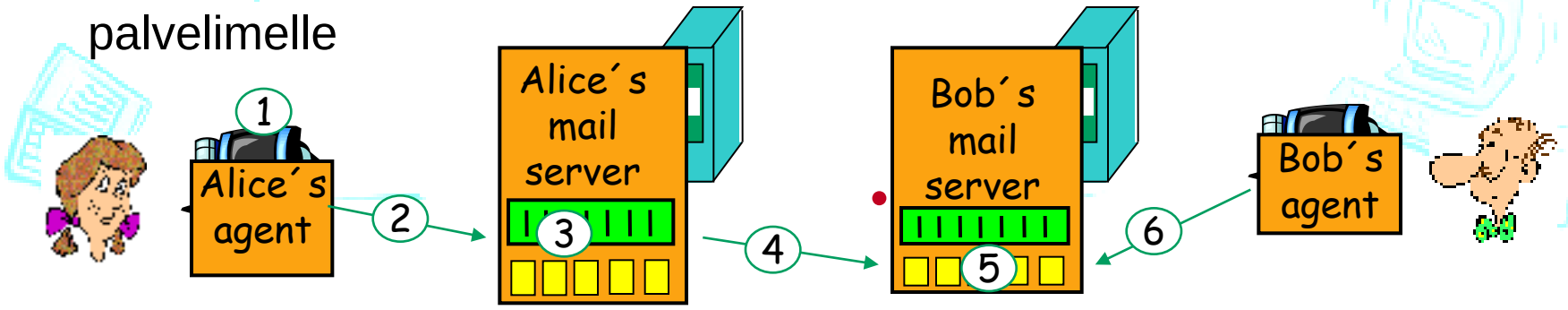
- Postiohjelma (user agent)
  - Postin lukeminen ja lähettäminen
  - Eudora, Outlook, elm, pine, Messenger, Pegasus, Kmail, …
  - Posti talletettuna omalle postipalvelimelle
- Postipalvelin (mail server)
  - Kullakin käyttäjällä on oma saapuvien postien laatikko
  - Yhteinen lähtevien postien laatikko
- Postiprotokolla SMTP
  - Protokolla, jolla postipalvelin välittää postin suoraan vastaanottajan postipalvelimelle



Esimerkki: Alice Bobille

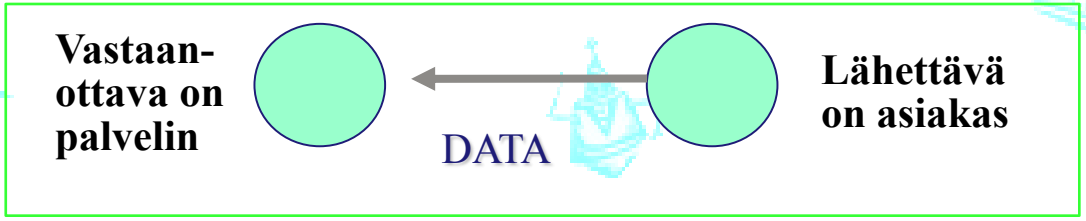
Fig 2.17 [KR12]

1. Alice kirjoittaa viestin posti-ohjelmalla: ”to:” **bob@someschool.edu**
2. Alicen postiohjelma lähettää viestin omalle postipalvelimelle (SMTP-protokollalla)
3. Alicen postipalvelin avaa TCP-yhteyden Bobin posti-palvelimelle
4. Alicen postipalvelin siirtää viestin SMTP-protokollalla Bobin postipalvelimelle käyttäen TCP-yhteyttä
5. Bobin postipalvelin laittaa viestin Bobin postilaatikkoon
6. Bob lukee viestin omalla postiohjelmalla (IMAP-prot.)



SMTP (Simple Mail Transfer Protocol) (RFC 821)

- Postipalvelimet kuuntelevat porttia 25
- Asiakas muodostaa säilyvän TCP-yhteyden palvelimeen
  - luotettava
  - yksi yhteys: lähetetään kaikki samalle palvelimelle menevät viestit
- Lähetyksessä: Kättely, Viestien välitys, Lopetus
- Pyyntö-vastaus-protokolla
  - Pyyntö: ASCII-tekstiä
  - Vastaus: status-koodi ja fraasi tekstinä
- Push-protokolla: työntää tietoa vastapäähän
- vrt. HTTP on ns. pull-protokolla



SMTP esimerkki

C – asiakas, lähettäjä
S – palvelija, vastaanottaja

S: 220 helsinki.fi
C: HELO princeton.edu
S: 250 Hello princeton.edu

SMTP:n kättely

C: MAIL FROM: <Bob@princeton.edu>
S: 250 <Bob@princeton.edu> OK
C: RCPT TO: <pekka.puupaa@cs.helsinki.fi>
S: 250 <pekka.puupaa@cs.helsinki.fi> OK
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: dataa … dataa
C: dataa … dataa
C: .
S: 250 Message accepted for delivery

Viesti(t)

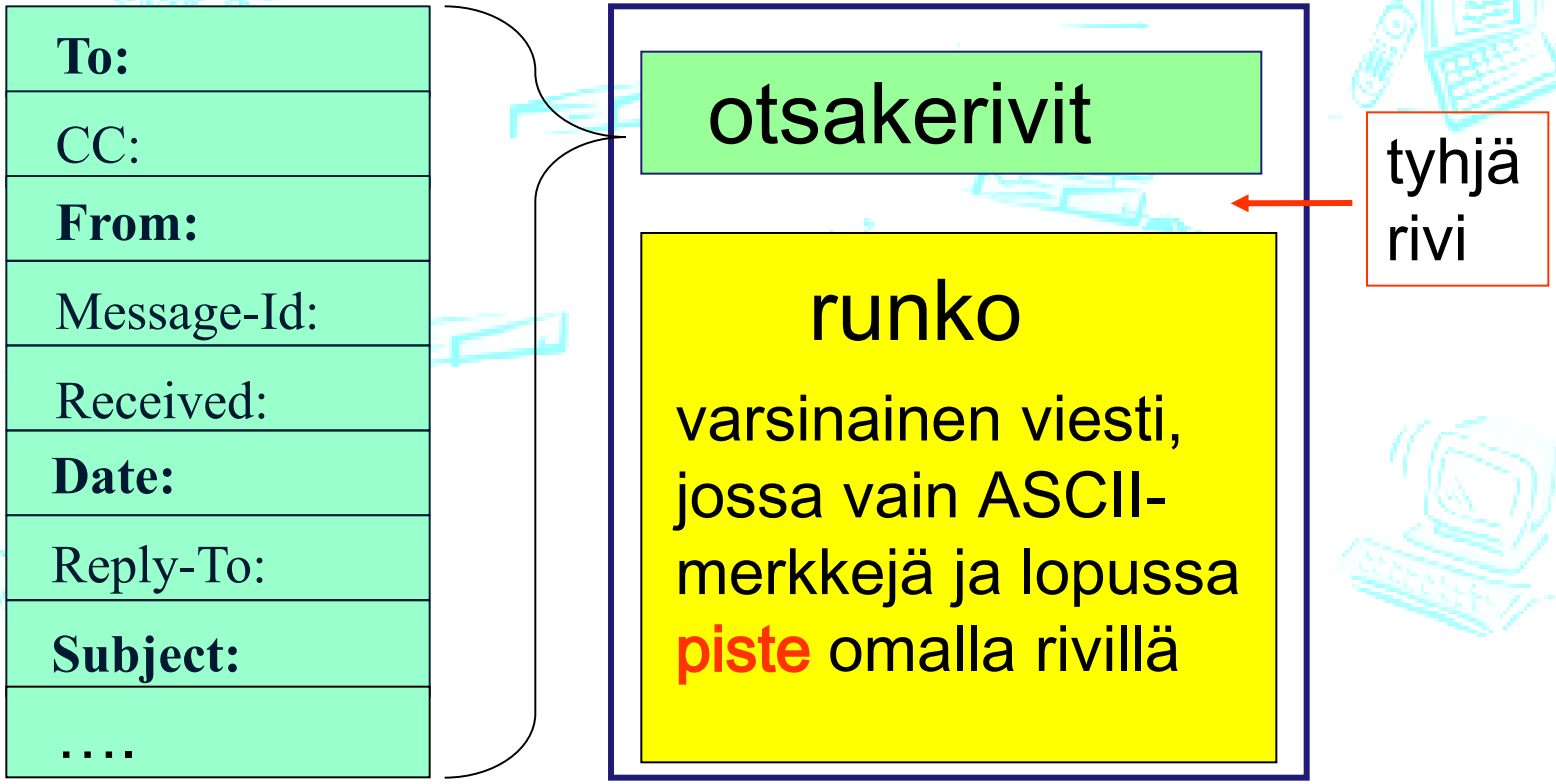
C: QUIT
S: 221 princeton.edu closing connection

SMTP:n lopetus

Sähköpostiviestin rakenne (on SMTP:lle vain dataa)

Eri asia kuin SMTP: eri standardit (RFC 822)

Esim.



SMTP:n rajoitteita

- Kaikki esitettävä 7-bittisenä ASCII:na

  = IRA, International Reference Alphabet

  Myös binääridata, esim. kuvat ja ääni

- **Yksittäinen viesti loppuu omalla rivillä olevaan pisteeseen**

  eli lopussa ASCII-merkit: CRLF**.**CRLF

  Vanha protokolla!

CR = carriage return

LF = line feed

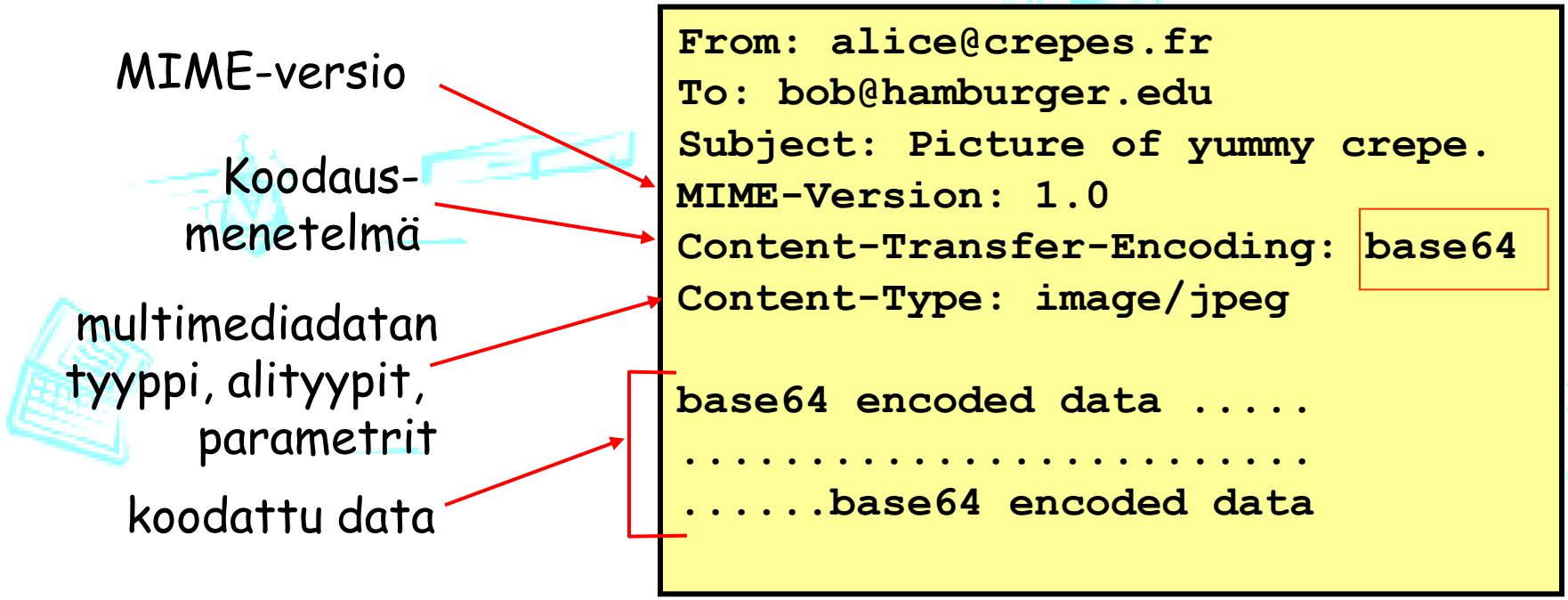
- Binääridata on koodattava s.e. siinä ei esiinny **CRLF.CRLF**

  MIME-laajennus

Multipurpose Internet Mail Extensions

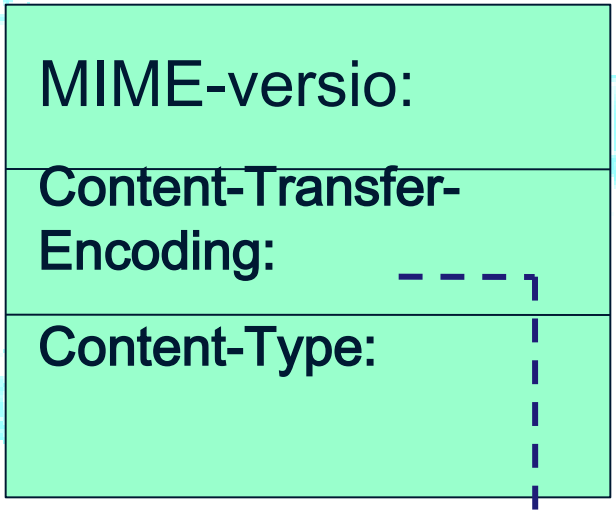
MIME (Multipurpose Internet Mail Extension) (RFC 2045, 2056)

- Kaikki on koodattava 7-bittiseksi ASCII-koodiksi
- Lisää kenttiä otsakkeeseen: vastaanottajan postiohjelma osaa käynnistää oikean sovelluksen viestin näyttämiseksi.



MIME

- MIME-sisältötyyppejä
  - text/plain; charset=us-ascii
  - text/html
  - image/gif, image/jpeg,
  - video/mpeg
  - application/postscript
  - application/msword
  - application/octetstream
  - multipart/mixed



- Base-64-koodaus
  - Sanoman 24 bitin ryhmät on jaettu 6 bitin osiksi,
  - jotka kukin on koodattu ASCII-merkiksi, 64 eri vaihtoehtoa (3:sta tavusta tulee 4 tavua)

Moniosainen MIME-viesti

- …
- Content-Type: multipart/mixed; Boundary=StartOfNextPart
- - - StartOfNextPart
- Hei Allu,
- sinulle kaunis kuva kissastani Villestä.
- - - StartOfNextPart
- Content-Transfer-Encoding: base64
- Content-Type: image/jpeg
- base64 encoded data ……...
- ……..
- ……base64 encoded data
- - - StartOfNextPart
- Haluatko muita kuvia!
- .

Nykyisin yleensä linkki www-sivulle, josta kuvan voi hakea!

Postinnoutoprotokollat

(mail access protocols)

Koska SMTP on 'PUSH'-protokolla, sitä ei voi käyttää sanomia haettaessa ('PULL').

Posti omalta postipalvelimelta postiohjelmaan

POP3: Post Office Protocol versio 3

Viestien lataamiseen omalle koneelle, ei postikansioita

IMAP: Internet Mail Access Protocol

Monipuolisempi: postikansiot (folders), lataa vain otsikot, viestien säilytys postipalvelimella

HTTP: Esim. TKTL:lla käytettävä IlohaMail, Hotmail, …

Web-palvelija käyttää IMAP-palvelijaa (eli yksi kone enemmän!) ja noutaa postin siltä IMAP–protokollalla.

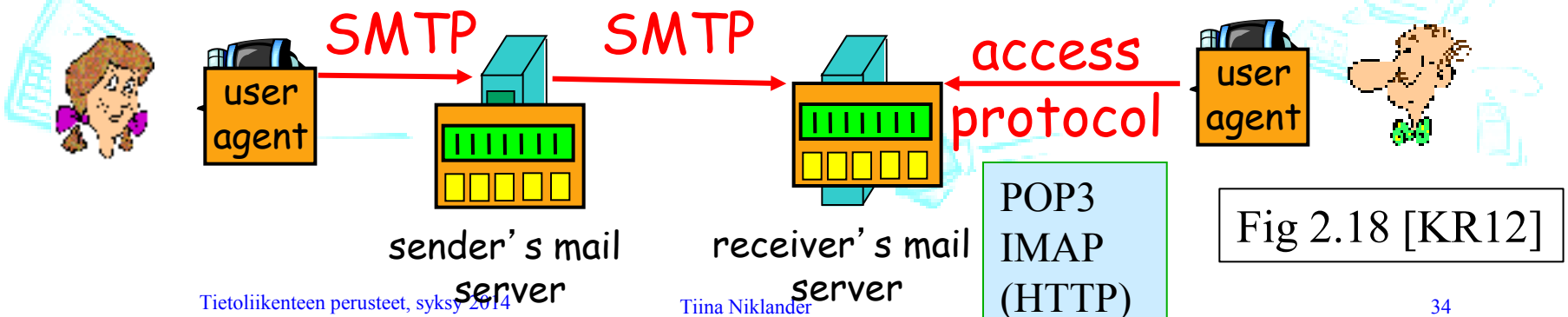


Fig 2.18 [KR12]

ESMTP

(Extended Simple Mail Transfer Protocol) RFC 2821
(uusin versio RFC 5321 (lokakuu 2008))

- Runsaasti laajennoksia jo 1995 (RFC 1868)
  * 8BITMIME — 8 bit data transmission, RFC 1652
  * ATRN — Authenticated Turn, RFC 2645
  * SMTP-AUTH — Authenticated SMTP, RFC 2554
  * CHUNKING — Chunking, RFC 3030
  * DSN — Delivery status notification, RFC 1891
  * ETRN — Extended Turn, RFC 1985
  * HELP — Supply helpful information, RFC 821
  * PIPELINING — Command pipelining, RFC 2920
  * SIZE — Message size declaration, RFC 1870
  * STARTTLS — Transport layer security, RFC 3207

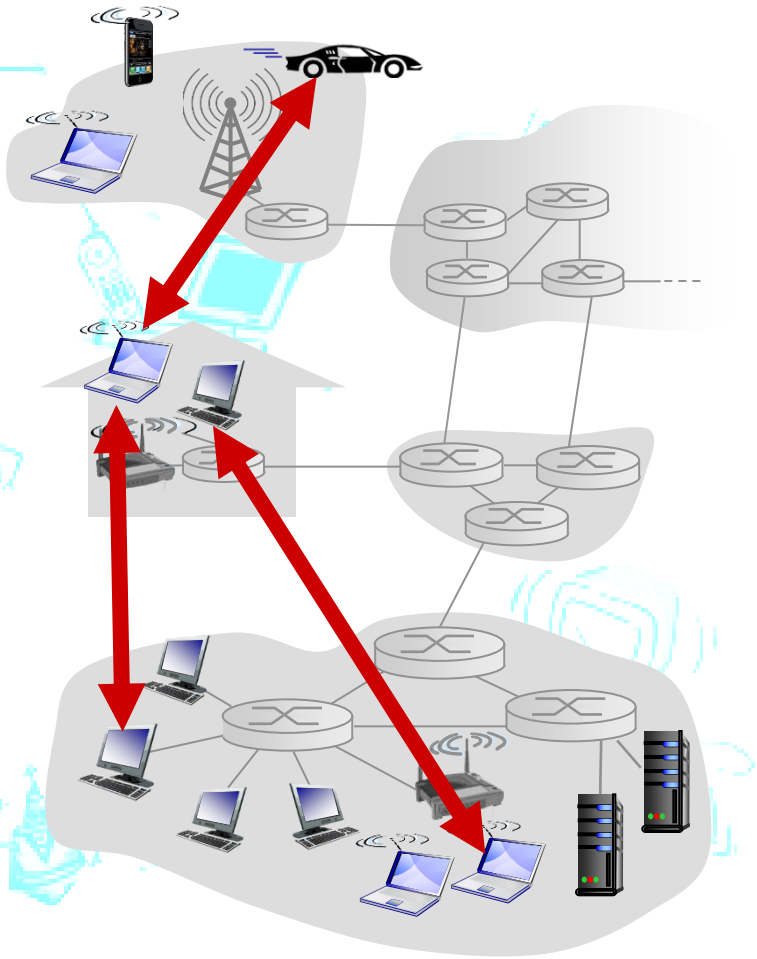
- EHLO aloittaa

VERTAISTOIMIJAT PEER-TO-PEER (P2P)

Vertaistoimijaverkko

- **Kone on satunnaisesti Internetissä** (no always-on)
  - IP-osoitekin voi vaihdella kerrasta toiseen
  - Kukin kone sekä palvelija että asiakas!
- **Skaalautuvuus, kuormantasaus**
- *Esimerkkejä:*
  - Tiedostojen jakaminen (BitTorrent)
  - Multimedia, kuten IPTV (KanKan)
  - VoIP (Skype)

Fig 2.2b [KR12]



Vertaistoimijat: tiedoston jakaminen

- **Isäntäkoneet asiakkaan ja palvelijan roolissa**
  - Jaetaan uusi versio käyttöjärjestelmästä, korjaustiedosto ohjelmaan, MP3-tiedostoja, videoleikkeitä, …
  - Jokainen vertainen voi toimia jakelijana
- **Miten löytää vertaistoimija(t)?**
  - Keskitetty hakemisto: kiinteä IP-osoite, josta voi kysellä
  - Kyselyn tulvitus: kysellään potentiaalisilta toimijoilta
  - Hiukan keskitetty hakemistopalvelu, joka tekee jatkokyselyt
- **Kun kohde löytynyt, kopiointi suoraan sieltä**
  - Kyselyn tuloksena IP-osoite
  - Nouto esim. HTTP-protokollaa käyttäen

BitTorrent-liikenne jo 30% Internetin koko liikenteestä?

Skaalautuvuus

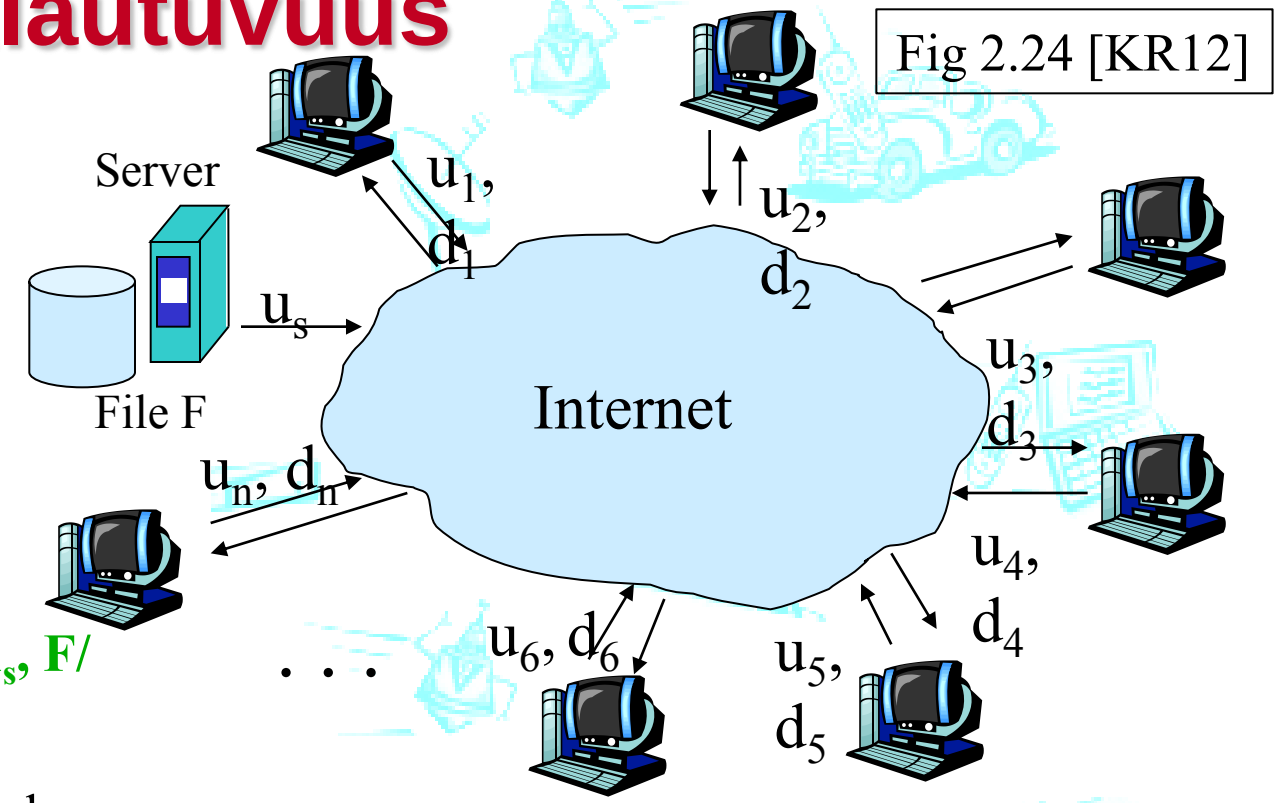


Fig 2.24 [KR12]

Asiakas-palvelinmalli:

Palvelimen siirrettävä n*F bittiä =>
siirtoaika = nF/u_s.

Hitain asiakas d_{min} saa tiedoston ajassa F/d_{min}

Siirtoaika = max (nF/u_s, F/d_{min})

Kun n kasvaa, palvelimen kuorma kasvaa ja siirtoaika kasvaa.

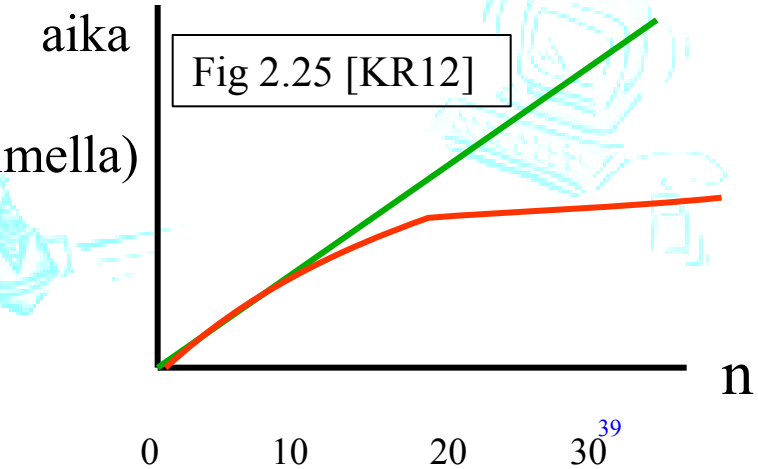


Fig 2.25 [KR12]

Vertaistoimijamalli (alussa tiedosto on palvelimella)

Siirtoaika = $max\{F/u_s, F/d_{min}, nF/(u_s + \Sigma u_i)\}$

Summamerkki: total upload rate

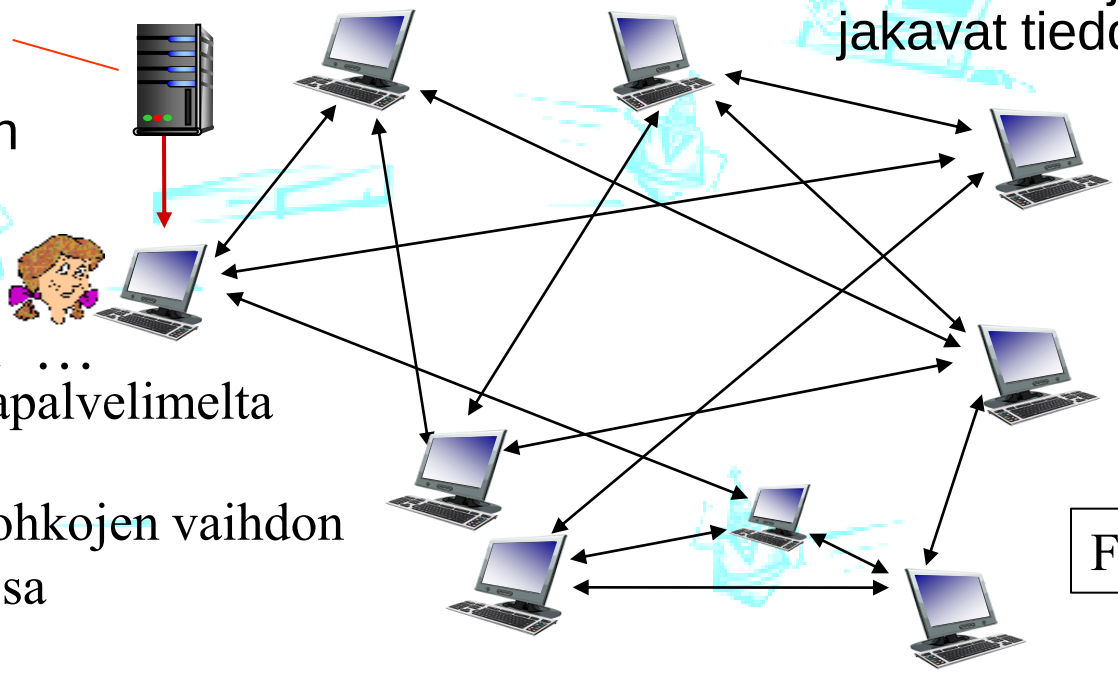
F/u_s lähetys kerran

BitTorrent
Tiedostonjakoverkko

- Tiedostot jaettu yhdenkokoisiin lohkoihin (256KB)
  Vertaistoimijat lataavat ja samaan aikaan jakavat yhden ”ryöpyn” (torrent) lohkoja

tracker, seurantapalv.: pitää kirjaa torrent-ryhmän jäsenistä

torrent: ryhmä vertaistoimijoita, jotka jakavat tiedoston lohkoja

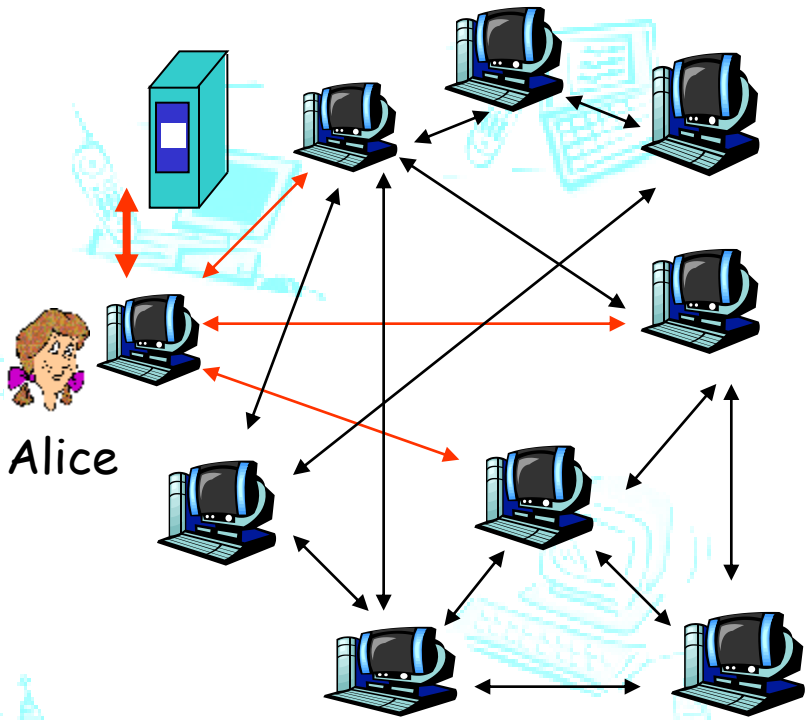


Alice saapuu …
… saa seurantapalvelimelta vertaislistan
… ja aloittaa lohkojen vaihdon vertaisten kanssa

Fig 2.26 [KR12]

BitTorrent – lohkojen vaihto

- Naapureilta kysellään lohkolistoja ja pyydetään lähettämään lohkoja (harvinaisimmat ensin)
- Itse lähettää
  - 4:lle, jotka lähettävät itselle suurimmalla nopeudella (arvio 10 s välein) (*tit-for-tat)*
  - ja 30 s välein satunnaiselle naapurille kokeeksi
- Vapaa matkustus -ongelma (free-riding)
- BitTorrentissa paljon muita piirteitä!



Hajautettu tiiviste (Distributed Hash Table, DHT)

avain	arvo
hetu	Henkilön nimi
Eloku-van nimi	IP osoite
Tiiviste/ Tunniste	Tunniste (ID)

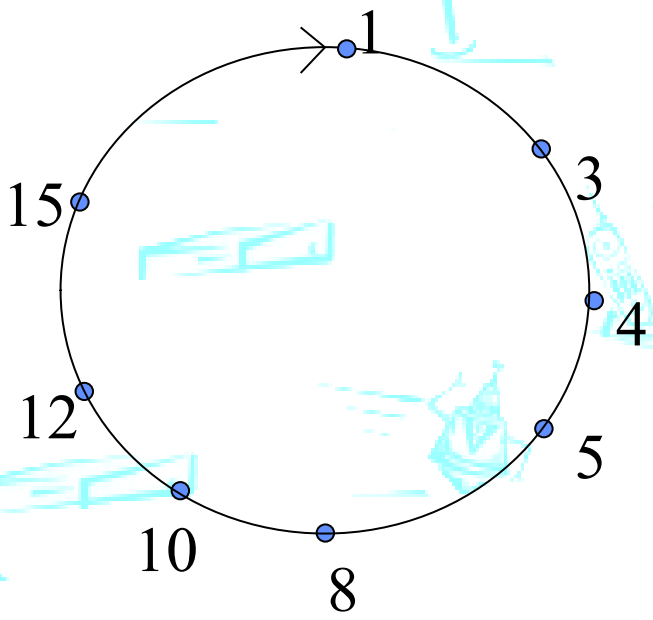
- Hajautettu tietokanta vertaisverkoille
  - alkiot ovat (avain, arvo) pareja
  - ei yhtä keskitettyä tietokantapalvelinta, vaan
  - tietokannan alkiot jaettu (miljoonille) vertaisille
    - Annetaan numeeriset tunnisteet sekä avaimille että vertaisille ja jaetaan näiden numeroiden perusteella
- Kyselyt aina avaimella
  - vastauksena tulee avaimeen liittyvä(t) arvo(t)
- Vertaisverkon solmut voivat lisätä alkioita

Numeeriset tunnisteet ja tiivisteet

- Numeroidaan vertaiset kokonaislukuarvoilla $[0,2^{n-1}]$
  - kukin tunniste on **n** bittiä.
- Lasketaan avaimille numeroarvo samalle arvovälille käyttäen hajautusfunktiota (hash function) ja tarvittaessa jakojäännöstä (jaetaan $2^{n)}$
  - esim: avain = hash(“Led Zeppelin IV”)
- Sijoittelu vertaisille (kirjan sääntö): numerojärjestyksessä lähimmälle seuraajalle renkaana.
  - eli kun n=4 ja vertaisilla tunnisteet: 1,3,4,5,8,10,12,14;
    - avain = 13, säilytysvastuu annetaan vertaiselle 14
    - avain = 15, säilytysvastuu annetaan vertaiselle 1

Circular DHT

Fig 2.27a [KR12]

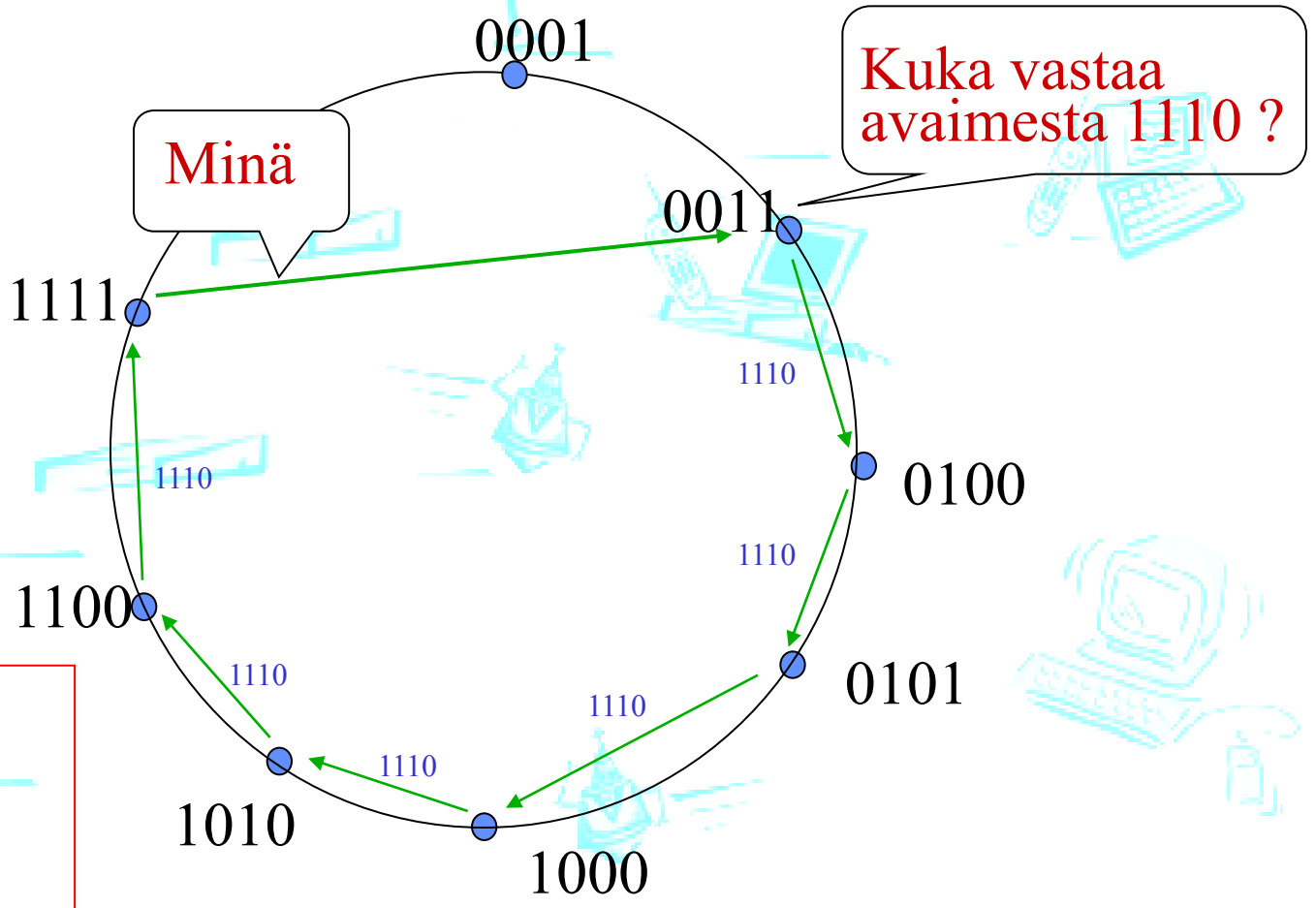


- Kukin vertainen tietää vain edeltäjän ja seuraajan numeerisen tunnisteen (ja IP:n)
- Näin muodostaa uusi päällysverkko (overlay network)

Circular DHT esimerkki

Fig 2.27a [KR12]

Kyselyyn vastaamiseen tarvitaan keskimäärin *O(N)* viestiä, kun verkossa on *N* solmua



Muista: Sijoitettu solmulle joka numeerisesti sama tai lähin suurempi

Circular DHT with shortcuts

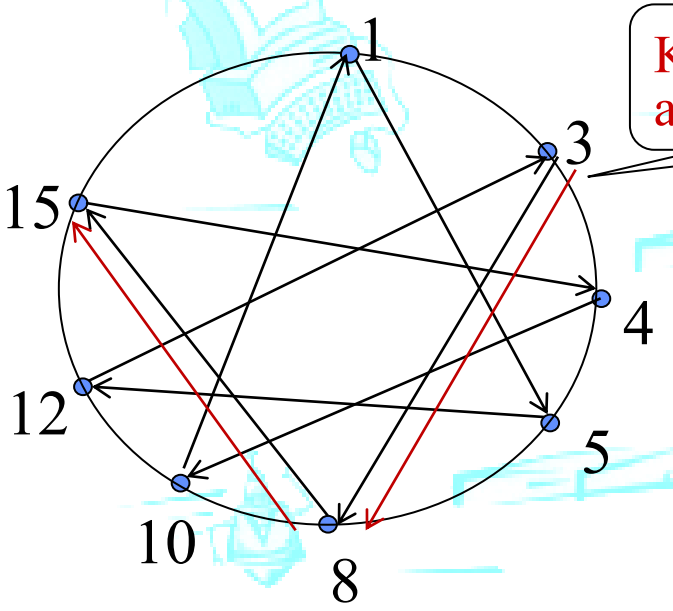
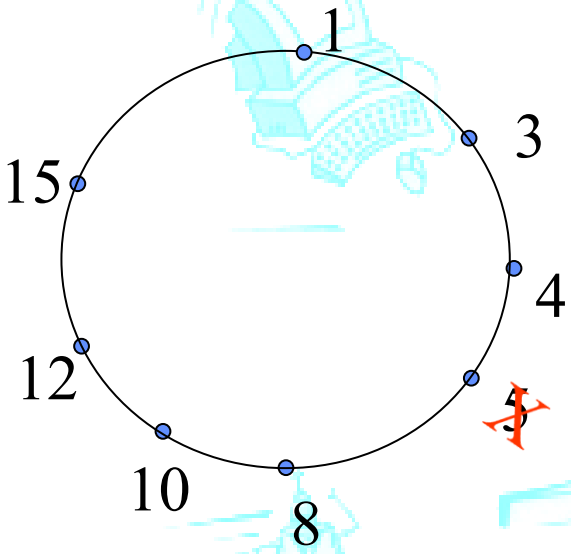


Fig 2.27b [KR12]

- Nyt jokainen solmu pitää kirjaa seuraajan ja edeltäjän lisäksi myös muutamasta oikopolusta.
- Kyselyssä tarvittavien viestien määrä putosi 6:sta 2:een.
- Teoria: oikopolut voidaan määrätä siten, että kirjanpidossa O(log N) naapuria ja kyselyissä O(log N) viestiä

Vertaisten vellominen (Peer churn): näitä tulee ja menee



- Solmut voivat liittyä ja poistua
- Jokaisen täytyy tietä kaksi seuraajaa ja edeltäjää
- Niiden mukanaolo tarkistettava (ping) säännöllisesti
- Jos muutoksia, niin tee tarvittavat päivitykset ja kysele muilta puuttuvat tiedot

- Esimerkki: Solmu 5 poistuu yllättäen
- Solmu 4 havaitsee solmun 5 poistumisen; kirjaa solmun 8 lähimmäksi seuraajaksi; kysyy solmulta 8 sen lähimmän seuraajan ja kirjaa sen itselleen seuraajan seuraajaksi.
  - Muutkin solmut joutuvat päivittämään – mitkä?
- Mitä, jos solmu 13 haluaa liittyä mukaan?

Kertauskysymyksiä

- Asiakas-palvelija-malli? Vertaisverkkomalli?
- Kuinka asiakas löytää palvelimen?
- Miten KJ osaa antaa bitit oikealle sovellukselle?
- Miten koneen nimestä saadaan selville sen IP-osoite?
- Miten HTTP-protokolla toimii?
- Miksi SMTP ei riitä, vaan tarvitaan POP3 tai IMAP?
- Mitä hyötyä on proxy-palvelimesta?
- Miksi käytetään evästeitä?
- Mikä on pistoke ja missä sitä käytetään?

Ks. myös kurssikirja s.195–197.