



# Luento 10: Kaikki yhteen ja langaton linkki

To 28.11.2013

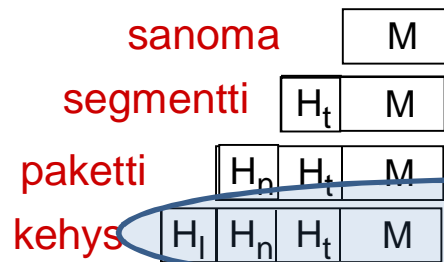
Tiina Niklander

Kurose&Ross  
Ch5.7 ja 6.1-6.3

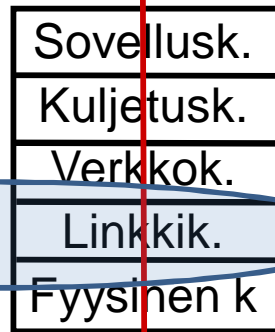
Pääasiallisesti kuvien  
© J.F Kurose and K.W. Ross,  
All Rights Reserved

# Luennon sisältöä

Lähettäjä (sender)



message,  
segment  
datagram  
frame



kytkin

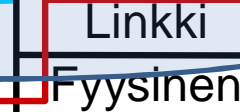
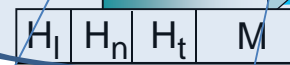
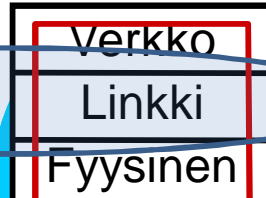
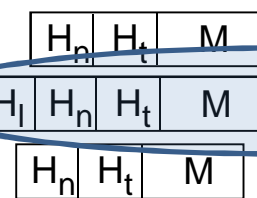
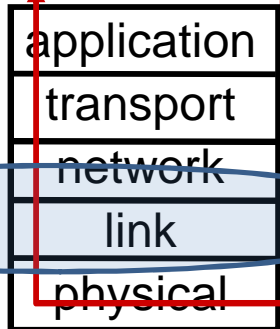
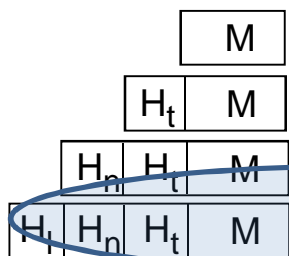
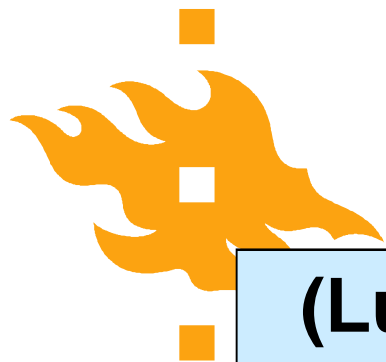


Fig 1.24 [KR12]

Vastaanottaja (recipient)



reititin



# Sisältö

**(Lukua 5:) Kaikki yhteen  
Keskitin ja kytkin**

**(6) Langaton: WLAN (ja CDMA)**



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

---

## Kytkin - esimerkki

Ch 5.6



# Kytkin: kehyksien välitys - mihin linkkiin?

---

Miten kytkin osaa välittää kehyksen juuri oikeaan piuhaan?

Se kerää itse ('oppii') tarvittavat tiedot

**takaperinoppimista** (backward learning): saa saapuvasta kehyksestä tiedon, mistä linkistä **lähettäjä** löytyy jatkossa

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

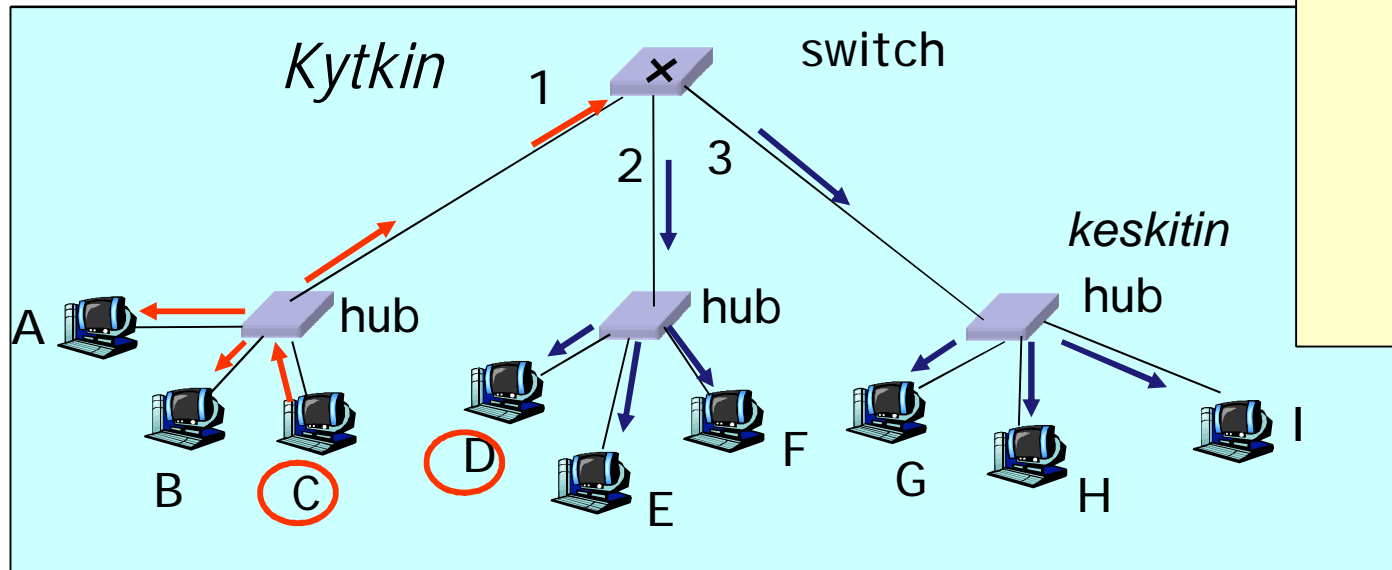
Lähde  $X$  ja kohde  $Y$  jo  
taulussa

$X$  ja  $Y$  samassa portissa =>  
hylkää kehys (on jo  
oikeassa verkon osassa)

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



# Esimerkki: C lähettää kehyksen D:lle

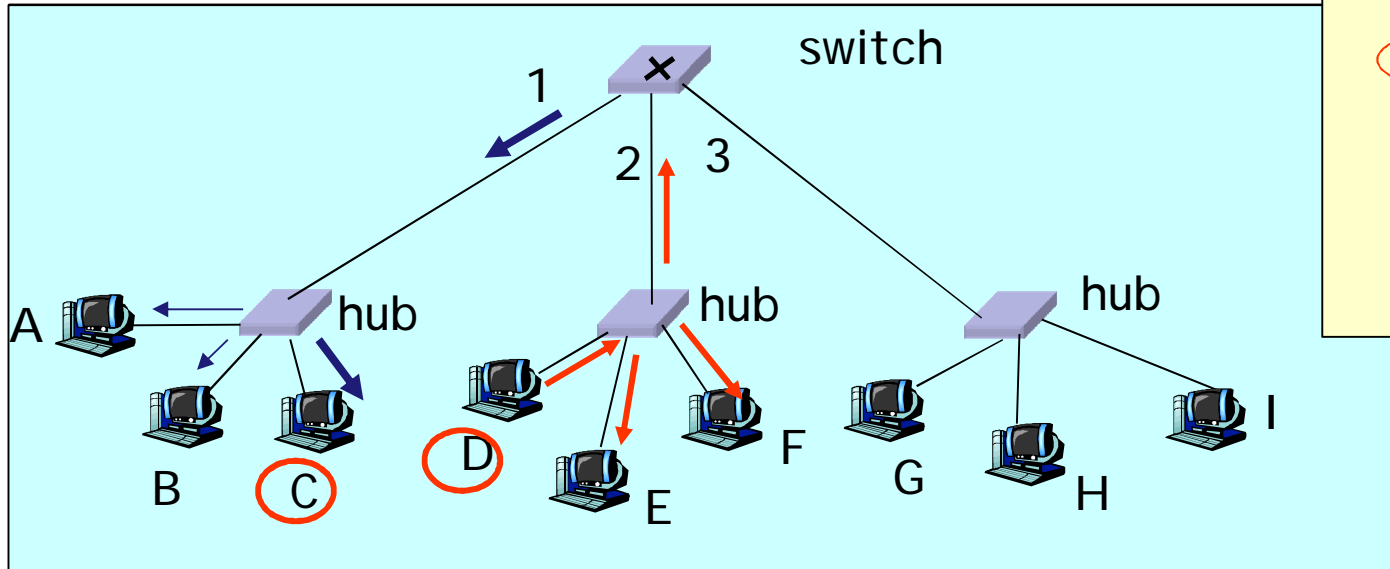


address	interface
A	1
B	1
E	2
G	3
C	1

- Kytkin 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

- Kytin 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

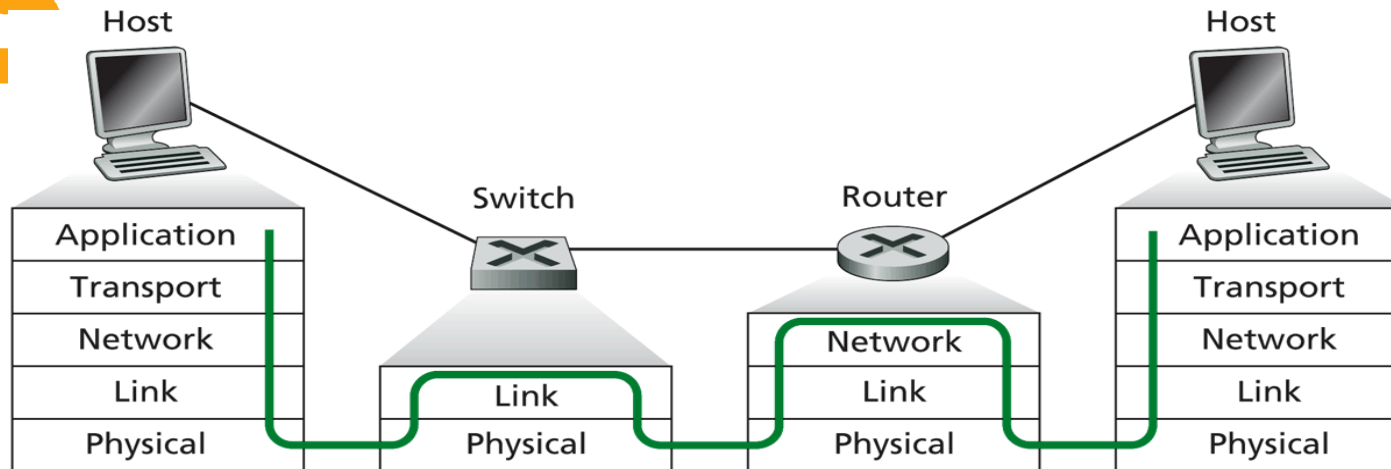


Fig 5.24 [KR12]

◆ 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



# Linkkikerros

---

## Kerrosten yhteistoiminta

Ch 5.7 (vain 6ed.)



# Skenaario: opiskelija yhdistää koneensa langattomaan verkkoon ja pyytää sivua [www.google.com](http://www.google.com)

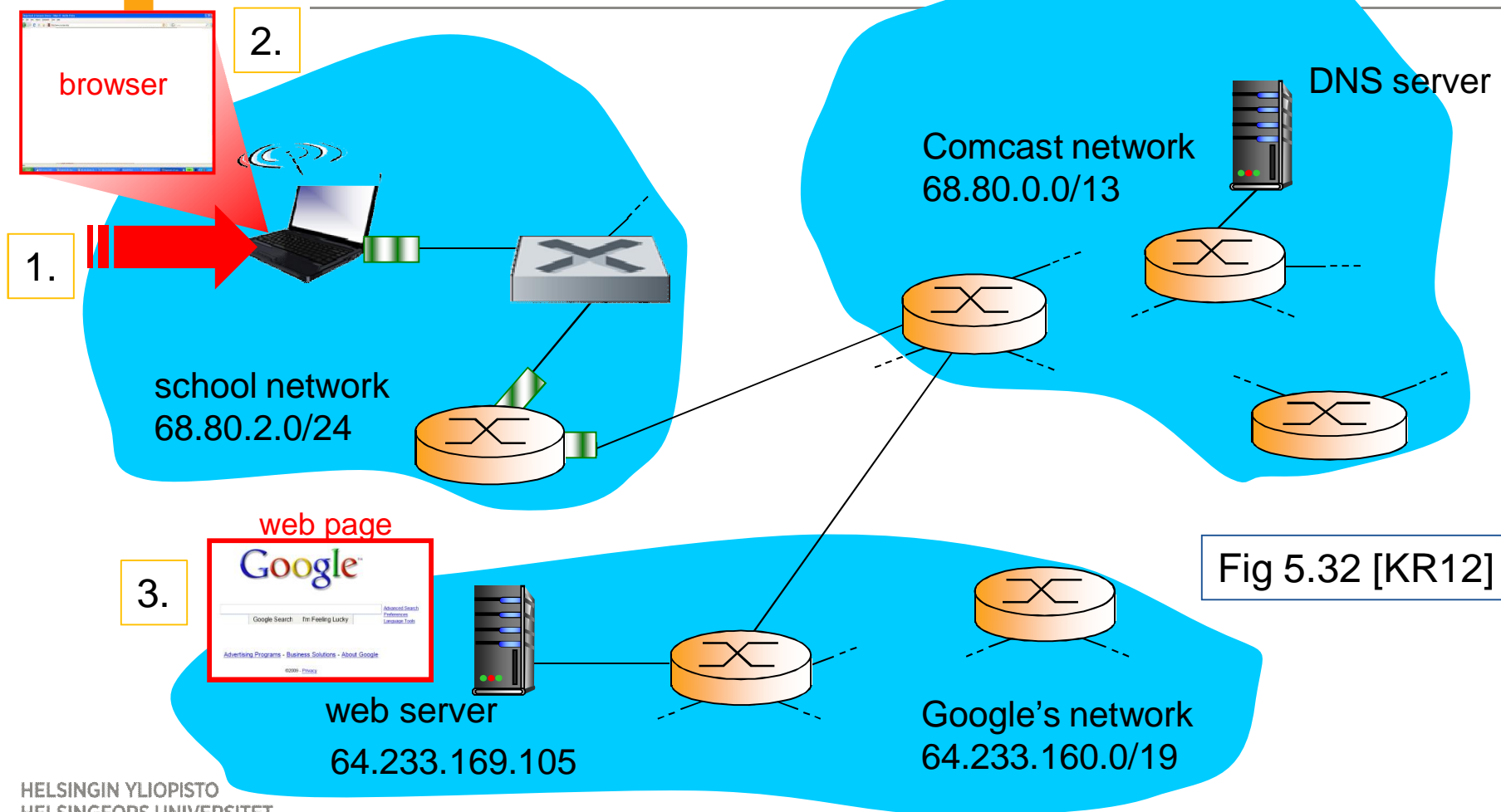


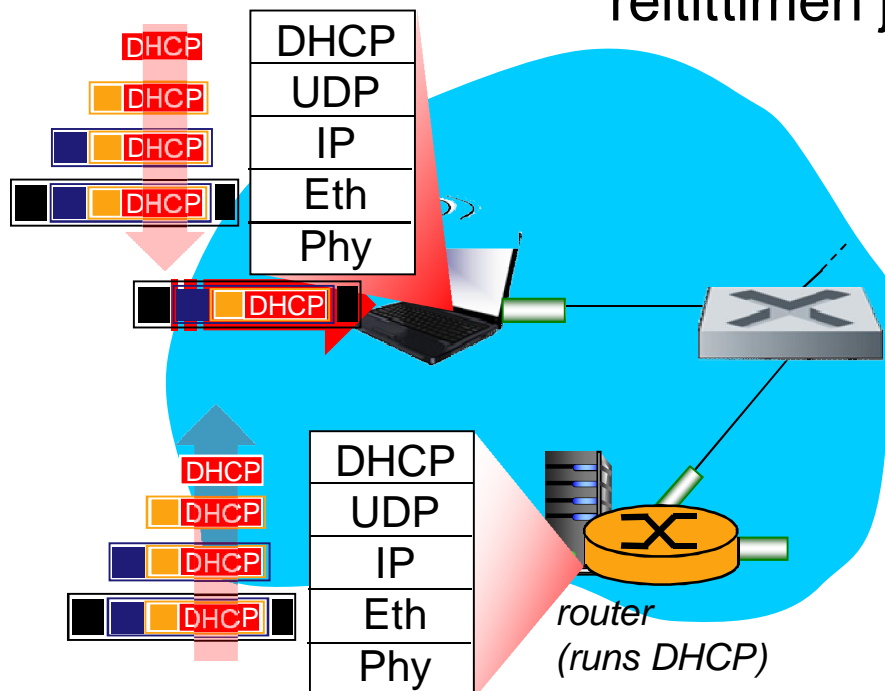
Fig 5.32 [KR12]



# Vaihe 1: yhteys verkkoon (Internet)

## DHCP pyyntö

❖ Kannettavan täytyy saada IP-osoite sekä reitittimen ja nimipalvelun osoitteet; *DHCP*

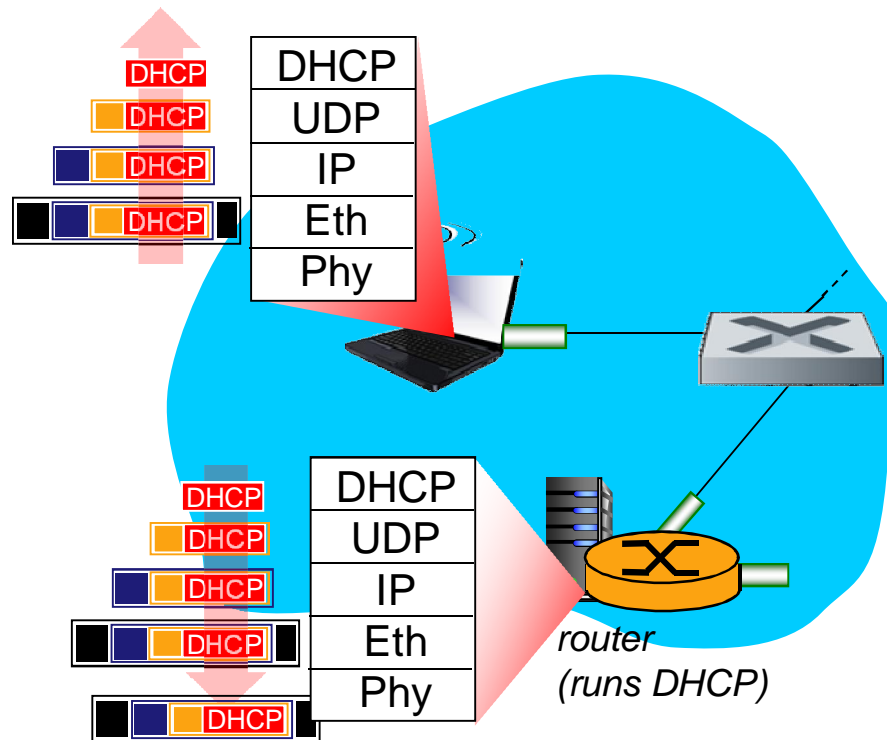


- ❖ DHCP-pyyntö paketoidaan (*encapsulated*) **UDP**, paketoidaan **IP**, paketoidaan **802.3** Ethernet
- ❖ Ethernet kehys (frame) lähetetään kaikille (*broadcast*) (kohde: FFFFFFFFFFFFFFFF), **DHCP** -palvelin (esim. reitittimessä) saa viestin
- ❖ Paketoinnit puretaan (*demuxed*) Ethernet → IP → UDP → DHCP



# Vaihe 1: yhteys verkkoon (Internet)

## DHCP vastaus



DHCP-palvelin muodostaa **DHCP ACK**-sanoman, jossa on kannettavalle IP-osoite, reitittimen osoite, aliverkkopeite + nimipalvelimen nimi ja osoite

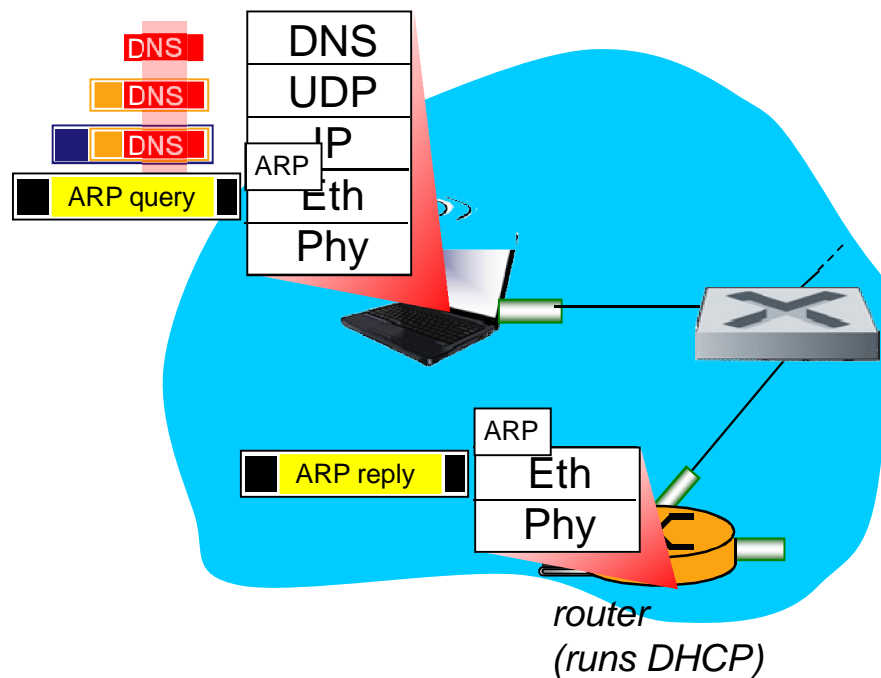
- ❖ Sanoma paketoidaan kerroksittain DHCP-palvelimella, kehys siirretään verkossa (**kytkin oppii!**) kannettavalle
- ❖ Kannettava (DHCP:n asiakas) vastaanottaa DHCP ACK sanoman

*Kannettavalla on nyt IP-osoite. Se tietää myös nimipalvelimen nimen ja osoitteen, sekä reitittimen ip-osoitteen*



## Vaihe 2: ARP (välivaihe, tarvitaan DNS-viestin lähetykseen verkkoon)

Ennen *HTTP*-pyynnön lähetystä tarvitaan *www.google.com*:in IP-osoite: *DNS*

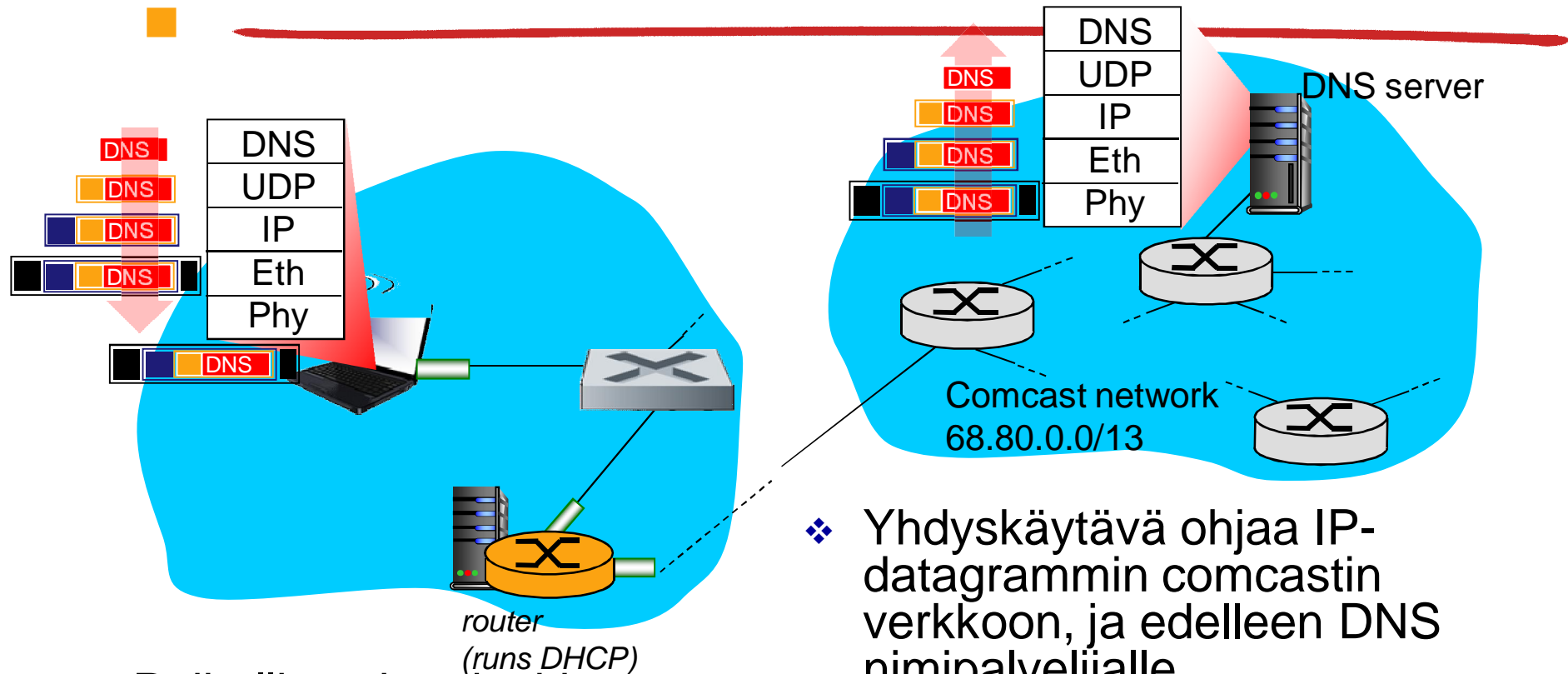


- ❖ Kannettava tekee DNS-pyyntöä, joka paketoituu UDP → IP → Ethernet. Mutta eri verkossa, joten tarvitaan vielä reitittimen MAC-osoite; **ARP**
- ❖ **ARP-kysely** yleislähetystenä, reititin vastaanottaa ja kertoo oman MAC-osoitteensa **ARP-vastausviestinä**
- ❖ Kannettavalla on nyt reitittimen MAC-osoite, joten se voi lähettää DNS-pyyntöä sisältävän kehyksen





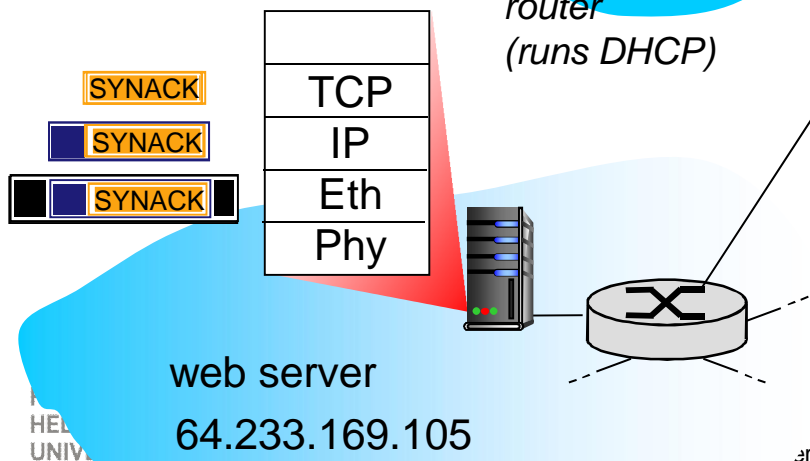
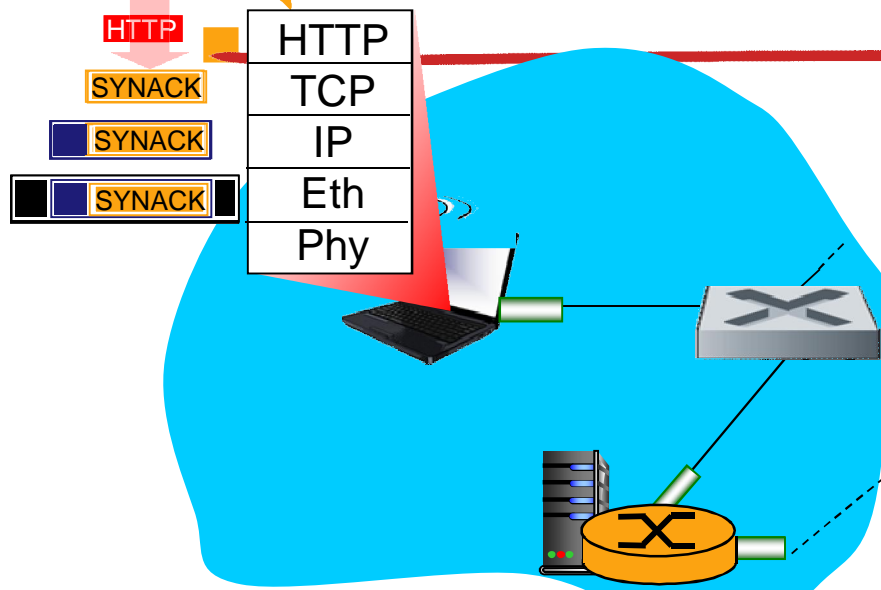
## Vaihe 2: jatketaan DNS-kyselyä. Nyt se voidaan lähettää



- ❖ Paikallisverkon kytkin ohjaa kehyksen yhdyskäytäväreitittimelle

- ❖ Yhdyskäytävä ohjaa IP-datagrammin comcastin verkkoon, ja edelleen DNS nimipalvelijalle
- ❖ DNS-palvelin vastaa asiakkaalle ja kertoo `www.google.com:n` IP-osoitteet

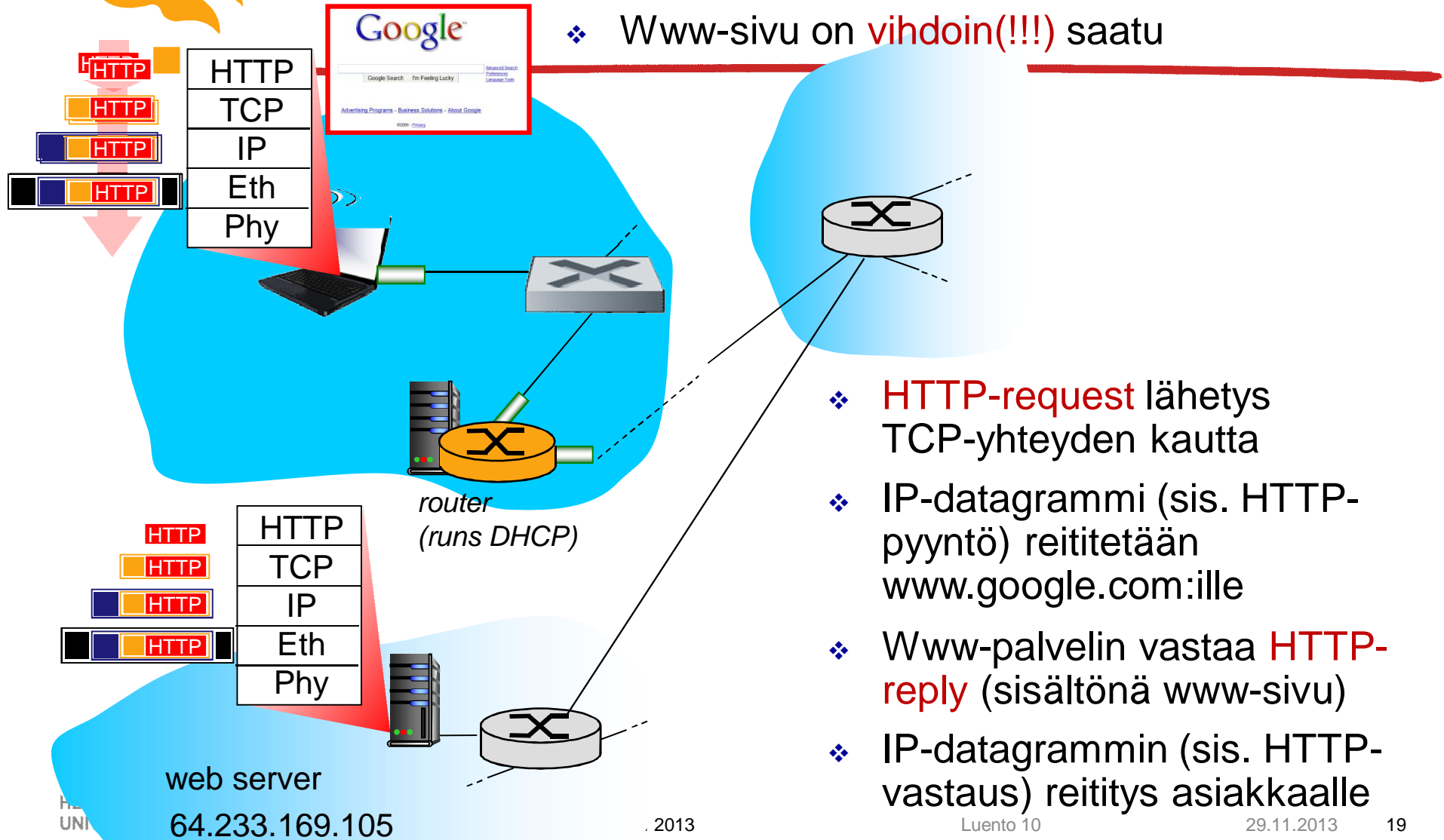
# Vaihe 2: TCP-yhteys



- ❖ HTTP-pyynnön lähetystä varten, asiakas avaa **TCP-yhteyden** palvelimeen
- ❖ TCP **SYN segmentti** (kättelyn 1. vaihe) reititetään www-palvelimelle
- ❖ Wwv-palvelin vastaa viestillä **TCP SYNACK**
- ❖ TCP **yhteys muodostettu!**

HELV  
UNIV  
web server  
64.233.169.105

# Vaihe 2: HTTP pyyntö ja vastaus





# Langaton verkko

---

## Langaton verkko

Ch 6.1, 6.2, 6.3-6.3.2

(ei: 6.2.1, 6.3.3-)



# Langattoman verkon komponentit

Fig 6.1 [KR12]

## Tukiasema

LAN-yhteys  
pääsy Internetiin

## Langattomat linkit

koneesta tukiasemaan  
koneesta koneeseen  
Rajattu kuuluvuusalue

## Isäntäkoneet

Laptop, PDA, IP-puhelin  
Suorittaa sovelluksia  
kiinteä tai liikkuva

## Haasteet

virhealtis linkki  
liikkuva työasema

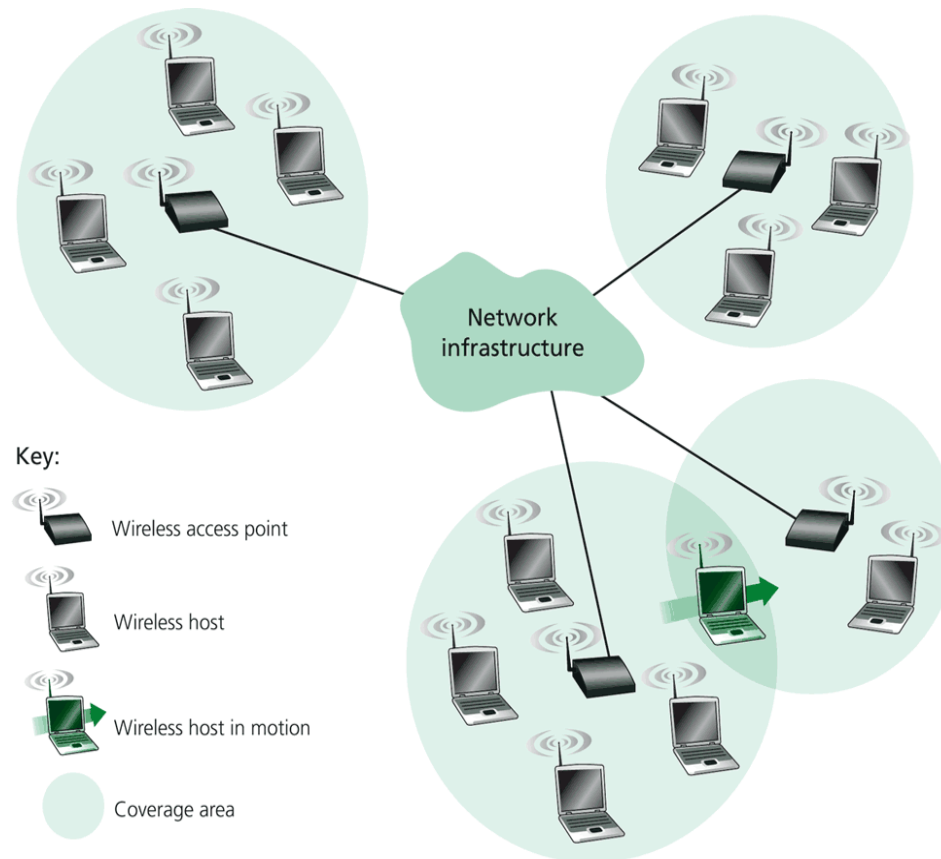
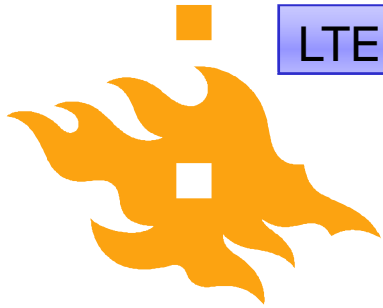


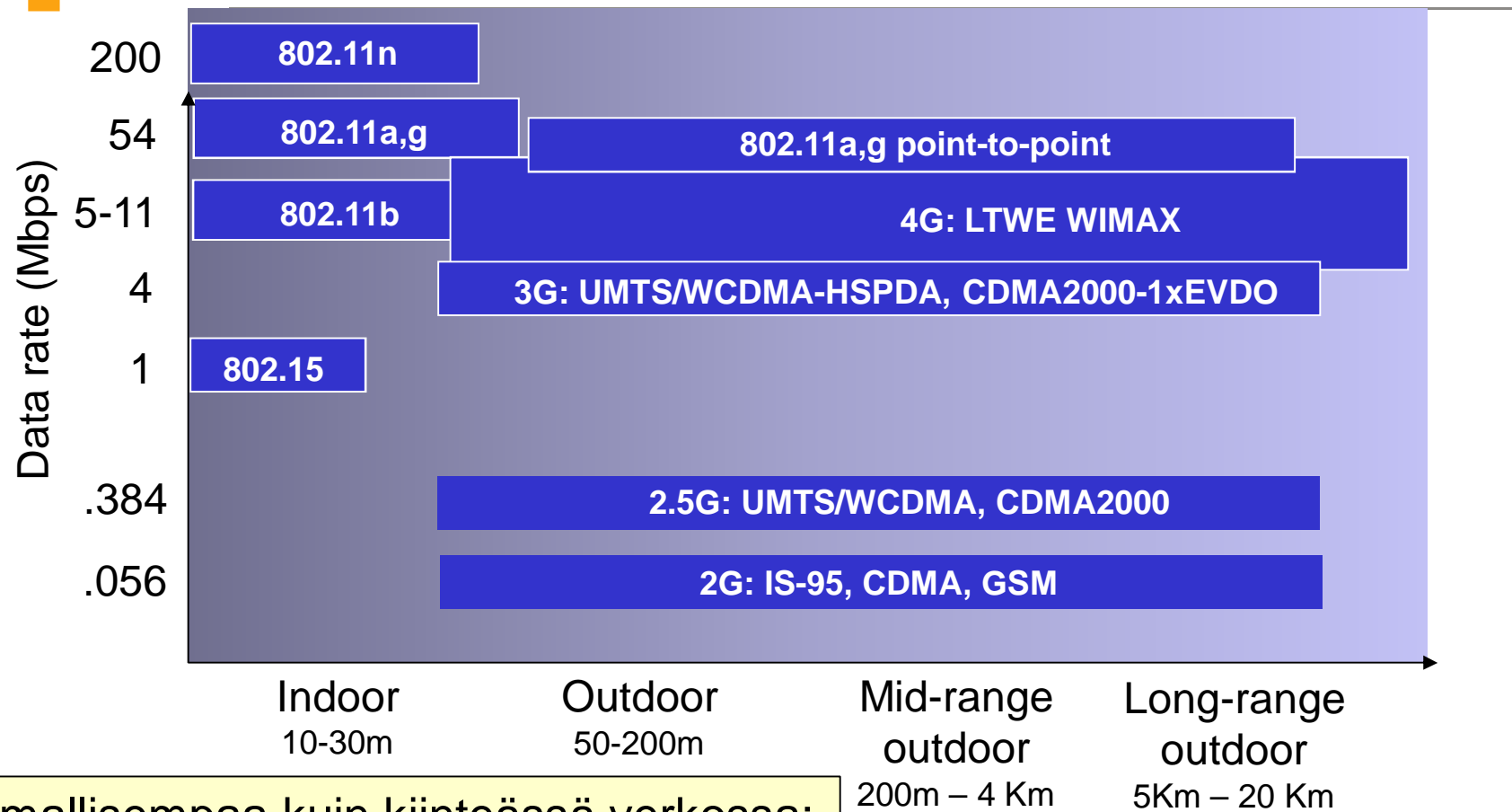
Figure 6.1 ♦ Elements of a wireless network

LTE Advanced: 4G, 1 Gbps



# Langattoman linkin ominaisuuksia

Fig 6.1 [KR12]



Ongelmallisempaa kuin kiinteässä verkossa:

- signaalin vaimeneminen, heijastukset
- muiden laitteiden aiheuttamat häiriöt



# Langattoman verkon tekniikat (IEEE)

---

<i>Tekniikka</i>	<i>IEEE standardi</i>	<i>Nimi</i>
Wireless personal area network (WPAN)	IEEE 802.15.1	Bluetooth
Low-rate WPAN (LR-WPAN)	IEEE 802.15.4	ZigBee
Wireless local area network (WLAN)	IEEE 802.11	WiFi
Wireless metropolitan area network (WMAN)	IEEE 802.16	WiMAX

---



# Ad hoc -verkko

Fig 6.8 [KR12]

Liikkuville koneille ...

Ei tukiasemia

Keskustelu omalla kuuluvuusalueella olevien koneiden kanssa

Ei valmiita palveluja

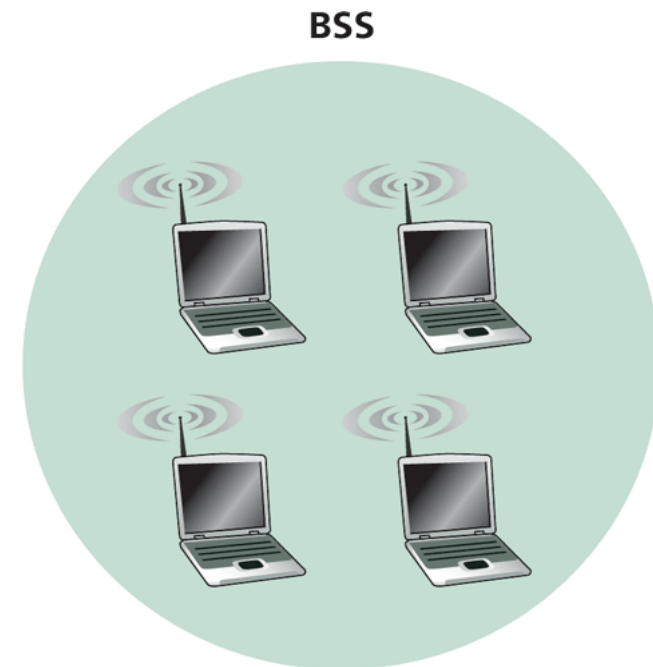
Reititys, IP-osoitteet, DNS, ..

Itseorganisoituva

Jonkun tuotettava tarvittavat palvelut

Ketä läsnä?

Reititys kuuluvuusalueelta toiselle?





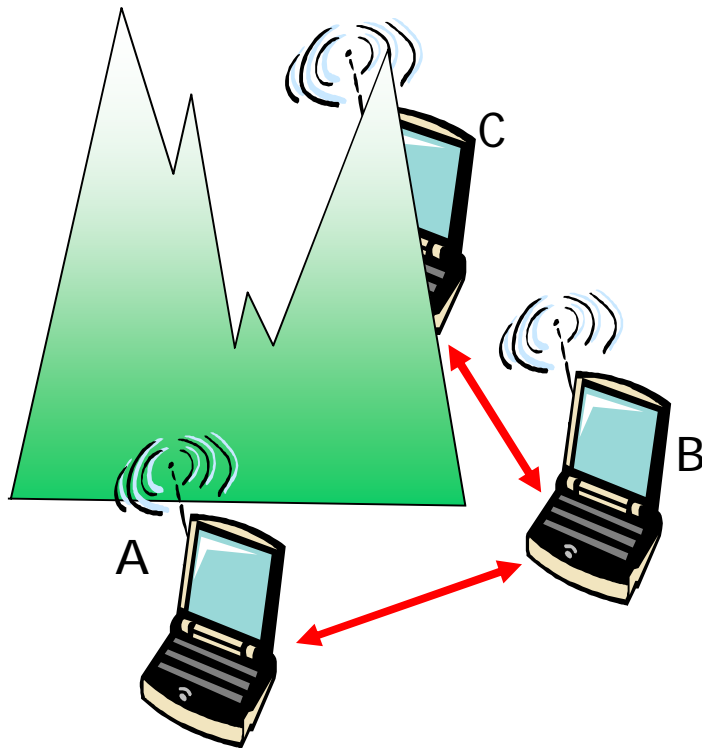


# Kätkeyn aseman ongelma

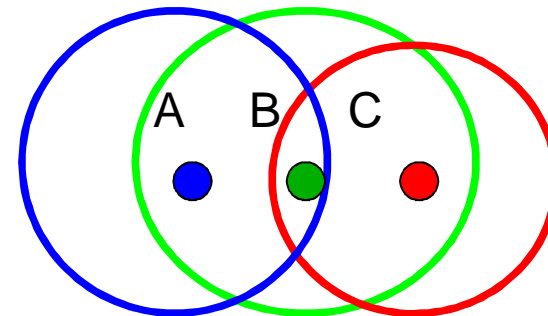
(Hidden terminal)

Fig 6.4a [KR12]

Asemat A ja C eivät kuule toisiaan eivätkä huomaa, milloin toinen lähettää samaan aikaan ja syntyy törmäys.



Miten asema voi tietää, menikö sen lähetyks perille?

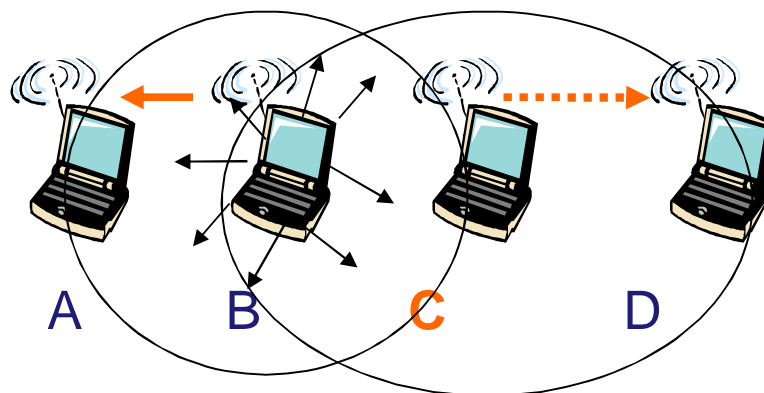




# Kuulee muun lähetyksen (Exposed terminal)

C ei voi lähettää D:lle, koska kuulee itse B:n lähetyksen eli joku on jo lähettämässä

Vaikka tämä lähetyk ei lainkaan häiritsisi C:n lähettämistä D:lle eikä B:n lähettämistä A:lle





## Langaton linkki

Kokeessa vain

- Perusidea
- Otsakkeiden osoitteet
- Ei pieniä detaljeja tai numeroita

# IEEE 802.11 WLAN (Wi-Fi)

Ch 6.3

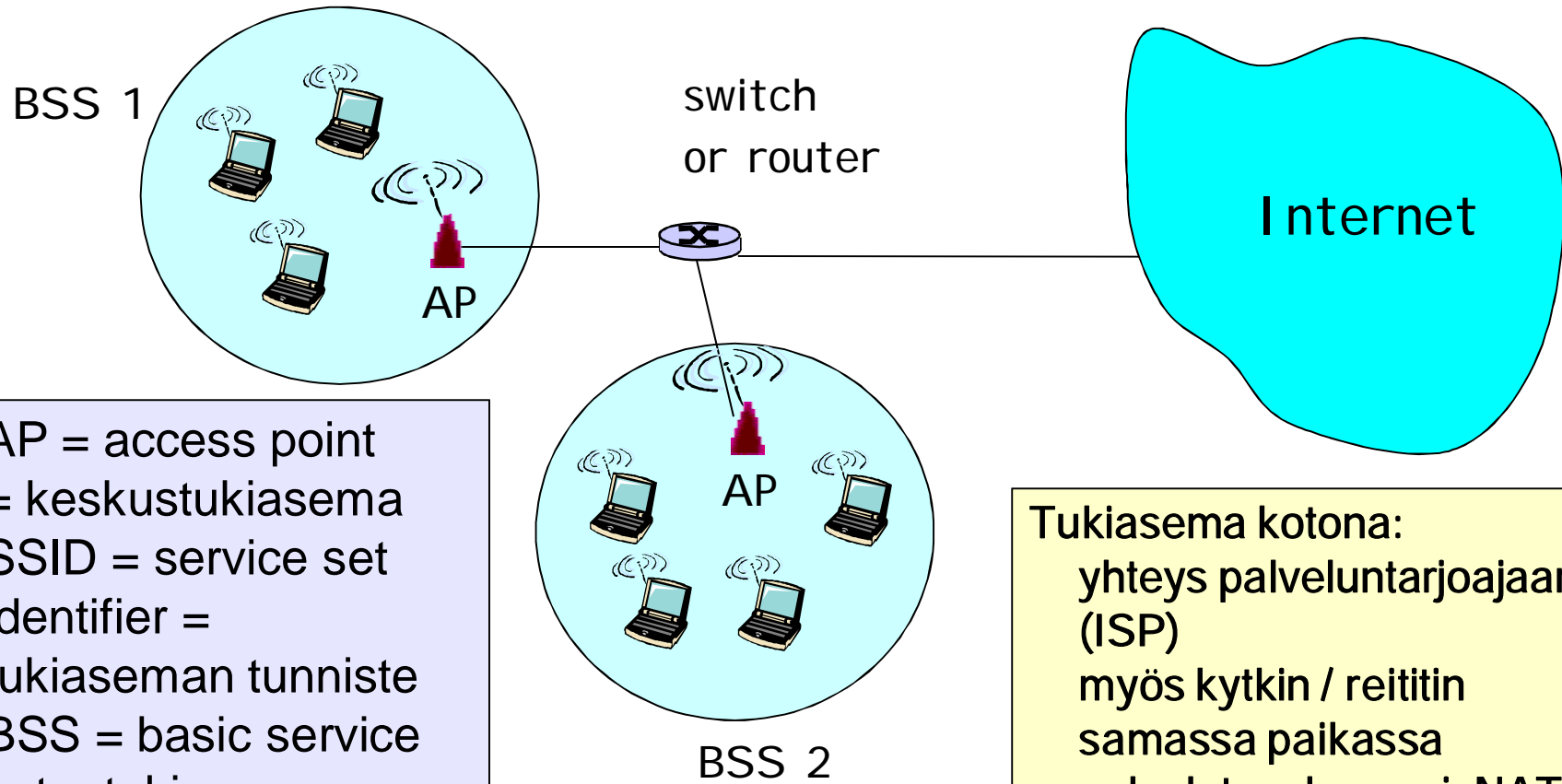
(ei tällä kurssilla 6.3.3-)



# IEEE 802.11 -lähiverkko

(infrastructure wireless LAN, Wi-Fi)

Fig 6.7 [KR12]

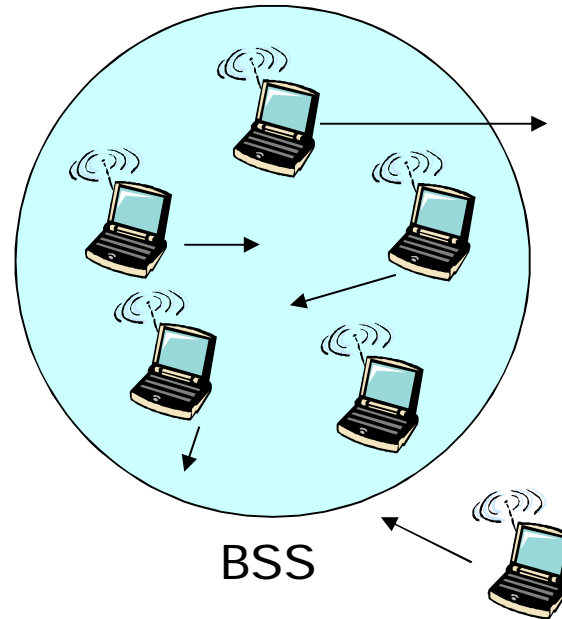


AP = access point  
= keskustukiasema  
SSID = service set  
identifier =  
tukiaseman tunniste  
BSS = basic service  
set = tukiaseman  
palvelemat koneet

Tukiasema kotona:  
yhteys palveluntarjoajaan  
(ISP)  
myös kytkin / reititin  
samassa paikassa  
palvelut: palomuri, NAT,  
DHCP



# Ad hoc-verkko



MANET (Mobile ad hoc network)

VANET (Vehicular ad hoc network)

Ei mitään infrastruktuuria ja solmut voivat liikkua

Solmujen on itse hoidettava kaikki toiminnot mm. **reititys**, jos eivät ole saman kuuluvuusalueen sisällä.



# IEEE 802.11: Kanavat

Standard	Frequency Range	Data Rate
802.11b	2.4 GHz	up to 11 Mbps
(802.11a	5 GHz	up to 54 Mbps)
801.11g	2.4 GHz	up to 54 Mbps

Alue 2.4 GHz - 2.2485 GHz

Jakaantuu 11 limittäiseen kanavaan (Eurooppa 13 ja Japani 14)

Esim. kanavat 1, 6 ja 11 eivät mene keskenään päällekkäin

Tukiaseman kanava on konfiguroitavissa

Naapuritukiasemalla saattaa olla sama kanava

Linkin käytössä **CSMA/CA**

Kaikissa sama linkkitason kehysrakenne



# 802.11: Kanavan valinta (1)

---

- Koneen kuuluvuusalueella voi olla useita tukiasemia
- Kone liittyy tiettyyn tukiasemaan (associate)
  - 'näkymätön' lanka ko. tukiasemaan
- Kone skannaa kanavat (passiivinen selaus)
  - Kuuntelee **merkkikehyksiä** (beacon frames), joilla tukiasemat mainostavat itseään
  - Kehyksessä tukiaseman nimi (SSID, Service set id) ja MAC-osoite
- Tai kone itse lähettää yleislähetyksenä kyselykehysten (probe) kaikille kantaman sisällä oleville tukiasemille. (aktiivinen selaus)
- Tukiasemat vastaavat ja kertovat nimensä ja MAC-osoitteensa.



## 802.11: Kanavan valinta (2)

---

- Standardi ei määrittele tukiaseman valintaa varten mitään erityistä algoritmia, vaan laitevalmistajat voivat toteuttaa sen eri tavoin
  - Yleensä valitaan voimakkaimmalla signaalilla lähettävä tukiasema
- Yhteys valittuun asemaan
  - Mahdollinen autentikointi (tukiasema konfiguroitavissa)
    - Käyttö vain sallituilta MAC-osoitteilta, tunnus, salasana, ..
- Saa asemalta IP-osoitteen DHCP:llä
- Saa asemalta DNS-palvelijan IP-osoitteen DHCP:llä

"WiFi Jungle"





# 802.11: Linkkitason protokolla (1)

---

CSMA kuten Ethernet (carrier sense multiple access)

Ei vuoronjakelua, vaan kilpailutilanne: lähetä, kun on lähetettävää (random access)

Kuuntele ennen lähetystä, että linkki on vapaa

Mutta ei CD (collision detection)

Ei huomaa törmäyksiä eikä keskeytä kehyksen lähetystä

Käyttää **kuittauksia**: jos kuittausta ei tule (=törmäys), lähetetään uudestaan

**Pyritään välttämään törmäyksen syntymistä**

**CSMA /CA (collision avoidance)**



## 802.11: Linkkitason protokolla (2)

---

### Miksi ei yritä huomata törmäystä?

Vaikea lähettää ja ottaa vastaan yhtäaikaan. Saapuva signaali on vaimentunut matkalla ja voi siksi olla hyvinkin paljon heikompi kuin lähetettävä signaali.

Ei voi huomata törmäystä, jossa toinen lähettävä solmu on oman kuuluvuusalueen ulkopuolella (**hidden terminal**)

Tai voi luulla törmäykseksi, vaikka lähetys ei sotkisikaan omaa lähetystä (**exposed terminal**)

# 802.11: CSMA/CA

## Lähetys

### 1. Jos kanava vapaa

Kuuntele DIFS aikayksikköä  
Lähetä kehys kokonaan

### 2. Jos kanava varattu

→ Käynnistä peruutuslaskuri (backoff)  
random(max), jota vähennetään vain  
kun kanava on vapaa,  
Lähetä, kun laskuri nollassa  
Jos ei tule kuittausta, niin yritä  
← uudestaan  $\text{max} = 2 * \text{max}$

## Vastaanotto

Jos kehys OK

Odota SIFS aikayksikköä

Lähetä ACK (linkkikerroksen ACK)

Fig 6.10 [KR12]

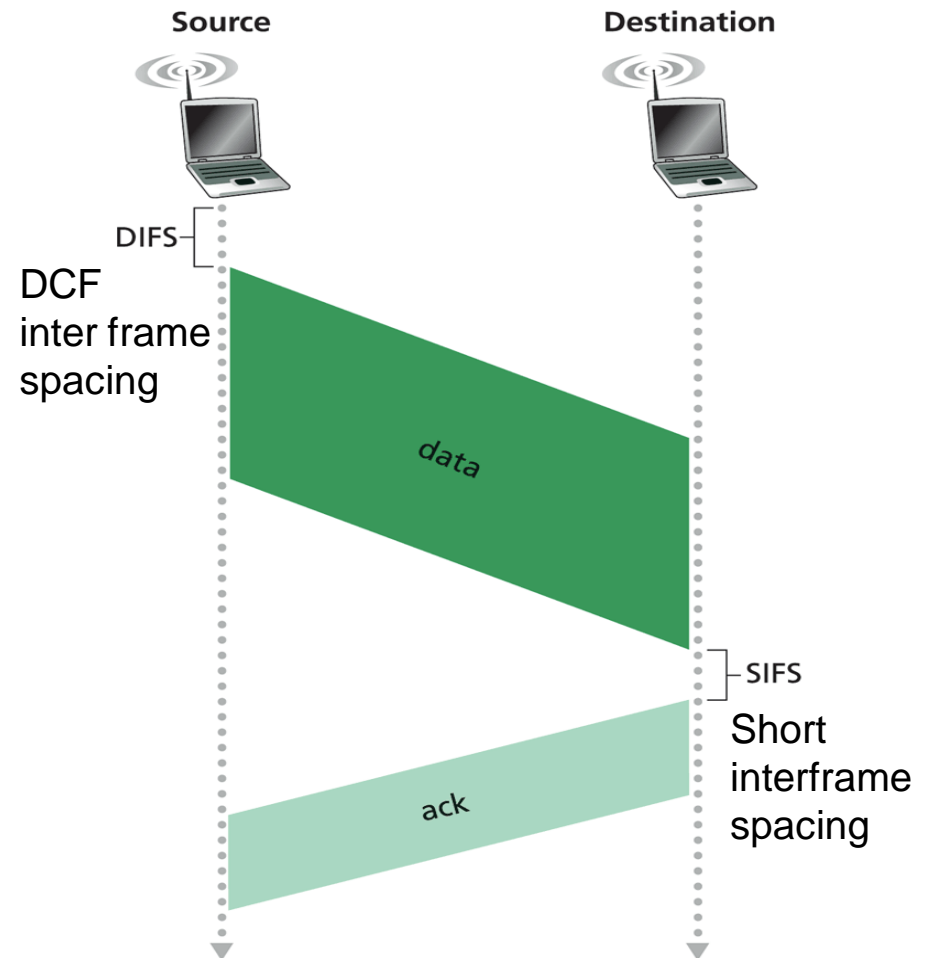
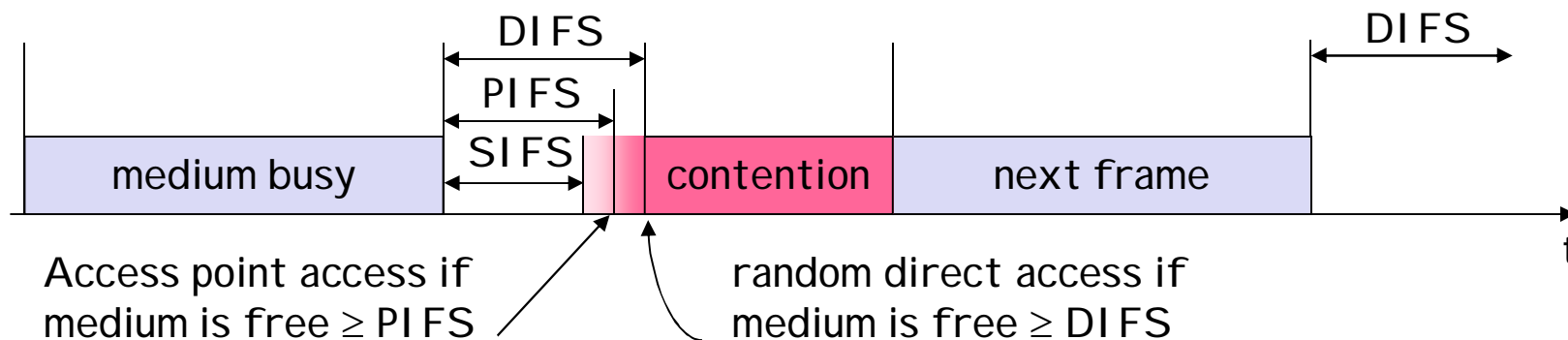


Figure 6.8 ♦ 802.11 uses link-layer acknowledgmer



# Kehysten väliä käytetään niiden priorisointiin

- Kehysväli (inter frame spacing, IFS): Korkean prioriteetin kehystyypeille sallitaan lyhyempi kehysväli
  - SIFS (Short Inter Frame Spacing)
    - Korkein prioriteetti; ACK, CTS, polling response
  - PIFS (Point Coordination Function Spacing)
    - Keskipälin prioriteetti, tietyille aikarajoitteisille viesteille, PCF:lle
  - DIFS (Distributed Coordination Function Spacing)
    - Alin prioriteetti, tavalliselle dataliikenteelle





# DIFS ja SIFS kestoja (mikrosek.)

Ei  
kokeessa

## DIFS DCF Interframe Space (DIFS)

Aika, jolloin odotetaan

$$\text{DIFS} = \text{SIFS} + (2 \cdot \text{slot time})$$

Tämä jälkeen voidaan lähettää frame

802.11b Slot time 20  $\mu\text{s}$ , DIFS 50  $\mu\text{s}$

802.11g Slot time 9 tai 20  $\mu\text{s}$ , DIFS 28 tai 50  $\mu\text{s}$

## SIFS Short Interframe Space

Aika datakehysten ja sen ackin välillä

Tarvitaan että ehditään siirtyä kuuntelusta lähetykseen (tai toisinpäin)

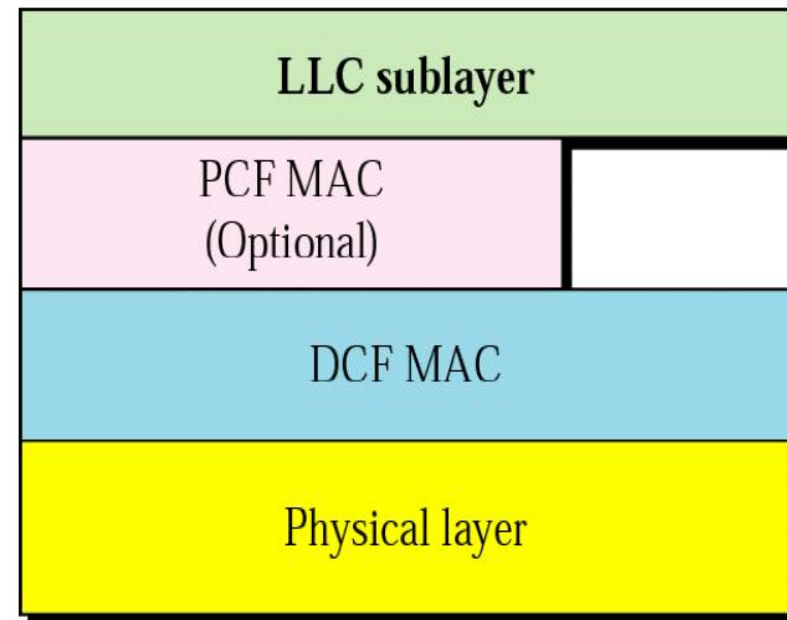
802.11b 10  $\mu\text{s}$ , 802.11g 10  $\mu\text{s}$



# Koordinoitiefunktiot

Ei  
kokeessa

- DCF (Distributed Coordination Function)
  - Kilpailua, ei priorisointia
- PCF (Point Coordination Function)
  - Keskitetty medianhallinta-funktio
  - Toimii infrastruktuuri-tilassa
  - Kukin tilaaja saa lähetyvuoron (myös DCF periaate)

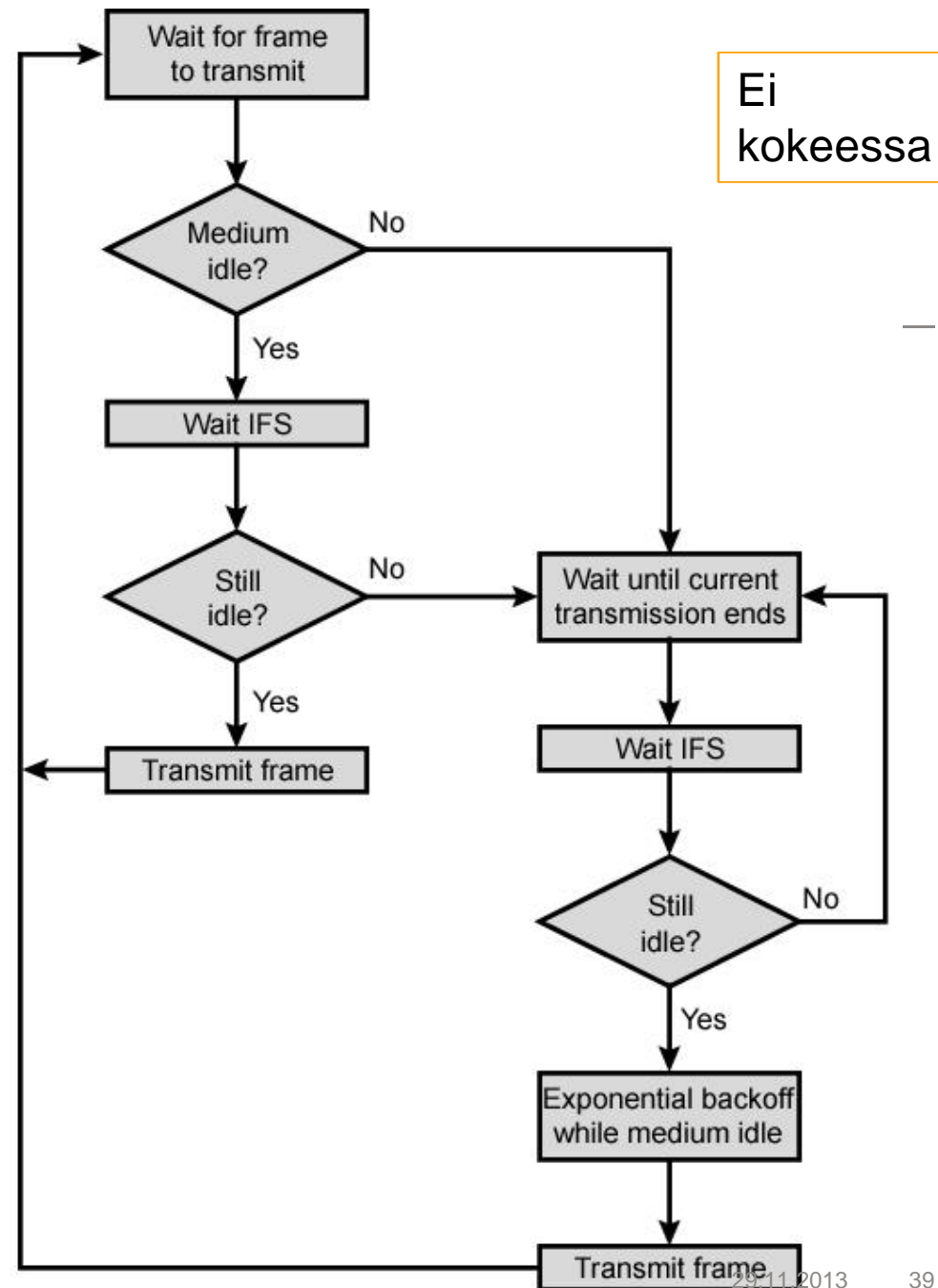




# IEEE 802.11 MAC Logic

IFS: Inter Frame Space (= DIFS, SIFS, or PIFS)

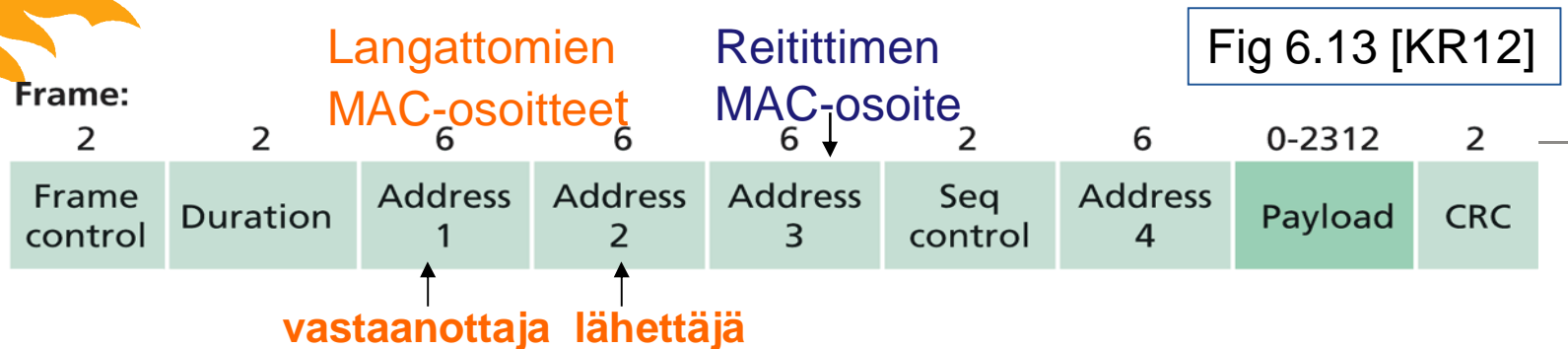
Reference: W. Stallings: Data and Computer Communications, 7th ed



Ei kokeessa



# 802.11: Kehyksen rakenne



## 4 osoitekenttää

isännän ja tukiaseman MAC-osoitteet (kenttä 1 ja 2)

Sen reitittimen osoite, jossa tukiasema on kiinni (kenttä 3)

Reitittimen ja tukiaseman välillä tavallinen kehys (esim. Ethernet)

Tukiasema on 'näkyvätön' reitittimelle, reititin luulee saavansa kehyksen suoraan isäntäkoneelta

Kenttä 4 käytössä vain ad hoc -verkossa

Request To Send  
Clear To Send

Lähetyksen kesto (duration)

Jos RTS/CTS kehys, varauksen kesto (lähetys-kuittaus)

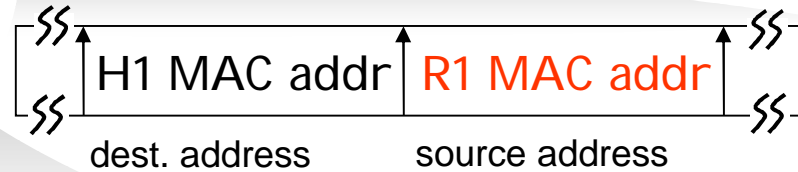
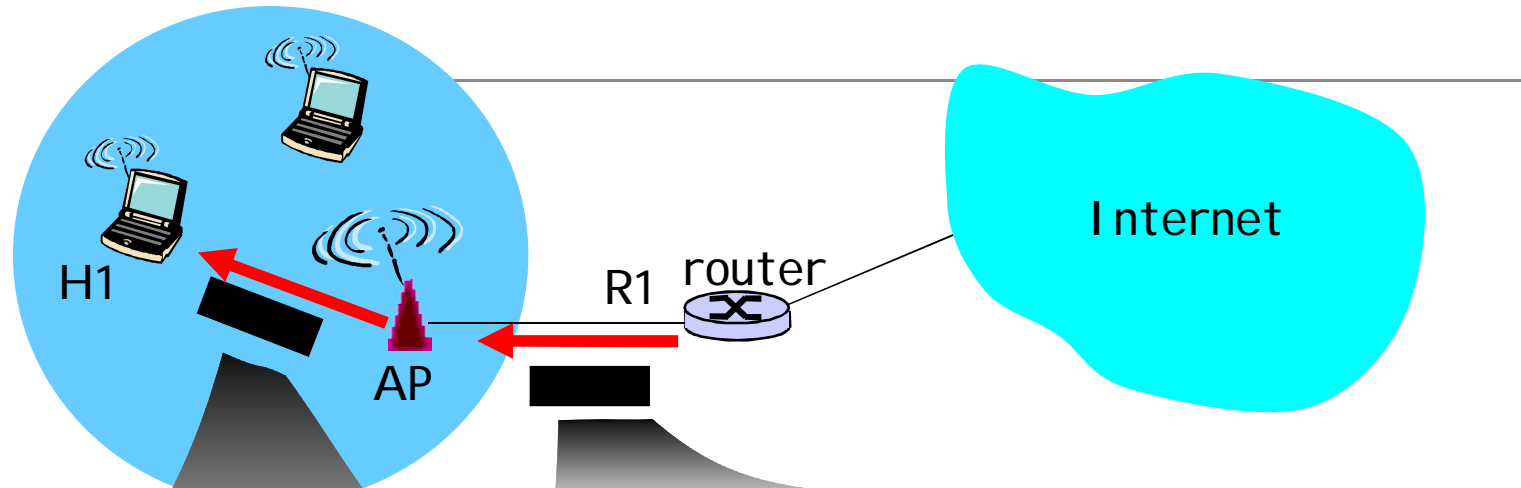
Seq control

Järjestysnumeroa tarvitaan kuittauksia varten

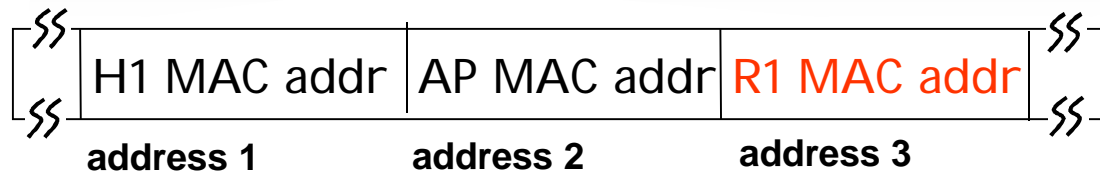




# Osoitteiden käyttö: Internetistä langattomalle



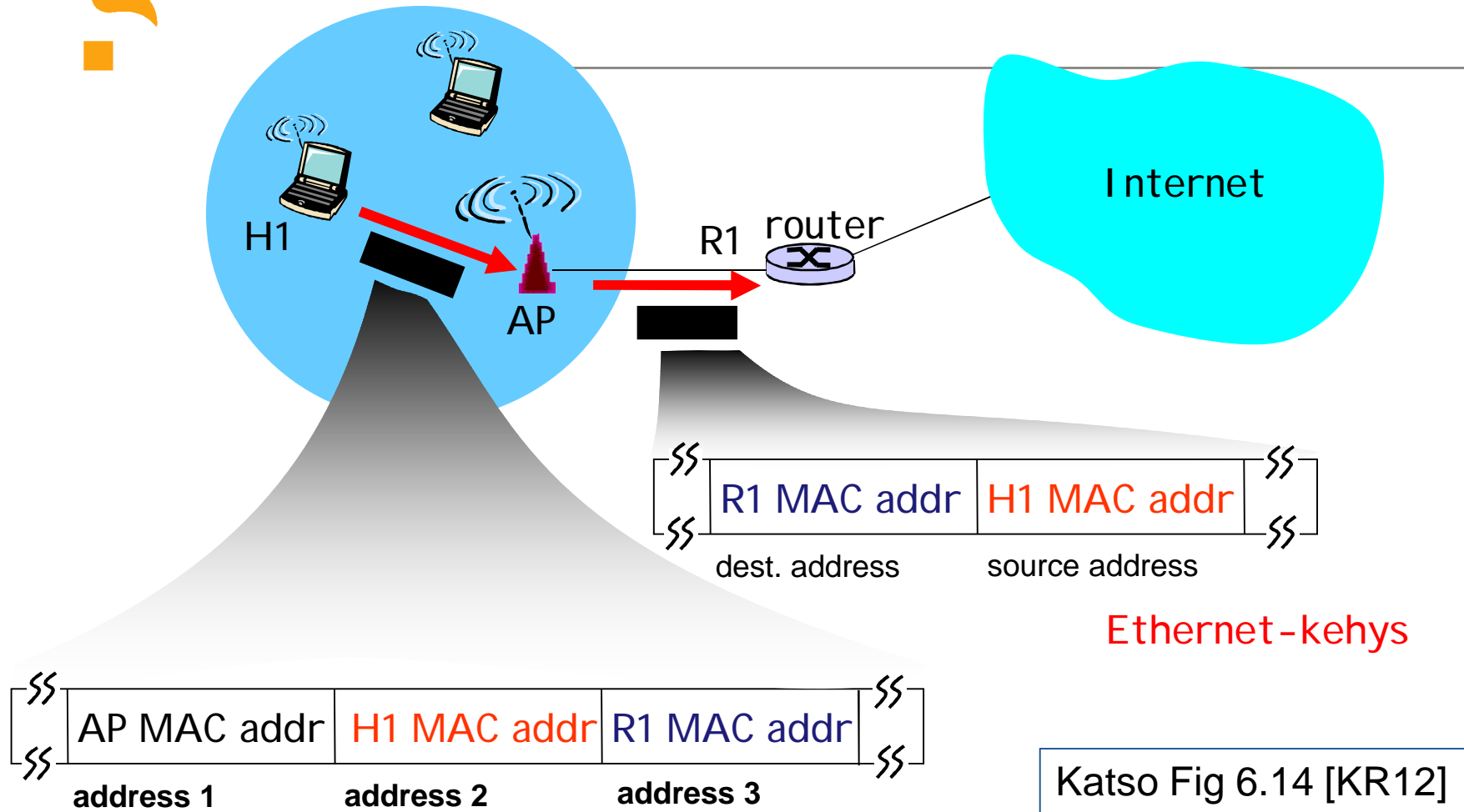
Ethernet-kehys



Katso Fig 6.14 [KR12]



# Osoitteiden käyttö: langattomalta Internetiin





# 802.11: Kehyksen rakenne

## Frame control

Type, Suptype

miten kehystä tulkittava: RTS/CTS/ACK/ data?

ToAP ja FromAP

miten osoitekenttiä tutkittava: lähettäjä /vastaanottaja  
ad hoc?

WEP (Wired Equivalent Privacy) ja WPA (WiFi Protected Acces)

Käyttääkö kryptausta (Huom. WEPin tietoturva surkea → ÄLÄ KÄYTÄ)

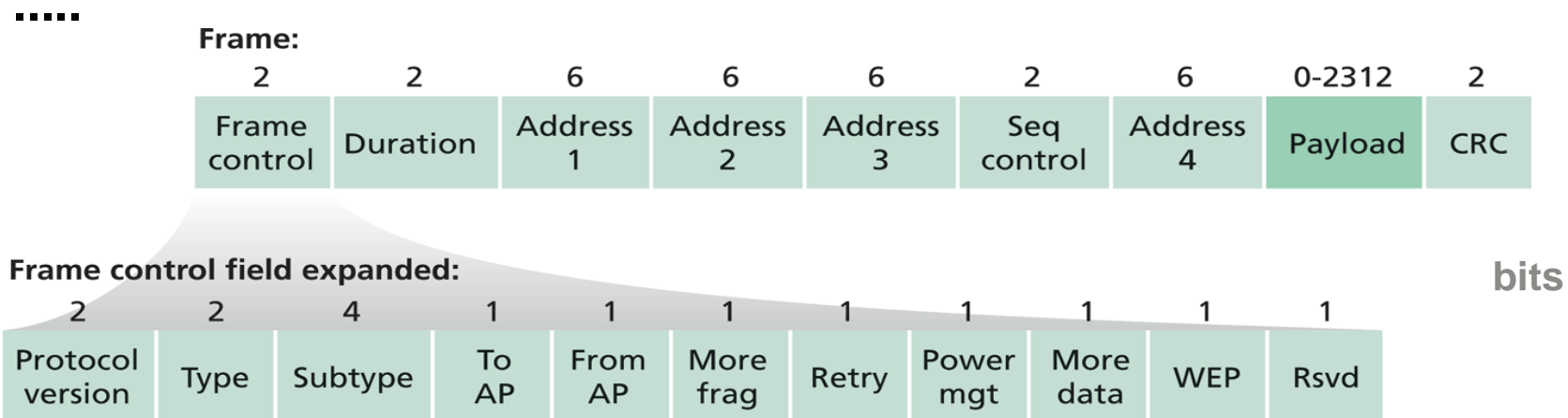


Figure 6.13

The 802.11 frame



# Kertauskysymyksiä

---

Miksi WLAN:ssa ei hyödytä käyttää törmäysten havaitsemista?

Miten sitten tiedetään, onko törmäystä tapahtunut?

Miten WLAN:ssa hoidetaan linkin yhteiskäyttö?

Miksi WLAN-kehyksessä kaksi osoitetta ei oikein riitä?

Onko törmäys lainkaan mahdollinen, jos käytetään RTS/CTS-varausmenetelmää?

Ks. myös kurssikirja s. 579-580



# CDMA (Code Division Multiple Access)

---

yksi kanava

usea samanaikainen lähetys

kukin koko kanavan taajuudella!

yhden bitin lähetyisaika jaetaan pienempiin osiin  
(aikasiruihin)

- 64 tai 128 sirua bittiä kohden

kullakin asemalla oma 'sirukuvio' 1-bitin lähetykseen

- 0-bitti on tämän komplementti (merkitään siksi **-1**)

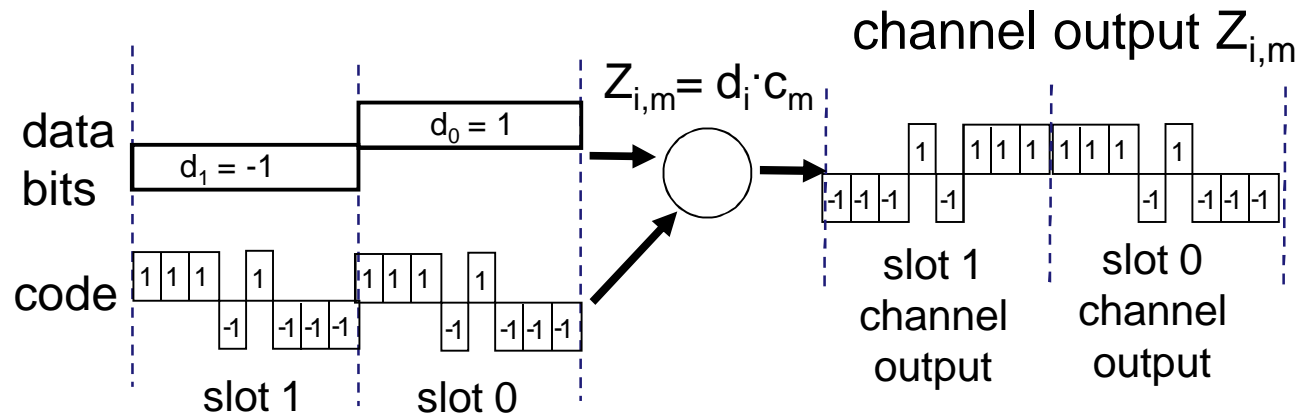
Bittikuviot ortogonaalisia: bittikuvioiden sisätulot nolliä



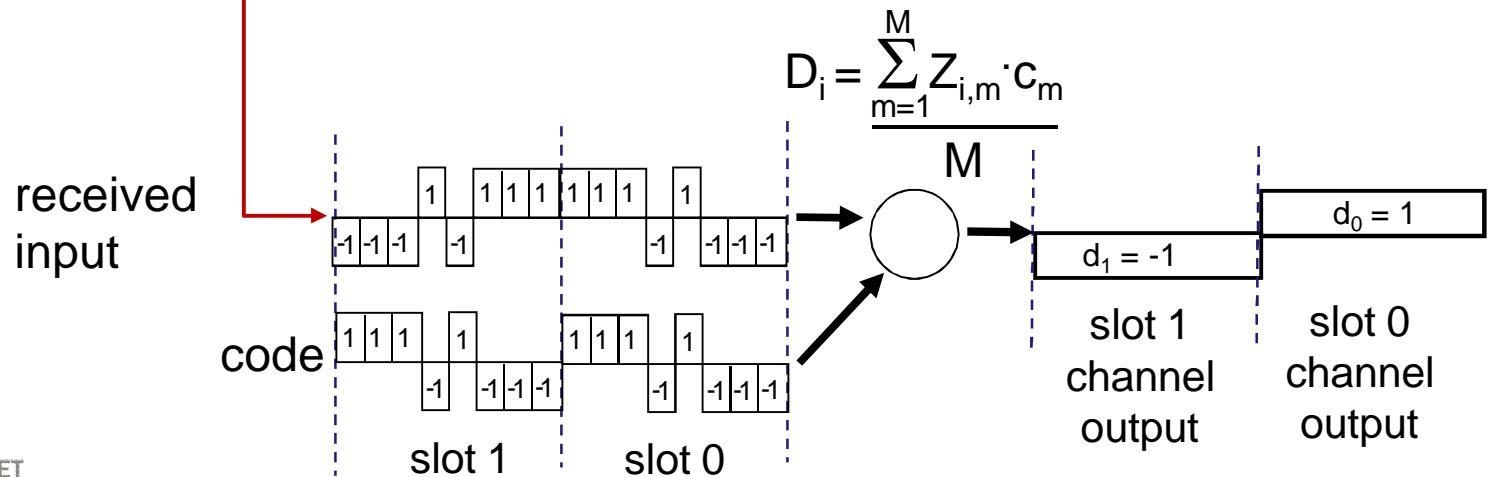
# CDMA encode/decode

Fig 6.5 [KR12]

sender



receiver



# CDMA: kahden lähettäjän interferenssi

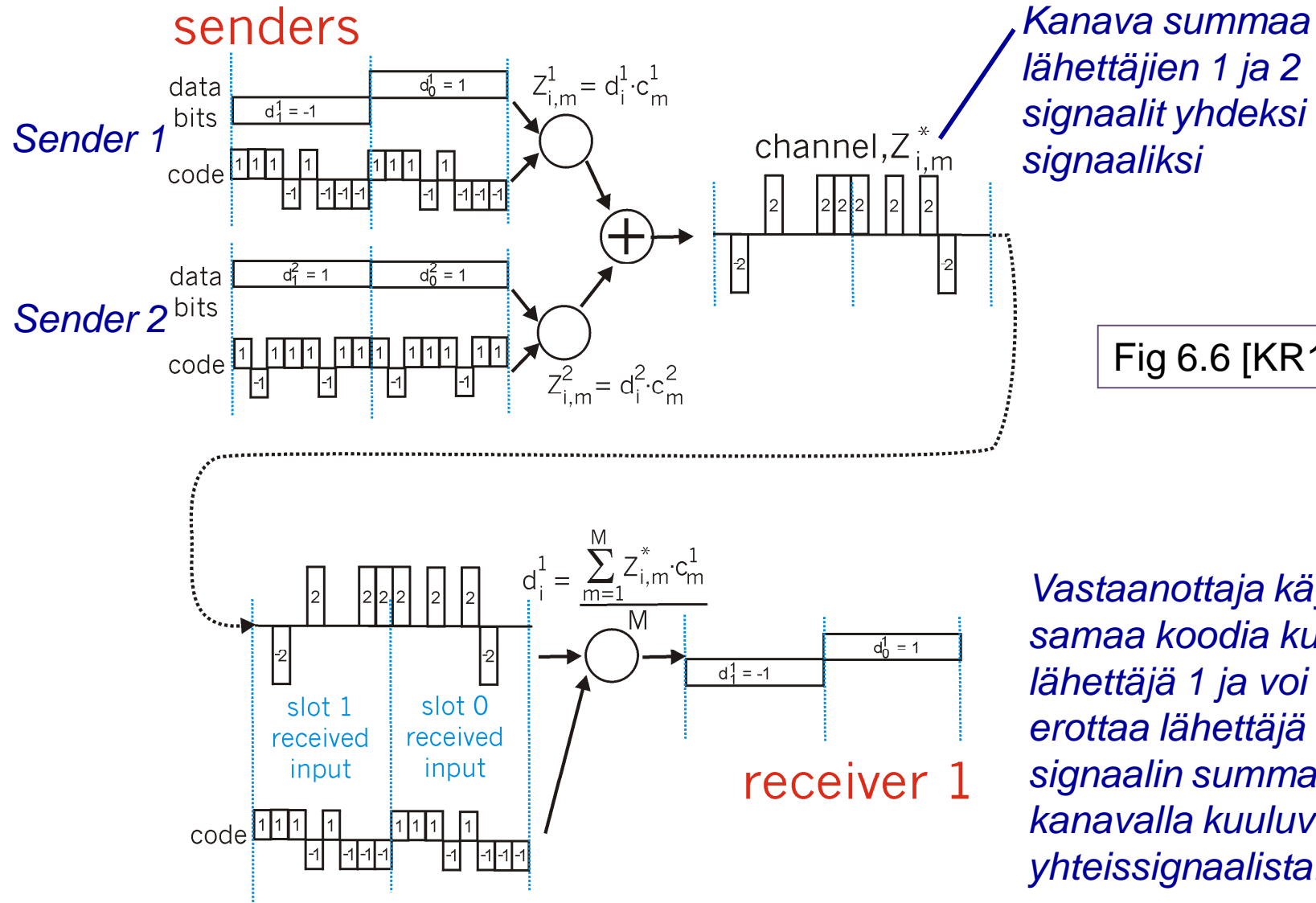


Fig 6.6 [KR12]



## Esimerkki: 8-siruinen koodi

aseman A 1-bitti: 00011011 = -1 -1 -1 1 1 -1 1 1

0-bitti: 11100100 = 1 1 1 -1 -1 1 -1 -1

aseman B 1-bitti: 00101110 = -1 -1 1 -1 1 1 1 -1

0-bitti: 11010001 = 1 1 -1 1 -1 -1 -1 1

aseman C 1-bitti: 01011100 = -1 1 -1 1 1 1 -1 -1

0-bitti: 10100011 = 1 -1 1 -1 -1 -1 1 1

aseman D 1-bitti: 01000010 = -1 1 -1 -1 -1 -1 1 -1

0-bitti: 10111101 = 1 -1 1 1 1 1 -1 1

$0 = -1$
$1 = 1$

Ps. Oikeasti käytetään 64 tai 128 sirua





# Kaikki sirukuviot parittain ortogonaalisia:

---

$$A \bullet B = 0 = 1/m \sum A_i B_i \text{ (sisätulo)}$$

$$A \bullet A = 1$$

$$-A \bullet A = -1$$

=> yhteissignaalista löydetään eri asemien omat lähetykset!

$$A:n \text{ 1-bitti: } 00011011 = -1 \ -1 \ -1 \ 1 \ 1 \ -1 \ 1 \ 1$$

$$B:n \text{ 1-bitti: } 00101110 = -1 \ -1 \ 1 \ -1 \ 1 \ 1 \ 1 \ -1$$

$$A \bullet B = 1+1+-1+-1+1+-1+1+-1$$

$$= 0 \Rightarrow \text{keskenään ortogonaalisia}$$



# Yhteissignaali

---

- kukin asema lähettää omat 1-bittinsä ja 0-bittinsä
- kun moni lähettää samanaikaisesti tuloksena on yhteissignaali S.
  - lähetettyjen signaalien ‘summa’
- aseman datan ‘purkaminen’ yhteissignaalista
  - $A$  = aseman oma bittikuvio
  - $S \bullet A$  tuottaa aseman lähettämän bitin
    - kerrottuna bitin aikasirujen lukumäärällä



## Esimerkki: Mitä C lähetti?

---

$$S = (-2 \ -2 \ 0 \ -2 \ 0 \ -2 \ 4 \ 0)$$

$$C = (-1 \ 1 \ -1 \ 1 \ 1 \ 1 \ -1 \ -1)$$

$$S \bullet C = (2 \ -2 \ 0 \ -2 \ 0 \ -2 \ -4 \ 0) \\ = -8 \Rightarrow -1$$

eli C lähetti 0-bitin



## Esimerkki jatkuu: Mitä B lähetti?

---

$$S = (-2 \ -2 \ 0 \ -2 \ 0 \ -2 \ 4 \ 0)$$

$$B = (-1 \ -1 \ 1 \ -1 \ 1 \ 1 \ 1 \ -1)$$

$$S \bullet B = (2 \ 2 \ 0 \ 2 \ 0 \ -2 \ 4 \ 0)$$

$$= 8 \Rightarrow 1$$

eli B lähetti 1-bitin



## Esimerkki jatkuu: Entä mitä A lähetti?

---

$$S = (-2 \ -2 \ 0 \ -2 \ 0 \ -2 \ 4 \ 0)$$

$$A = (-1 \ -1 \ -1 \ 1 \ 1 \ -1 \ 1 \ 1)$$

$$S \bullet A = (2 \ 2 \ 0 \ -2 \ 0 \ 2 \ 4 \ 0)$$
$$= 8 \Rightarrow 1$$

eli A lähetti 1-bitin

## Lähetikö myös D jotain?



# Käytännössä CDMA on vaativa toteuttaa

---

64 tai 128 bitin ortogonaalisia koodeja edellyttää signaalien voimakkuuksien vertailua ja yhteenlaskua => signaalien heikkeneminen eri etäisyyksillä otettava huomioon

tarkat ajoitukset

tunnettava lähettäjien sirukoodit