



Luento 2: Internetin ydin ja protokollapino

1.11.2012

Tiina Niklander

Kurose&Ross
Ch1

Luennon sisältöä

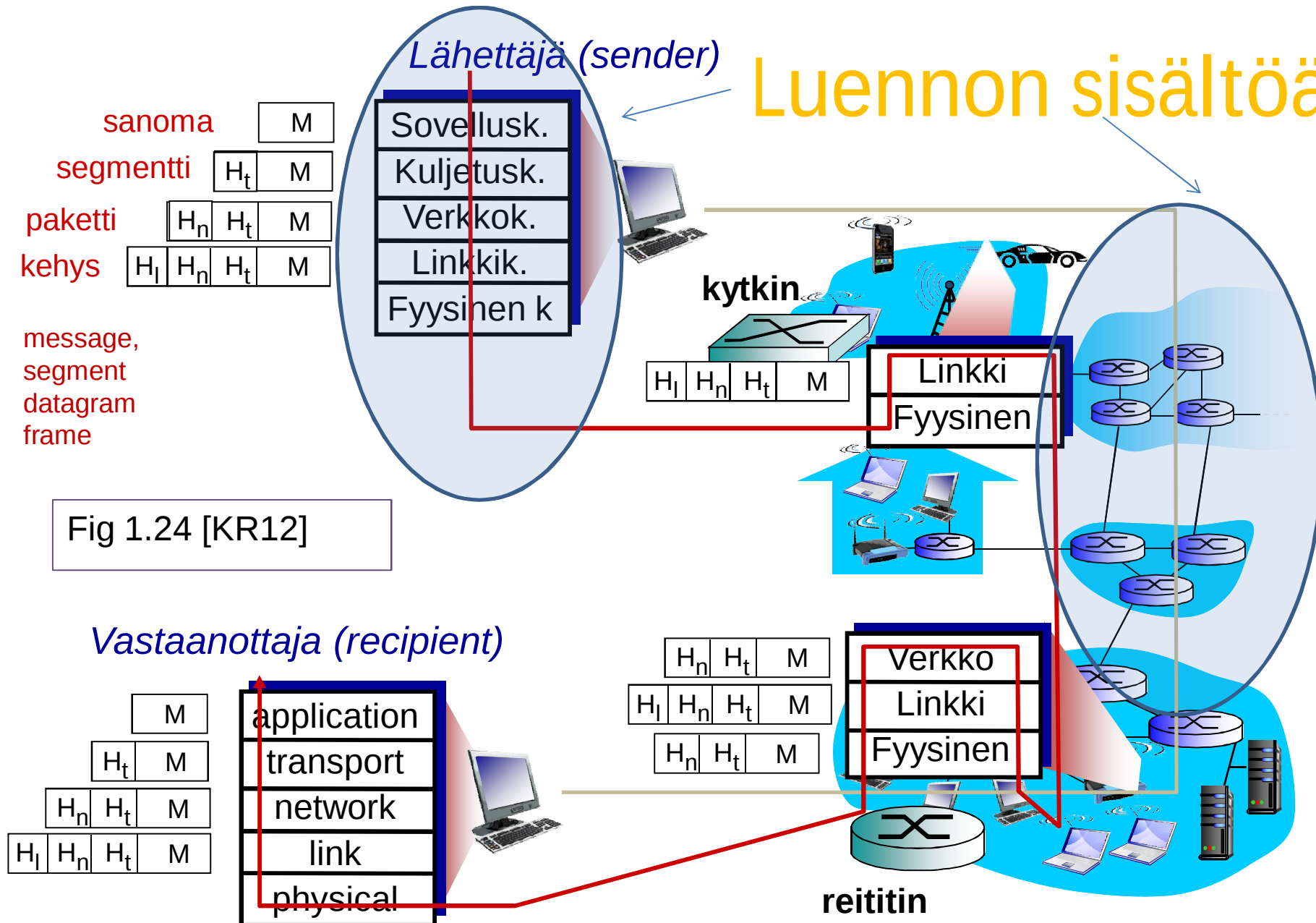


Fig 1.24 [KR12]



Tietoliikenteen perusteet

Verkon syövereissä,
reititys
(network core)



Verkon syövereissä: verkon ydin (core)

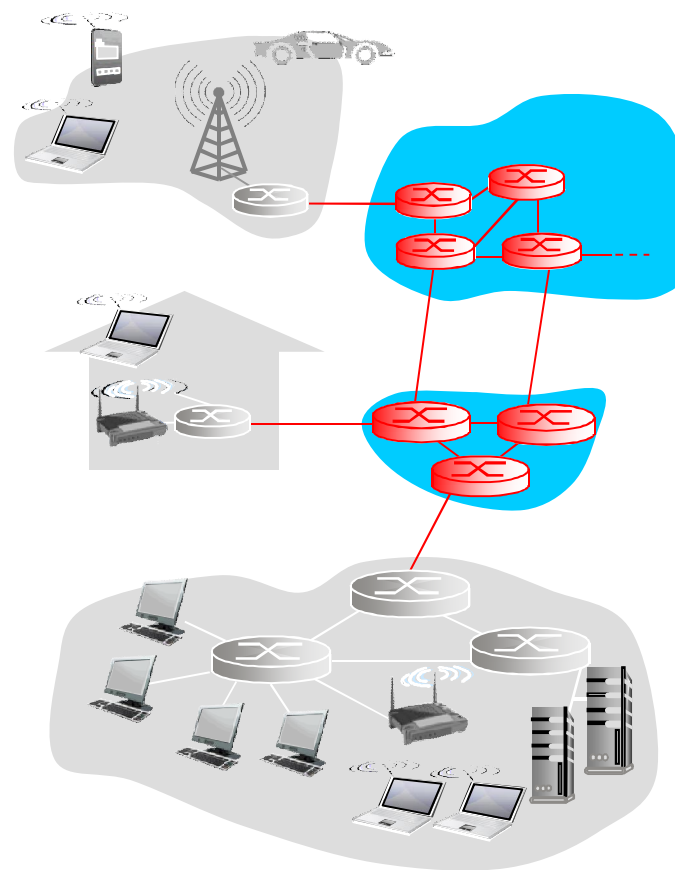
Fig 1.10 [KR12]

Toisiinsa liitettyjen reittimien verkko (mesh)

isäntäkoneet pilkkovat
sovelluskerroksen sanomat
paketeiksi (*packets*)

Reitittimet siirtävät kunkin
paketin seuraavalle linkki
(hop) kerrallaan, näin
muodostuu polku lähettäjältä
vastaanottajalle

Koko linkin kapasiteetti
siirrettävälle paketille



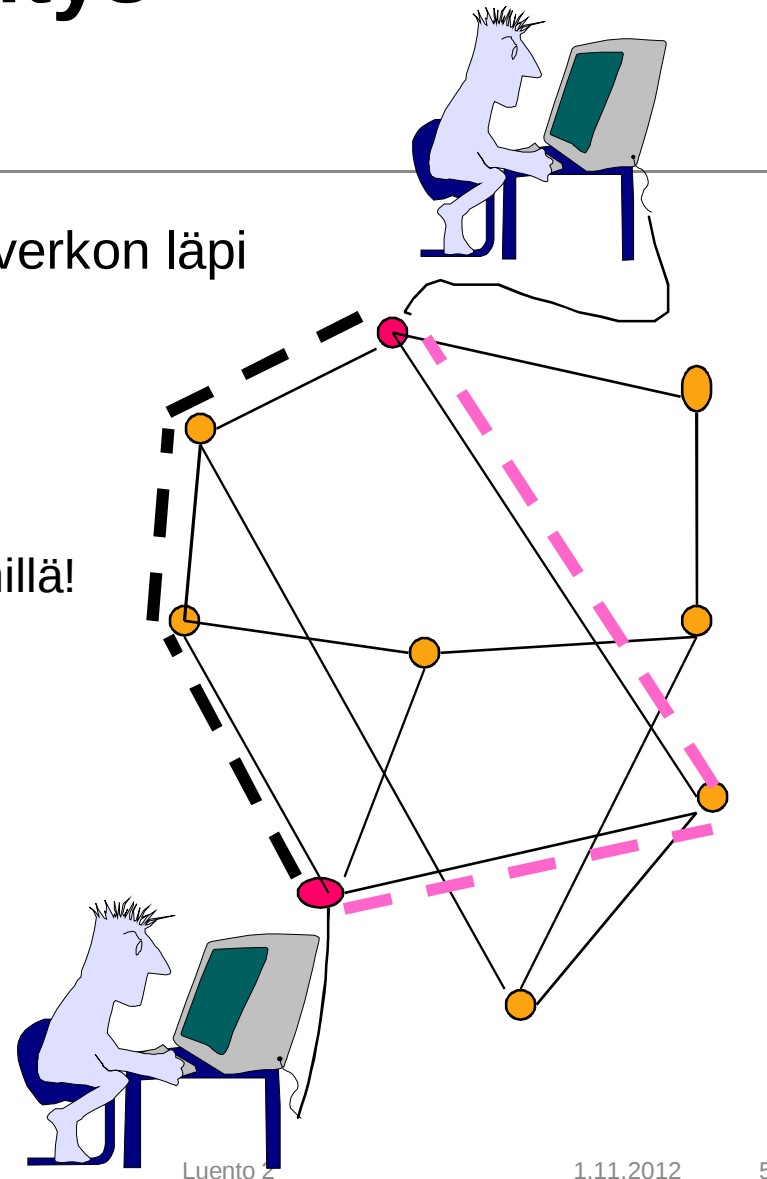


Reitittimet, reititys

- Miten sanoma kuljetetaan verkon läpi
 - lähettävältä koneelta vastaanottavalle koneelle?
- Verkkojen verkko,
 - verkot on yhdistetty reitittimillä!

Piirikytkentä: varaa ensin linkit, joita pitkin kaikki data kulkee

Pakettikytkentä: kuljeta data verkossa pieninä paketteina ja reititä kukin paketti itsenäisesti



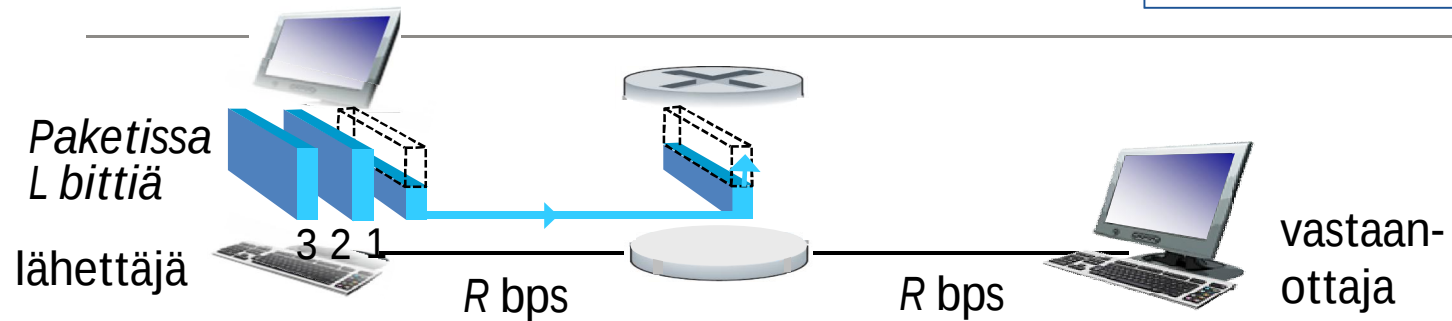
- jokainen paketti reititetään erikseen





Pakettikytkentä: etappivälitys

Fig 1.11 [KR12]



Etappivälitys (store and forward)

paketti vastaanotetaan kokonaan ja vasta sitten lähetetään eteenpäin

Paketin (koko L bittiä) lähettämiseen linkin (nopeus R bps) yli kuluu L/R sekuntia

Esimerkki yhden linkin yli:

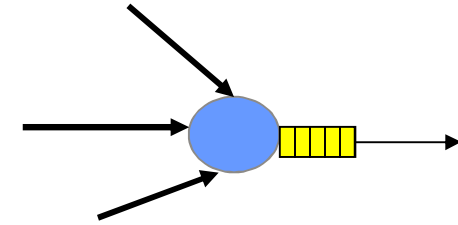
$L = 7.5$ Mbits

$R = 1.5$ Mbps

siirtoaika linkin yli = 5 sec



Pakettikytkentä (packet switching)



Ei varata resursseja eikä siis reittiä etukäteen,

- Varaus tarvittaessa (on-demand)
- Tilastollinen kanavointi (Statistical multiplexing)

jokainen paketti reititetään erikseen => paketit voivat kulkea eri reittejä lähettäjältä vastaanottajalle

Yhteenlaskettu siirtotarve voi ylittää lähtevän linjan siirtonopeuden

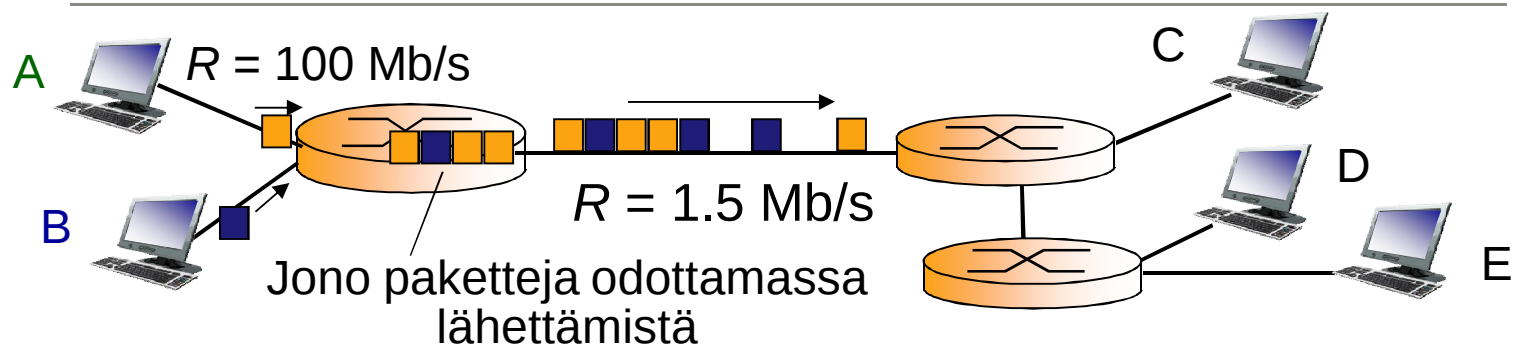
Paketti joutuu odottamaan vuoroaan reitittimen muistissa (jonotusviive, queueing delay)

Ruuhka (congestion) => jopa paketin häviäminen



Pakettikytkentä: jonotus ja katoaminen

Fig 1.12 [KR12]



Jonotus (queuing) ja katoaminen (loss):

Jos paketteja saapuu hetkellisesti nopeammin kuin niitä ehditään lähettää eli saapumistiheys (arrival rate) > lähetystiheys (transmission rate)

 paketit jonottavat lähetysvuoroaan

 jos reittimen muisti täyttyy, se kadottaa saapuvia paketteja

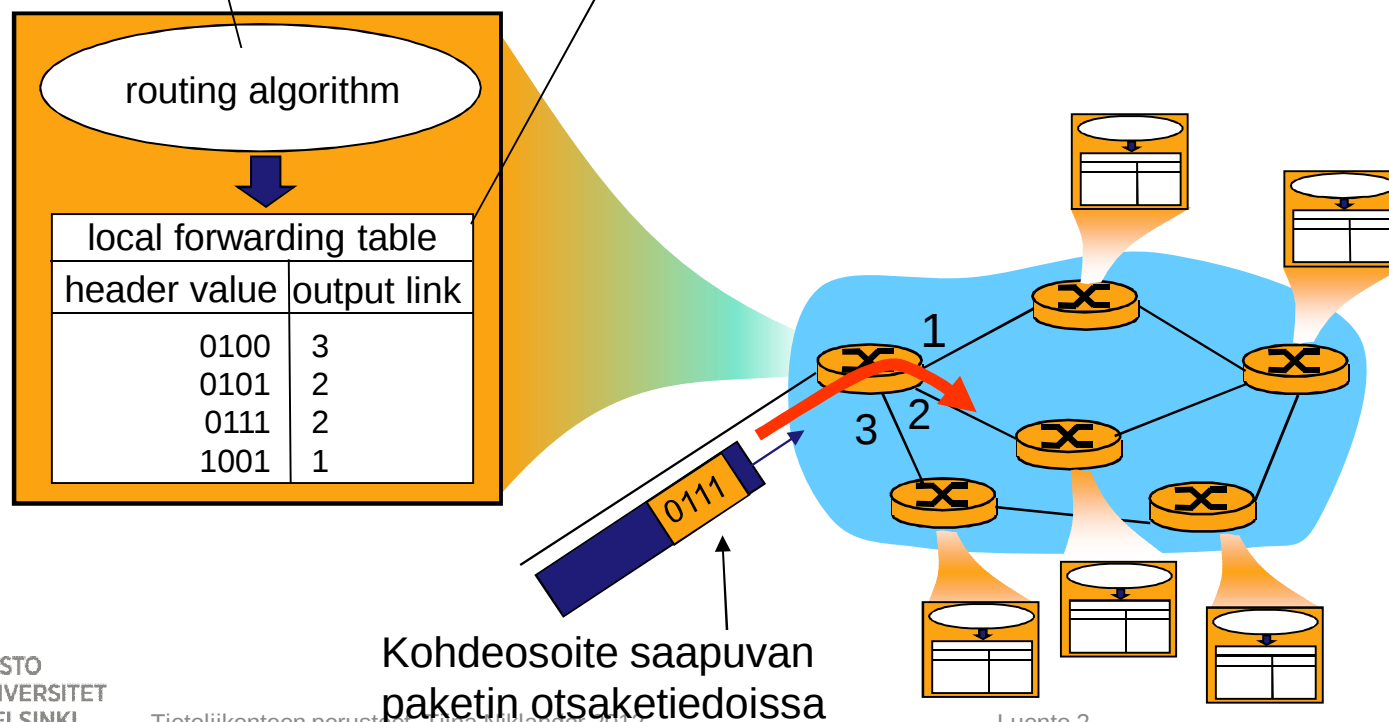


Reititys

Fig 4.2 [KR12]

Reititys (routing): Reititysalgoritmit laskevat parhaat reitit ja päivittävät taulukkoa

Edelleenlähetys (forwarding): Reitittimessä taulukko => mihin linkkiin kukin kohdeosoite on ohjattava





Siirtonopeus, siirtoaika

Siirtonopeus

miten nopeasti dataa lähetetään (bittejä generoidaan)
linjalle

Montako bittiä per aikayksikkö lähetetään

bps = bittejä sekunnissa

Siirtoaika

kauanko datamäärän lähettäminen linkille kestää
(s.e. viimeinenkin bitti on lähetetty linkille)

Esim. 10 MB dataa ja siirtonopeus on 1 Mbs => siirtoaika
 $= 10 \text{ MB} * 8 \text{ b/B} / 1 \text{ Mb/s} = 80 \text{ s}$ sekuntia



Kauanko kestää...

Kauanko kestää lähettää

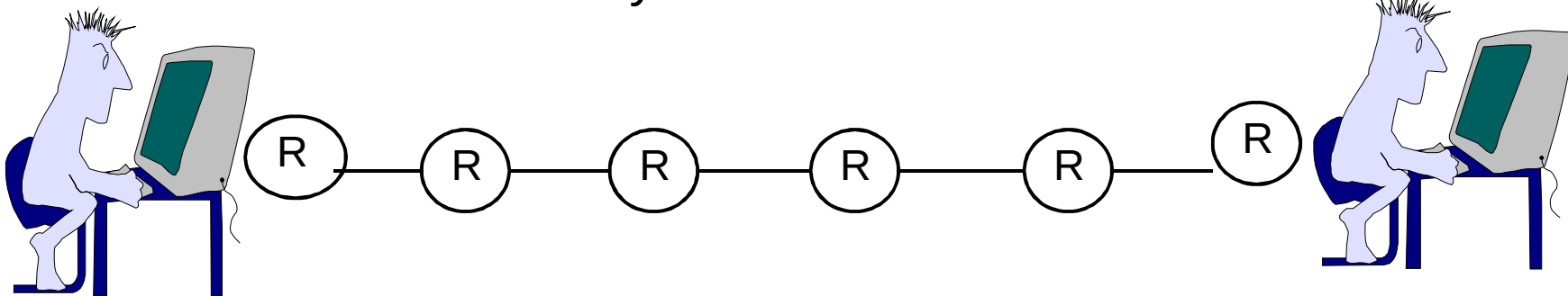
yksi 4 kb:n paketti

pakettikytkentäisessä verkossa, jossa

linkin siirtonopeus on 1 Mbps

ja paketti kulkee

5 linkin yli





Ratkaistaan:

siirtoaika yhdellä linkillä on

$$4 \text{ kb} / 1000 \text{ kb/s} = 0.004 \text{ s} = 4 \text{ ms}$$

joten siirto 5 linkin yli

$$5 * 4 \text{ ms} = 20 \text{ ms}$$

Jonotusviipeitä ei ole otettu huomioon



Pakettikytkentä: Sanoma vs. paketit

Miksi ei lähetetä koko sanomaa kerralla?



- Siirtovirhe
 - Sanoma: koko virheellinen sanoma lähetettävä uudestaan
 - Paketti: Vain yksi virheellinen paketti lähetettävä uudestaan
- Yleisrasite (overhead) = otsake, jossa mm. lähettäjän ja vastaanottajan osoitteet
 - Sanoma: yksi otsake riittää
 - Paketti: jokaisessa paketissa oma otsake



Sanoma vs. paketit (jatkuu)

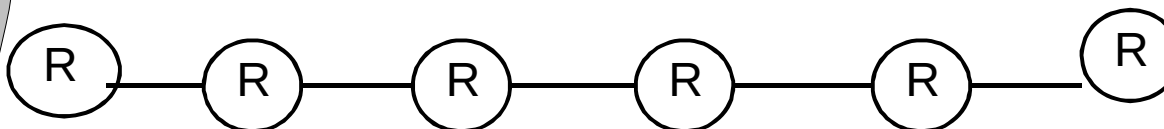
Esim. Sanoman koko 400 kb, linkin nopeus 1 Mbps

- Kun koko sanoma lähetetään 5 linkin yli, niin aikaa kuluu

$$5 * 400 \text{ kb} / 1 \text{ Mbps} = 5 * 0.4 \text{ Mbps} / 1 \text{ Mbps} = 2 \text{ s} = 2000 \text{ ms}$$

- Kun sanoma pilkotaan 4 kb:n paketeiksi, niin aikaa 100 paketin välittämiseen kuluu

$$416 \text{ ms!! } (= 100 * 4 \text{ kb} / 1 \text{ Mps} + (5-1) * 4 \text{ kb} / 1 \text{ Mps})$$





Miksi noin?

Paketteja voi lähettää samanaikaisesti eri linkeillä
etappivälitys (store-and-forward)

Koko sanoma (100 pakettia) siirretty 1. linkin yli
 $400 \text{ Kb} / 1 \text{ Mbps} = 400 \text{ ms}$

Kun viimeinen paketti on siirretty 1. linkin yli, lähes
kaikki edeltävät paketit ovat jo perillä.

Nyt vielä viimeinen paketti on siirrettävä 4 linkin yli
 $4 * 4 \text{ Kb} / 1 \text{ Mbps} = 16 \text{ ms}$

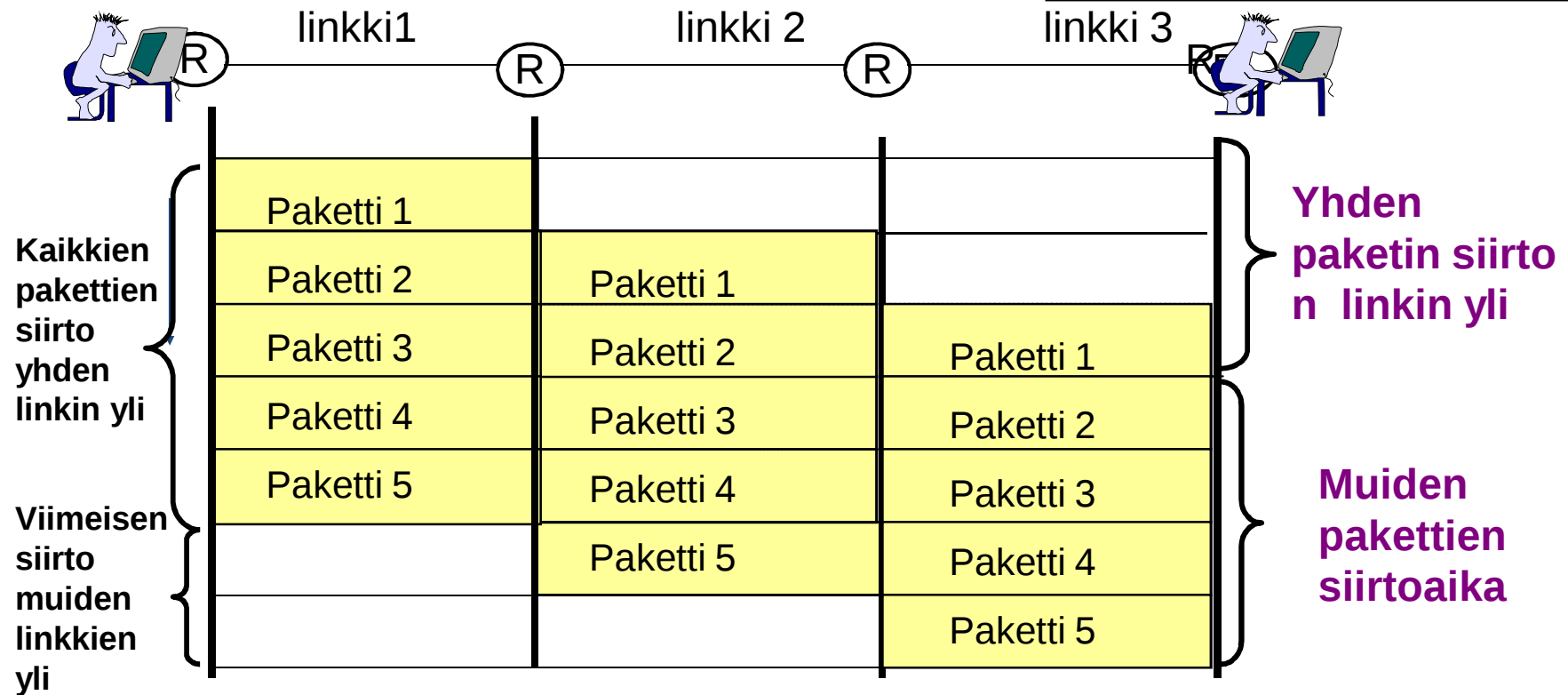
Siis: $400 \text{ ms} + 16 \text{ ms} = 416 \text{ ms}$

Pakettivälitys: siirtoaika

Olkoon siirtoaika a:

a) $ka + (n-1)a = (k+n-1)a$

b) $na + (k-1)a = (n+k-1)a$



Sanoman siirtoaika, kun sanomassa on k pakettia ja linkkejä on n kappaletta

a) k:n paketin siirto 1. linkin yli + viimeisen paketin siirto n-1 linkin yli.

b) 1. paketin siirto n:n linkin yli + muiden k-1 paketin siirto yhden linkin yli

Animaatio:

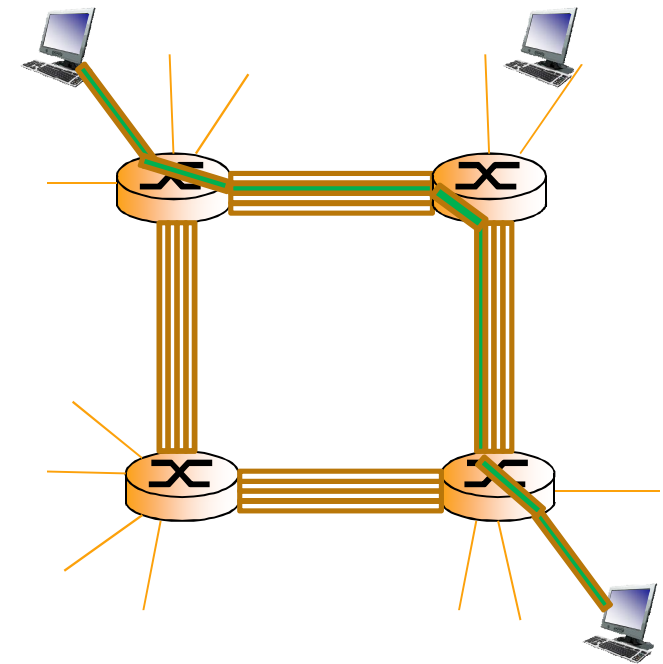
http://wps.aw.com/aw_kurose_network_4/63/16303/4173752.cw/index.html



Piirikytkentä (circuit switching)

Fig 1.13 [KR12]

- Varaa yhteydelle omat resurssit päästä-päähän koko yhteyden ajaksi
 - Varataan puskurit, linjakapasiteetti
 - Yhteydenmuodostus ("call")
 - Yhteydenpurku ("shutdown")
- Resurssit varattuna, vaikka niitä ei käytettäisi
- Takaa tasaisen nopeuden
 - puhelinverkko

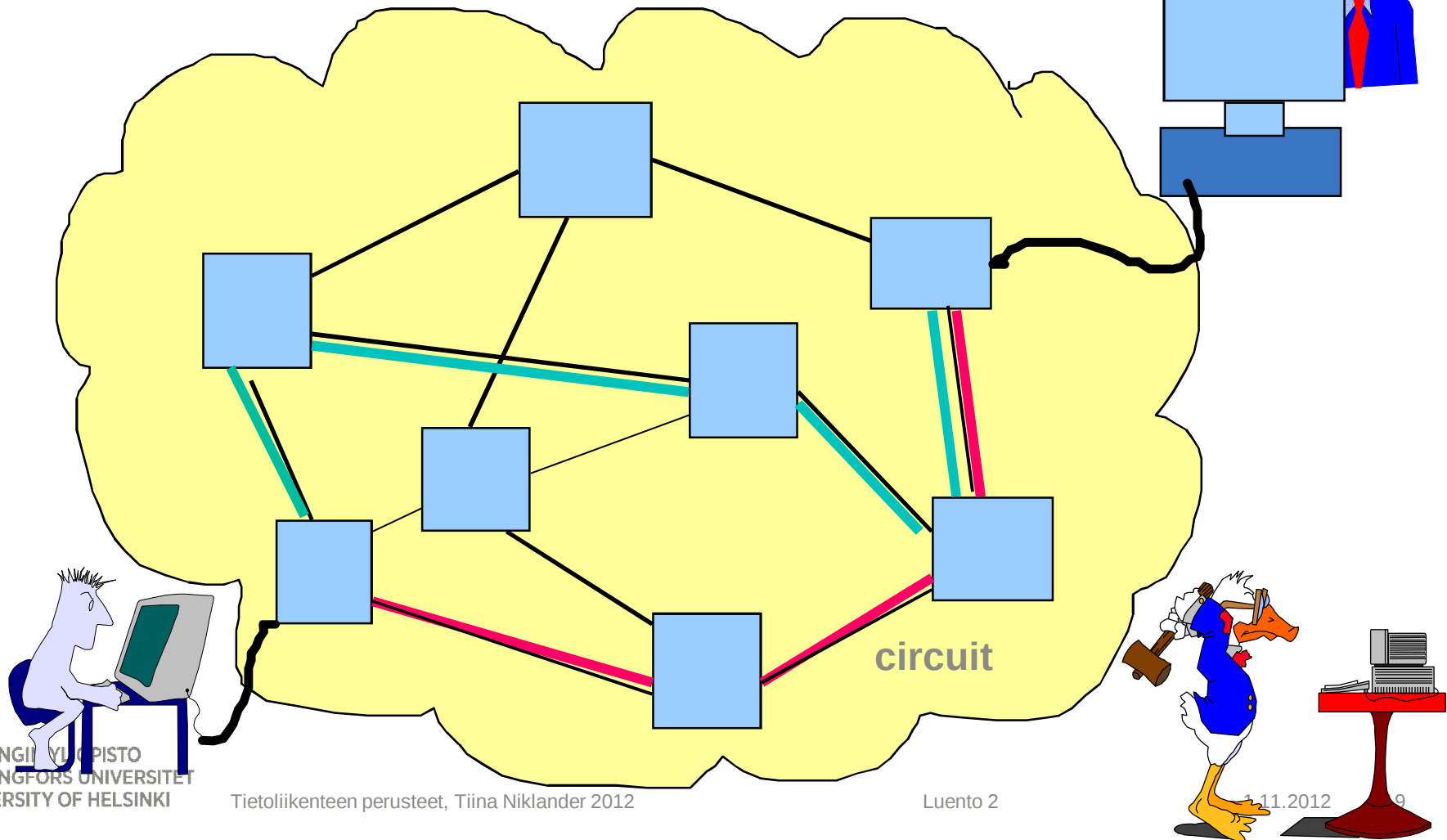


vrt: vesipisteiden yhdistäminen letkuilla ja veden valutus



Piirikytkentä

- ensin varataan resurssit yhteyttä varten
- sitten koko datan siirto yhteyttä pitkin
- vapautetaan resurssit





Piirikytkentä: kanavointi (multiplexing)

Fig 1.14 [KR12]

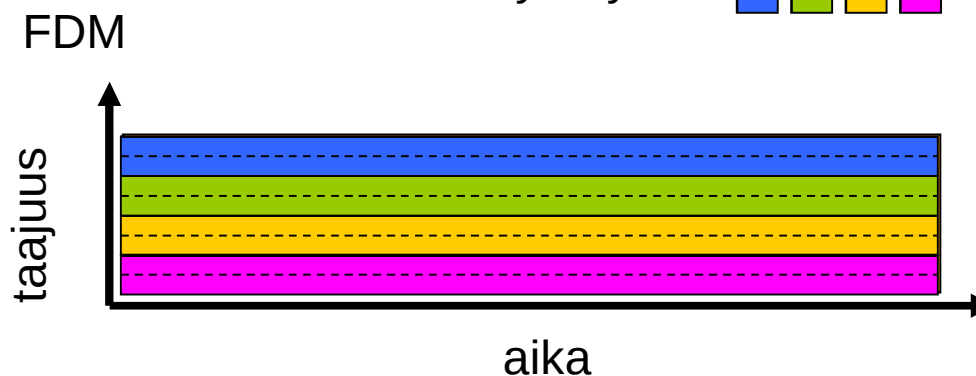
Linkille on limitetty usean yhteyden sanomia

Esimerkki:

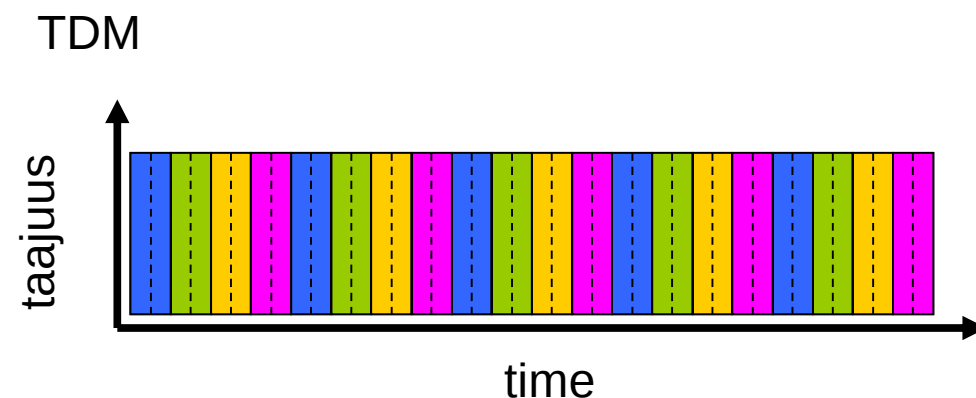
4 yhteyttä



Taajuusjako, FDM (frequency-division multiplexing): linkin kaistanleveys (taajuudet) jaettu käyttäjien kesken



Aikajako, TDM (time-division multiplexing): jokainen saa käyttöönsä koko kaistanleveyden tietyn aikajakson ajaksi





Kauanko kestää ...

Kauanko kestää lähettää

640 kbitin tiedosto

piirikytkentäistä verkkoa käyttäen, kun

linjan siirtonopeus on 1.536 Mbps

ja linjalla käytetään TDM:ää, jossa on

24 aikaviipaletta (eli 24 käyttäjää)

ja yhteyden muodostamiseen kuluu

500 ms?



Ratkaistaan

Yhdelle yhteydelle on käytössä

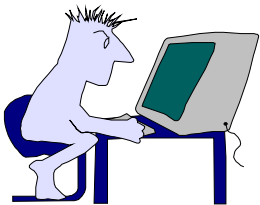
$$1.536 \text{ Mbps}/24 = 64 \text{ kbps}$$

joten siirtoon kuluu $640 \text{ kb}/64 \text{ kbps} = 10 \text{ s}$

Kun yhteyspiirin muodostus vie 0.5 s

niin aikaa kuluu yhteensä $10.5 \text{ s}.$

Huom! Aika ei riipu välissä olevien linkkien
lukumäärästä! (ei käsittelyviivettä)





Pakettikytkentä vs. piirikytkentä

Pakettikytkentä sallii enemmän samanaikaisia käyttäjiä!

Esimerkki: 1 Mb/s linkki

Jokainen käyttäjä:

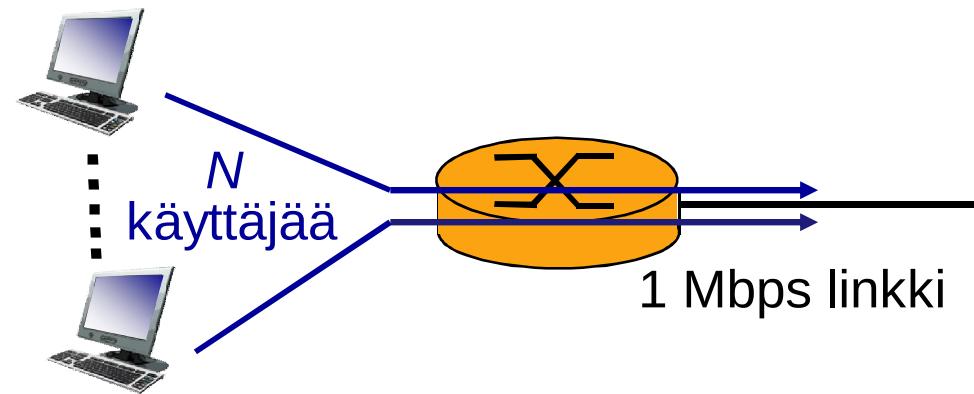
100 kb/s kun “aktiivinen”
aktiivinen 10% ajasta

Piirikytkentä:

10 käyttäjää

Pakettikytkentä:

kun 35 käyttäjää, todennäköisyys
että > 10 aktiivista samaan aikaan
on vähemmän kuin 0.0004



Q: Mistä tulee tuo arvo 0.0004?

$$\begin{aligned} & \text{tn (aktiivisia } > 10) = \\ & 1 - \left(\sum_{k=0}^{10} \binom{35}{k} (0.1)^k (0.9)^{35-k} \right) \end{aligned}$$



Tietoliikenteen perusteet

Protokolla,
protokollapino



Protokollien kerrostaminen

- Protokolla = yhteyskäytäntö
 - Mitä sanomia, missä tilanteessa ja missä järjestyksessä lähetetään
 - Miten saatuihin sanomiin reagoidaan
 - Sanomien syntaksi ja semantiikka
- Protokollapino = protokollien kerrosrakenne
 - Toiminnot on jaettu kerroksiin järkevästi
 - Alemman kerroksen toiminnot ovat ylemmän käytössä
 - Palvelu ja sen toteutus erotettu
- Kukin protokolla toimii yhdellä kerroksella ja toteuttaa tämän kerroksen jonkin palvelun



Miksi kerrosrakenne?

Monimutkaisuuden hallinta

- Kerroksittainen viitemalli (reference model) helpottaa asiakokonaisuuksiin viittaamista
 - Kullakin kerroksella omat selkeät tehtävänsä
 - Kerroksissa omat 'lisä'toiminnot
- Voi käyttää olemassaolevia alemman kerroksen toimintoja
- Kerrosten rajapinnat (interface) hyvin määriteltynä
 - Kaksisuuntainen 'palveluluukku': mitä tekee, kuinka on käytettävissä
- Joustavuus
 - Pino koottavissa erilaisista protokollista
 - Kerroksen toteutusta voi muuttaa, kunhan rajapinnat ennallaan
- Jos kerroksia on paljon, se voi vaikuttaa suorituskykyyn
 - Sama työ toistamiseen, esim. virhetarkistus
 - Kutsumekanismi: kopiointia paikasta toiseen, ..



Esimerkki: lentomatustus

Fig 1.21 [KR12]

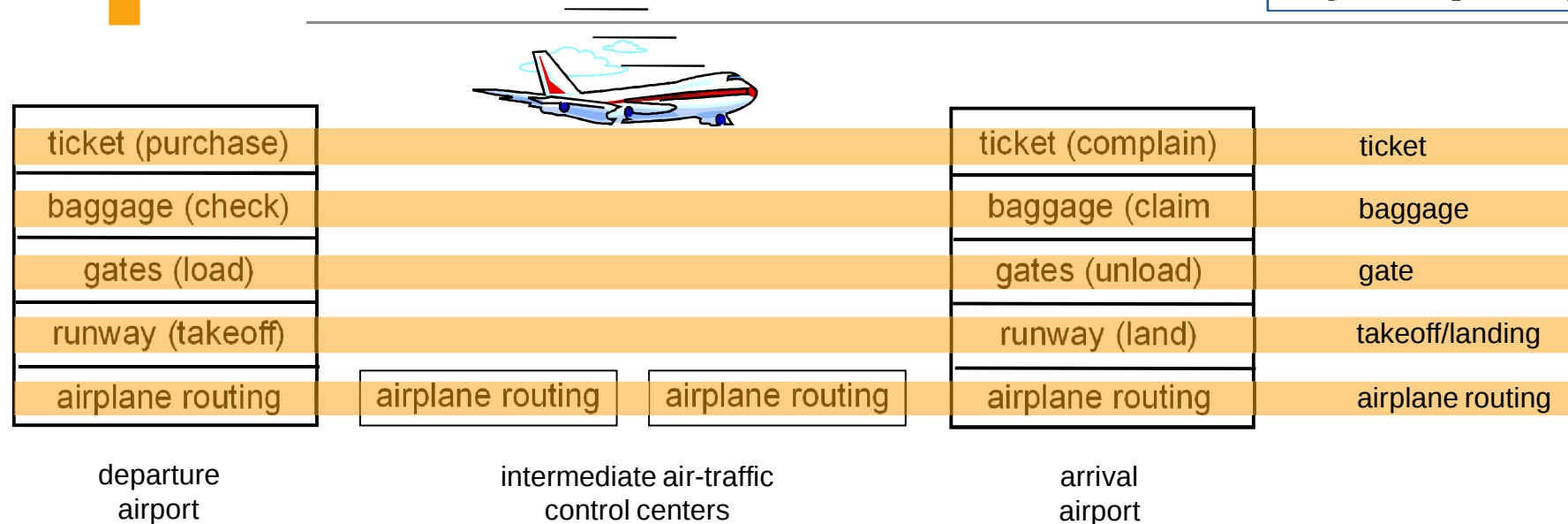


Useita vaiheita, jotka tehtävä järjestyksessä



Esimerkki: lentoyhtiön palvelurakenne

Fig 1.22 [KR12]



Kerrokset (layers): kukin kerros toteuttaa jonkin palvelun

- oma palvelurajapinta ja sisäinen toiminta
- käyttää alemman kerroksen palveluja oman toteuttamiseen



Internet-protokollapino

1969: aluksi TCP ja IP
samassa nipussa

1980: uusittu TCP, UDP ja IP

Lähtökohdat

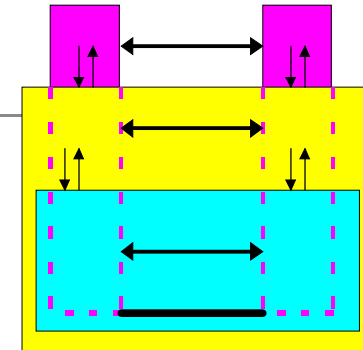
- Tarve yhdistää monia hyvin erilaisia verkkoja
- Vikasietoisuus

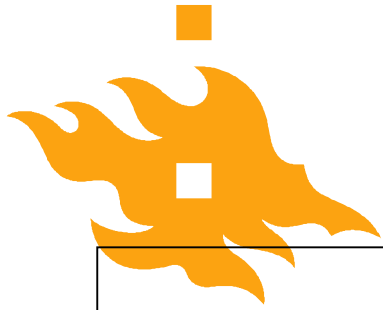
De-facto-standardi

- Ensin toimiva protokollatoteutus, sitten viitemalli
- RFC-julkaisuja, standardeja

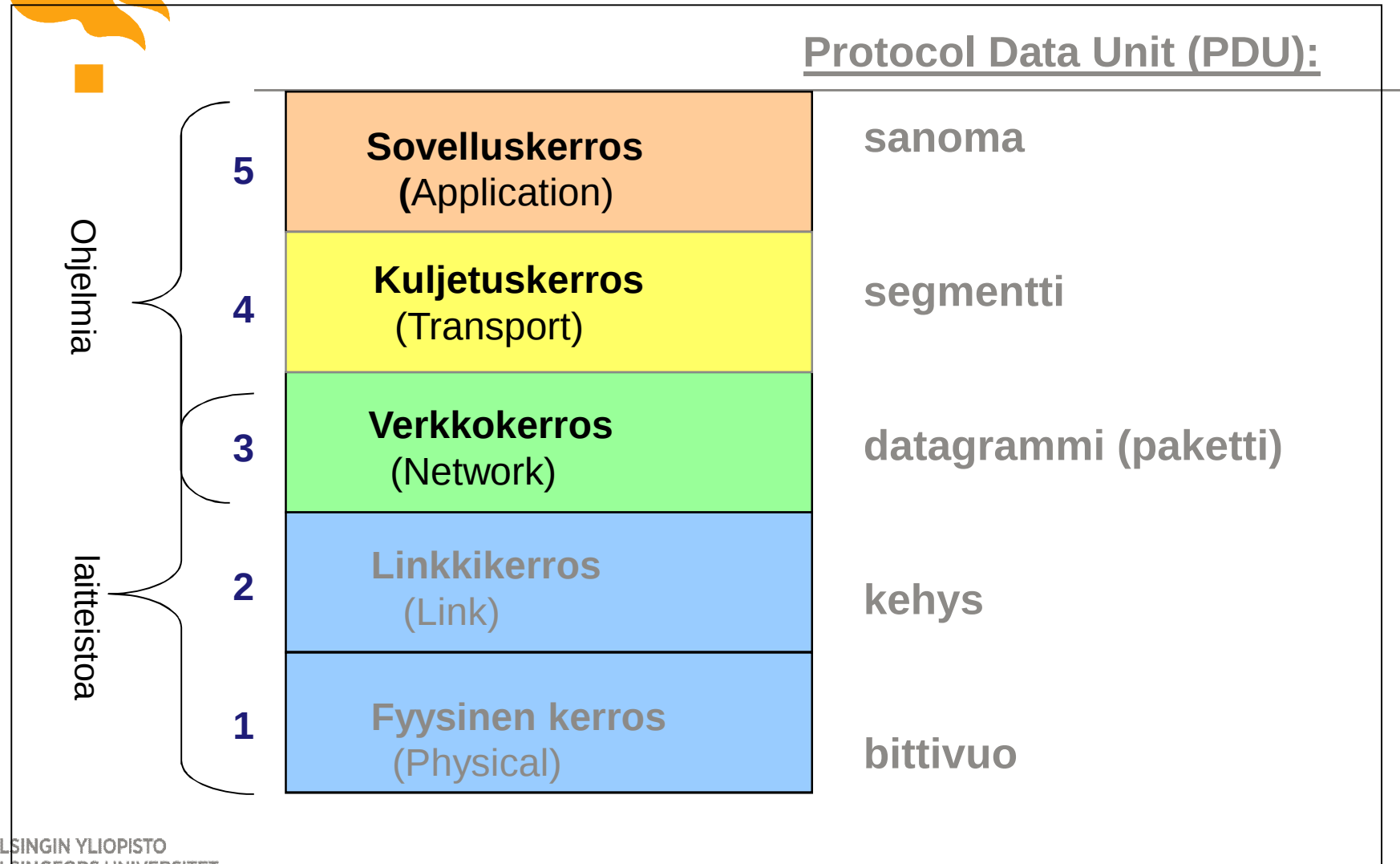
Tulos

- Koneilla yksikäsitteiset IP-osoitteet
- Pakettikytkentäinen verkko: IP-pakettien välitystä
- Yhteydellinen (TCP) ja yhteydetön (UDP) palvelu.





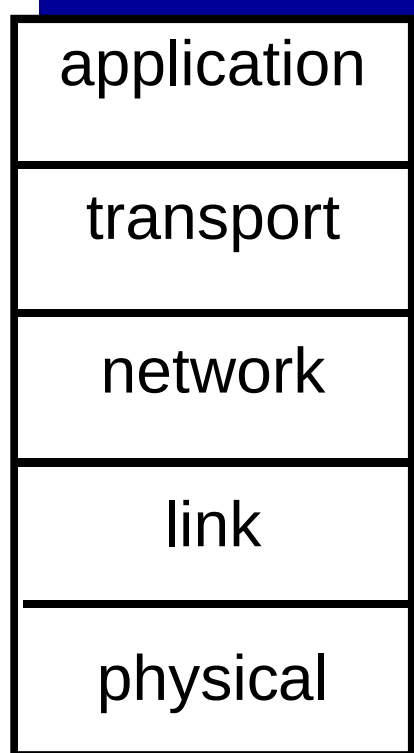
Internet-protokollapino





Internet-protokollapino

Fig 1.23a [KR12]



Sovellus: verkkosovellusten omat protokollat

HTTP, DNS, SMTP, FTP,

Kuljetus: segmenttien siirto prosessilta prosessille
("päästä-päähän") TCP, UDP

siirtää sanomien bittivirtaa segmentin kokoisina lohkoina

Verkko: pakettien reititys verkossa, siirto
lähettäjäkoneelta vastaanottajan koneelle

IP, reititysprotokollat

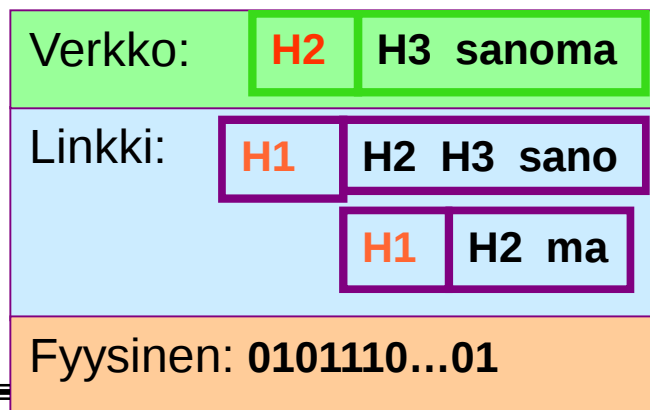
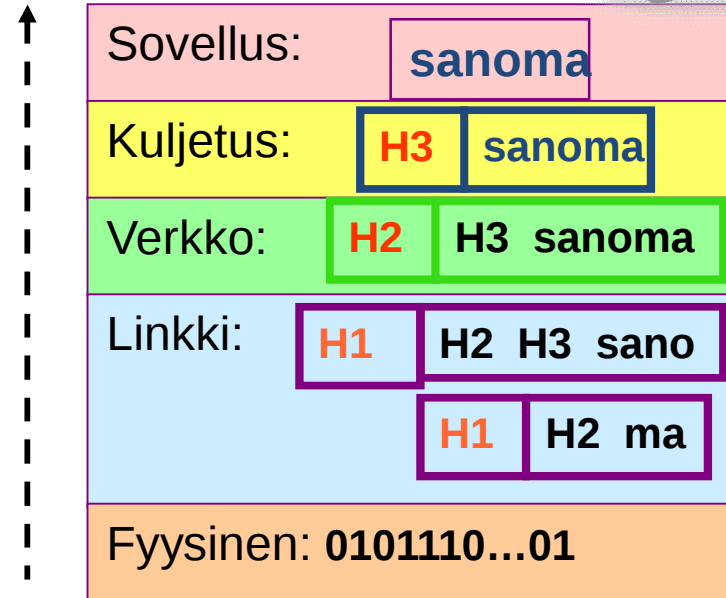
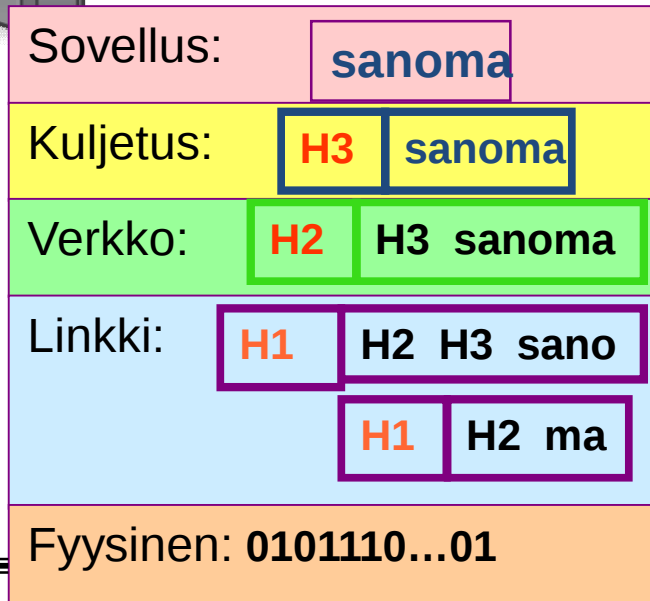
muodostaa segmenteistä paketteja, voi pilkkoa

Linkki: siirtää paketit kehyksinä kahden koneen välillä

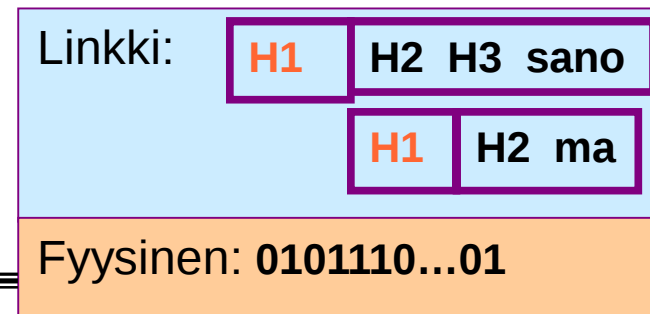
Ethernet, WiFi, PPP

Fyysinen: generoi, siirtää ja vastaanottaa bittejä
koneelta toiselle

Kapselointi (encapsulation)



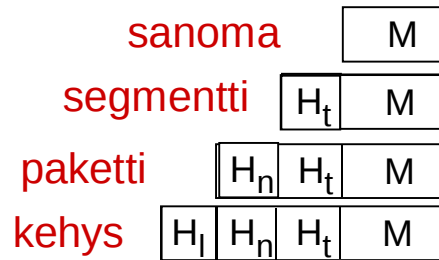
Reititin



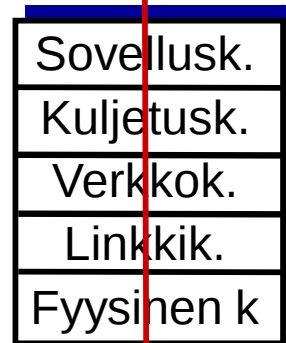
Linkkitason kytkin

Kurssin tärkein kuva vrt ed.kalvo

Lähettäjä (sender)



message,
segment
datagram
frame



kytkin

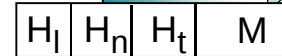
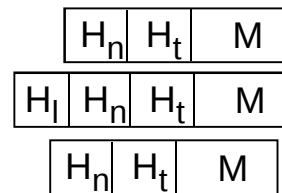
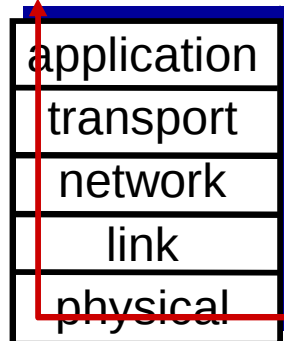
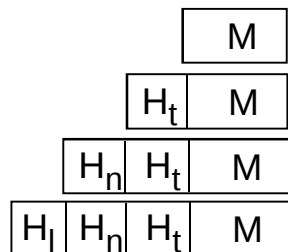


Fig 1.24 [KR12]

Vastaanottaja (recipient)



reititin



ISO OSI-viitemalli

7-kerroksinen malli

ISO = International Standardization Organization

OSI = Open Systems Interconnection

yhdistää koneita, jotka 'avoimia' kommunikointiin toisten kanssa

Käsitteellisesti ehjä malli,

1978 -> 1982 viitemalli

1983 -> toiminnallisia standardeja

1995 uudistuksia

mutta ei paljoakaan käytössä

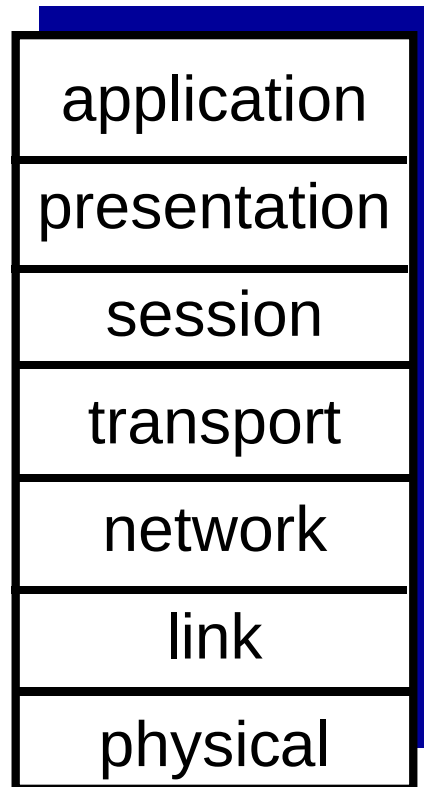
Katoavaa kansanperinnettäkö? Vai vasta tulossa?





ISO OSI-viitemalli

Fig 1.23a [KR12]

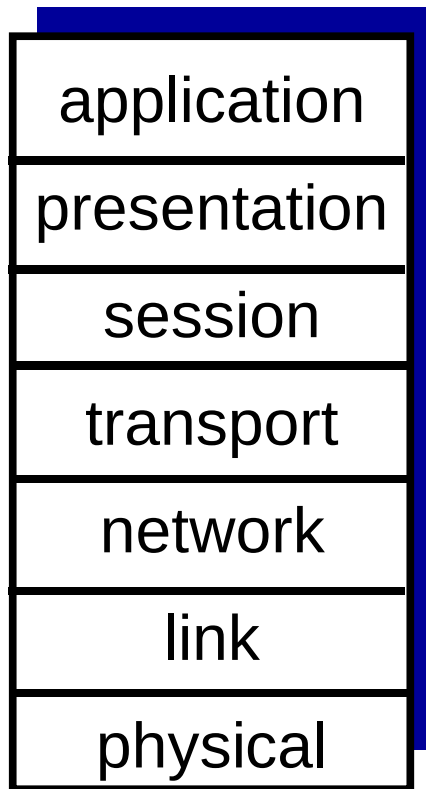


- Esitystapakerros (Presentation)
 - Huolehtii tiedon esitysmuodosta
 - Tiedon esitystapa voi olla erilainen eri koneissa
 - Käytettävästä siirtosyntaksista sopiminen
 - Muuttaa tiedon siirtosyntaksin mukaiseksi
 - abstrakti tietorakenne (esim. henkilötietue) siirtomuotoon
- Salaus ja tiivistys haluttaessa



ISO OSI-viitemalli

Fig 1.23a [KR12]



- Istuntokerros (Session)
 - Jäsentää tietojen vaihtoa istunnossa
 - kommunikointitavasta sopiminen: kaksi- vai yksisuuntainen
 - lähetysvuoronsäätely
 - Tahdistaa kommunikointia esim. tiedostonsiirrossa
 - Tahdistuspisteet: jos yhteys katkeaa, voi jatkaa siitä mihin jäi
- Esitystapa- ja istuntokerrosten toiminnallisuus (+ paljon muuta) rakennettavissa TCP/IP-kerrosten päälle = väliohjelmistot (middleware)



Kertauskysymyksiä

Isäntäkone vs. reititin?

Protokolla vs. palvelu?

Vertaisverkkomalli vs. asiakas-palvelin malli?

Fyysinen siirtomedia?

Piiri- ja pakettikytkentä? Hyödyt ja haitat?

Viipeet ja pakettien katoamiset

Internet-protokollakerrokset ja niiden tehtävät?

Miksi kerrosrakenne?

Mitä protokollakerroksia eri laitteissa tarvitaan?

Ks . myös kurssikirja ss. 94-96.



TKT:n luokkakoneelta maailmalle

TKTL:n oma verkko? (osa HYn verkko)

Helsingin yliopiston verkko

<http://www.helsinki.fi/atk/yhteydet/images/>

FUNET

http://www.csc.fi/hallinto/funet/esittely/runkoverkko/index_html

NORDUNET

http://www.nordu.net./maps/map_nordunet.png

<http://stats.nordu.net/stat-q/load-map/ndn-map,,traffic,peak>

(Kotimaan liikenne FUNETista FICIXin kautta eteenpäin)



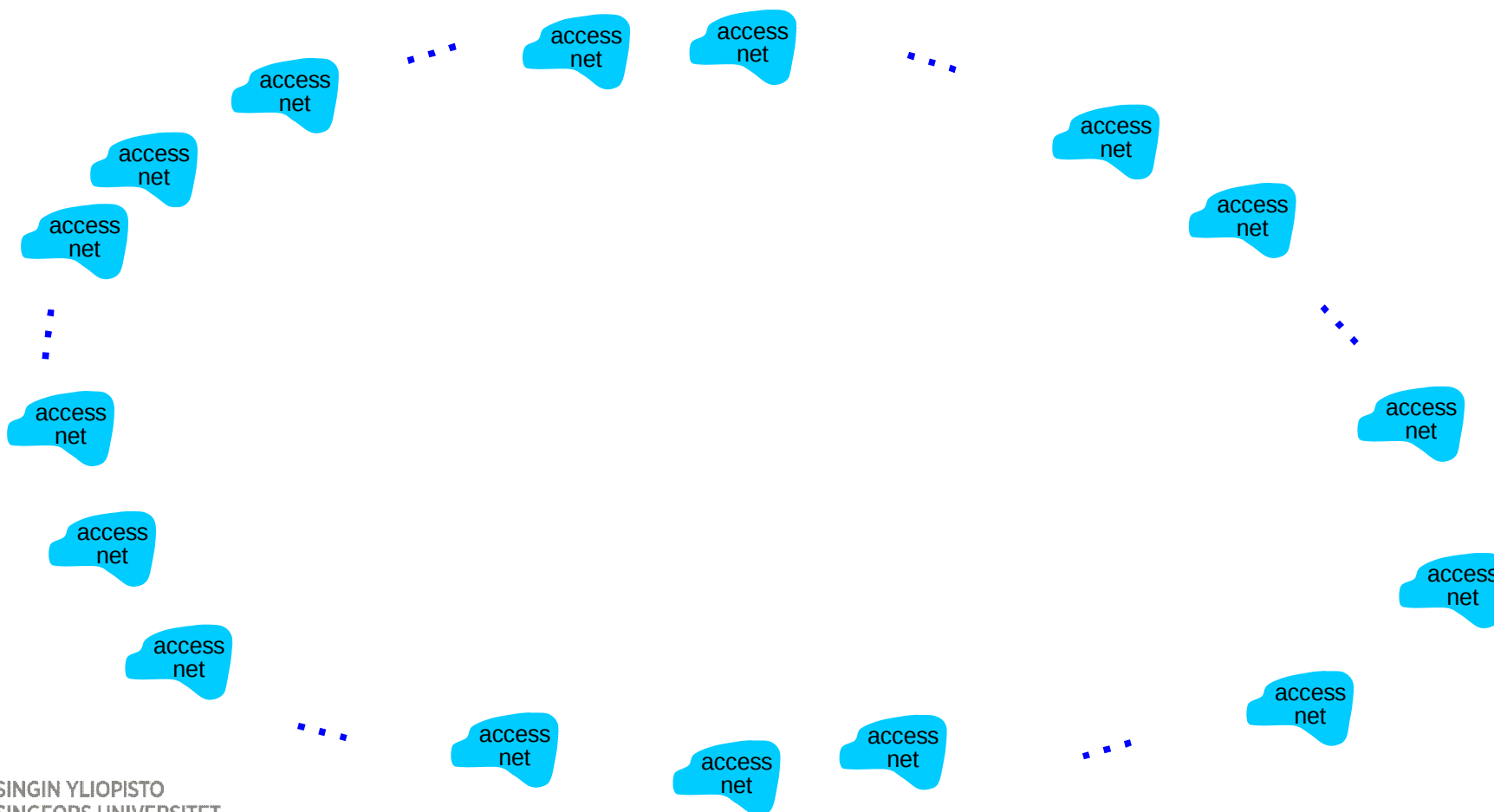
Internetin rakenne: verkkojen verkko

- ❖ Päätelaitte on kytketty palveluntarjoajan (Internet Service Providers, ISP) verkon kautta Internetiin
- ❖ Palveluntarjoajien verkot on kytketty edelleen toisiinsa.
 - ❖ Näin kaksi Internetiin kytkettyä laitetta voi lähettää paketteja toisilleen
- ❖ Tällainen verkkojen verkko on monimutkainen
 - ❖ Taustalla sekä talous että kansalliset politiikat
- ❖ Katsotaan rakennetta askel askeleelta



Internetin rakenne: verkkojen verkko

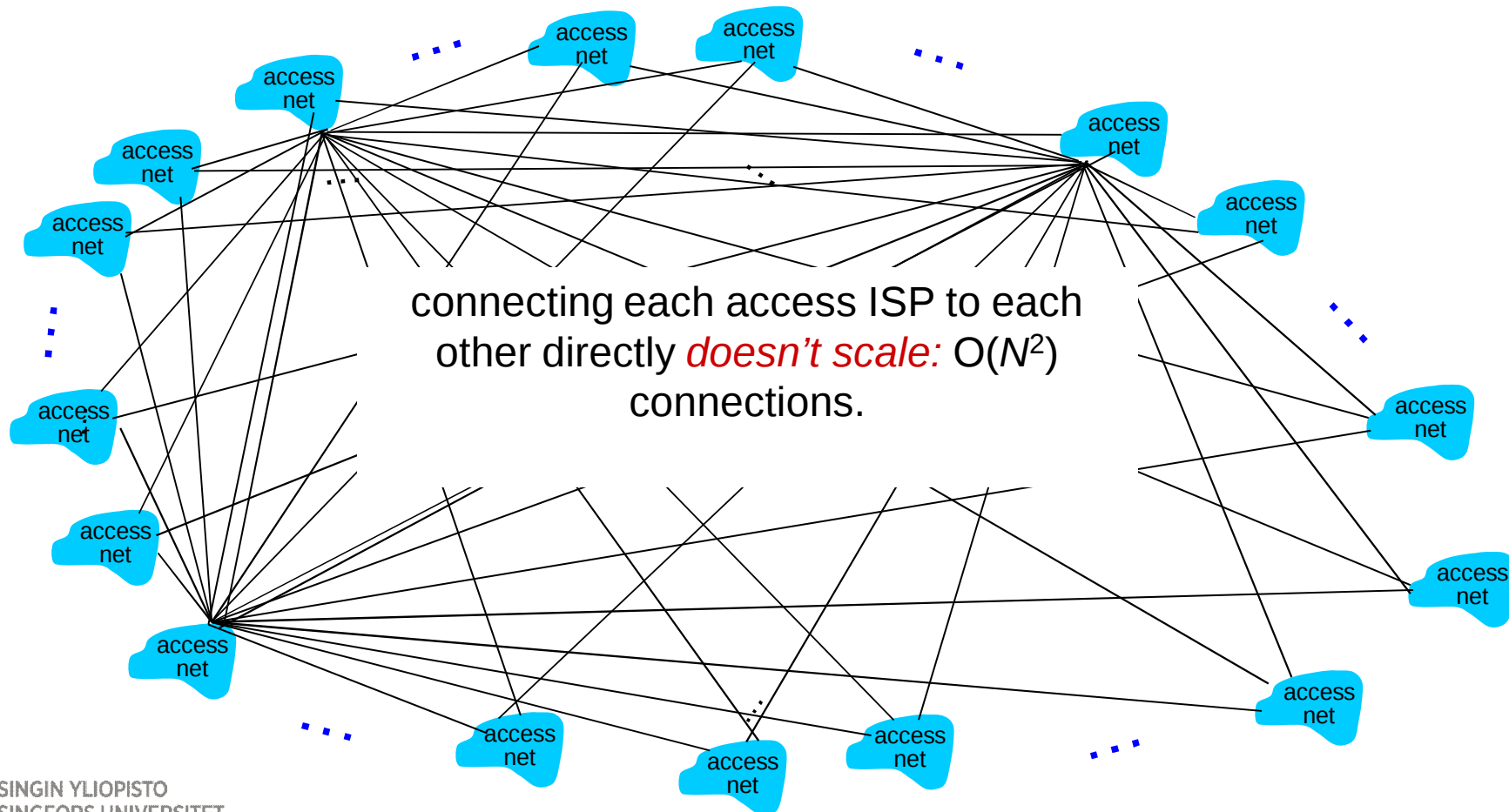
Ongelma: Miten miljoonat palveluntarjoajien verkot yhdistetään?





Internetin rakenne: verkkojen verkko

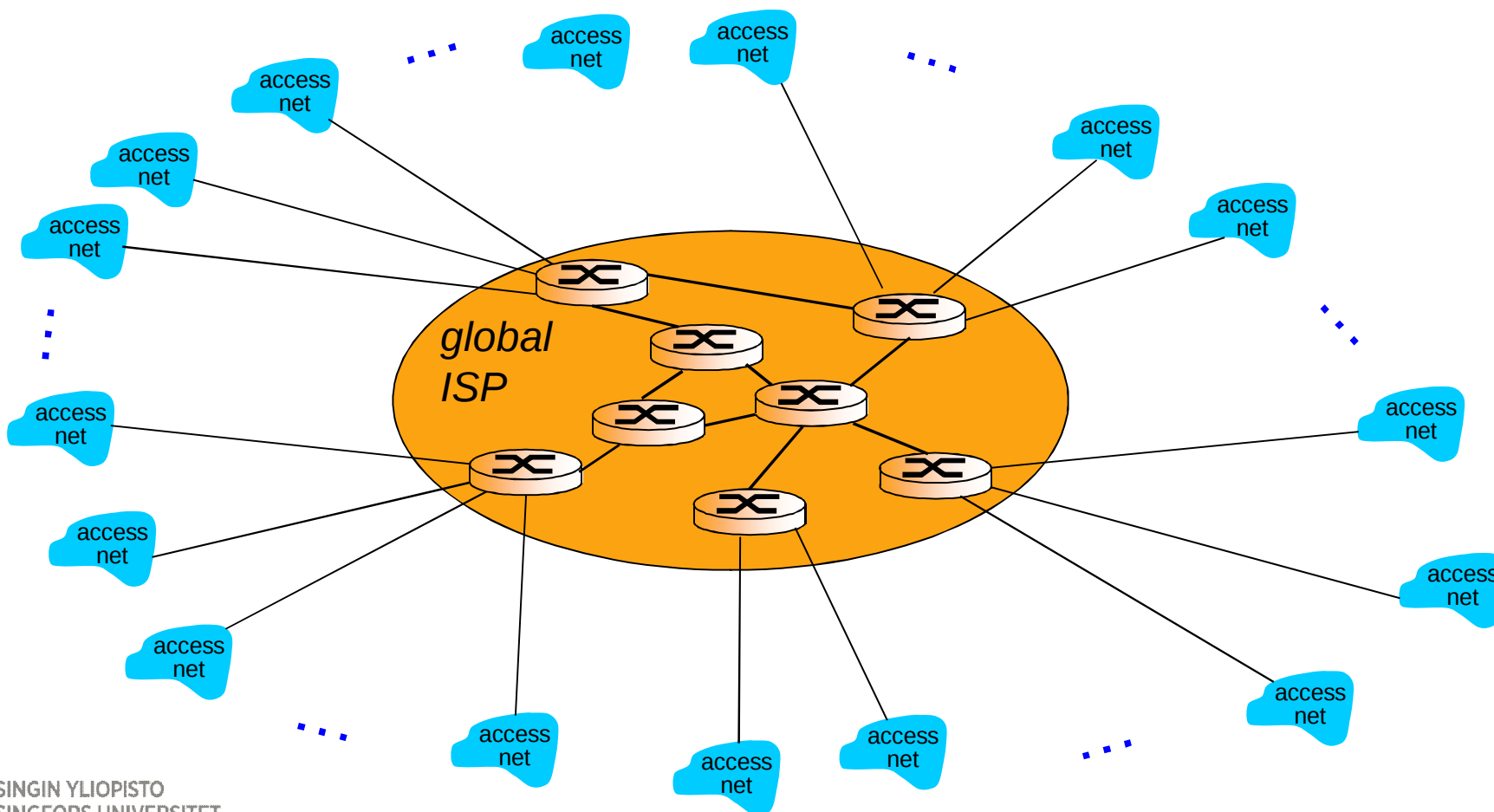
Optio: yhdistetään jokainen ISP suoraan kaikkiin muihin?





Internetin rakenne: verkkojen verkko

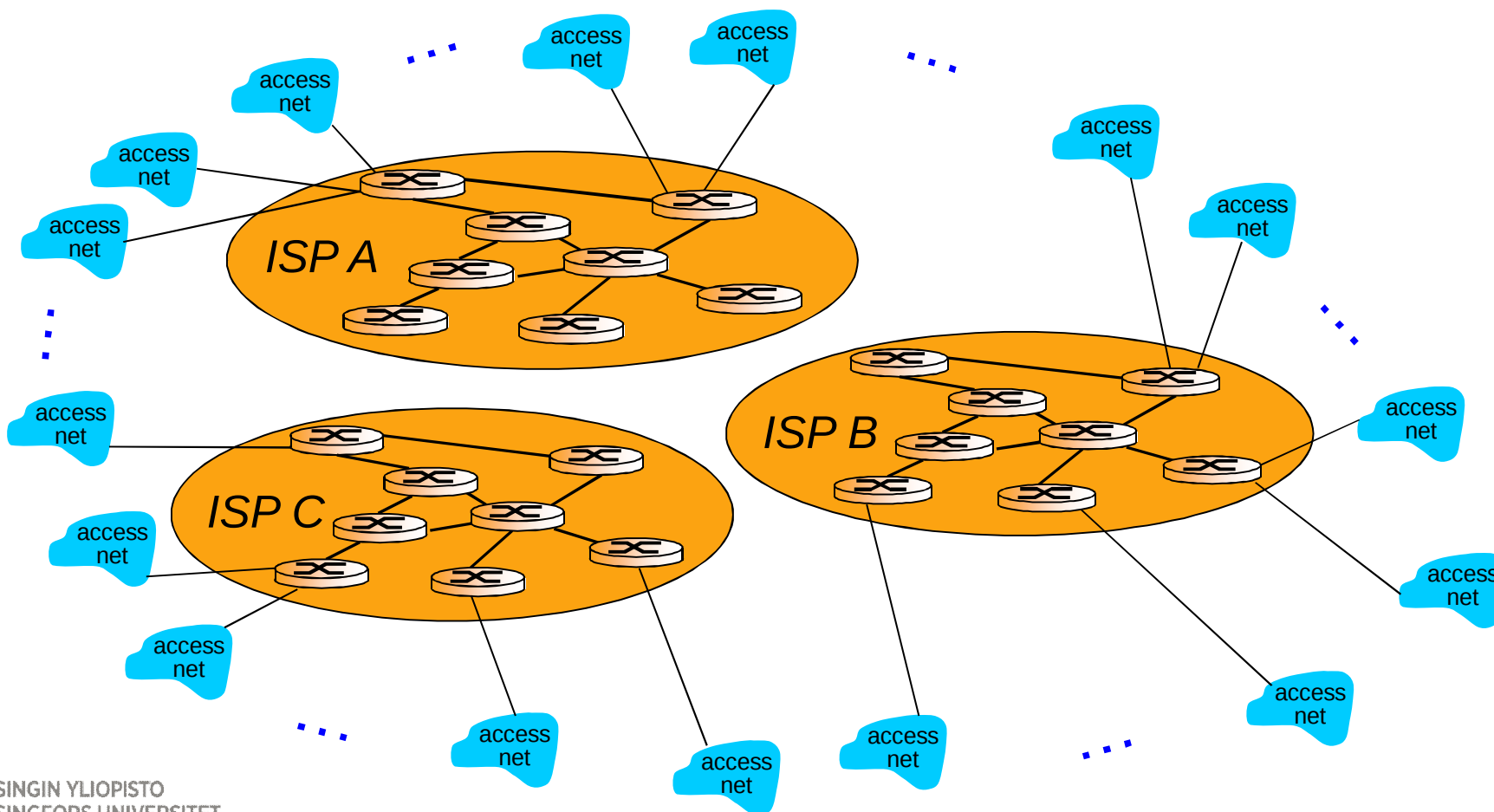
Optio: yhdistetään kaikki yhteen keskitettyyn välitysverkkoon?





Internetin rakenne: verkkojen verkko

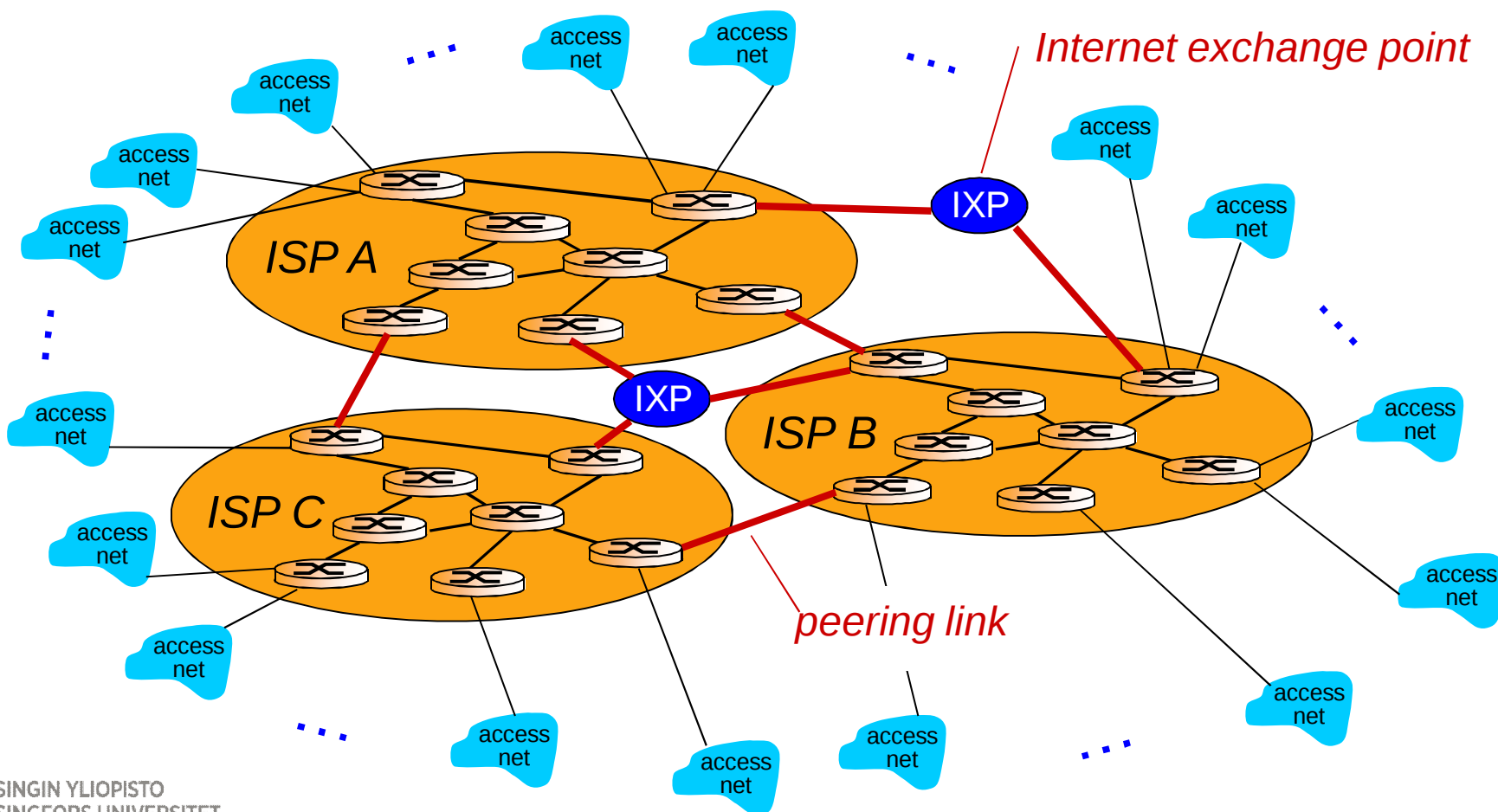
Mutta: jos välitysverkko tuottaa voittoa, tulee kilpailijoita ...





Internetin rakenne: verkkojen verkko

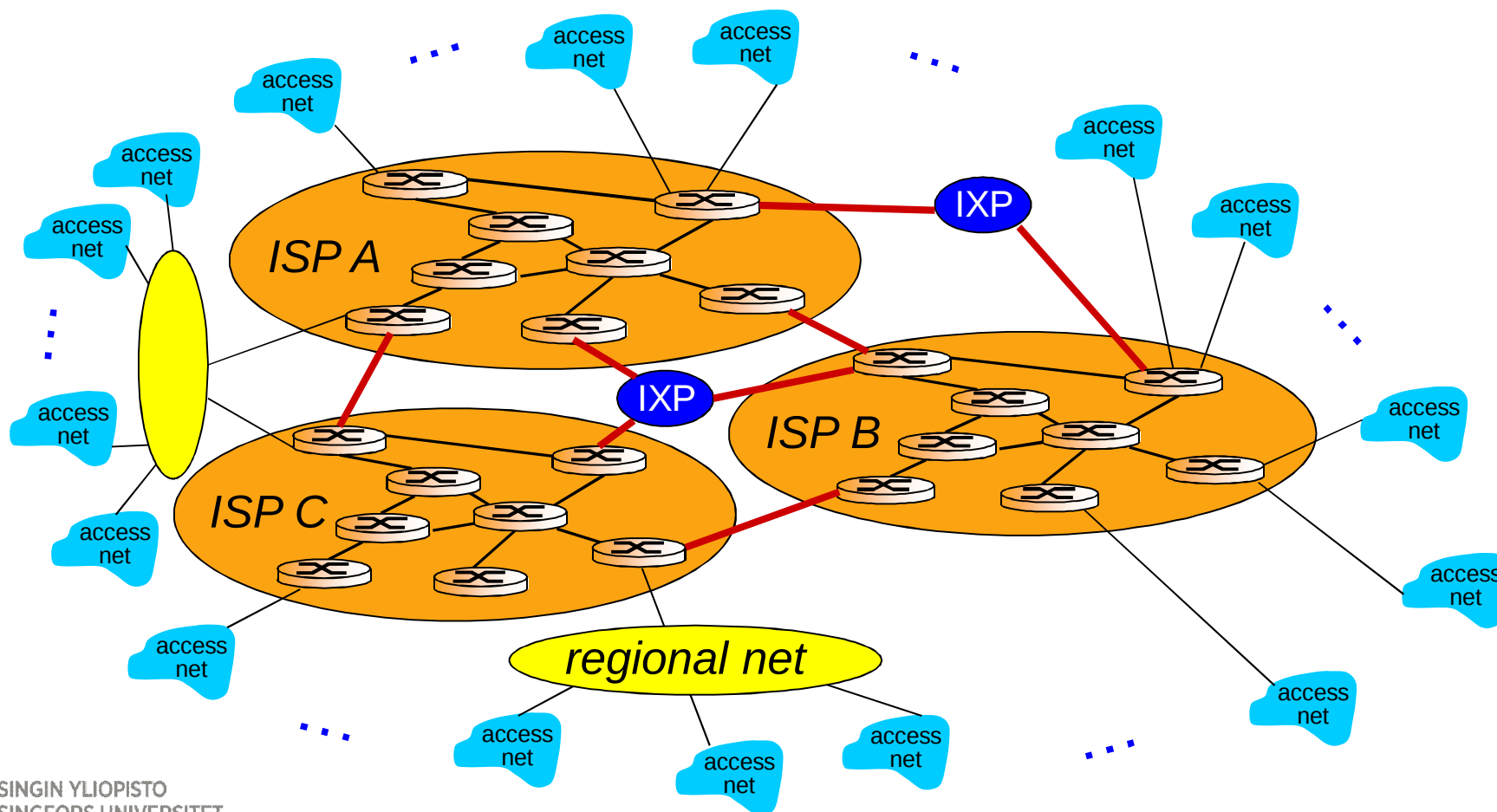
kilpailijoita... jotka pitää yhdistää toisiinsa





Internetin rakenne: verkkojen verkko

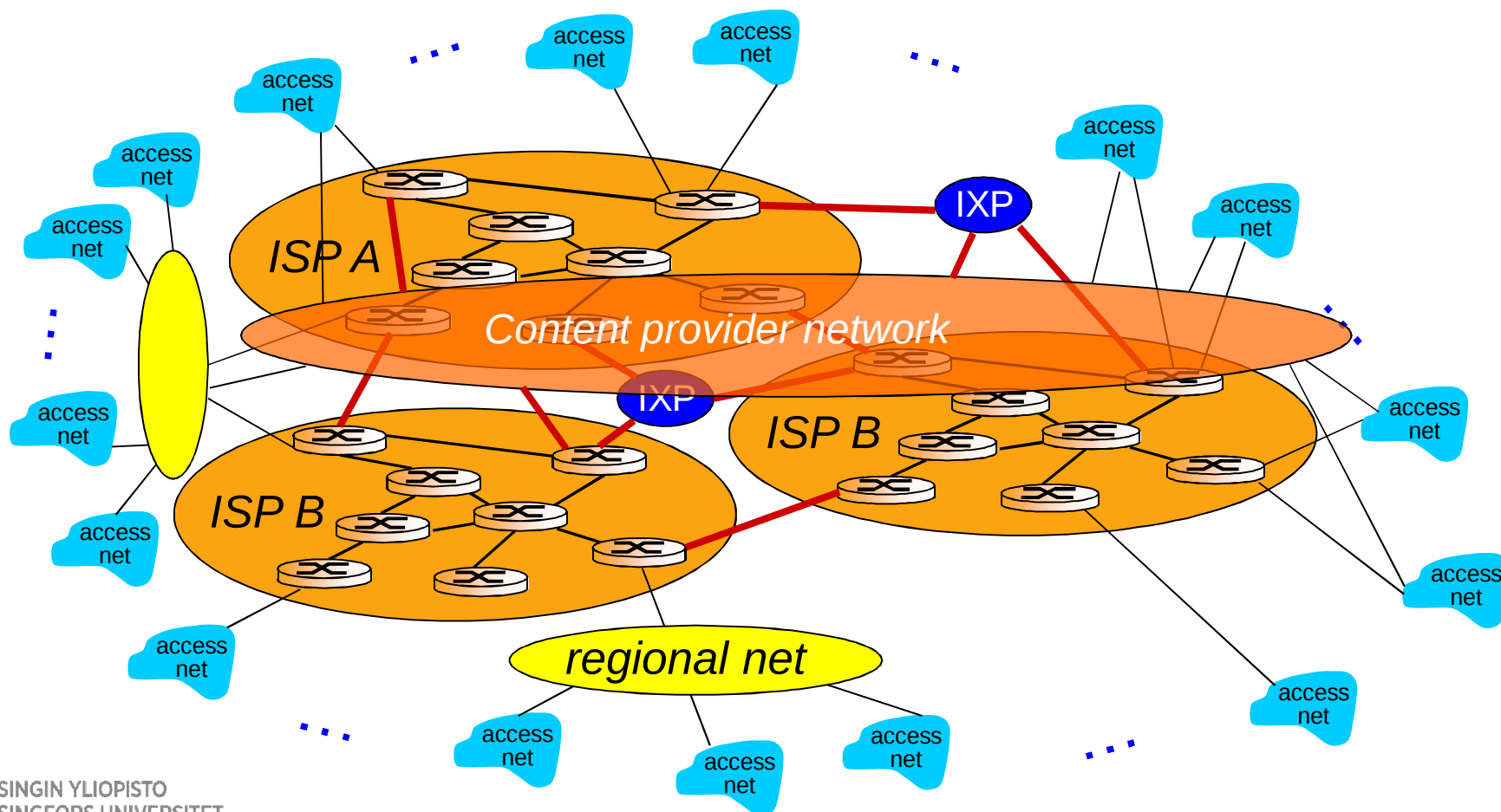
... ja alueverkkoja, joiden kautta yhteydet palveluntarjoajiin





Internetin rakenne: verkkojen verkko

... ja sisältötuottajien (e.g., Google, Microsoft, Akamai) omia verkkoja





Internetin rakenne: verkkojen verkko

Fig 1.15 [KR12]

sisimpänä: pieni joukko
maailmanlaajuisesti hyvin
kytkeytyneitä verkkoja

“tier-1” kaupalliset ISPt

(esim. Sprint, AT&T, NTT),
kansainvälisiä, hyvä kattavuus
(coverage)

**Sisällöntarjoajan verkko
(content provider network)**

(esim. Google): yksityinen
verkko, joka yhdistää
palvelinkeskuksia Internetiin,
usein ohittaa kokonaan tai
osittain tier-1 ISPt

