



Luento 9: Linkkikerros

26.11.2012

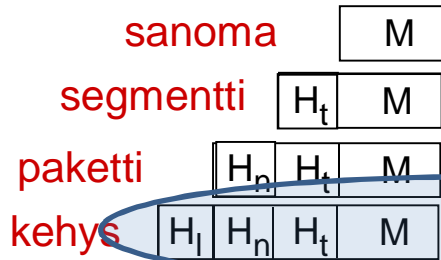
Tiina Niklander

Kurose&Ross
Ch5.1-5.4 ja 5.7

Pääasiallisesti kuvien
© J.F Kurose and K.W. Ross,
All Rights Reserved

Luennon sisältöä

Lähettäjä (sender)



message,
segment
datagram
frame

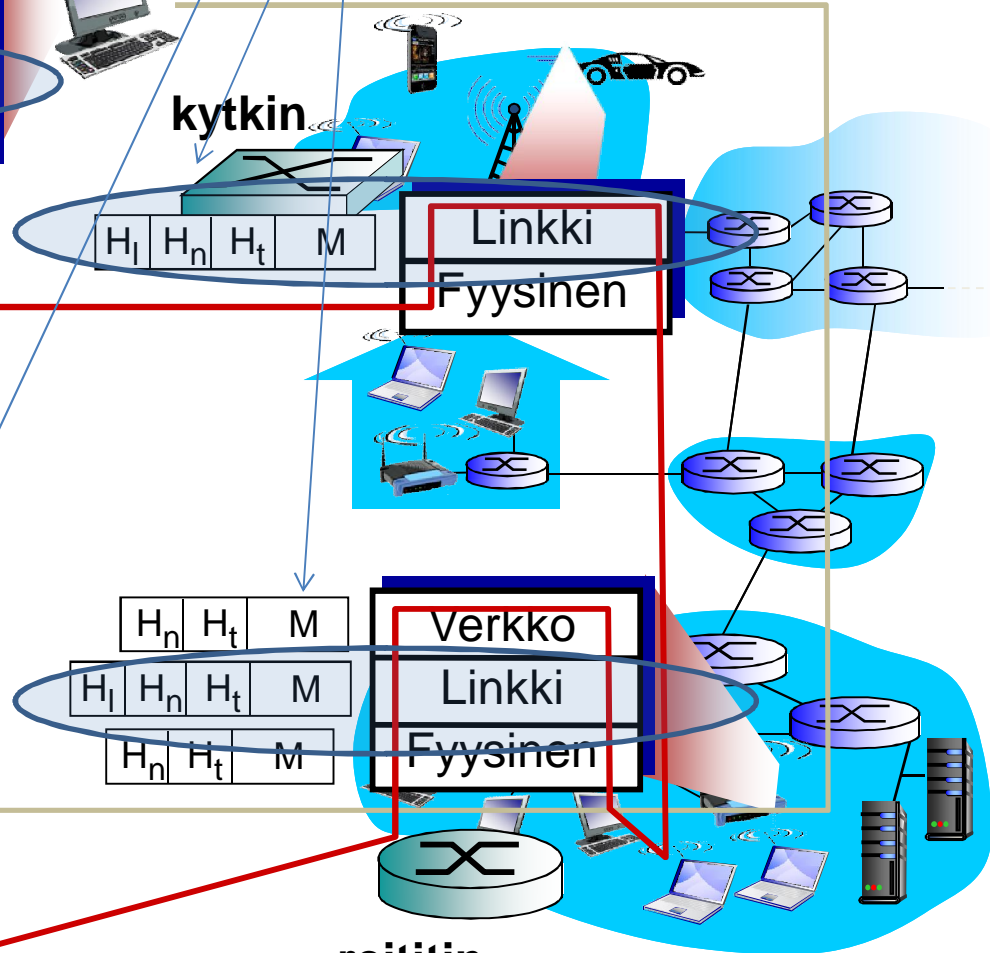
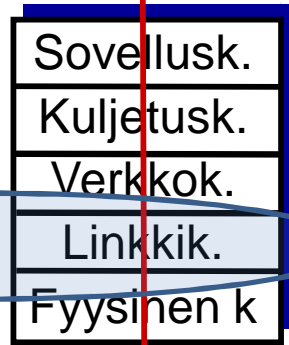
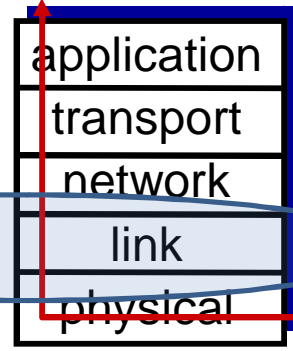
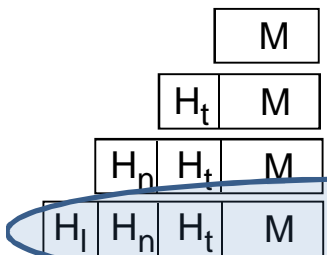
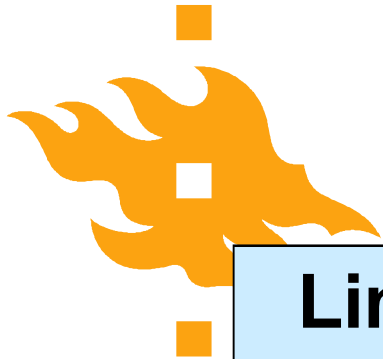


Fig 1.24 [KR12]

Vastaanottaja (recipient)





Sisältö

Linkkikerroksen tehtävät
Virheiden havaitseminen ja korjaaminen
Yhteiskäyttöisen kanavan varaus
Osoittaminen linkkikerroksella
Ethernet



Oppimistavoitteet:

- Osata selittää linkkikerroksen toiminnallisuus (MAC-osoitteet, bittivirheiden havaitseminen) ja ARP-protokollan käyttö.
- Osata selittää yhteiskäyttöisen siirtokanavan varaus ja käyttö
- Osata selittää, kuinka koneita voi yhdistellä lähiverkoiksi
- Osata selittää reitittimen, kytkimen ja keskittimen erot



Linkkikerros

Linkkikerroksen tehtävät

Ch 5.1



Linkkikerros

Laitetoimintoa

Siirtää paketin fyysistä linkkiä pitkin koneelta (solmulta (node)) toiselle

langallinen / langaton

bitit sisään, bitit ulos

Kapseloi paketin siirtoon sopivaan muotoon

Siirtokehys (frame)

Lähiverkossa linkkejä voi yhdistää keskittimillä tai kytkimillä

Käytetään fyysisiä osoitteita

'reititystä' ilman IP-osoitteita

Fig 5.1 [KR12]

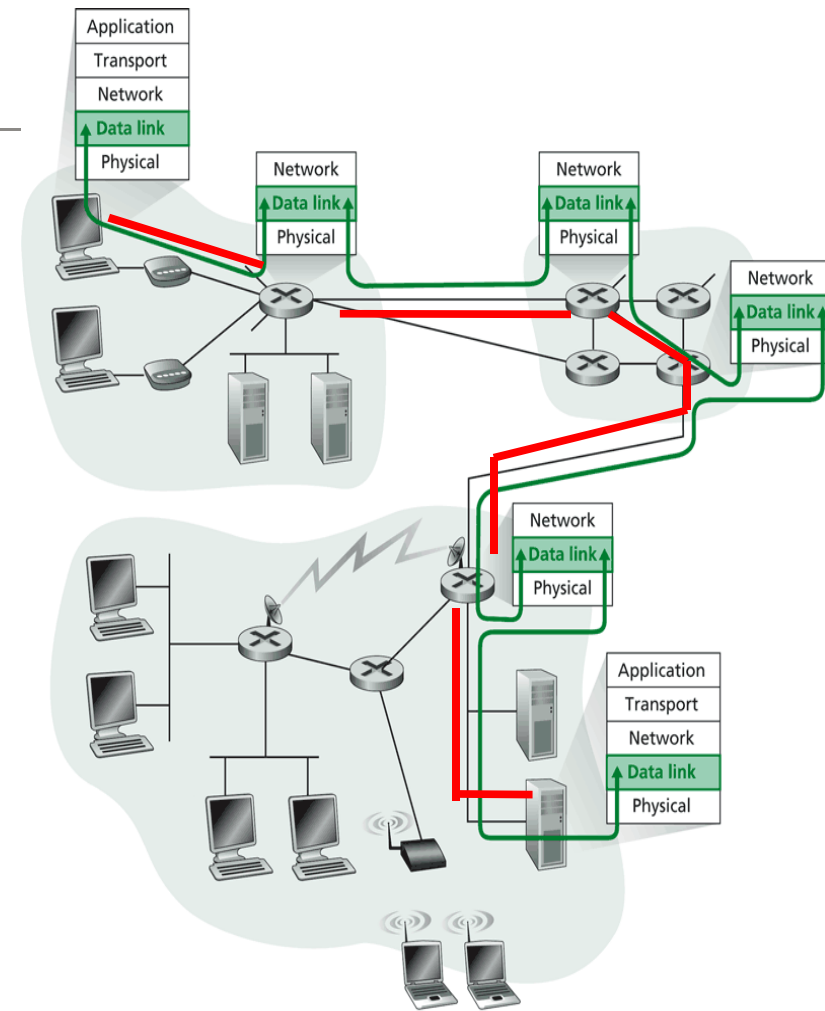


Figure 5.1 ♦ The link layer



Linkkikerroksen tehtäviä

NIC (Network Interface Card)
linkki- ja fyysinen kerros

Vuonvalvonta, puskurointi

Kytkimessä on useita erinopeuksisia linkkejä

Yksisuuntainen /kaksisuuntainen liikenne

Yksisuuntainen: lähetysvuorojen hallinta

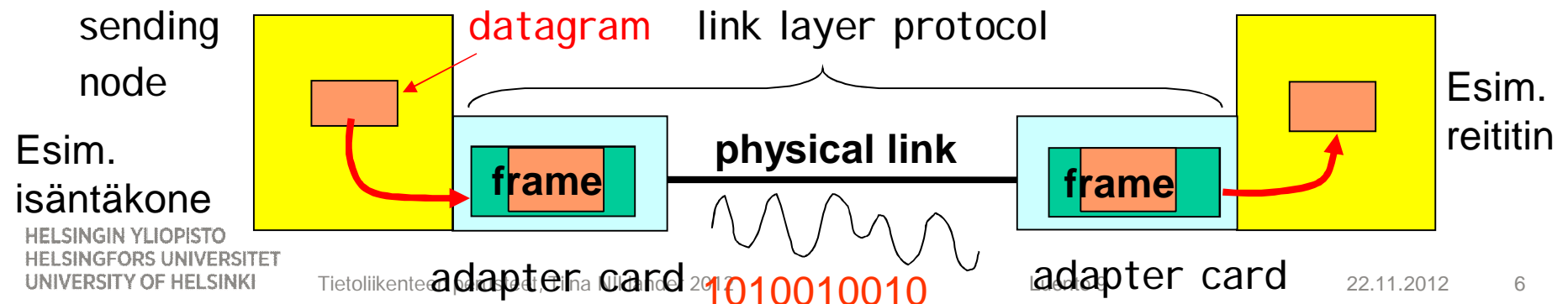
Virhevalvonta

signaali vaimenee, taustakohina häiritsee, ...

Kehyksessä on tarkistustietoa (error detection and correction bits)

Vastaanottava solmu korjaa, jos pystyy

Jos ei pysty, pyytää uudelleen tai hävittää





Linkkikerroksen tehtäviä

otsake

data

lopuke

Kehystys (framing)

Kehyksen rakenne ja koko riippuu siitä, millainen linkki on kyseessä

Otsake, data, lopuke

Kohteen ja lähteen osoittaminen

Yhteiseen linkkiin voi olla liitettynä useita laitteita

Käytössä laitetaso MAC-osoite (Medium access control)

Yhteisen linkin varaus ja käyttö (link access)

Esim. langaton linkki, keskittimiin yhdistetyt linkit

Luotettava siirto

Langattomilla linkeillä suuri virhetodennäköisyys

Linkkitaso huolehtii oikeellisuudesta

Miksi tästä täytyy huolehtia vielä kuljetuskerroksella?

Jotkut linkkityypit eivät huolehdi lainkaan!

Jos kehys hävitettävä ..

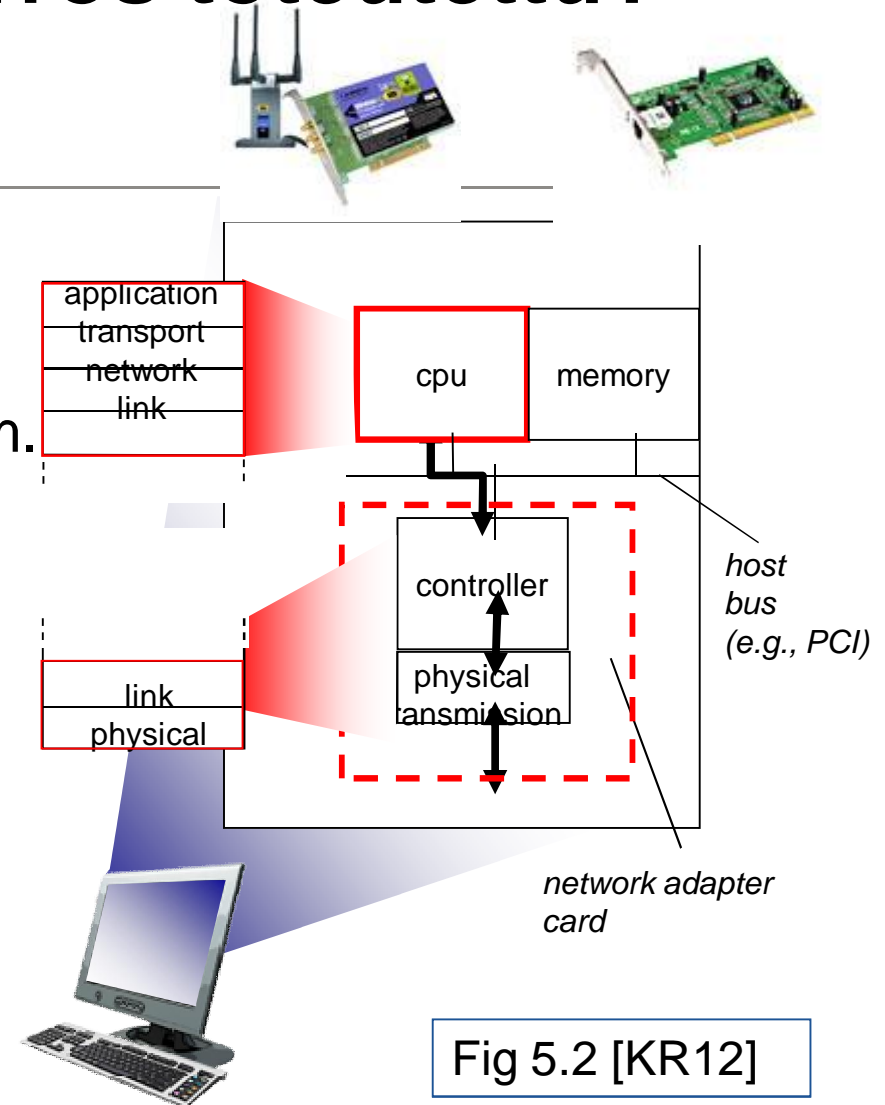


Miten linkkikerros toteutettu?

Jokaisessa koneessa!
Linkkikerroksen toteutus
usein “sovittimessa” (esim.
verkkokortti)

Linkki ja fyysinen kerros
joka liitetään koneen
väylään

Yhdistelmä laitteistoa,
ohjelmistoa ja laitteisto-
ohjelmisto (firmware)





Linkkikerros

Virheiden havaitseminen ja korjaaminen

Ch 5.2



Bittitason virheet

- *Yhden bitin virheitä siellä täällä tai peräkkäisten bittien virheryöppyjä (burst)*
- Virheiden esiintymistiheys BER (bit error rate)
 - Mitä suurempi tiheys, sitä lyhyempiä kehyksiä käyttöön
- Havaitsemiseen lisäbittejä
 - Feedback/backward error control
 - Tietoliikenteessä yleensä: hylkää virh. ja uudelleenlähetys
- Korjaamiseen enemmän lisäbittejä
 - Forward error correction (FEC) (esim. Hamming-koodi)
 - Esim. CD, DVD, Blu-Ray, viivakoodit, satelliityhteydet, digitelevisio, ... (Reed-Salomon-koodi)



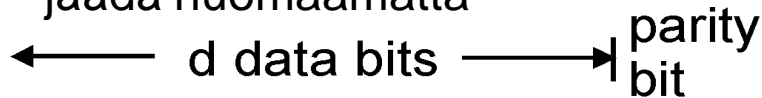
Pariteettitarkistus

Pariteettibitti

Parillinen vs. pariton pariteetti

Virheryöpyssä jopa 50% voi

jäädä huomaamatta



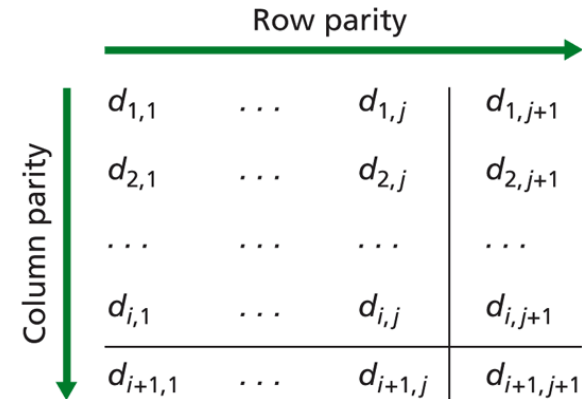
Kaksiulotteinen pariteetti

Erikseen horisontaalinen (parillinen)

ja vertikaalinen (pariton) pariteetti

Pystyy korjaamaan yhden bitin virheen.

Hamming-koodi (*kts. TiTo*)



No errors

1	0	1	0	1	1
1	1	1	1	0	0
0	1	1	1	0	1
0	0	1	0	1	0

Correctable single-bit error

1	0	1	0	1	1
1	0	1	1	0	0
0	1	1	1	0	1
0	0	1	0	1	0

Parity error →

Parity error ↓

Fig 5.5 [KR12]

Two-dimensional even parity



Tarkistussumma

Internet-checksum

Yhteenlasketaan 16 bitin kokonaisuuksia, yhden komplementti
Kuljetuskerros laskee ja tarkastaa UDP- ja TCP-protokollissa
Huom. IP sekä UDP/TCP ja UDP optionaalinen
Ei kovin tehokas; linkkikerros ei käytä

CRC (cyclic redundancy check)

Linkkikerroksella paljon käytetty virheenpaljastusmenetelmä,
helppo toteuttaa laitteistotasolla, luotettava
Perustuu polynomien aritmetiikkaan
tunnetaan myös nimellä polynomikoodi (polynomial code)
Useita tarkistusbittejä; havaitsee usean bittivirheen ryöpyn.



CRC

cyclic redundancy check

Käsittelee databittejä yhtenä kokonaislukuna

Sovittu **virittäjäpolynomi** G

bittejä yksi enemmän kuin lisättäviä tarkistusbittejä ($=r$ kpl) eli $r+1$

Lähettäjä

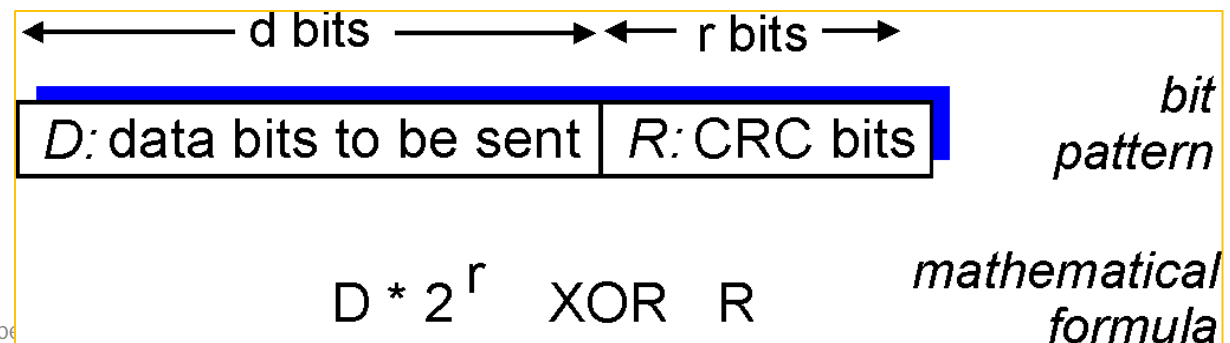
Asettaa tarkistusbitit R s.e. datan bitit ($=D$) + niiden perään liitetyt tarkistusbitit ovat jaollisia virittäjällä G (**modulo 2-aritmetiikka**)

Vastaanottaja

Jakaa samoin saamansa bittijonon ($D+R$) virittäjällä G .

Jos **jakojäännös $\neq 0$** , niin on virhe.

Fig 5.6 [KR12]





CRC esimerkki

Fig 5.7 [KR12]

G pituus 4b



G D r = 3

Tavoite (virheetön kun):

$$D \cdot 2^r \text{ XOR } R = nG$$

Ekvivalenttimuoto:

$$D \cdot 2^r = nG \text{ XOR } R$$

Edelleen sama:

Kun $D \cdot 2^r$ jaetaan G:llä, täytyy jäännöksen R olla:

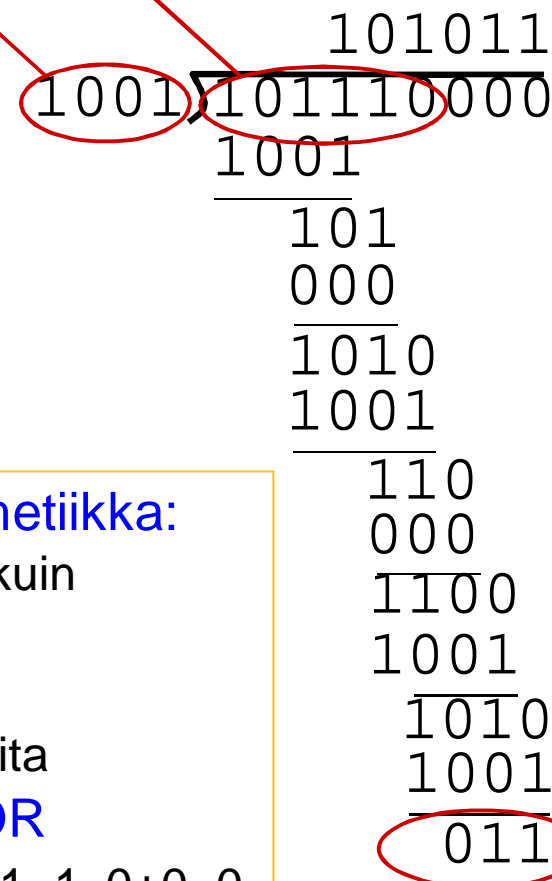
$$R = \text{remainder} \left[\frac{D \cdot 2^r}{G} \right]$$

Modulo 2-aritmetiikka:

vähennyslasku kuin yhteenlasku
ei lainaamista,
ei muistinumeroita

= bittitason XOR

$$1+1=0, 1+0=0+1=1, 0+0=0$$





Standardoituja virittäjäpolynomeja IEEE

$$G_{\text{CRC-12}} = x^{12} + x^{11} + x^3 + x^2 + x + 1$$

$$G_{\text{CRC-16}} = x^{16} + x^{15} + x^2 + 1$$

$$G_{\text{CRC-32}} = x^{32} + x^{26} + x^{23} + \dots + x^4 + x^2 + x + 1$$

$$= 1\ 0000\ 0100\ 1100\ 0001\ 0001\ 1101\ 1011\ 0111$$

($r+1=33$ bittiä)

Virittäjäpolynomin merkitsevin bitti aina =1

Havaitsee

kaikki virheryöpyt, joiden pituus < tai = kuin virittäjän pituus

lähes kaikki virheryöpyt, joiden pituus on suurempi



CRC polynomin määrittäminen

Polynomi kuvaa virheiden jakautumista

Maksimoi virheiden havaitseminen

Minimoi ylimääräinen tarkistus

CRC hyödyllinen koska se on tehokas toteuttaa ja havaitsee myös purskevirheet (jotka eivät ole tasajakautuneita)

Ei sovi tietoturvakäyttöön

Pariteettibitti voidaan nähdä triviaalina 1-bitin CRC:nä

Lisätietoja:

http://en.wikipedia.org/wiki/Mathematics_of_CRC



Linkkikerros

Yhteiskäyttöinen kanava

Ch 5.3



Yksi kanava: jaettu vai 'oma'?

Kaksipisteyhteys (point-to-point)

PPP-protokolla, puhelinyhteys (dial-up access)

Ethernet-piuha kytkimen ja isäntäkoneen välissä

Yleislähetysyhteys (broadcast)

Alkuperäinen Ethernet, Ethernet keskittimen ja

isäntäkoneen välissä, kaapelimodeemiyhteys (upstream),

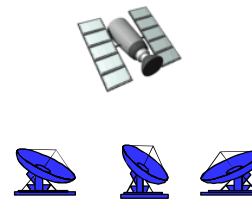
WLAN, satelliitti,



shared wire (e.g.,
cabled Ethernet)



shared RF
(e.g., 802.11 WiFi)



shared RF
(satellite)



humans at a
cocktail party
(shared air, acoustical)

Fig 5.8 [KR12]



Lähetysvuorojen jakelu

- Yksi yhteinen kanava lähettäjiille
 - Lähetys onnistuu vain, jos yksi kerrallaan lähettää
- Jos useampi lähettää yhtäaikaan, syntyy yhteentörmäys
 - Kaikki solmut saavat useita signaaleja, "bittimössöä"
 - Törmänneet sanomat tuhoutuvat ja ne on lähetettävä uudelleen
- Multiple Access Protocol
 - Tapa, jolla solmu päättää, voiko se lähettää
 - Kuinka solmun on toimittava törmäystilanteessa
 - Neuvottelu samassa kanavassa!





Multiple Access Protocol tavoitteet:

Pieni yleisrasite

Kun vain yksi lähettäjä, se pystyy hyödyntämään koko kanavan siirtonopeuden R bps

Tasapuolisuus

Kun M lähettäjä, kukin saa keskimäärin saman osuuden linjan siirtonopeudesta (R/M bps)

Toimintavarmuus

Yksikään solmu ei ole erikoisasemassa, koordinaattorina

Ei kellojen sykronointia tms

Hajautettu vuoroista sopiminen

Kustannustehokkuus

Yksinkertainen ja halpa toteuttaa



Lähetysvuorojen jakelu

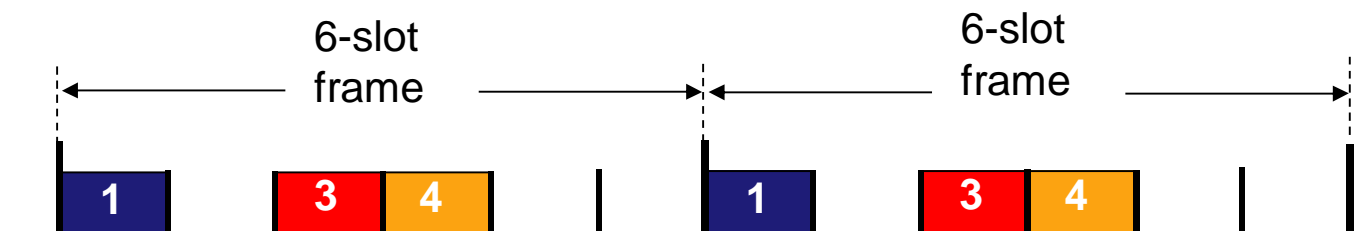
- 1) **Kanavanjakoprotokollat** (channel partitioning protocol)
Jaa kanavan käyttö 'viipaleisiin' (time slots, frequency, code)
Kukin solmu saa oman viipaleensa
TDMA, FDMA, CDMA
"Käytä sinä tätä puolta, minä tätä toista"
- 2) **Kilpailuprotokollat** (random access protocols)
"Se ottaa, joka ehtii."
Jos sattuu törmäys, yritä myöhemmin uudelleen.
Aloha, CSMA, CSMA/CD
- 3) **Vuoronantoprotokollat** (taking-turns protocols)
Jaa käyttövuorot jollakin sovitulla tavalla:
vuorokysely (polling), vuoromerkki, ...
"Minä ensin, sinä sitten."



Kanavajako: TDMA

time division multiple access

- Vuorotellen:
 - Vakiokokoinen aikaviipale (time slot) kullekin kanavaan kytketylle asemalle (station) kerran jaksossa
 - Aikaviipaleen pituus on yhden kehyksen lähetysaika
- Varattuna, vaikka ei lähetettävää
- Esim 6 solmua: viipaleissa 1,3,4 on paketteja ja viipaleissa 2,5,6 ei ole

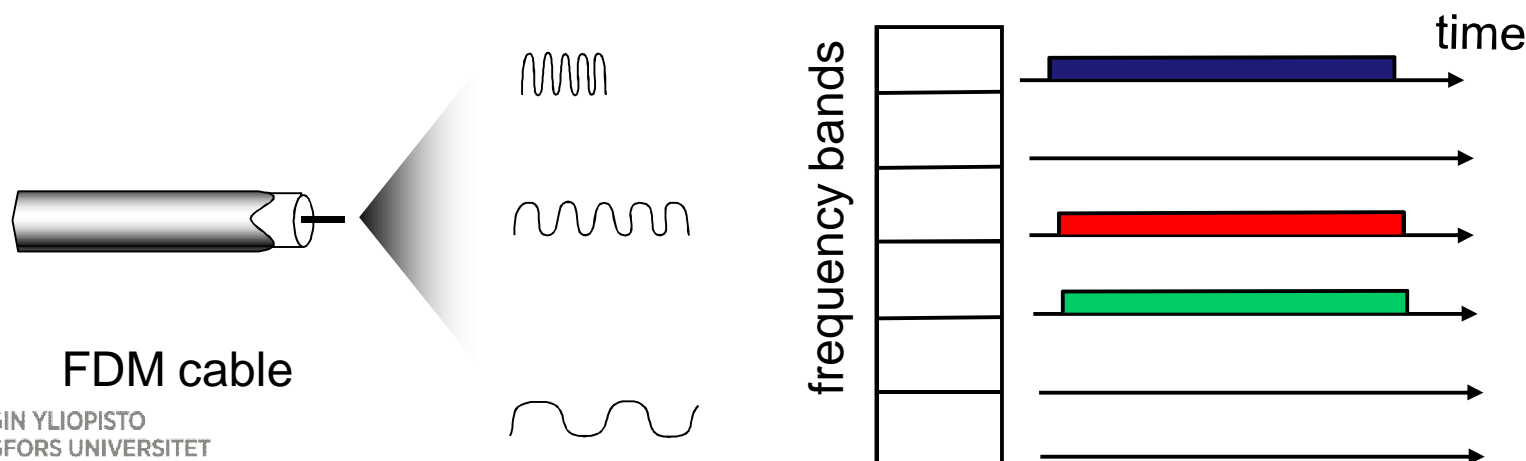




Kanavanjako: FDMA

frequency division multiple access

- Kanavan taajuusalueet jaettu kanavan käyttäjien (varaajien) kesken
 - Jokainen asema saa kiinteän taajuusalueen (fixed frequency band), vain osa kanavasta (**R/M bps**)
- Varattuna, vaikka ei lähetettävää
- Esim. alueilla 1,3,4 on paketteja, ja alueilla 2,5,6 ei





Kanavanjako: CDMA

Code Division Multiple Access

- Radiolinjoilla käytettävä koodinjakoon perustuva protokolla
 - Kullakin asemalla oma yksilöllinen tapansa koodata bitit 1 ja 0
- Asemat voivat lähettää yhtäaikaan koko kanavan taajuudella
 - Kaikkien signaalit saavat yhdistyä linkillä
 - Asemat pystyvät erottelemaan yhteissignaalista itselleen kuuluvat bitit (yksilöllinen koodaustapa)
 - Signaali levitetään laajalle spektrialueelle



Kilpailuprotokollat

(random access protocols)

Kun asema haluaa lähettää

Se kuuntelee ensin, onko joku muu asema jo lähettämässä

Jos ei, lähettää heti täydellä nopeudella

Jos kaksi aloittaa yhtäaikaan => törmäys

Odota satunnainen aika ja yritä uudestaan (random access)

Protokolla määrittää

Miten törmäys huomataan

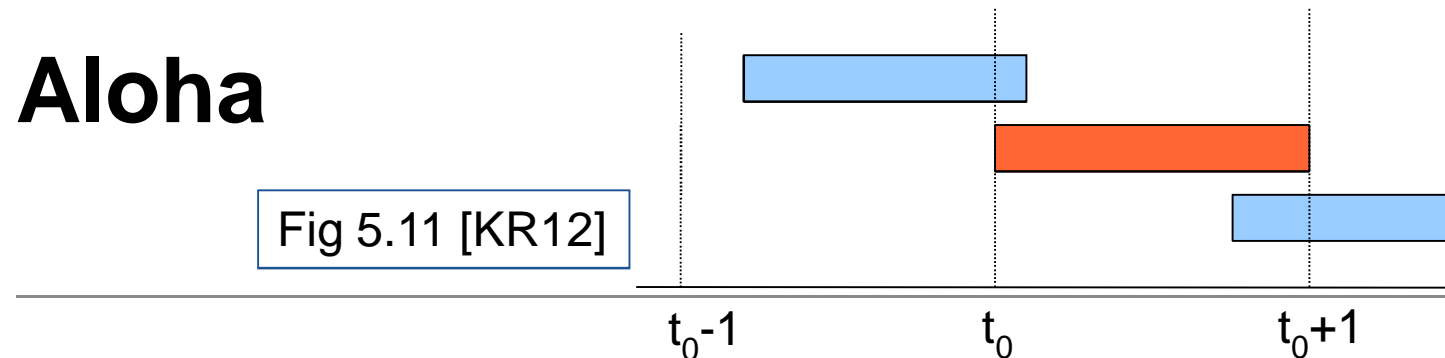
Miten törmäyksestä toivutaan

Esim. (viipaloitu)ALOHA, CSMA, CSMA/CD, CSMA/CA



Aloha

Fig 5.11 [KR12]



Kehitetty Hawaijilla, 70-luvulla radiotietä varten

Lähetä heti, kun on lähetettävää

Ei mitään kuuntelua ennen lähetystä

Kuuntele sitten, onnistuiko lähetys

Lähiverkossa törmäys havaitaan 'heti', sillä siirtoviive on pieni (toisin kuin satelliitilla)

Jos törmäys, niin odota satunnainen aika ja yritä uudelleen

Törmäyksen td. suuri

Max tehokkuus ~ 18%



Viipaloitu ALOHA

(slotted ALOHA)

Oletukset:

- ❖ Kaikki siirtokehukset samankokoiset
- ❖ Aikaviipale yhden kehyksen lähetysaika
- ❖ Solmu aloittaa lähetyksen aina aikaviipaleen alusta
- ❖ Solmut ja niiden kellot on synkronoitu
- ❖ Kaikki solmut havaitsevat yhteentörmäykset (collision)

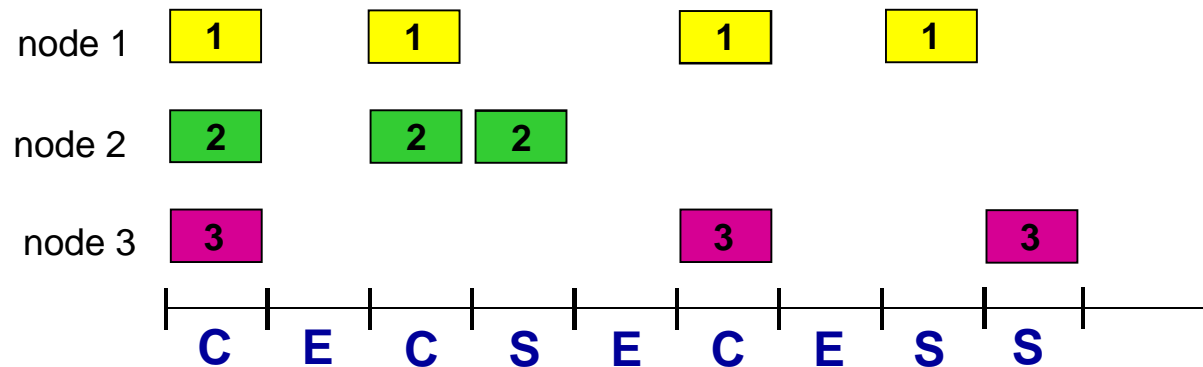
Toiminta:

- ❖ Valmis siirtokehys lähetetään heti seuraavassa aikaviipaleessa
 - *Ei yhteentörmäystä:* Solmu voi lähettää seuraavan kehyksen seuraavassa aikaviipaleessa
 - *Yhteentörmäys:* Solmu yrittää lähetystä uudelleen seuraavassa aikaviipaleessa todennäköisyydellä p. Yrittää niin kauan kunnes onnistuu.



Viipaloitu ALOHA (slotted ALOHA)

Fig 5.10 [KR12]



Suorituskyky kaksinkertaistuu (Alohaan verrattuna)

Jos paljon lähettäjiä **max. ~37 % tehokkuus**

Siis 37% tyhjiä, 37% onnistumisia, 26% törmäyksiä



Kilpailuprotokollat: Lähetyskanavan kuuntelu

Kuuntele ennen kuin lähetät

Asema tutkii, onko kanava jo käytössä (carrier sense)

Jos siirtotie on vapaa, saa lähettää

Jos siirtotie on varattu, odota satunnainen aika ja yritä uudelleen

CSMA (Carrier Sense Multiple Access)
Useita variaatioita

Ei aina paljasta jo alkanutta lähetystä

Aina huomaaminen ei ole mahdollista

Esim. satelliittikanavan kuuntelu ei paljasta, onko jokin muu maa-asema jo aloittanut lähetyksen

Langattomassa lähiverkossa lähettäjän ympäristön kuuntelu ei kerro, onko vastaanottaja saamassa sanomia muilta



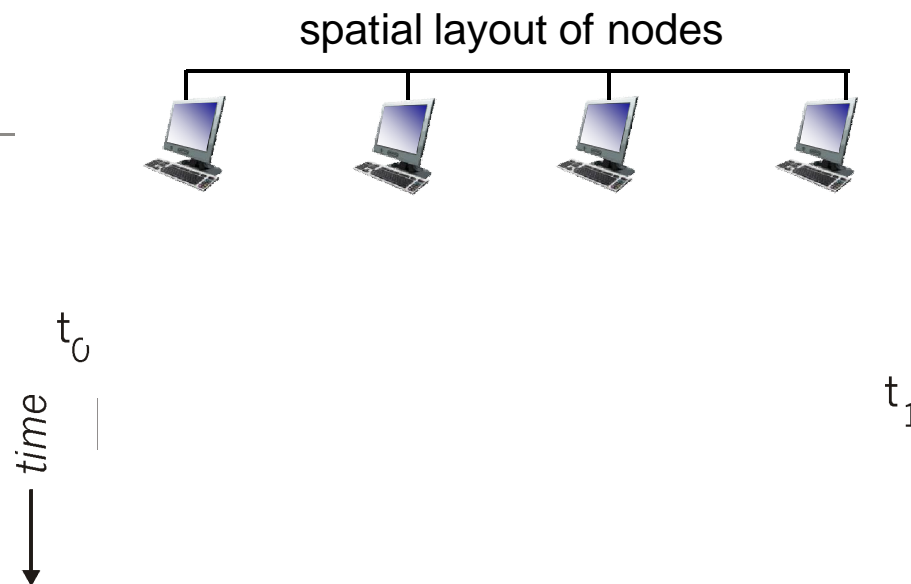
CSMA yhteentörmäys (collision)

Fig 5.12 [KR12]

Yhteentörmäyksiä voi tapahtua: Etenemisviiveen takia ei huomata toisen signaalia ajoissa

Yhteentörmäyksen seuraus: Paketin lähetys epäonnistuu ja lähetyisaika menee hukkaan

Solmujen etäisyydet ja etenemisviiveet vaikuttavat yhteentörmäysten todennäköisyyteen





CSMA/CD

(Carrier Sense Multiple Access with Collision Detection)

Asema kuuntelee myös lähettämisen jälkeen

Langallinen LAN: törmäys => signaalin voimakkuus muuttuu

– Esim. Ethernet

Langaton LAN: hankalaa

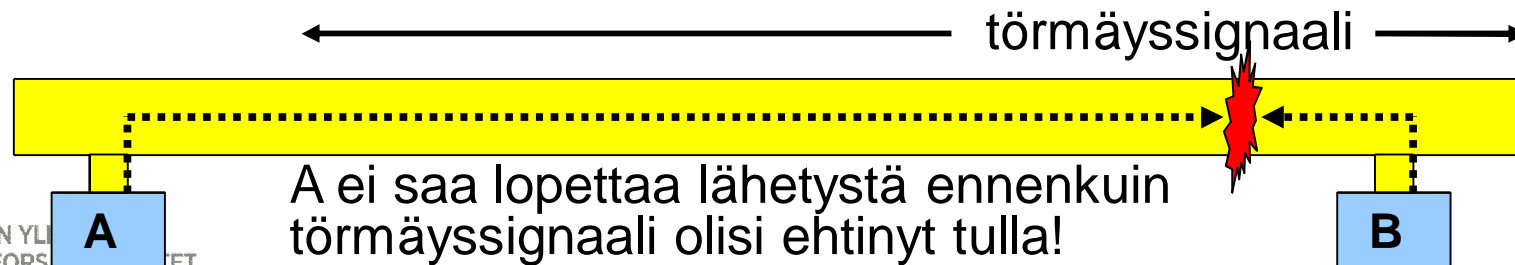
Jos törmäys, keskeytä lähettäminen **heti**

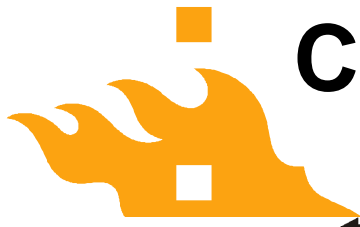
ja yritä uudestaan satunnaisen ajan kuluttua

Näin törmäyksen aiheuttama hukka-aika pienenee

Kauanko kuunneltava?

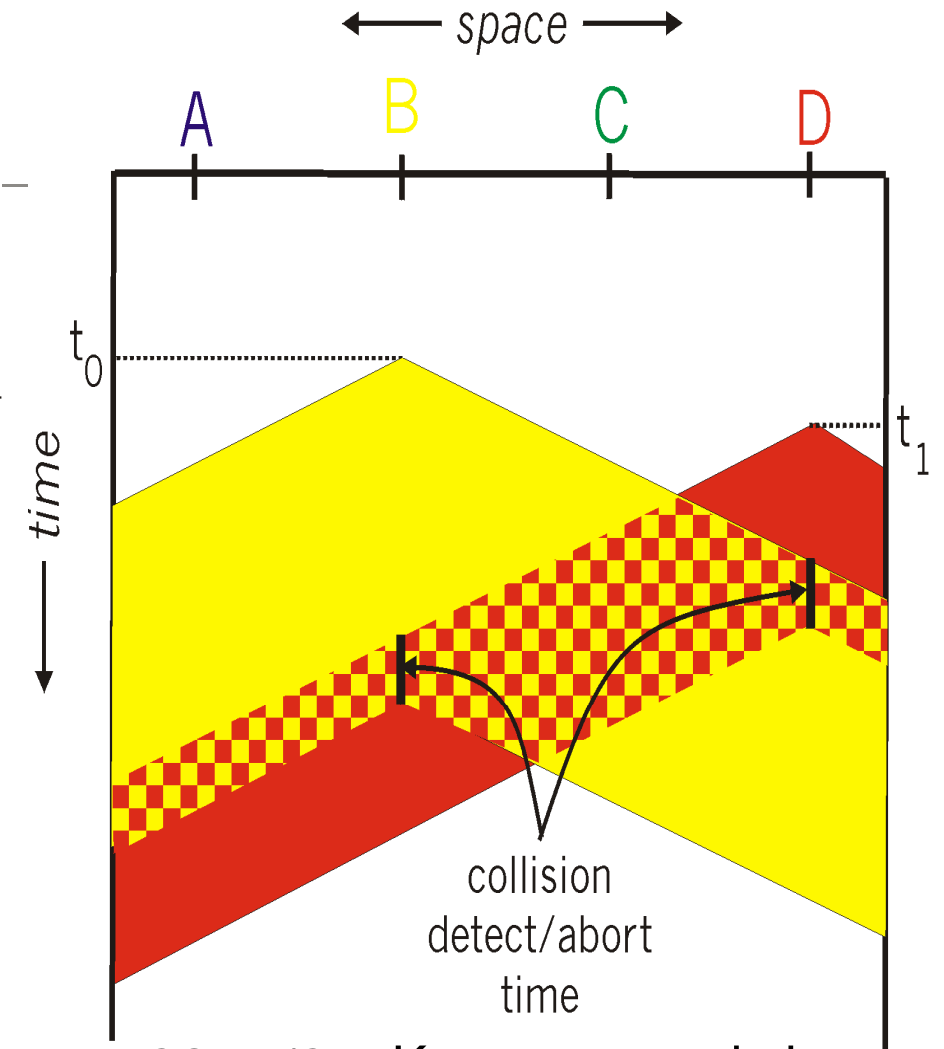
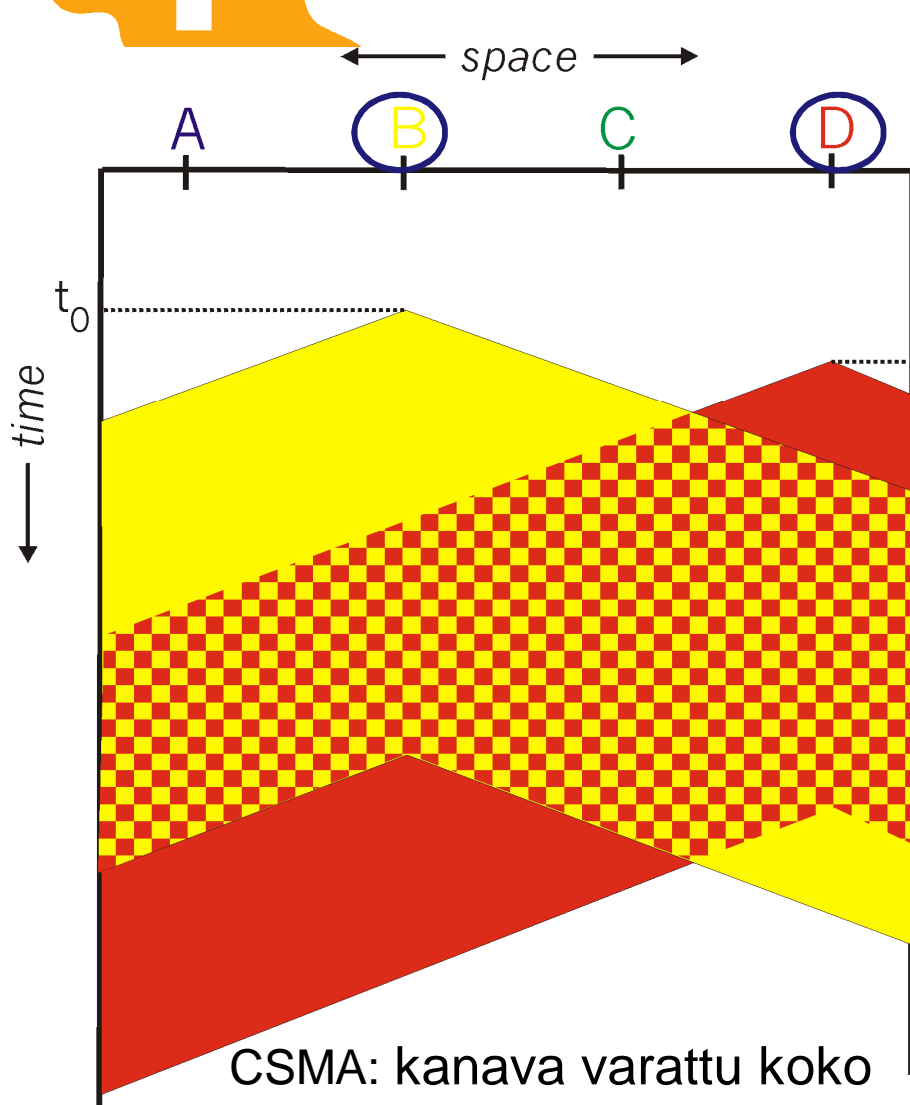
2* maksimi etenemisviive solmujen välillä





CSMA ja törmäys

Fig 5.12 ja 5.13 [KR12]





Vuoronantoprotokollat

Yhdistä edellisten parhaita puolia

Älä pidä kapasiteettia turhaan varattuna

Älä aiheuta törmäystä

Vuorokysely, pollaus

Isäntäasema kyselee vuorotellen jokaiselta asemalta, onko sillä lähetettävää (polling)

Isäntä kuuntelee signaalia, osaa päätellä, milloin lähetys loppuu

Vuoromerkki (token)

Se, jolla on vuoromerkki, saa lähettää

Jos ei ole lähetettävää, niin vuoromerkki siirtyy seuraavalle

Kummastakin useita versioita

Ongelmia: lisäviive, 'single point of failure', ..

Montako kehystä yhdessä vuorossa saa lähettää



Linkkikerros

Linkkikerroksen osoitteet

Ch 5.4



Linkkikerroksen fyysinen osoite

32 bitin IP-osoite verkkokerroksella

Reitityksen tapa viitata koneeseen

Erilaisilla linkkikerroksilla omat tapansa osoittaa oikea linkki (~ verkkokortti)

Siirtokehys on kuljetettava fyysisen linkin yli jollekin toiselle samaan verkkoon (LAN) kytketyistä laitteista

MAC-osoite (Media Access Control Address)

Käytetään myös nimiä LAN-osoite, fyysinen osoite, laiteosoite, Ethernet-osoite, ...

Liitetty valmistusvaiheessa kiinteästi laitteeseen

Analogia:

IP-osoite ~ katuosoite

MAC-osoite ~ henkilötunnus



MAC-osoite

Lähes 300 biljoonaa erilaista osoitetta.

Lähes 17 miljoonaa valmistajanumeroa, kuhunkin mahdollista lähes 17 miljoonaa osoitetta.

MAC spoofing

48-bittinen (6 tavua)

24 b kertoo valmistajan ja 24 b identifioi ohjainkortin (adapter)

IEEE jakaa valmistajanumerot

Kiinteä - Liitetty mukaan valmistuksessa

Säilyy, vaikka laite toiseen verkkoon (toisin kuin IP-osoite)

Ohjain

Kuulee kaikki kanavalla kulkevat kehykset

Välittää omalle koneelle vain sen MAC-osoitteella tai yleislähetysosoitteella

FF-FF-FF-FF-FF-FF merkityt lähetykset

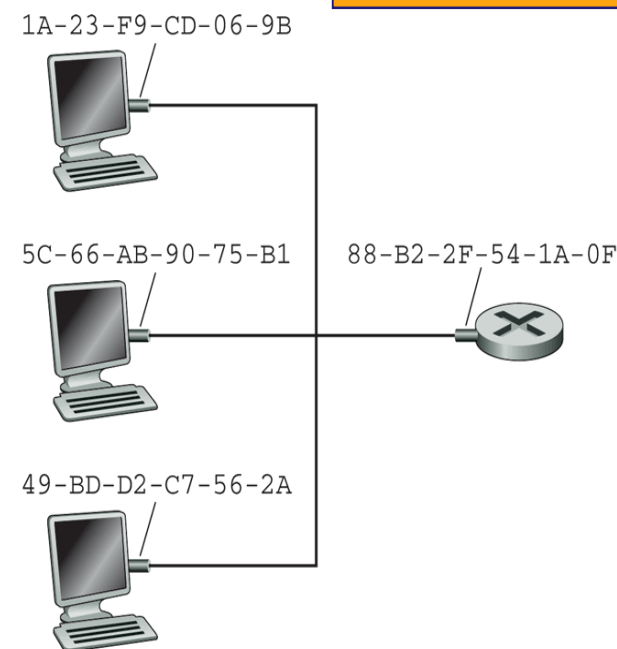
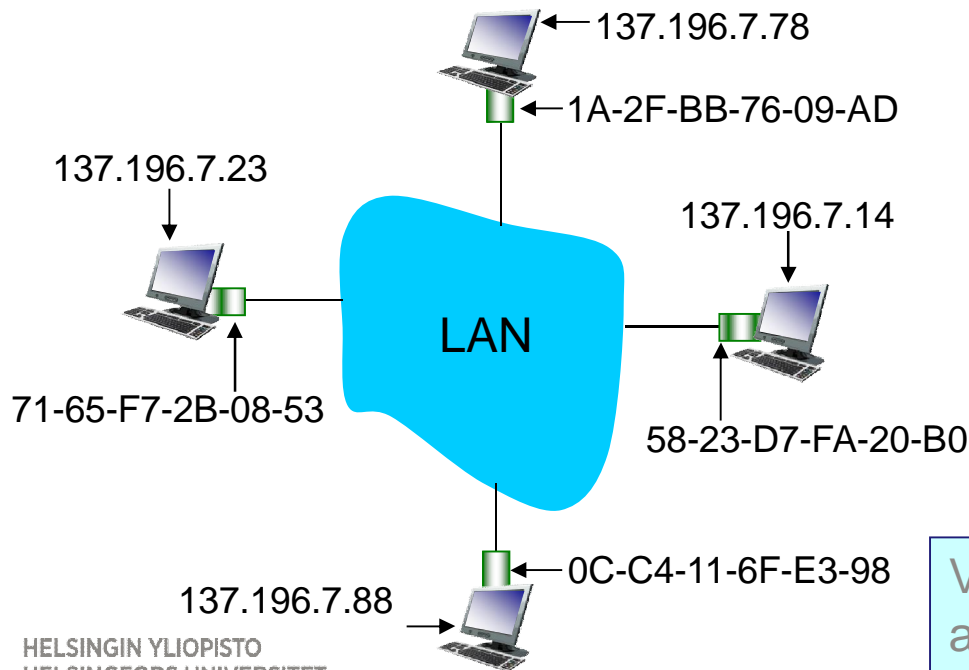


Fig 5.17 [KR12]



Koneen MAC-osoitteen selvittäminen: **ARP**: address resolution protocol

Miten selvittää koneen
MAC-osoite, kun
tiedetään IP-osoite?



ARP taulu: jokaisella
koneella (isäntä, reititin)
oma taulu kullekin aliverkolle

- IP/MAC osoitteen muunnos
saman aliverkon koneille:
< IP-osoite; MAC-osoite; TTL >
- TTL (Time To Live):
Voimassaoloaika (tyypillisesti
20 min), ajan kuluttua
muunnoksen voi unohtaa

Vrt: sovelluskerroksen **DNS**, jonka
avulla voi selvittää IP-osoitteen nimelle



ARP-protokolla

(Address Resolution Protocol)

MAC-
yleislähetysosoite:
FF-FF-FF-FF-FF-FF

ARP-protokolla lähettää **yleislähetysosoitteella** kyselyn, jonka kaikki vastaanottavat.

Oman osoitteensa tunnistava laite **vastaa kyselijän MAC-osoitteeseen** ja kertoo oman MAC-osoitteensa

"aa-bb-cc-dd-ee-ff", "FF-FF-FF-FF-FF-FF"

"Kenen IP-osoite on "xx:yy:zz:vv"?"

"kk-ll-mm-nn-oo-pp", "aa-bb-cc-dd-ee-ff"

ARP-taulun sisältö kootaan suorituksen aikana, aluksi tyhjä

IPV6:ssa Neighbour Discovery Protocol (NDP)



Lähettäminen toiseen verkkoon

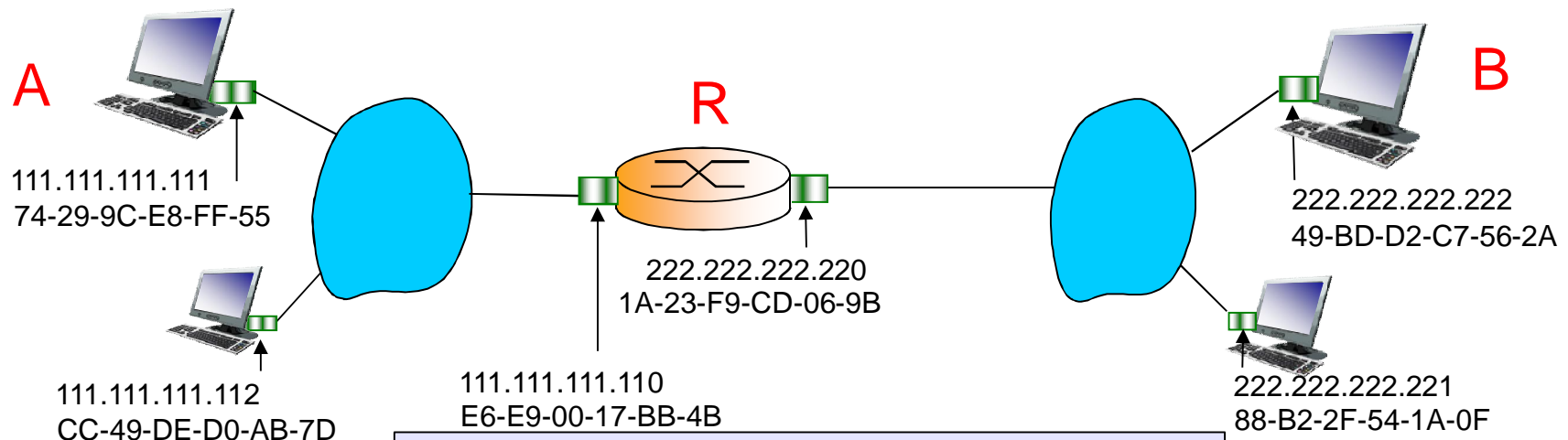
Kts Fig 5.19 [KR12]

Ensin omalle reitittimelle sen MAC-osoitteella ja reititin ohjaa eteenpäin

Reititystaulussa on verkko-osoite, jonne paketti seuraavaksi ohjattava

Katso kohdeverkon ARP-taulusta kohteen MAC-osoite

Jos ei ole taulussa, tee ARP-kysely kohdeverkon koneille



Reitittimellä on useita ARP-tauluja.



Lähettäminen toiseen verkkoon

Lähettäjä A

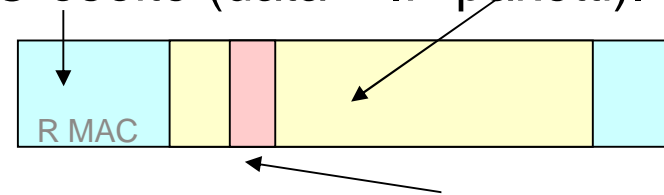
Muodosta IP-paketti, jossa Source IP = A, Dest. IP = B
Etsi ARP-taulusta **reitittimen** IP-osoitetta vastaava MAC-osoite
Luo siirtokehys, osoitteena reitittimen MAC-osoite (data = IP-paketti).
Verkkokortti lähettää siirtokehyyksen.

Reititin R

Verkkokortti ottaa siirtokehyyksen vastaan.
Ota IP-paketti kehyksestä ja tutki otsakkeesta kohteen IP-osoite (B)
Katso reititystaulusta, mihin verkkoon seuraavaksi (mille reitittimelle)
Koska omassa verkossa, etsi kohdeverkon ARP-taulusta kohteen
MAC-osoite
Muodosta siirtokehys, osoitteena B:n MAC-osoite (data = IP-paketti)

Vastaanottaja B

Verkkokortti ottaa kehyksen vastaan; ohjaa IP-paketin verkkokerrokselle.





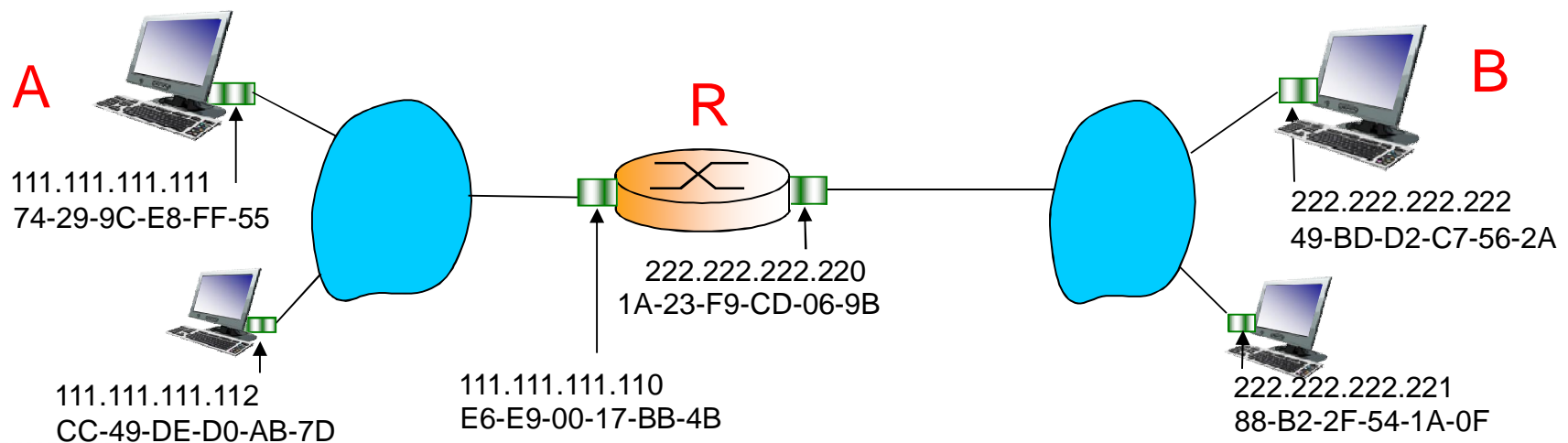
Lähettäminen toiseen verkkoon: vaiheittainen esimerkki

Lähetetään paketti (datagram) A:sta B:hen

Keskitytään osoitteisiin – IP (paketti) ja MAC (kehys)

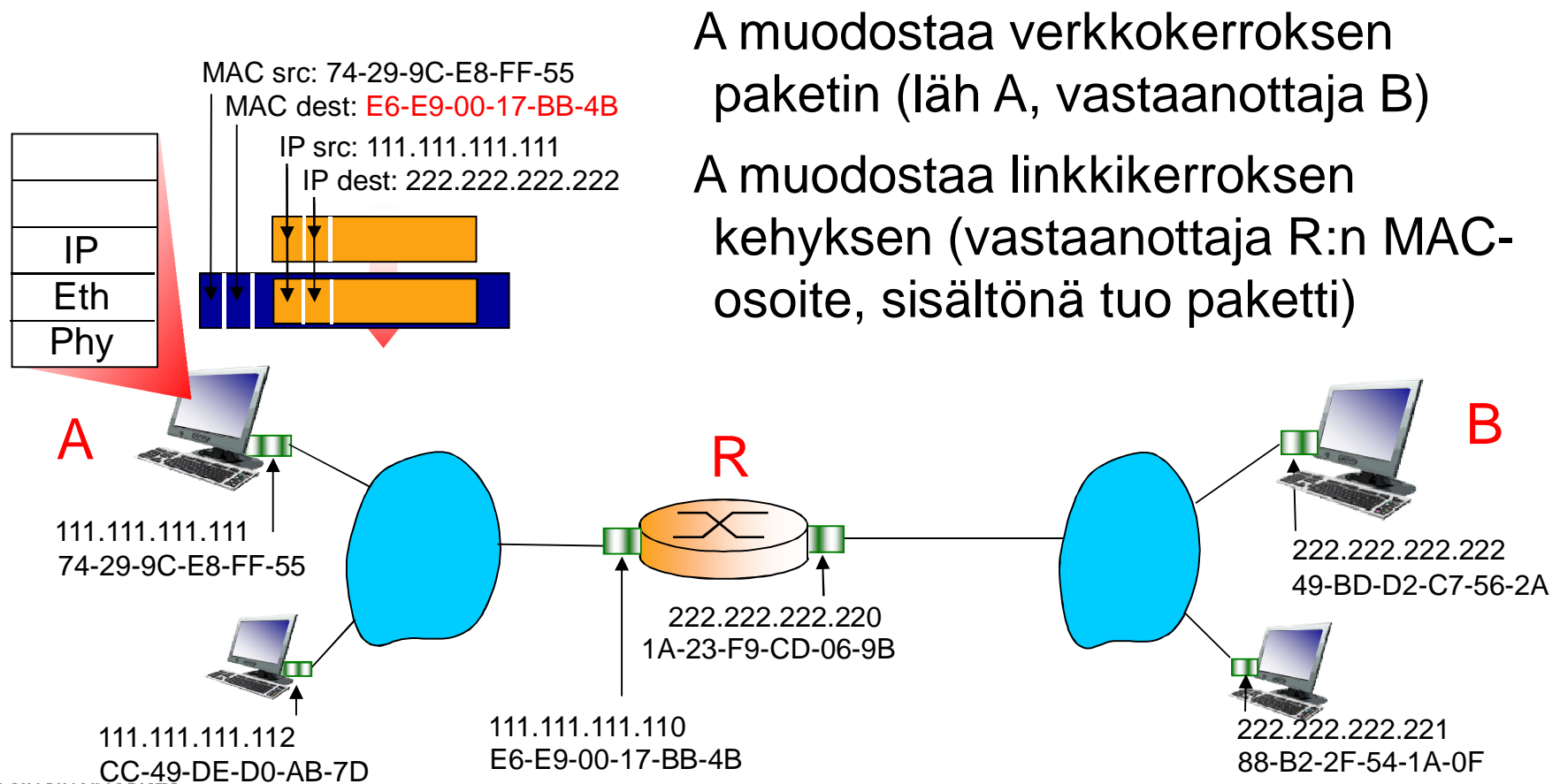
Oletetaan, että A tietää jo B:n IP-osoitteen,

Oletetaan myös, että A tuntee oman aliverkkonsa reitittimen R IP-osoitteen ja MAC-osoitteen (Miten?)





Lähttäminen toiseen verkkoon: vaihe 1



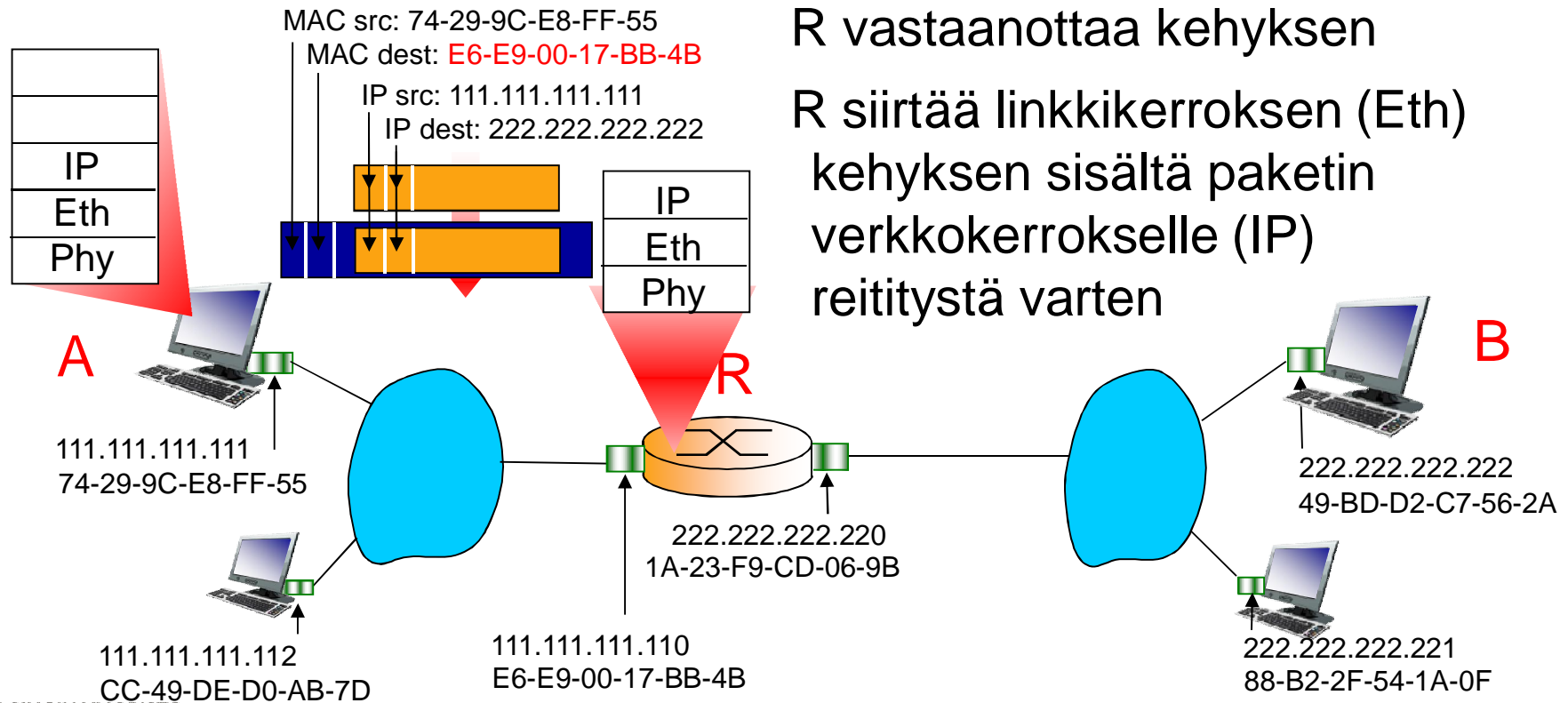


Lähttäminen toiseen verkkoon: vaihe 2

A lähettää kehyksen

R vastaanottaa kehyksen

R siirtää linkkikerroksen (Eth)
kehyksen sisältä paketin
verkkokerrokselle (IP)
reititystä varten

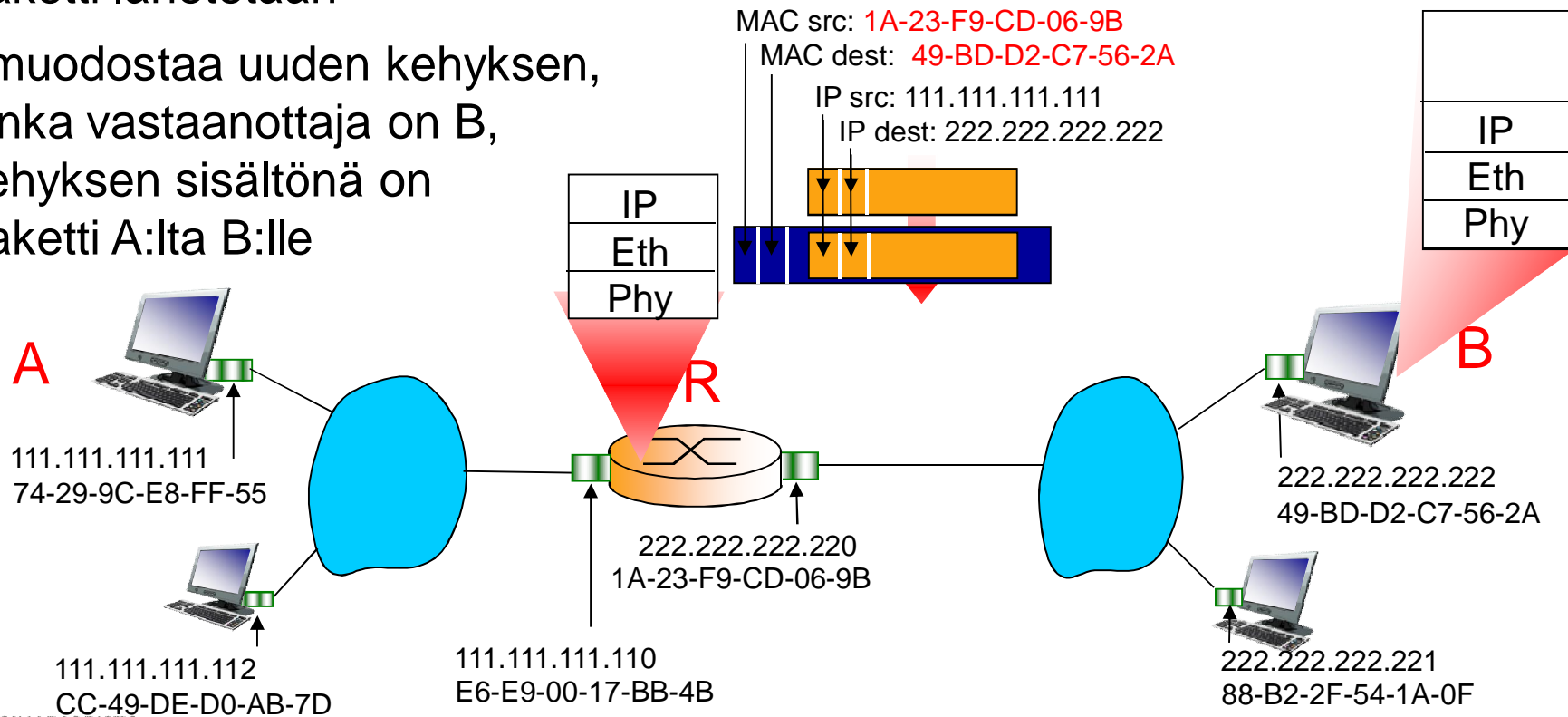




Lähtettäminen toiseen verkkoon: vaihe 3

R katsoo reitititystaulusta, minne paketti lähetetään

R muodostaa uuden kehyksen, jonka vastaanottaja on B, kehyksen sisältönä on paketti A:lta B:lle

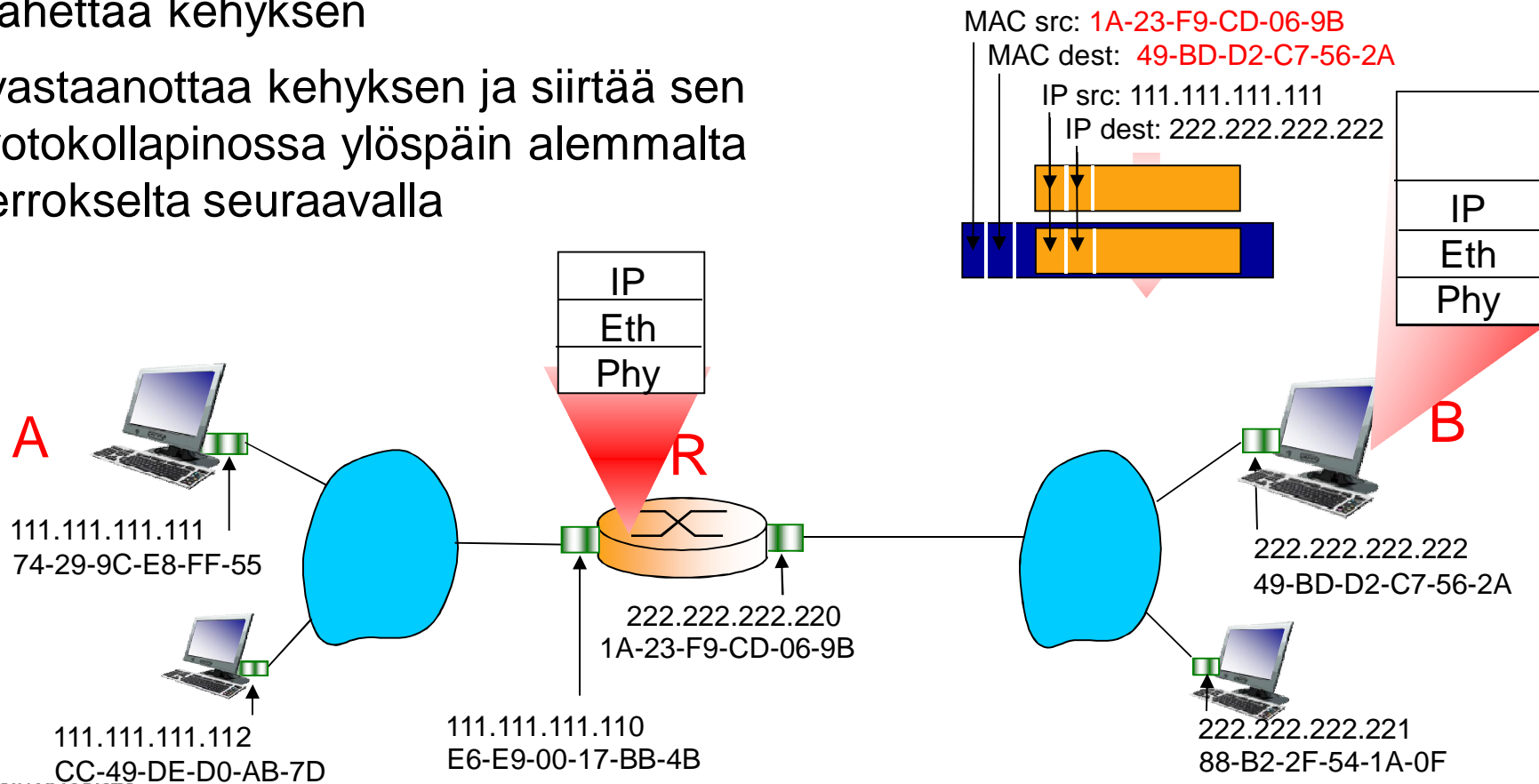




Lähettäminen toiseen verkkoon: vaihe 4

R lähettää kehyksen

B vastaanottaa kehyksen ja siirtää sen
protokollapinossa ylöspäin alemmalta
kerrokselta seuraavalla





Linkkikerros

Ethernet

Ch 5.5



Ethernet

Yleisin lähiverkkoteknologia

Yksinkertainen, edullinen, helppo laajentaa

Lähiverkko syntyy kytkemällä koneet keskittimeen tai kytkimeen

IEEE:n standardoima LAN-verkko

Klassinen Ethernet (10 Mbps): CSMA/CD (kuulosteluväylä)

Fast Ethernet (FE, 100 Mbps), Gigabit Ethernet (GE), 10 Gigabit Ethernet, 100 GB Ethernet (pian??), 1 TB Ethernet (joskus??!)

- Yleensä kytkentäisiä kaksipisteyhteyksiä

Muita lähiverkkostandardeja

- Token Ring (vuororengas)
- FDDI (Fiber Distributed Data Interface)
- WLAN (langaton lähiverkko)

Ethernet Timeline (2003)

- * 10 Megabit Ethernet 1990
- * 100 Megabit Ethernet 1995
- * 1 Gigabit Ethernet 1998
- * 10 Gigabit Ethernet 2002
- * 100 Gigabit Ethernet 2006**
- * 1 Terabit Ethernet 2008**
- * 10 Terabit Ethernet 2010**

April 24, 2008

Terabit Ethernet around 2015

**Bob Metcalfe (ethernet
coinventor)**

**gave a keynote speech,
"Toward Terabit Ethernet."**



10BaseT ja 100BaseT

10 Mbps tai 100Mbps (Fast Ethernet, FE)

T = Twisted Pair eli kierretty parikaapeli

Maks. etäisyys keskittimeen 100 m

Keskitin (hub) *toistaa bitit heti sellaisenaan muille*

Fyysisen tason toistin (repeater); yleislähetys

Signaalin vahvistus

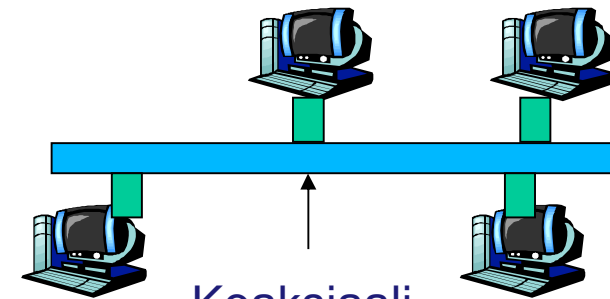
Verkkokortit käsittelevät törmäykset

Maks. 30 konetta / keskitin

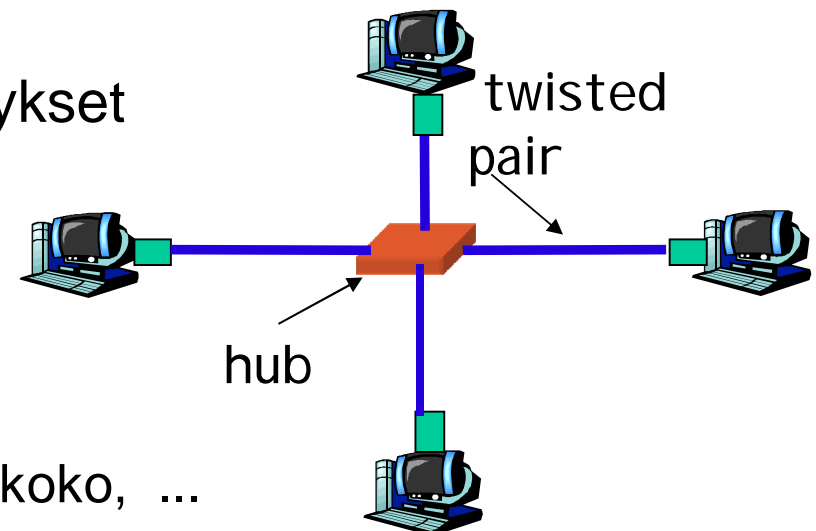
Keskitin osaa jättää huomiotta vikaantuneen kortin

Kerää myös tietoa liikenteestä

Törmäysten lkm, keskim. kehyskoko, ...



Koaksiaali-
kaapeli max.
500 m





Gigabitin Ethernet (GE):

1 Gbps, 10 Gbps, 40 Gbps, 100 Gbps

Edelleen sama kehysformaatti

Taaksepäin yhteensopiva

Yhteiskäyttöiset linkit edelleen OK

Koneiden yhdistely keskittimen välityksellä
CSMA/CD

Kaksipisteyhteydet

ei törmäyksiä

koneet yhdistetty **kytkimien** kautta

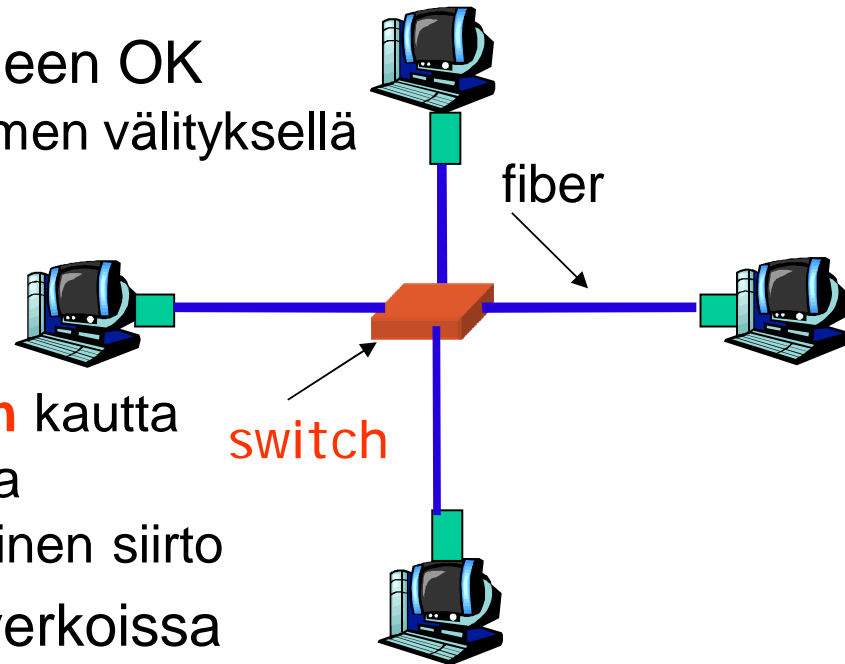
pitkät välimatkat mahdollisia

kaksisuuntainen täysivauhtinen siirto

Käytetään yleisesti runkoverkoissa

verkkojen yhdistely (reititin -> reititin)

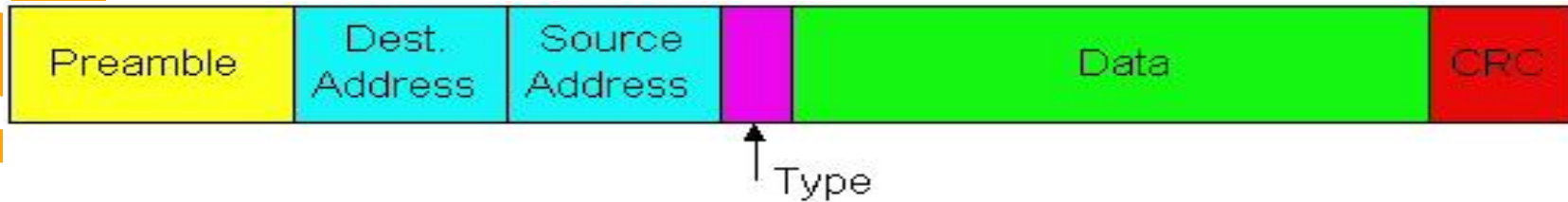
valokaapeli, myös cat5/cat6 parikaapeli





Ethernet-kehys

Fig 5.20 [KR12]



Tahdistuskuvio (preamble) (8 B)

- 7 ensimmäistä tavua: 10101010 kellojen tahdistusta varten
- 8. tavu: 10101011 kertoo varsinaisen kehyksen alkavan

Kohteen ja lähteen MAC-osoitteet (6 + 6 B)

Type (2 B)

- verkkoprotokolla, jolle vastaanottaja luovuttaa kehyksen datan
- IP, ARP, jokin muu esim, Apple Talk, Novell IPX, ..

Data (**46** ... 1500 B)

Ethernet MTU = 1500 B

CRC (4 B eli 32 bittiä)

tarkistusbitit, tahdistuskuvio mukana laskennassa



Kehyksen minimipituus

Data-osan pituus min 46 B

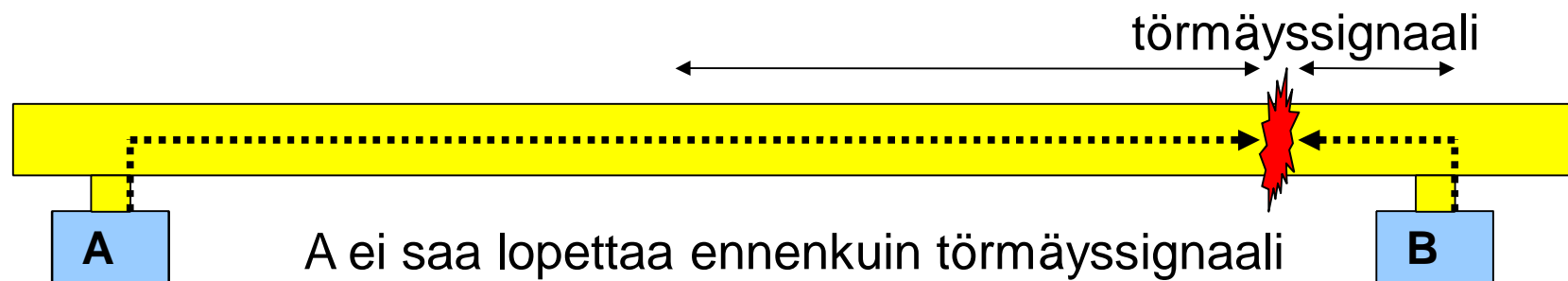
Tarvittaessa täytetäviä (pad), jotka vastaanotto poistaa

Lähettäjän **ehdittävä huomata mahdollinen törmäys**

Kehyksen lähetys ei saa päättyä ennenkuin alku on perillä ja mahdollinen törmäysääni kuuluu

- Alku perillä -> loppukin onnistuu

Lähetyksen minimikesto = $2 \times$ etenemisviive





Linkkikerros

Keskitin, kytkin

Ch 5.6



Keskitin (hub)

fyysinen kerros (layer-1)

Käsittelee bittejä

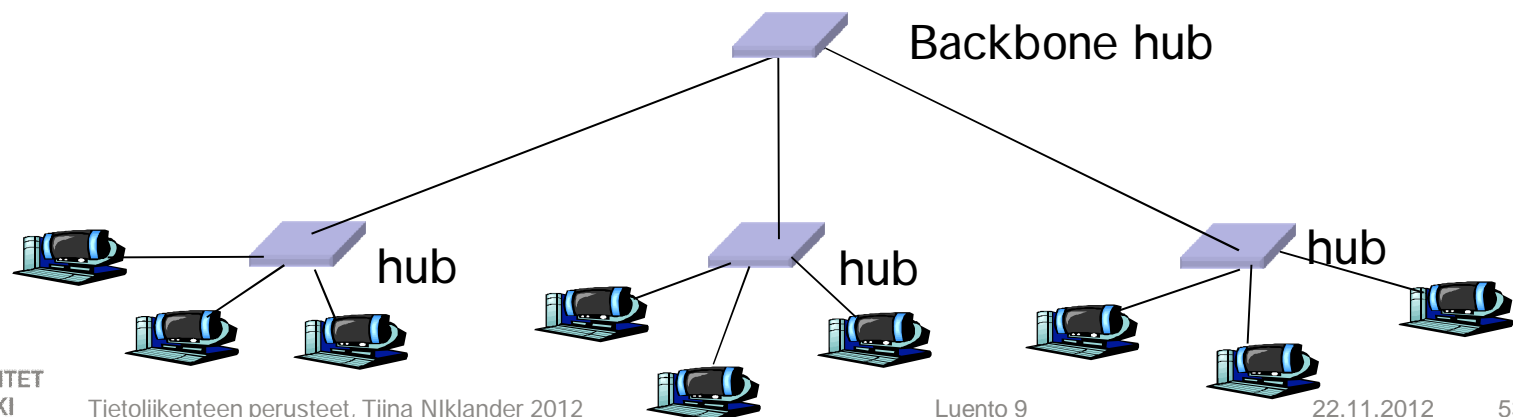
Toistaa saamansa bitit heti kaikille muille linkeille

Signaalin vahvistus

Yhteinen törmäysalue → vain pieniin verkkoihin

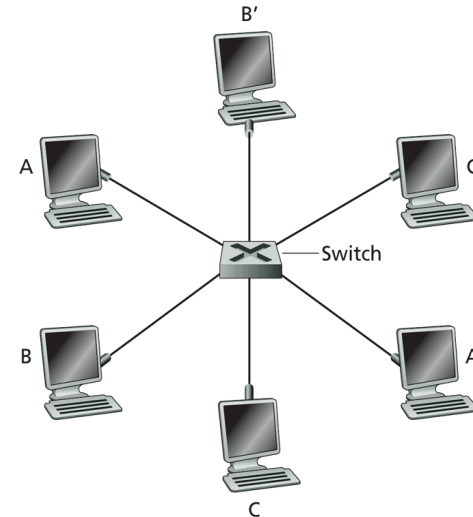
Yhdistää vain saman teknologian laitteita

Ei esim. 10 Mbps ja 100 Mbps samaan keskittimeen





Kytkin (switch) linkkikerros (layer-2)



- Käsittelee siirtokehyksiä,
useita yhtäaikaista yhteyksiä
- Vastaanottaa ja lähettää kokonaisia kehyksiä
- Etappivälitys (store and forward) (yleensä)
- Ei törmäyksiä
- Suora piuha koneelta kytkimeen
- Kytkin lähettää ulos vain yhdelle piuhalle
- Voi yhdistää erilaisia verkkosegmenttejä
- Kytkimessä esim. 10/100 Mbps portteja
- Puskurointia
- Tuntumaton (transparent)
- Sopeutuu itse verkon muutokseen
- 'plug-and-play, self-learning



Kytkin: kehyksien välitys - mihin linkkiin?

Miten kytkin osaa välittää kehyksen juuri oikeaan piuhaan?

Se kerää itse ('oppii') tarvittavat tiedot

takaperinoppimista (backward learning): saapuva kehys kertoo, mistä linkistä **lähettäjä** saavutetaan

Ylläpitää kytkentätaulukkoa

(MAC-osoite, linkki, TTL)

TTL-aikaleima: poista ne, joita ei ole käytetty esim. 60 minuutin aikana



Kytkentätaulu (switching table): -tietojen keruu saapuvista kehyksistä

Aluksi taulu on tyhjä

Saapuva kehys

Lähteen MAC-osoite x ,
kohteen MAC-osoite y ,
tuloportti p , yms

Lähde X ei ole taulussa =>

Lisää (X, p, TTL) tauluun
eli **kytkin oppii, että
osoite X on
saavutettavissa portin
 p kautta**

Lähde X on taulussa =>
päivitä TTL

Kohde Y ei ole taulussa =>

Lähetetään kehys kaikkiin
muihin portteihin =
tulvitus (flooding)

Opitaan myöhemmin Y :n
oikea portti jostain sen
lähettämästä kehyksestä

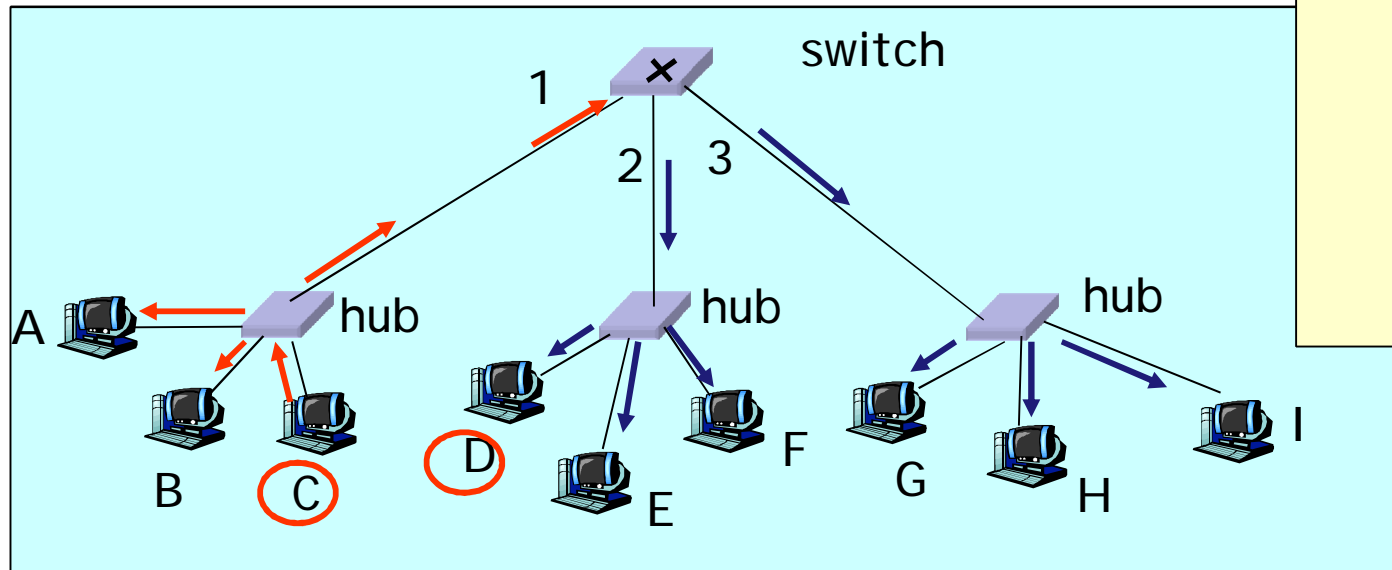
Lähde X ja kohde Y jo
taulussa

X ja Y samassa portissa =>
hylkää kehys (on jo
oikeassa aliverkossa)

X ja Y eri porteissa =>
lähetä kehys Y :n porttiin



Esimerkki: C lähettää kehyksen D:lle

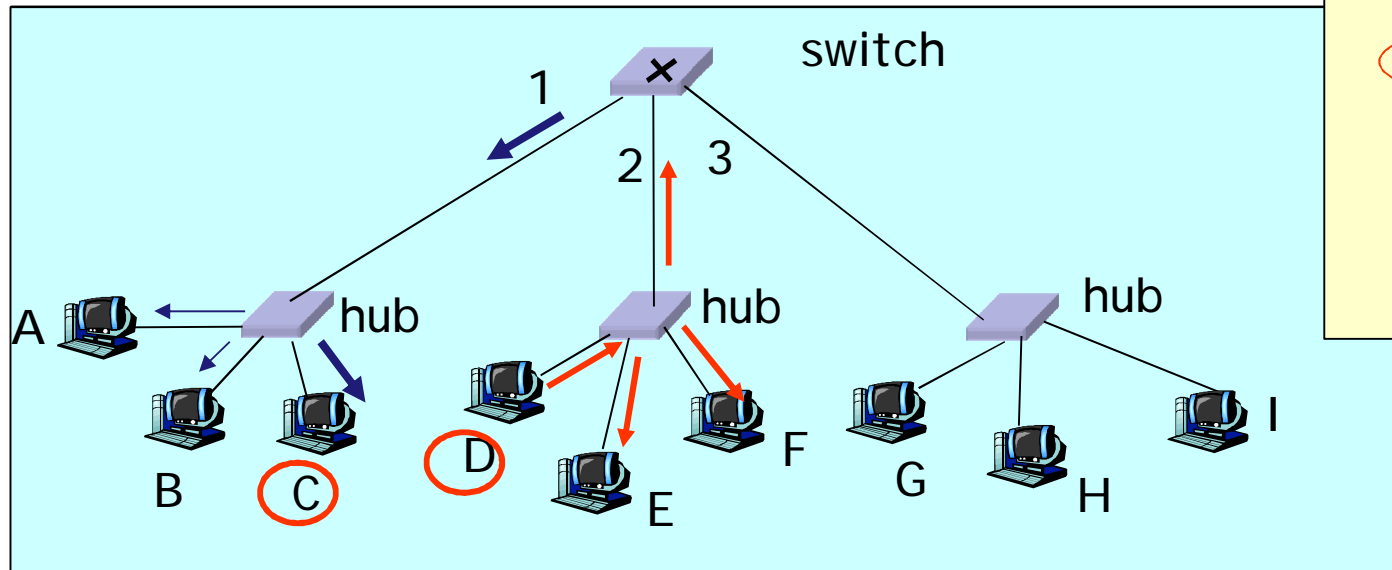


address	interface
A	1
B	1
E	2
G	3
C	1

- Kytin vastaanottaa kehyksen (A ja B kuulevat myös)
 - Merkitsee tauluun C:n MAC-osoitteen ja portin 1
 - Koska D ei ole taulussa, tulvittaa linkeilla 2 ja 3.
- D vastaanottaa kehyksen (E, F, G, H, I kuulevat myös)



Esimerkki jatkuu: D lähettää kehyksen C:lle



address	interface
A	1
B	1
C	1
E	2
G	3
D	2

- Kytkin vastaanottaa kehyksen (E ja F kuulevat myös)
 - Merkitsee tauluun D:n MAC-osoitteen ja portin 2
 - C:n osoite on taulussa, joten lähettää kehyksen linkkiin 1
- C vastaanottaa kehyksen (A ja B kuulevat myös)



Tulvitus (flooding)

Tulvitus voi olla ongelma

Kehykset voivat jäädä kiertämään silmukoissa

Koko verkko tukkeutuu

Siis silmukoita ei saa muodostua!

Verkon loogisen rakenteen pitää olla puu.

Virittävä puu (Spanning tree)

Lyhimmin poluin virittävä puu Dijkstran algoritmilla



Suorakytkentä (cut-through switching)

Jotkut kytkimet voivat välittää kehyksen bitit ulos sitä mukaa kuin itse ne saavat

Välityspäätöksen tekoon riittää tutkia otsakkeesta kohdeosoite

Ei siis enää etappivälitteistä (store-and-forward)

Pienentää latenssiaikaa

Ei kuitenkaan mahdollomasti ...

100 Mbps:n linjalla odotusta maksimissaan noin 0.12 ms



Vertailua

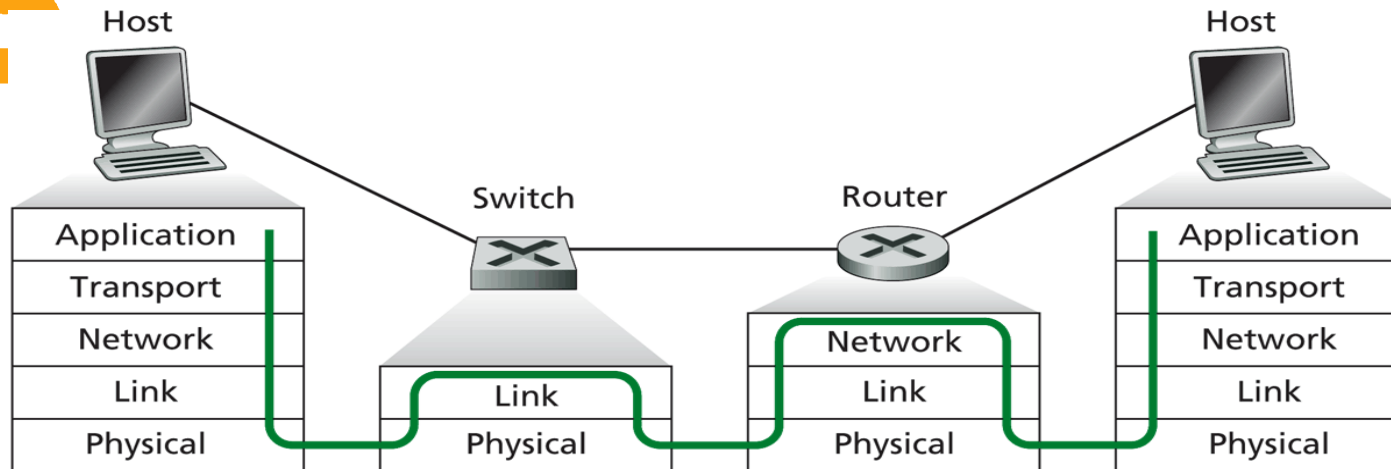


Fig 5.24 [KR12] ♦ Packet processing in switches, routers, and hosts

Table 5.1 [KR12]

	Keskitin (hub)	Kytkin (switch)	Reititin (router)
Traffic isolation	no	yes	yes
Plug and play	yes	yes	no
Optimal routing	no	no	yes
Cut through	yes	yes	no



Kertauskysymyksiä

Miten lähiverkko rakennetaan?

Reititin vs. kytkin vs. keskitin?

IP-osoite vs. MAC-osoite?

ARP-protokolla ja ARP-taulu?

Takaperinoppiminen ja kytkentätaulu?

Bittivirheiden havaitseminen?

CRC?

Lähetyskanavanjako?

CSMA/CD?

ks. kurssikirja s. 501

