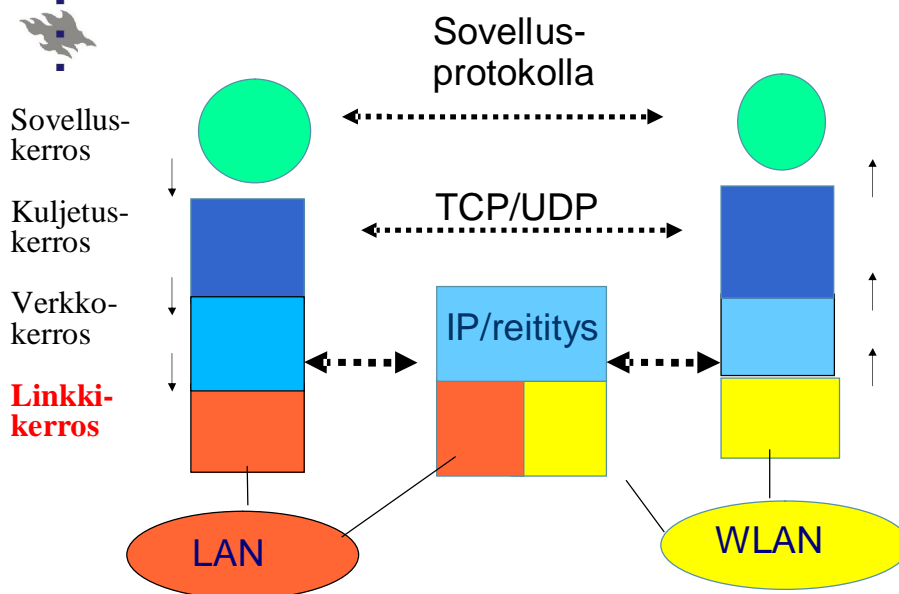




# Tietoliikenteen perusteet

## Linkkikerros

Kurose, Ross: Ch 5.1- 5.6





## Sisältö

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



### 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

- n Laitetoimintoa
- n Siirtää paketin fyysistä linkkiä pitkin koneelta (solmulta (node)) toiselle
  - langallinen / langaton
  - bitit sisään, bitit ulos
- n Kapseloi paketin siirtoon sopivaan muotoon
  - n Siirtokehys (frame)
- n Lähiverkossa linkejä voi yhdistää keskittimillä tai kytkimillä
  - n Käytetään fyysisiä osoitteita
  - n 'reititystä' ilman IP-osoitteita

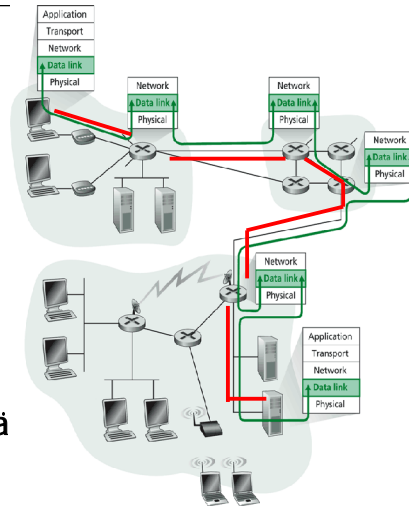


Figure 5.1 • The link layer



# Linkkikerroksen tehtäviä

- n **Kehystys (framing)**
  - Kehyksen rakenne ja koko riippuu siitä, millainen linkki on kyseessä
  - Otsake, data, loppuke
- n **Kohteen ja lähteen osoittaminen**

otsake	data	loppuke
--------	------	---------
- n **Yhteisen linkin varaus ja käyttö (link access)**
  - Esim. langaton linkki, keskittimiin yhdistetyt linkit
- n **Luotettava siirto**
  - Langattomilla linkeillä suuri virhetodennäköisyys
  - Linkkitaso huolehtii oikeellisuudesta
  - Miksi tästä täytyy huolehtia vielä kuljetuskerroksella?
  - Jotkut linkkityypit eivät huolehdi lainkaan!
  - Jos kehys hävitettävä ..



## Linkkikerroksen tehtäviä (2)

### n Vuonvalvonta, puskurointi

Kytkimessä on useita erinopeuksisia linkejä

### n Virhevalvonta

signaali vaimenee, taustakohina häiritsee, ...

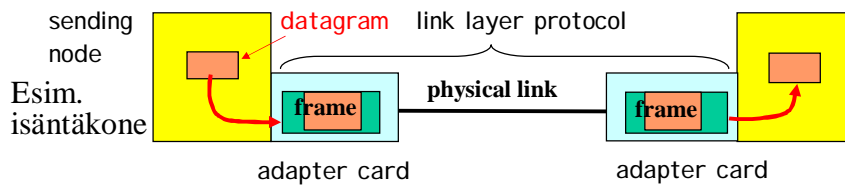
Kehyksessä on tarkistustietoa (error detection and correction bits)

Vastaanottava solmu korjaa, jos pystyy

Jos ei pysty, pyytää uudelleen tai hävittää

### n Yksisuuntainen /kaksisuuntainen liikenne

Yksisuuntainen: lähetysvuorojen hallinta

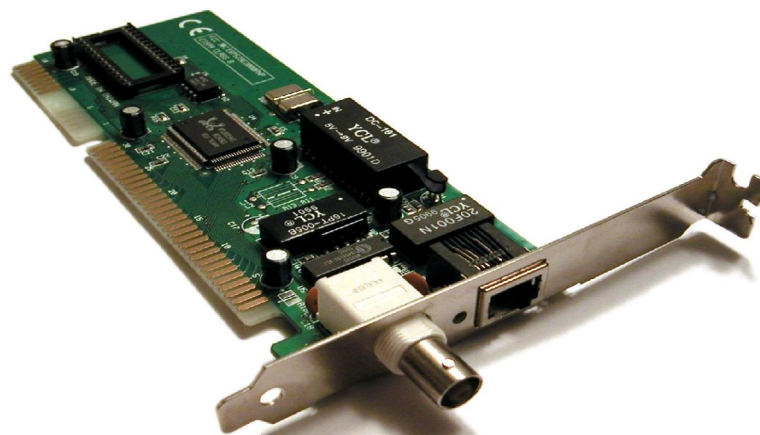


Tietoliikenteen perusteet /2008/ Liisa Marttinen

7



## A 1990s Ethernet network interface card.



[http://upload.wikimedia.org/wikipedia/commons/9/9e/Network\\_card.jpg](http://upload.wikimedia.org/wikipedia/commons/9/9e/Network_card.jpg)

Tietoliikenteen perusteet /2008/ Liisa Marttinen

8



## Linkkikerros

# Virheiden havaitseminen ja korjaaminen

Ch 5.2



## Bittitason virheet

- n Yhden bitin virheitä siellä täällä tai peräkkäisten bittien virheryöppyjä (burst)
- n Virheiden esiintymistiheys BER (bit error rate)
  - n Mitä suurempi BER, sitä lyhyempiä kehyksiä kannattaa käyttää
- n Havaitsemiseksi lisäbittejä
  - n feedback/backward error control
- n Korjaamiseksi enemmän lisäbittejä
  - n Forward error correction (FEC)
  - n Esim. CD, DVD, viivakoodit, satelliittiyhteydet, digitelevisio, ...
- n Tietoliikenne yleensä tyytyy vain havaitsemaan virheet
  - n Virheelliset hylätään ja korjauksena on uudelleenlähetyt



## Pariteettitarkistus

### Pariteettibitti

Parillinen vs. pariton pariteetti  
Virheryöpyssä jopa 50% voi jäädä huomaamatta

← d data bits → parity bit

0111000110101011 | 0

### Kaksiulotteinen pariteetti

Erikseen horisontaalinen (parillinen) ja vertikaalinen (pariton) pariteetti  
Pystyy korjaamaan yhden bitin virheen.

### Hamming-koodi

Korjaa yhden bitin virheen

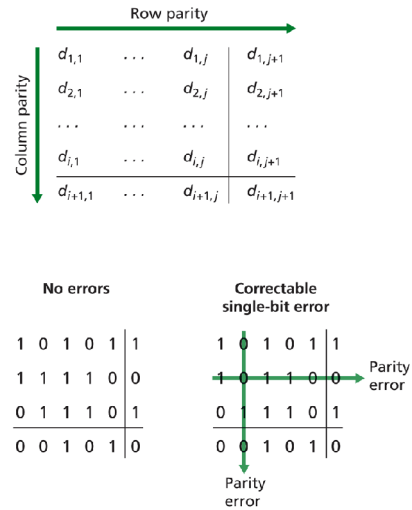


Figure 5.6 ♦ Two-dimensional even parity



## Tarkistussumma

### Internet-checksum

Yhteenlasketaan 16 bitin kokonaisuuksia, yhden komplementti  
Kuljetuskerros laskee ja tarkastaa UDP- ja TCP-protokollissa  
Ei ole kovin tehokas; linkkikerros ei käytä

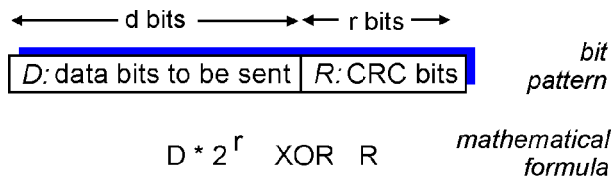
### CRC (cyclic redundancy check)

Yleisesti linkkikerroksella käytetty virheenpaljastusmenetelmä,  
helppo toteuttaa laitteistotasolla, luotettava  
Perustuu polynomien aritmetiikkaan  
tunnetaan myös nimellä polynomikoodi  
Useita tarkistusbittejä; havaitsee usean bittivirheen ryöpyn.



## CRC

- n Käsittelee databittejä yhtenä kokonaislukuna
- n Sovittu **virittäjäpolynomi G**
  - bittejä yksi enemmän kuin lisättäviä tarkistusbittejä (=r) eli r+1
- n Lähettäjä
  - Asettaa tarkistusbitit R s.e. datan bitit (=D) + niiden perään liitetyt tarkistusbitit ovat jaollisia virittäjällä G (**modulo 2-aritmetiikka**)
- n Vastaanottaja
  - Jakaa samoin saamansa bittijonon (D+R) virittäjällä G.
  - Jos **jakojännös != 0, niin on virhe.**



Tietoliikenteen perusteet /2008/ Liisa Marttinen

13

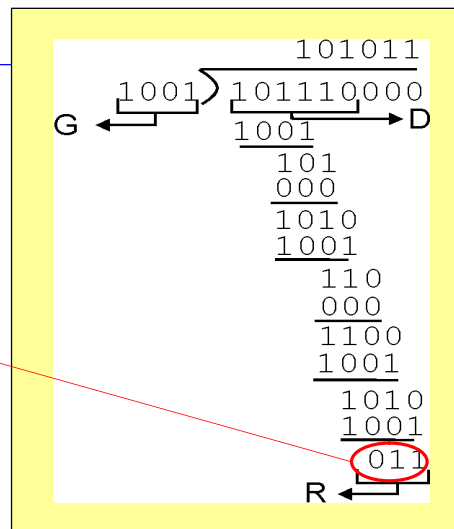


## CRC-esimerkki

Data: 101110  
 G: 1001, polynomina  
 $1*x^3 + 0*x^2 + 0*x^1 + 1*x^0$   
 <D,R>: 101110????

Lähetä: 101110**011**

**Modulo 2-aritmetiikka**  
 vähennyslasku yhteenlaskuja  
 ei lainaamista, ei muistinumeroita  
 = bittitason XOR  
 $1+1=0, 1+0=0+1=1, 0+0=0$



KuRo08:Fig 5.8

Tietoliikenteen perusteet /2008/ Liisa Marttinen

14



## Standardoituja virittäjäpolynomeja

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

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

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

= 1 0000 0100 1100 0001 0001 1101 1011 0111

Virittäjäpolynomin merkitsevin bitti = 1

### Havaitsee

kaikki virheryöpyt, joiden pituus < tai = kuin virittäjän pituus  
lähes kaikki virheryöpyt, joiden pituus on suurempi



## Linkkikerros

# Yhteiskäyttöinen kanava

Ch 5.3



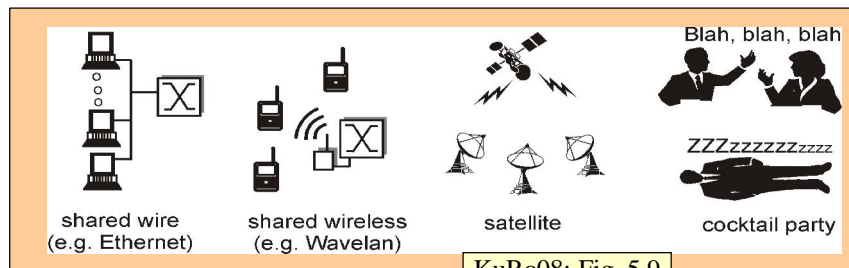
## Yksi kanava

### n Kaksipisteyhteys (point-to-point)

- n PPP-protokolla, puhelinyhteys (dial-up access)
- n Ethernet-piuha kytkimen ja isäntäkoneen välissä

### n Yleislähetysyhteys (broadcast)

- n Alkuperäinen Ethernet, Ethernet keskittimen ja isäntäkoneen välissä, kaapelimodeemiyhteys (upstream), WLAN, satelliitti,



KuRo08: Fig. 5.9

Tietoliikenteen perusteet /2008/ Liisa Marttinen

17

## Lähetysvuorojen jakelu

### n Yksi yhteinen kanava lähettäjiille

- n Lähetys onnistuu vain, jos yksi kerrallaan lähettää

### n Jos useampi lähettää yhtäaikaan, syntyy yhteentörmäys

- n Kaikki solmut saavat useita signaaleja, "bittimössöä"
- n Törmäykset sanomat tuhoutuvat ja ne on lähetettävä uudelleen

### n Multiple Access protocol

- n Tapa, jolla solmu päättää, voiko se lähettää
- n Kuinka solmun on toimittava törmäystilanteessa



Tietoliikenteen perusteet /2008/ Liisa Marttinen

18



## Tätä tavoitellaan

### n Pieni yleisrasite

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

### n Tasapuolisuus

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

### n Toimintavarmuus

- n Yksikään solmu ei ole erikoisasemassa, koordinaattorina
- n Ei kellojen sykronointia tms
- n Hajautettu vuoroista sopiminen

### n Kustannustehokkuus

- n Yksinkertainen ja halpa toteuttaa



## Lähetysvuorojen jakelu

### 1) Kanavanjakoprotokollat (channel partitioning protocol)

Jaetaan 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)

Jaetaan käyttövuorot jollakin sovitulla tavalla:  
pollaus, vuoromerkki, ...  
"Minä ensin, sinä sitten."



## 1) Kanavanjako: TDMA

**TDMA: Time Division Multiple Access**

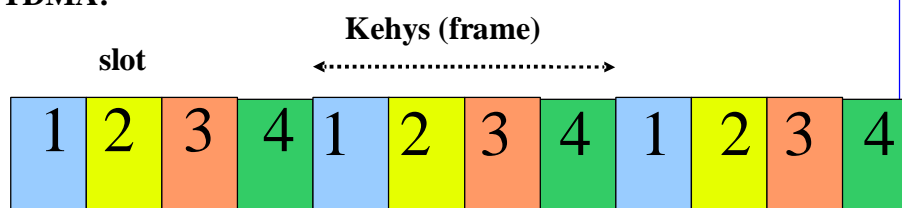
Anna aikaviipale kullekin kanavaan kytketylle vuorotellen

Koko kanava on hetken yksityiskäytössä => R/M bps

Ehtii lähettää yhden kehyeellisen (data frame)

Vaikka lähetettävää ei olisikaan, aikaviipale on silti varattuna

**TDMA:**



Tietoliikenteen perusteet /2008/ Liisa Marttinen

21



## Kanavanjako: FDMA

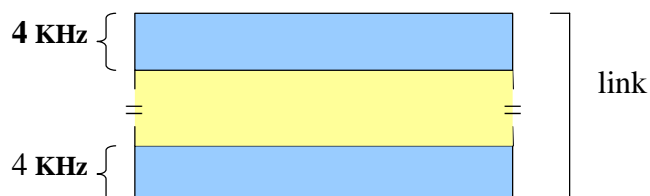
FDMA (Frequency Division Multiple Access)

Jaetaan kanavan taajuusalueet kanavan käyttäjien (varaajien) kesken

Vain osa kanavasta yksityisessä käytössä => R/M bps

Varattuna, vaikka ei olisi lähetettävää

**FDM:**



Tietoliikenteen perusteet /2008/ Liisa Marttinen

22



## Kanavanjako: CDMA

- n CDMA (Code Division Multiple Access)  
Radiolinjoilla käytettävä koodinjakoon perustuva protokolla
  - n Matkapuhelimet, ..
  - n Kullakin asemalla oma tapansa koodata bitit 1 ja 0 (oma sirukoodi)
- n Asemat voivat lähettää yhtäaikaan koko kanavan taajuudella
  - n Kaikkien signaalit saavat yhdistyä linkillä
  - n Asemat pystyvät erottelemaan yhteissignaalista itselleen kuuluvat bitit (oma sirukoodi)
- n Tarkat ajoitukset



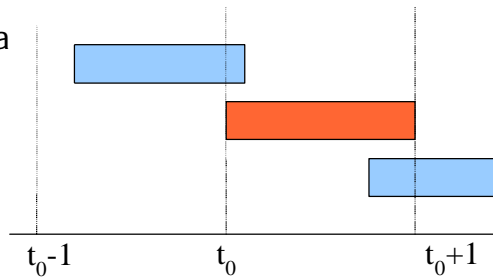
## 2) Kilpailuprotokollat

- n Kun asema haluaa lähettää
  - n Se kuuntelee ensin, onko joku muu asema jo lähettämässä
  - n Jos ei, lähettää heti täydellä nopeudella
- n Jos kaksi aloittaa yhtäaikaan => törmäys
  - n Odota satunnainen aika ja yritä uudelleen (random access)
- n Protokolla määrittää
  - n Miten törmäys huomataan
  - n Miten törmäyksestä toivutaan
- n Esim.
  - n ALOHA, viipale ALOHA (slotted ALOHA)
  - n CSMA (carrier sense multiple access)
  - n CSMA/CD (with collision detection)
  - n CSMA/CA (collision avoidance)



## Aloha

- n Hawaijilla, 70-luvulla radiotietä varten
- n Lähetä heti, kun on lähetettävää
  - n Ei mitään kuuntelua ennen lähetystä
- n Kuuntele sitten, onnistuiko lähetys
  - n Lähiverkossa törmäys havaitaan 'heti', sillä siirtoviive on pieni (toisin kuin satelliitilla)
- n Jos törmäys, niin odota satunnainen aika ja yritä uudelleen
- n Yksinkertainen
- n Törmäyksen td. suuri
  - n Max tehokkuus ~ 18%



## Viipaloitu Aloha (slotted Aloha)

- n Lähetysaika jaettu aikaviipaleiksi (slot)
  - n Kaikki siirtokehukset samankokoisia => siirtoaika aina vakiomittainen
- n Lähetys voi alkaa vain aikaviipaleen alussa
  - n Törmäykset täydellisiä => törmäysaika = yhden aikaviipaleen mittainen
- n Solmut synkronoitava: aikaviipaleen alku
- n Jos törmäys, niin kaikki solmut huomaavat
  - n Uudelleenyritys seuravalla viipaleella todennäköisyydellä p (ts. jättää yrittämättä seuraavalla viipaleella tn:llä 1-p)
  - n Yrittää, kunnes onnistuu
- n Suorituskyky kaksinkertaistuu (Alohaan verrattuna)
  - n Jos paljon lähettäjiä max., ~37 % tehokkuus
  - n Siis 37% tyhjiä, 37% onnistumisia, 26% törmäyksiä

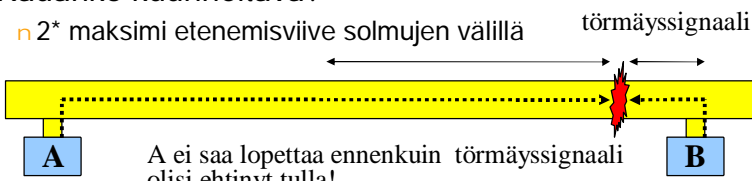
## Lähetyskanavan kuuntelu

- n Kuuntele ennenkuin lähetät
  - n Asema tutkii, onko kanava jo käytössä (carrier sense)
  - n Jos siirtotie on vapaa, saa lähettää
  - n Jos siirtotie on varattu, odota satunnainen aika ja yritä uudelleen
- n Ei aina paljasta jo alkanutta lähetystä
  - n Etenemisviiveen takia ei huomata toisen signaalia ajoissa
    - Seurauksena on törmäys
- n Tai huomaaminen ei ole mahdollista / järkevää
  - n Esim. Satelliittikanavan kuuntelu ei paljasta, onko jokin muu maa-asema jo aloittanut lähetyksen
  - n Langattomassa lähiverkossa lähettäjän ympäristön kuuntelu ei kerro, onko vastaanottaja saamassa sanomia muilta
- n CSMA (Carrier Sense Multiple Access)
  - n Useita variaatioita

Tietoliikenteen perusteet /2008/ Liisa Marttinen

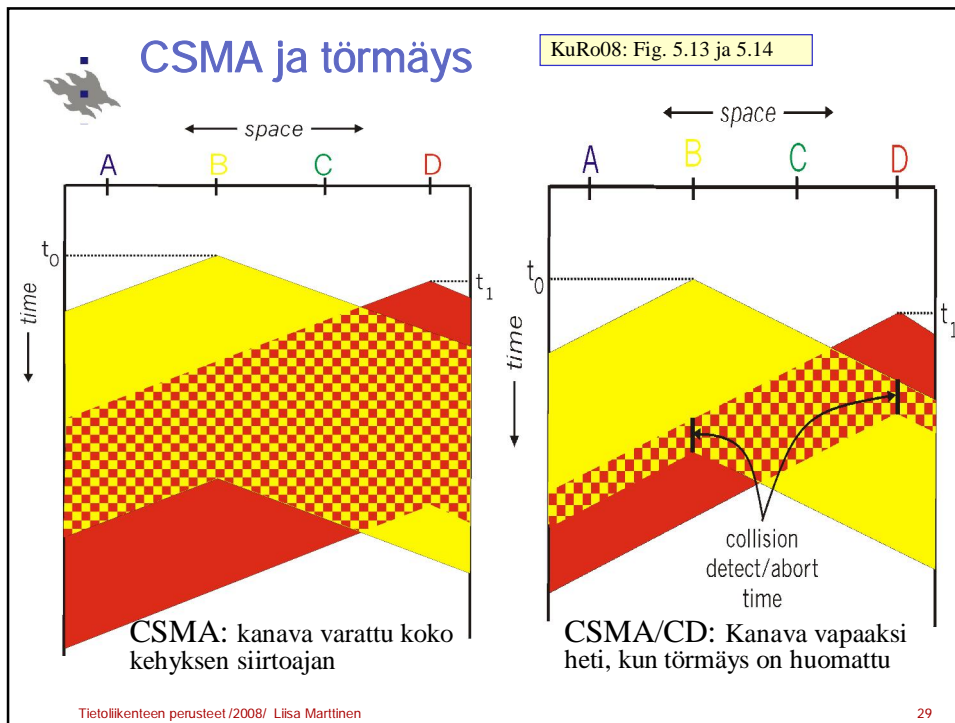
27

## CSMA/CD (with Collision Detection)

- n Asema kuuntelee myös lähettämisen jälkeen
    - n Langallinen LAN: signaalin voimakkuus muuttuu
      - Esim. Ethernet
    - n Langaton LAN: hankalaa
  - n Jos törmäys
    - n Niin keskeytä heti lähettäminen
    - n ja yritä uudestaan satunnaisen ajan kuluttua
    - n Näin törmäyksen aiheuttama hukka-aika pienenee
  - n Kauanko kuunneltava?
    - n  $2^*$  maksimi etenemisviive solmujen välillä
- 
- A ei saa lopettaa ennenkuin törmäyssignaali olisi ehtinyt tulla!

Tietoliikenteen perusteet /2008/ Liisa Marttinen

28



### 3) Vuoronantoprotokollat

- n Yhdistä edellisten parhaita puolia
  - n Älä pidä kapasiteettia turhaan varattuna
  - n Älä aiheuta törmäystä
- n Pollaus
  - n Isäntäasema kysyy vuorotellen jokaiselta asemalta, onko sillä lähetettävää (vuorokysely, polling)
  - n Isäntä kuuntelee signaalia, osaa päätellä, milloin lähetys loppuu
- n Vuoromerkki
  - n Se, jolla on vuoromerkki, saa lähettää
  - n Jos ei ole lähetettävää, niin vuoromerkki siirtyy seuraavalle
- n Kummastakin useita versioita
  - n Ongelmia: lisäviive, 'single point of failure', ..
  - n Montako kehystä yhdessä vuorossa saa lähettää

Tietoliikenteen perusteet /2008/ Liisa Marttinen 30



## Linkkikerros

# Linkkikerroksen osoitteet

Ch 5.4



## Linkkikerroksen fyysinen osoite

- n 32 bitin IP-osoite verkkokerroksella
  - n Reitityksen tapa viitata koneeseen
- n Erilaisilla linkkikerroksilla omat tapansa osoittaa oikea linkki (~ verkkokortti)
  - n Siirtokehys on kuljetettava fyysisen linkin yli jollekin toiselle samaan verkkoon (LAN) kytketyistä laitteista
- n **MAC-osoite** (Media Access Control Address)
  - n Käytetään myös nimiä LAN-osoite, fyysinen osoite, laiteosoite, Ethernet-osoite, ...
  - n 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.

### n 48 bittinen (6 tavua)

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

n IEEE jakaa valmistajanumerot

### n Kiinteä

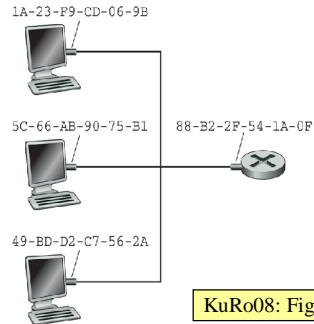
n Liitetty mukaan valmistuksessa

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

### n Ohjain

n Kuulee kaikki kanavalla kulkevat kehykset

n Välittää omalle koneelle vain sen MAC-osoitteella tai yleislähetysosoitteella FF-FF-FF-FF-FF-FF merkityt lähetykset



mm. Ethernet, Bluetooth, IEEE 802.11 langattomat verkot käyttävät



## Koneen MAC-osoitteen selvittäminen

### n Reititys: Paketissa on IP-osoite

n IP-osoitteen verkko-osa reitityksen perusteena

n Paketti saapuu kohdeverkon reitittimelle

### n Miten selvitetään IP-osoitetta vastaava MAC-osoite?

n Oikea verkko saavutettu, mutta mille koneelle?

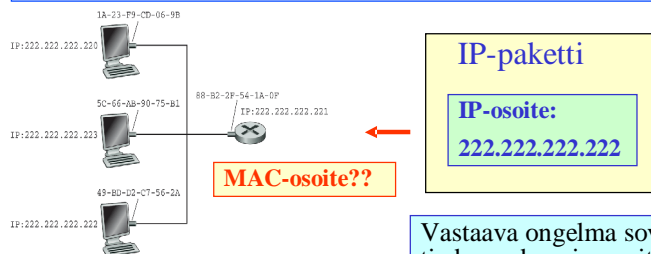


Figure 5.17 ♦ Each node on a LAN has an IP address, and each node's adapter has a MAC address.

Vastaava ongelma sovelluskerroksella: tiedossa domain-osoite (esim. URL), mutta tarvitaan IP-osoite!

Ratkaisu **DNS!**



## ARP-protokolla (Address Resolution Protocol)

n Ratkaisuna ARP-protokolla ja ARP-taulu  
n **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"?"

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

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

n **ARP-taulu** pitää tallessa kyselyjen vastauksia: IP-osoite, MAC-osoite, TTL)

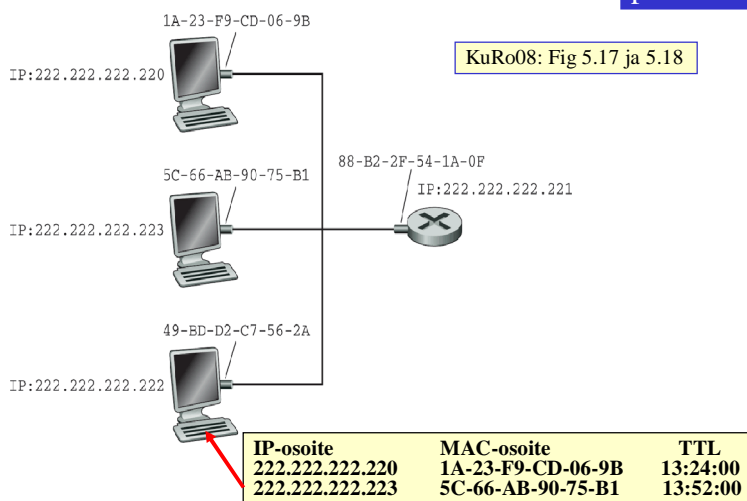
Kussakin koneessa (myös reitittimessä) jokaiselle aliverkolle oma taulunsa

Tiedot vanhenevat n. 20 minuutissa (time-to-live)



## MAC-osoitteet ja ARP-taulu

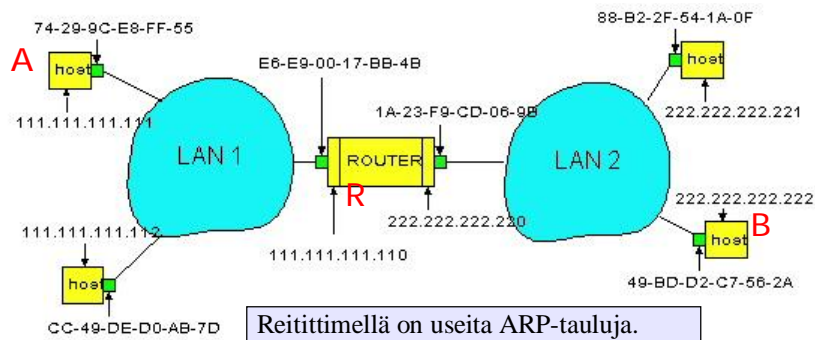
Minkä kerroksen protokolla?





## Lähtettäminen toiseen verkkoon (1)

- 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



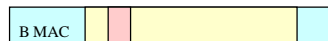
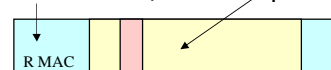
Tietoliikenteen perusteet /2008/ Liisa Marttinen

37



## Lähtettäminen toiseen verkkoon (2)

- Lähetäjä 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 siirtokeh്യksen.
- Reititin R**
  - Verkkokortti ottaa siirtokeh്യksen vastaan.
  - Ota IP-paketti keh്യksestä 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 siirtokeh്യks, osoitteena B:n MAC-osoite (data = IP-paketti)
- Vastaanottaja B**
  - Verkkokortti ottaa keh്യksen vastaan; ohjaa IP-paketin verkkokerrokselle.



Tietoliikenteen perusteet /2008/ Liisa Marttinen

38



## Linkkikerros

# Ethernet

Ch 5.5



## Ethernet

### n Yleisin lähiverkkoteknologia

- n Yksinkertainen, edullinen, helppo laajentaa
- n Lähiverkko syntyy kytkemällä koneet keskitimeen tai kytkimeen

### n IEEE:n standardoima LAN-verkko

- n Klassinen Ethernet (10 Mbps): CSMA/CD (kuulosteluväylä)
- n Fast Ethernet (FE, 100 Mbps), Gigabit Ethernet (GE), 10 Gigabit Ethernet, 100 GB Ethernet (pian??), 1 TB Ethernet (joskus??!)
  - Yleensä kytkentäisiä kaksipisteyhteyksiä

### n Muita lähiverkkostandardeja

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

## 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

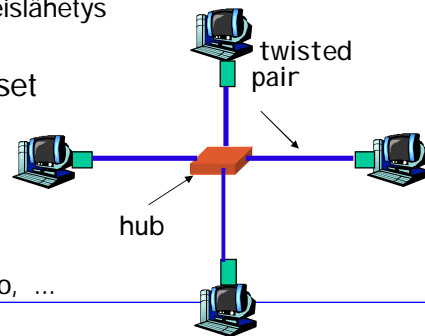
Verkkokortit käsittelevät törmäykset

Maks. 30 konetta / keskitin

Keskitin osaa jättää huomiotta vikaantuneen kortin

Kerää myös tietoa liikenteestä

Törmäysten lkm, keskim. kehyskoko, ...



## Gigabitin Ethernet (GE)

1 Gbps tai 10 Gbps

Edelleen sama kehysformaatti

Taaksepäin yhteensopiva

Yhteiskäyttöiset linkit edelleen OK

Koneiden yhdistely keskittimen välityksellä

CSMA/CD

Kaksipisteyhteydet

ei törmäyksiä

koneet yhdistetty **kytkimien** kautta

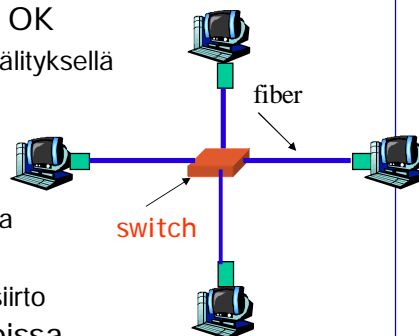
pitkät välimatkat mahdollisia

kaksisuuntainen täysivauhtinen siirto

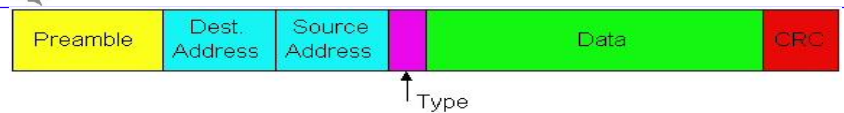
Käytetään yleisesti runkoverkoissa

verkkojen yhdistely (reititin -> reititin)

valokaapeli, myös cat5/cat6 parikaapeli



## Ethernet-kehys



### Tahdistuskuvio (preamble) (8 B)

7 tavussa 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)

tarkistusbitit, tahdistuskuvio mukana laskennassa

## Kehyksen minimipituus

### Data-osan pituus min 46 B

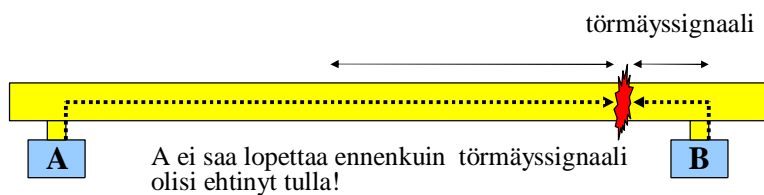
• Tarvittaessa täytetäviä (pad), jotka vastaanotto poistaa

### Lähtäjän ehdittävä huomata mahdollinen törmäys

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

- Alku perillä -> loppukin onnistuu

### Lähetysten minimikesto = $2 \times$ etenemisvive





## Epäluotettava siirto

- n Ethernet ei kättele, ei kuitaile
  - n Uudelleenlähetys vain, jos törmäys
- n Mutta tarkistussumma
  - n Hylkää kehyksen, jos siirrossa virheitä

### Verkkokerros

saa vain kelpollisia paketteja, antaa kuljetuskerrokselle

### Kuljetuskerros

TCP: huolehtii luotettavuudesta

UDP: välistä voi puuttua segmenttejä

### Sovelluskerros

Voi huolehtia halutessaan luotettavuudesta (vaikka UDP)



## Ethernet varaus: CSMA/CD

(klassinen Ethernet-verkko on yleislähetysverkko!)

- n Carrier Sense
  - n Kuuntele, onko väylä vapaa (96 b:n ajan)
  - n Jos vapaa, lähetä heti
  - n Muuten odota ja lähetä, kun linja vapautuu
- n Collision Detection
  - n Kun lähetetty, kuuntele onnistuiko
- n Törmäys?
  - n Huomaa signaalin voimakkuudesta
  - n Lopeta kehyksen lähetys heti
  - n Lopeta 48 bitin sotkusignaali (jam): muutkin huomaavat varmasti
- n Random Access
  - n Odota törmäyksen jälkeen satunnainen aika



## Törmäys

### n Binary Exponential Backoff

- n Kun kuorma kasvaa eli törmäykset lisääntyvät, uudelleenyritysten väli kasvaa

### n Odota törmäyksen jälkeen

- n  $N \cdot (512 \text{ bitin} = 64 \text{ tavun siirtoon kuluva aika} = \text{minimikehys})$
- n 1. törmäys:  $N = 0$  tai  $1$
- n 2. törmäys:  $N = 0, 1, 2$  tai  $3$
- n k:s törmäys:  $N = 0, \dots$  tai  $2^k - 1$
- n 10. törmäyksen jälkeen ei enää kasvata väliä  $[0-1023]$
- n 16 törmäyksen jälkeen luopuu ja ilmoittaa 'asiakkaalle' (eli verkkokerrokselle) epäonnistumisesta

10 Mbps:n linkillä 512 bitin siirtoon kuluu 51,2 mikrosekuntia

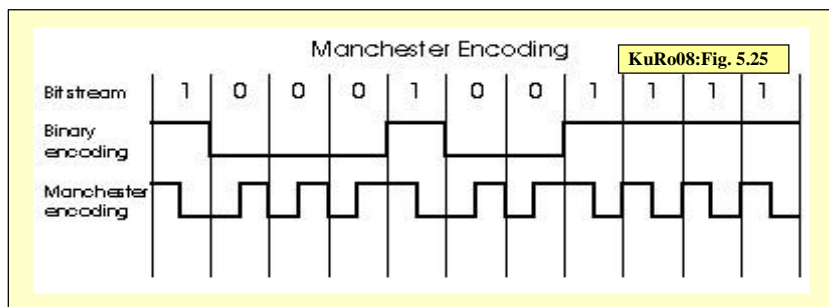


## Signaalin koodaus

### n Lähettäjän ja vastaanottajan kellopulssit on tahdistettava

### n Manchester-koodaus (10BaseT)

- n Ethernetissä ei ole kellopulssia, tahdistus osana bittijonoa
- n Jännitemuutos aina keskellä bittiä
  - 1-bitti: ylhäältä alas, 0-bitti: alhaalta ylös





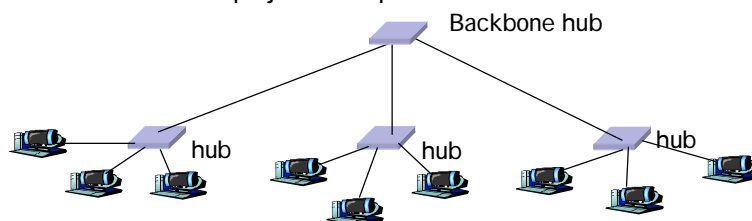
## Linkkikerros

# Keskitin, kytkin

Ch 5.6

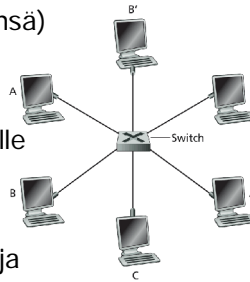
## Keskitin (hub)

- n Toimii fyysisellä kerroksella (layer-1)
  - n Käsittelee bittejä
- n Toistaa saamansa bitit heti kaikille muille linkeille
  - n Signaalin vahvistus
- n Yhteinen törmäysalue
  - n Sopii vain pieniin verkkoihin
- n Laitteet samanlaisia
  - n Ei esim. 10 Mbps ja 100 Mbps samaan



## Kytkin (switch)

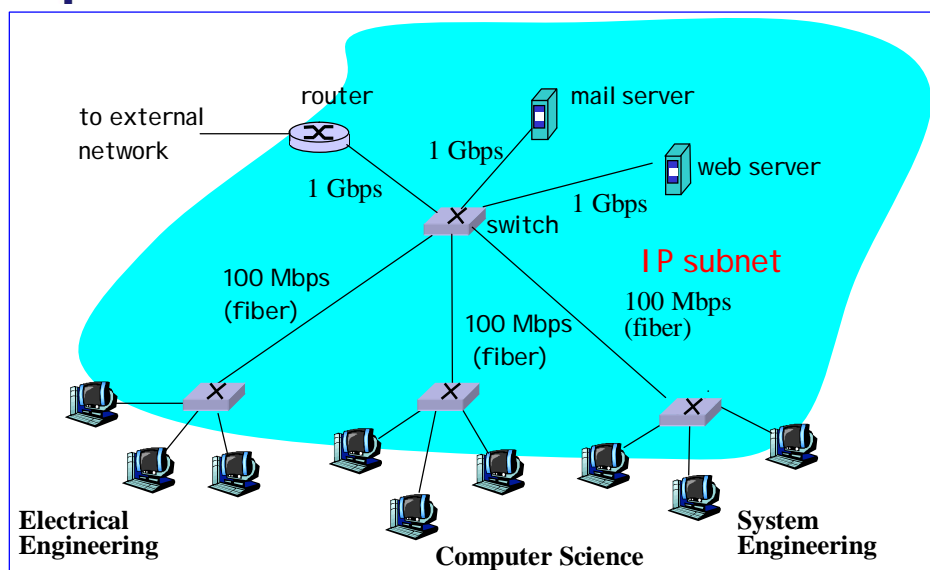
- n Toimii linkkikerroksella (layer-2)
  - n Käsittelee siirtokehyksiä, useita yhtäaikaista yhteyksiä
- n Vastaanottaa ja lähettää kokonaisia kehyksiä
  - n Etappivälitys (store and forward) (yleensä)
- n Ei törmäyksiä
  - n Suora piuha koneelta kytkimeen
  - n Kytkin lähettää ulos vain yhdelle puhalalle
- n Voi yhdistää erilaisia verkkosegmenttejä
  - n Kytkimessä esim. 10/100 Mbps portteja
  - n Puskurointia
- n Tuntumaton (transparent)
  - n Sopeutuu itse verkon muutoksiin
  - n 'plug-and-play, self-learning



A switch providing dedicated Ethernet access to six hosts

KuRo08: Fig. 5.24

## LAN, verkkosegmentit





## Kytkin ja kehyksen välitys

- n Miten kytkin osaa välittää kehyksen juuri oikeaan piuhaan?
- n Se kerää itse ('oppii') tarvittavat tiedot
  - takaperinoppimista** (backward learning): saapuva kehys kertoo, mistä linkistä **lähettäjä** saavutetaan
- n Ylläpitää kytkentätaulukkoa (MAC-osoite, linkki, TTL)
  - TTL-aikaleima: poista ne, joita ei ole käytetty esim. 60 minuutin aikana



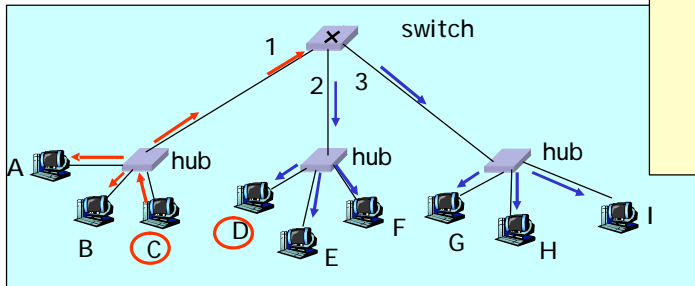
## Kytkentätaulu (switching table)

- n Aluksi taulu on tyhjä
- n Saapuva kehys
  - n **Lähteen MAC-osoite** x, kohteen MAC-osoite y, tuloportti p, yms
- n Lähde X ei ole taulussa
  - n Lisää (X, p, TTL) tauluun eli **kytkin oppii, että osoite X on saavutettavissa portin p kautta**
- n Kohde Y ei ole taulussa
  - n Lähetetään kehys kaikkiin muihin portteihin = **tulvitus** (flooding)
  - n Opitaan myöhemmin Y:n oikea portti jostain sen lähettämästä kehyksestä
- n Lähde X ja kohde Y ovat jo taulussa
  - n X ja Y samassa portissa => hylkää kehys (samassa alliverkossa)
  - n 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

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

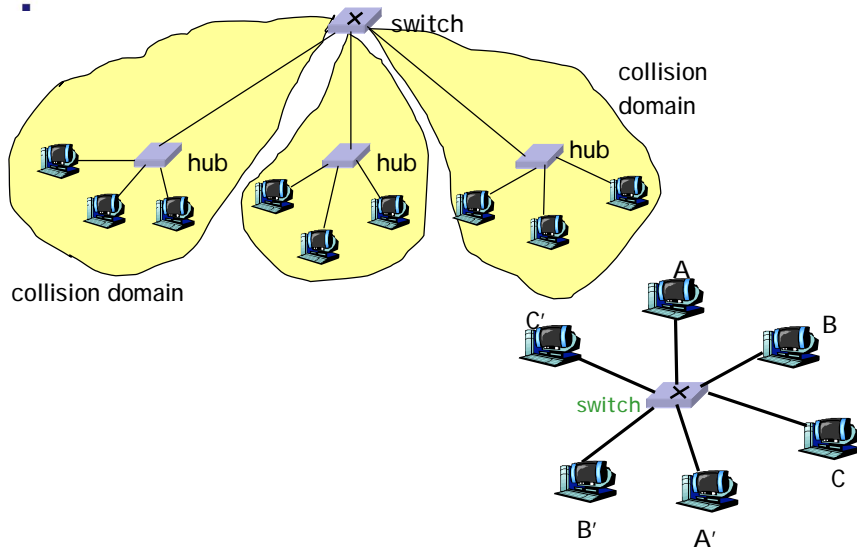


## Tulvitus (flooding)

- n Tulvitus voi olla ongelma
  - n Kehykset voivat jäädä kiertämään silmukoissa
  - n Koko verkko tukkeutuu
- n Siis silmukoita ei saa muodostua!
- n Verkon loogisen rakenteen pitää olla puu.
  - n Virittävä puu (Spanning tree)
  - n Lyhyimmin poluin virittävä puu Dijkstran algoritmilla



## Rajoitetut törmäysalueet / ei törmäyksiä



Tietoliikenteen perusteet /2008/ Liisa Marttinen

57



## Suorakytkentä (cut-through switching)

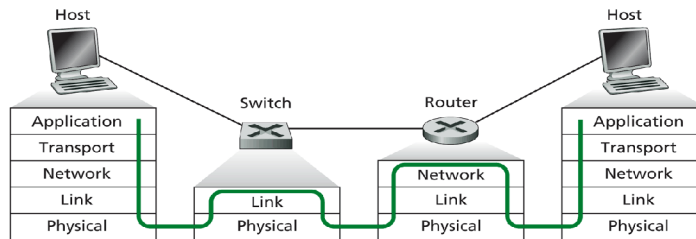
- n Jotkut kytkimet voivat välittää kehyksen bitit ulos sitä mukaa kuin itse ne saavat
  - n Välityspäätöksen tekoon riittää tutkia otsakkeesta kohdeosoite
  - n Ei siis enää etappivälitteistä (store-and-forward)
  
- n Pienentää latenssiaikaa
  - n Ei kuitenkaan mahdollomasti ...
  - n 100 Mbps:n linjalla odotusta maksimissaan noin 0.12 ms

Tietoliikenteen perusteet /2008/ Liisa Marttinen

58



## Vertailua



**Figure 5.33** ♦ Packet processing in switches, routers, and hosts

	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

KuRo08: Table 5.1



## Kertauskysymyksiä

- ☐ Miten lähiverkko rakennetaan?
- ☐ Reititin vs. kytkin vs. keskitin?
- ☐ IP-osoite vs. MAC-osoite?
- ☐ ARP-protokolla ja ARP-taulu?
- ☐ Takaperinoppiminen?
- ☐ KytKentätäulu?
- ☐ Bittivirheiden havaitseminen?
- ☐ Lähetykskanavanjako?
- ☐ CSMA/CD?

ks. kurssikirja s. 501

