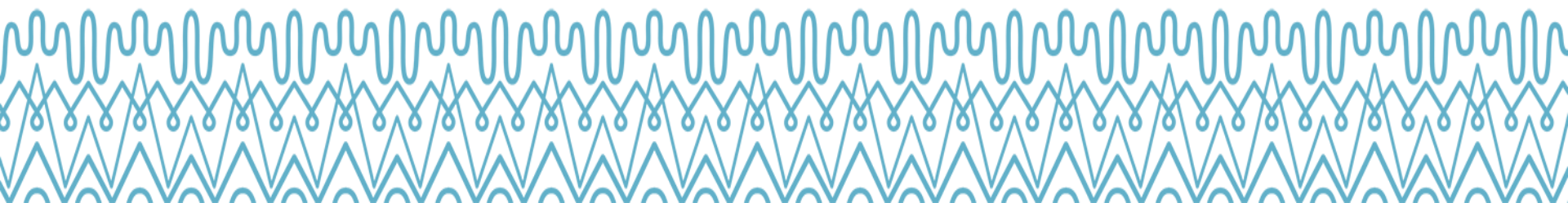


TK1100

0xA forelesning:

Linklaget



Linklaget (“Datalinjelaget”)

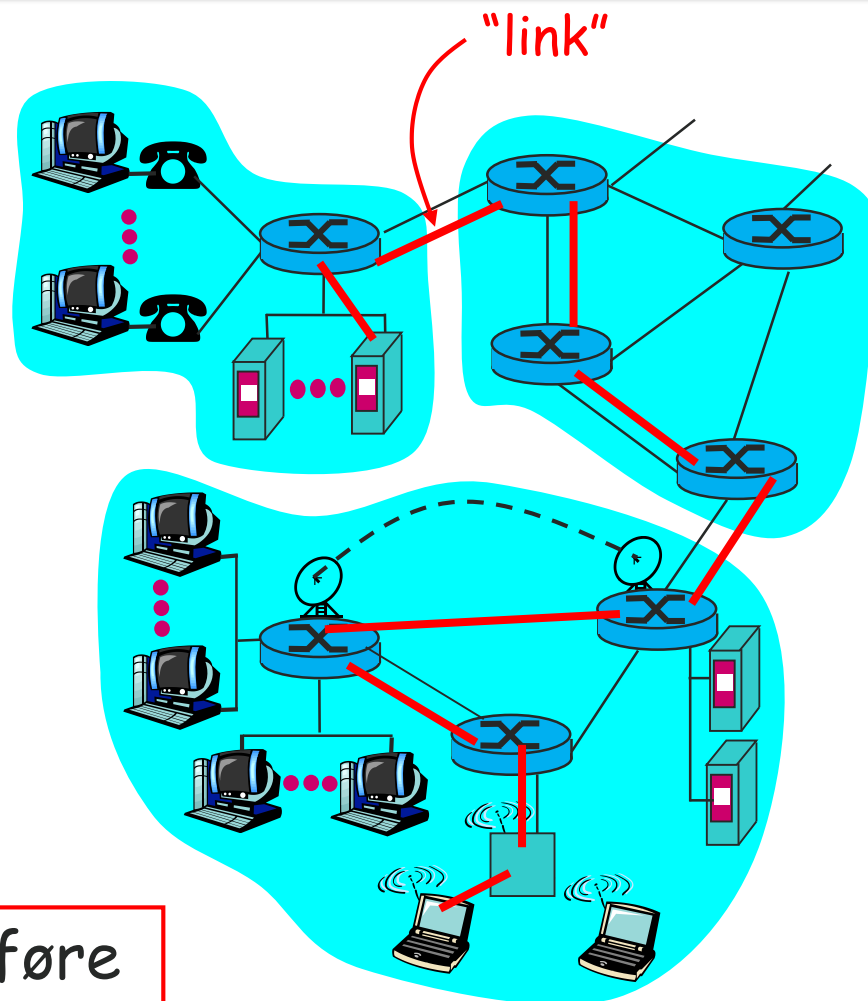
Mål:

- **forstå prinsippene** bak linklagstjenester:
 - feil**deteksjon** og feil**retting**
 - deling av en kringkastingskanal: **multippel aksess**
 - linklags**adressering**
 - pålitelig dataoverføring: *gjort! – se TCP*
 - flytkontroll: *gjort! – se TCP*
- ulike linklagsteknologier
 - Konsentrerer oss om **Ethernet** fordi dette er det vi treffer på i hverdagen

Litt terminologi:

- maskiner, rutere og svitsjer er **noder**
- kommunikasjonskanaler som forbinder nabonoder langs kommunikasjonsveien er **linker**
 - trådbundne (kablede) linker
 - trådløse (radio) linker
 - LAN
- En PDU kalles en **ramme (frame)**, innkapsler datagrammer

linklaget har ansvar for å overføre datagrammer fra en node til en nabonode over en link



Linklaget: sammenheng

- Datagram overføres av **ulike linkprotokoller** over ulike linker:
 - f eks
 - Ethernet på første link
 - Frame Relay på neste link
 - FDDI (fiber) på neste link
 - ...
 - 802.11 på siste link
- Hver linklagsprotokoll tilbyr ulike tjenester
 - f eks: én protokoll kan være pålitelig, en annen upålitelig

transportanalogi

- tur fra Halden til Trondheim
 - **tog**: Halden til Gardermoen
 - **fly**: Gardermoen til Værnes
 - **buss**: Værnes til Trondheim
- reisende = **datagram**
- **transportetappe** = **kommunikasjonslink**
- **transporttype** = **linklagsprotokoll**
- reisebyrå = **routingalgoritme**

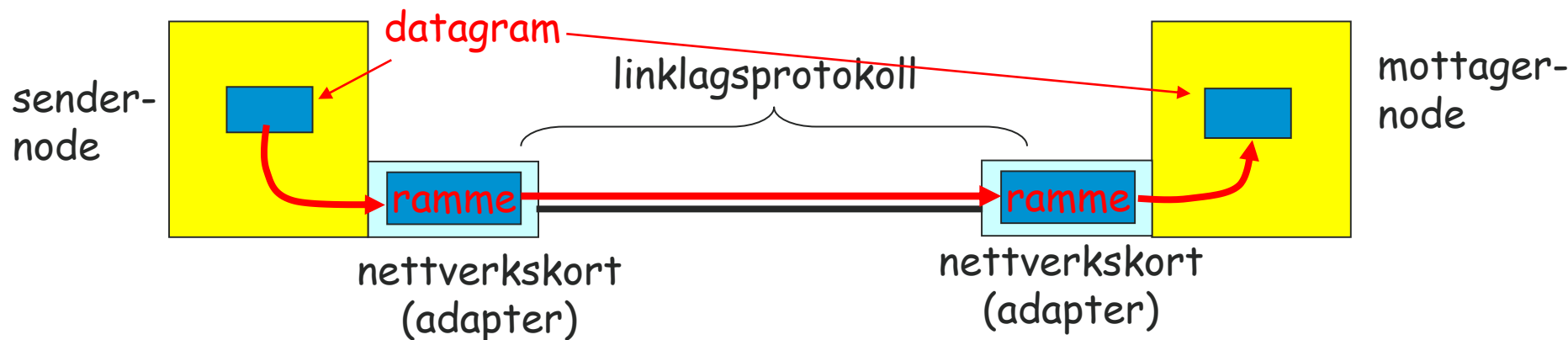
Linklagstjenester (1)

- **Omramming (framing) og link-aksess:**
 - innkapsling av datagram i rammer, legger til header og trailer
 - kanaltilgang hvis delt medium (MAC = medium access control)
 - **MAC-adresser** benyttes i rammeheader for å identifisere avsender og mottager
 - forskjellig fra IP-adresser!
- **Pålitelig leveranse mellom naboroder**
 - vi har alt sett på hvordan dette kan gjøres (Forelesning 08)!
 - lite nødvendig på link med lav bitfeilrate (fiber og noen typer kobberkabel)
 - trådløse linker: høy bitfeilrate

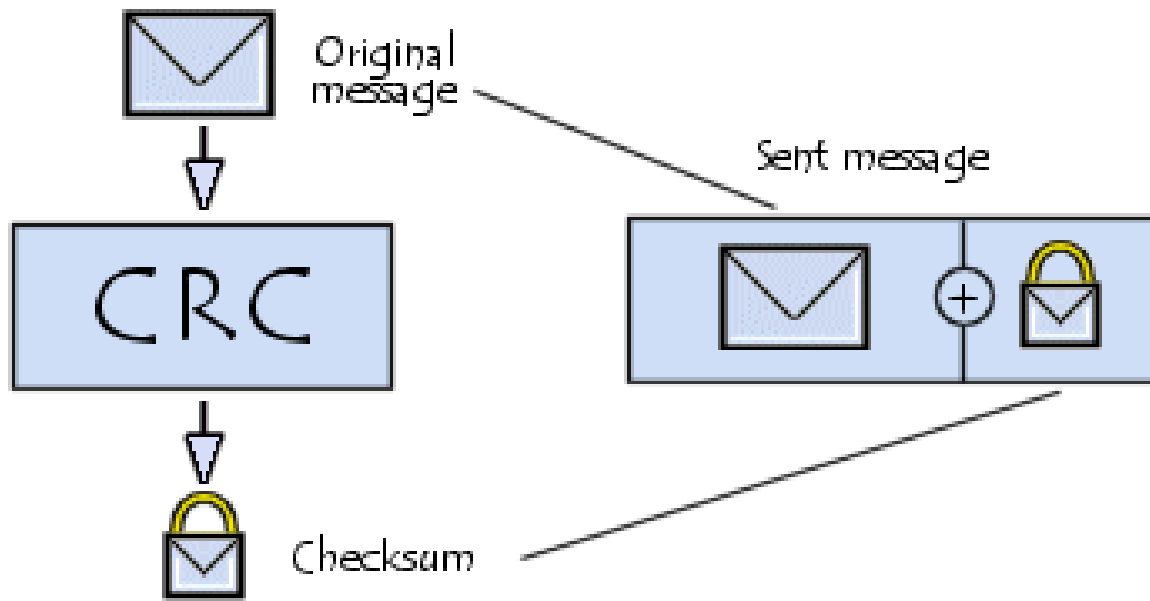
Linklagstjenester (2)

- *Flytkontroll:*
 - sender må ikke sende fortere enn mottager kan ta imot
- *Feildeteksjon:*
 - feil forårsakes av demping og elektrisk støy
 - mottager oppdager bitfeil:
 - gir enten beskjed til sender om å retransmittere
 - eller den bare kaster rammen
- *Feilretting:*
 - mottager identifiserer *og retter* bitfeil uten at sender må retransmittere
 - (Finnes i 802.11n)
- *Halv-duplex og full-duplex*
 - med halv-duplex kan noder i begge ender av linken sende, men ikke begge samtidig
 - med full-duplex kan begge sende og motta samtidig

Nettverkskort kommuniserer



- linklaget implementert i nettverkskort (NIC)
 - Ethernet-kort, 802.11-kort e.l.
- senderside:
 - innkapsling av datagram i en ramme
 - adderer bit for deteksjon av bitfeil, (sekvensnummer, flytkontroll etc.)
- mottagerside
 - ser etter bitfeil, re-transmisjon, flytkontroll etc.
 - ekstraherer datagram, leverer dette til mottagernode
- NIC er delvis autonomt
- Moderne nettverkskort støtter ofte også transport- og nettverkslagsfunksjonalitet

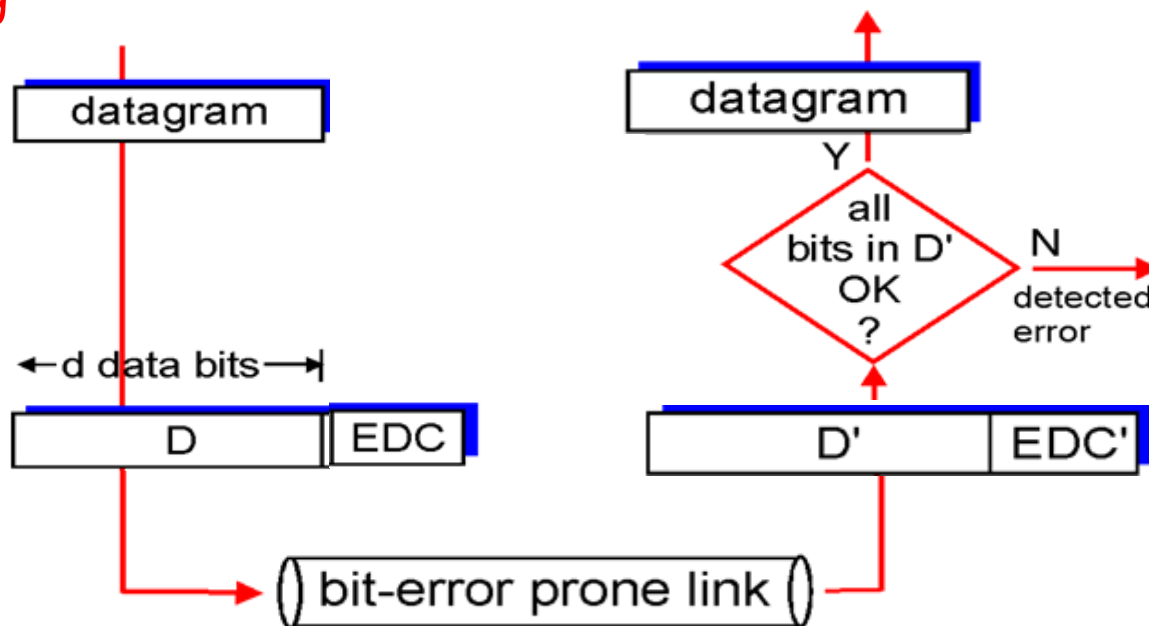


$$X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$$

FEILHÅNDTERING

Feildeteksjon

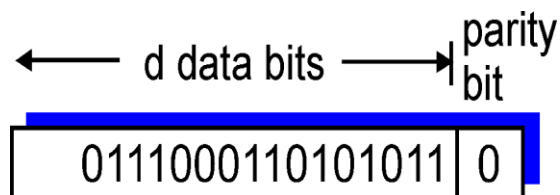
- **D** = Data beskyttet av feilsjekk, kan omfatte header-felter
- **EDC** = **E**rror **D**etection and **C**orrection bits (redundante bit)
- Feildeteksjon er ikke 100% pålitelig
 - protokollen kan overse feil (selv om det er sjelden)
 - **flere** feildeteksjons**bit** gir **bedre** deteksjon og mulighet for **retting**



Paritetssjekk

Ett-bits paritet:

Kan oppdage dersom ett bit er feil



Like paritet:

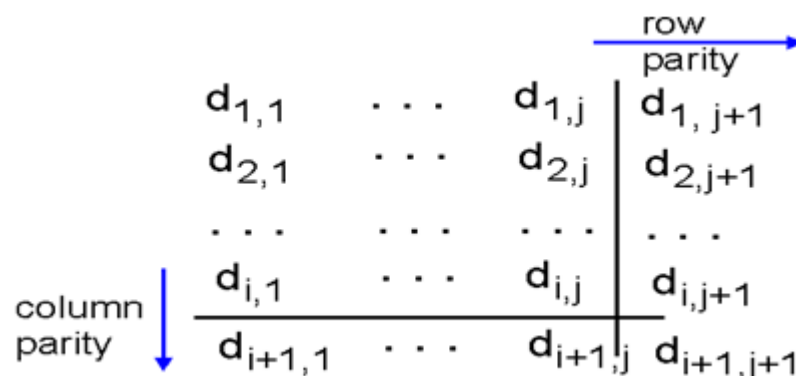
Det totale antall enere (inkl paritetsbit) skal være et **partall**

Odde paritet:

Det totale antall enere (inkl paritetsbit) skal være et **oddetall**

Todimensjonale paritetsbit:

Kan oppdage og rette dersom ett bit er feil



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

no errors

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

correctable single bit error

Internett-sjekksum (**repetisjon**)

Mål: oppdage bitfeil i mottatt segment

Sender:

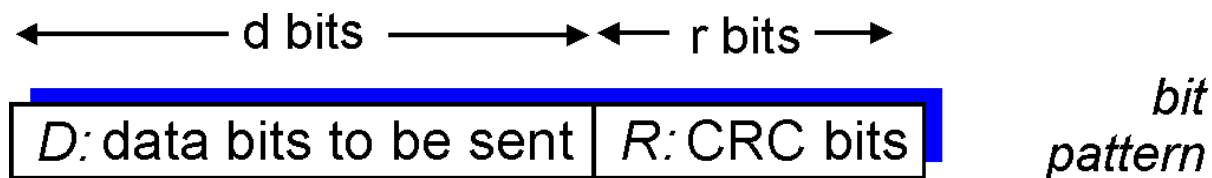
- behandler innholdet i segment/datagram som sekvens av 16-bits tall
- sjekksum: addisjon (eners komplement) av innholdet i segmentet
- sender legger sjekksum inn i UDP/TCP og IPv4 sjekksum-felt

Mottager:

- beregner sjekksum av mottatt segment
- ser om beregnet sjekksum er korrekt:
 - NEI → feil oppdaget
 - JA → ingen feil oppdaget. *Men det kan allikevel finnes feil...*

Cyclic redundancy check (CRC)

- ser databitene, D , som et binært tall
- velger et bitmønster, $r + 1$ bit langt \rightarrow generator, G
- mål: velge r CRC-bit, R , slik at
 - DR er delelig med G (modulo 2)
 - mottager kjenner G og dividerer DR med G .
Dersom divisjonen gir en rest, er det bitfeil
 - oppdager alle skur-feil mindre enn $r+1$ bit
- mye benyttet i praksis (ATM, HDCL, Ethernet, zip, ...)



$$D * 2^r \text{ XOR } R$$

mathematical formula

CRC eksempel

Ønsker:

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

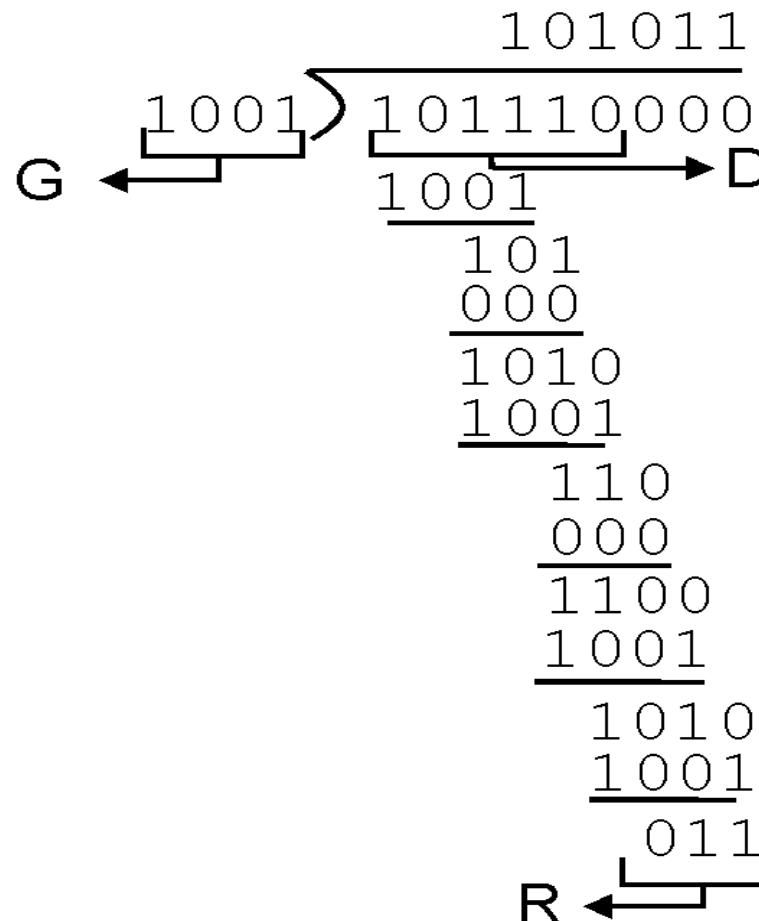
ekvivalent:

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

ekvivalent:

hvis vi deler $D \cdot 2^r$ med G er det *resten*, R , vi søker

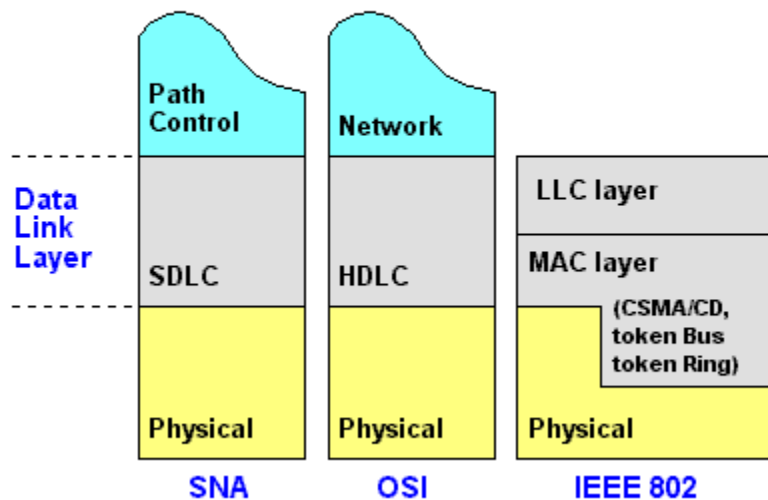
$$R = \text{rest}\left[\frac{D \cdot 2^r}{G}\right]$$



CRC-32

- Bruker noen forskjellige nøkler
 - $x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$
- Enkel å beregne i hardware
 - XOR-porter og skift-registre
- Oppdager alle burst-**feil** som er på 32 bit eller færre
- Oppdager $1 - 2^{-32} = 99,9999999767\%$ av alle feil som består av flere enn 32 bit
- Brukes også av ZIP, MPEG, PNG, m.fl.

From Computer Desktop Encyclopedia
© 1998 The Computer Language Co. Inc.

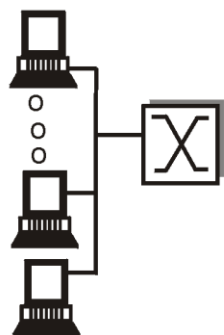
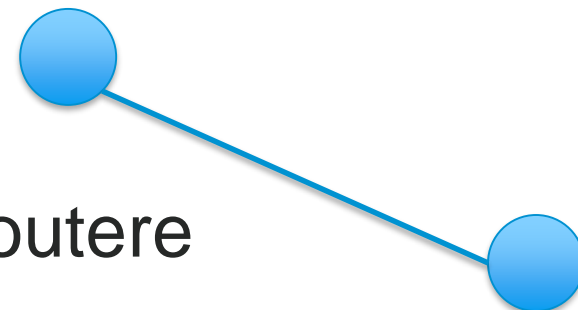


MEDIUM ACCESS CONTROL

Linker og protokoller for **multippel aksess**

To hovedtyper “linker”:

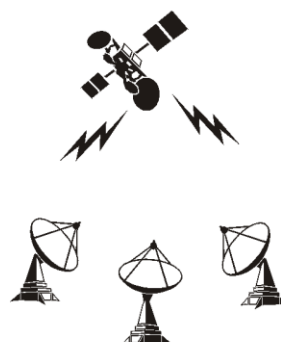
- **punkt-til-punkt**
 - PPP for oppringt aksess
 - punkt-til-punkt-link mellom to routere
- **kringkasting** (delt medium)
 - tradisjonelt ethernet
 - 802.11 trådløst lokalnett



shared wire
(e.g. Ethernet)



shared wireless
(e.g. Wavelan)



satellite



cocktail party

Multippel aksess-protokoller (MA)

- én delt kanal som alle kan lytte på (kringkasting)
- to eller flere samtidige transmisjoner fra noder resulterer i interferens
 - bare én må sende om gangen

multippel-aksessprotokoll

- distribuert algoritme som bestemmer hvordan noder deler en kanal, dvs at den bestemmer når en node kan sende
- kommunikasjon om kanaldeling må selv bruke kanalen

MAC-protokoller: taksonomi

Tre ulike tilnærminger til kanaldeling:

- **Kanalpartisjonering**
 - deler kanalen i mindre biter
 - Tidsluker
 - Frekvenser
 - koder
 - tildeler del av kanalen eksklusivt til en node
- **Random access**
 - kanal deles ikke opp, kollisjoner kan forekomme
 - har metoder for å håndtere kollisjoner
 - Ethernet (IEEE 802.3)
- **“Etter tur”**
 - unngår kollisjoner ved at kun en om gangen får sende
 - “Token Ring” (IEEE 802.5)
 - WiFi (IEEE 802.11 med Access Point)

Random access protokoller

- Når en node har pakker å sende
 - Sender med full kanalrate, R
 - ingen *a priori* koordinering mellom nodene
- to eller flere sender samtidig → “kollisjon”,
- **random access MAC protokoll** spesifiserer:
 - hvordan detekteres kollisjoner
 - hvordan håndtere kollisjoner (f eks ved hjelp av forsinkede retransmisjoner)
- Eksempler på random access MAC protokoller:
 - slotted ALOHA
 - ALOHA
 - CSMA, CSMA/CD, CSMA/CA

CSMA (**C**arrier **S**ense **M**ultiple **A**ccess)

CSMA: lytter før sending:

- Hvis kanalen er ledig: send rammen
- Hvis kanalen er opptatt: utsett transmisjonen
- Menneskelig analogi: før man begynner å snakke, hører man etter om andre snakker – ikke avbryt!

CSMA kollisjoner

kollisjoner kan forekomme:

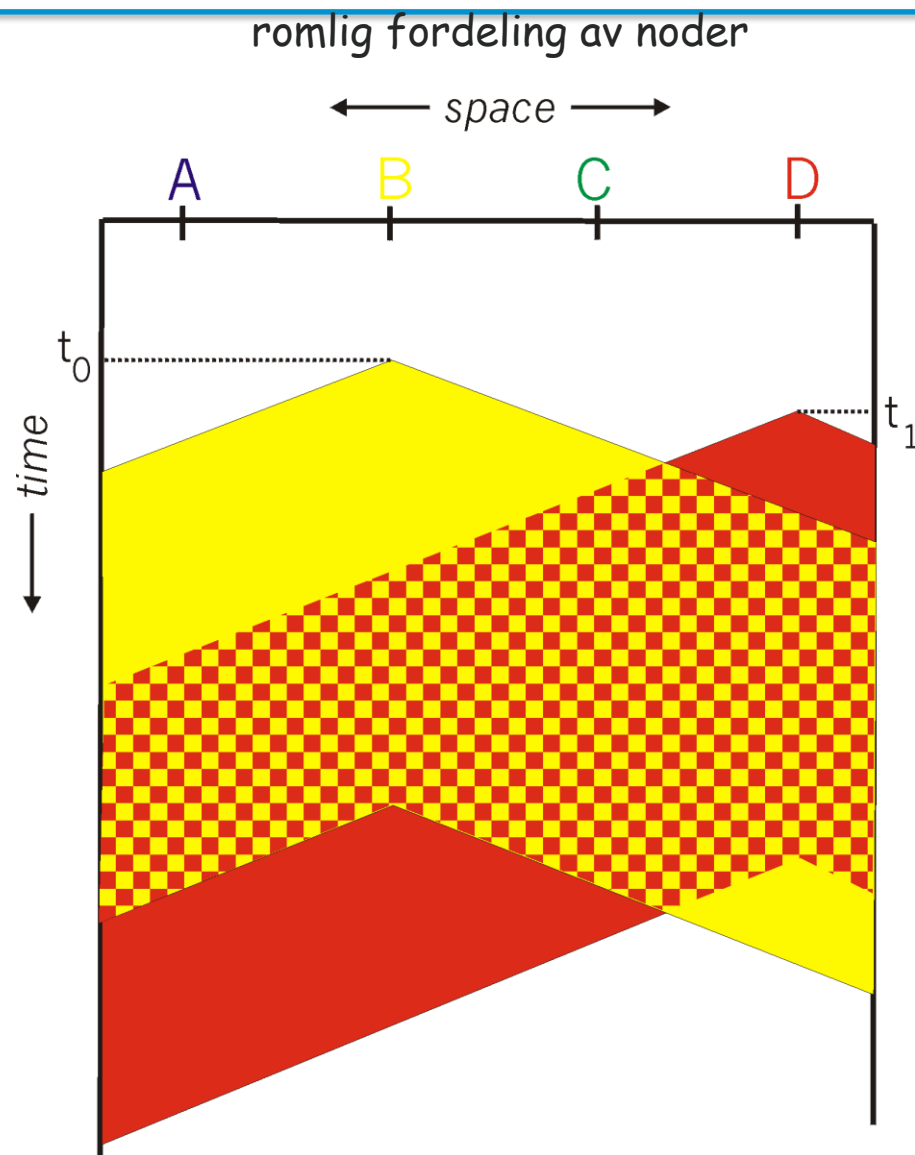
avstanden kan gjøre at en node ikke hører at en annen har startet sending

kollisjon:

hele sendetiden bortkastet

merk:

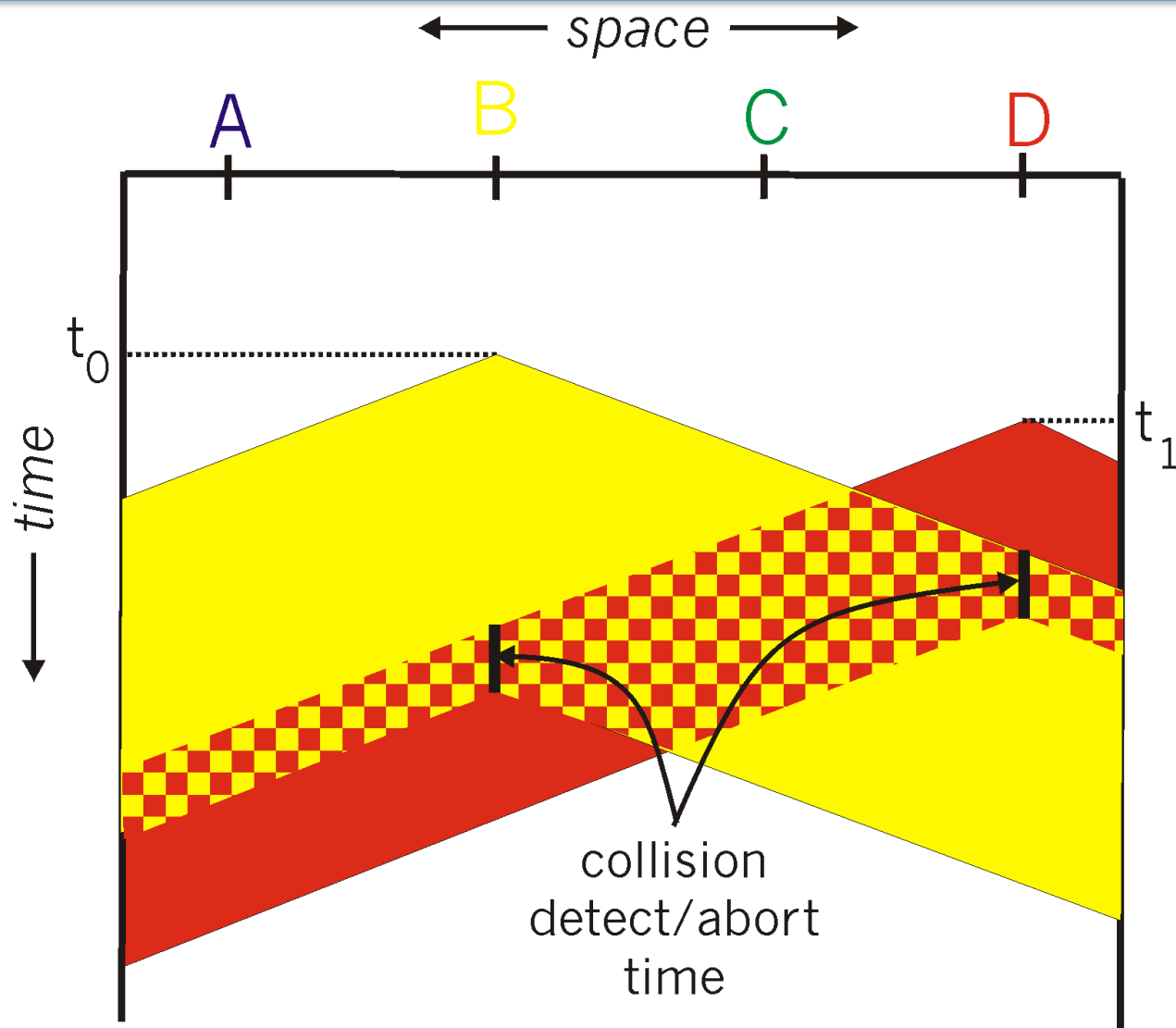
avstander og gangtid har betydning for sannsynligheten for kollisjoner



CSMA/CD (Collision Detection)

- CSMA/CD:** lytter på mediet før sending, venter hvis mediet er opptatt (som i CSMA)
- fortsetter å lytte mens man sender: kollisjoner *detektert* i løpet av kort tid
 - ved kollisjon avbrytes sendingen umiddelbart → reduserer sløsing med tid
- collision detection:
 - enkelt i kablede lokalnett: måler signalstyrken, sammenligner sendt og mottatt signal
 - vanskelig i trådløse lokalnett: mottager er vanligvis slått av mens man sender
 - menneskelig analogi: den høflige samtalepartner

CSMA/CD kollisjonsdeteksjon



“Etter tur” MAC-protokoller

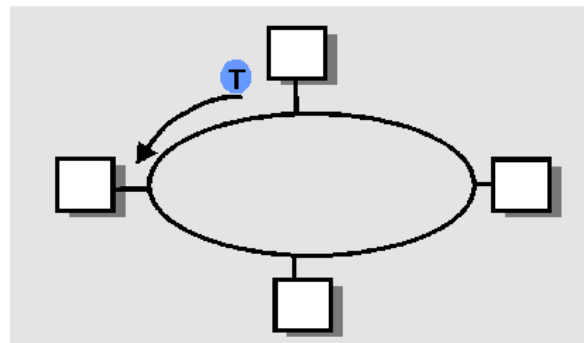
Alternativer til *tilfeldig tilgang*, finnes og brukes noen steder, f.eks. IEEE 802.5

Polling:

- master-node “inviterer” slavenoder til å sende
 - en om gangen
- ulemper:
 - overhead pga pollingen
 - latens (forsinkelse) – må vente på tur
 - single point of failure (master)

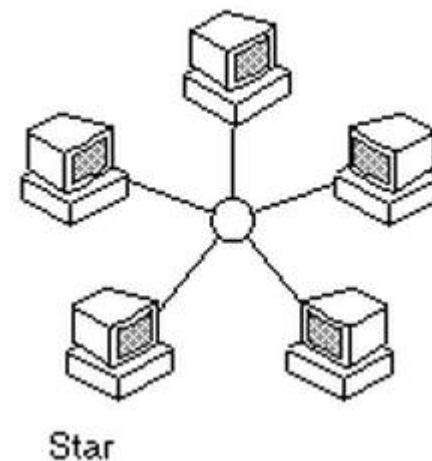
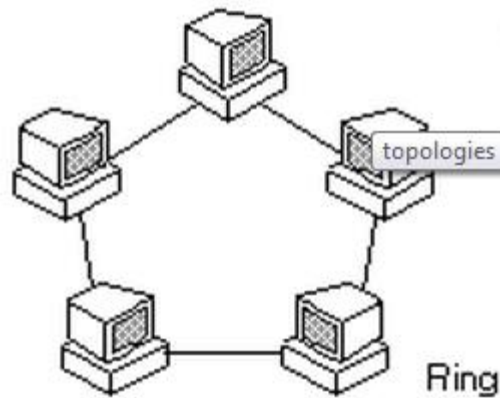
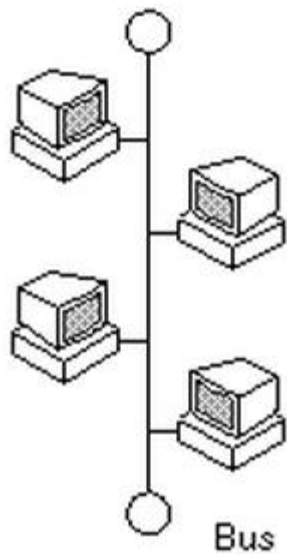
Token passing:

- ❑ en spesiell ramme - **token** (stafettpinne), sendes fra node til node
- ❑ token message
- ❑ ulemper:
 - token overhead
 - latens (forsinkelse)
 - single point of failure (token)



Ex: Konsekvenser av CSMA

- Delt medium medfører deling av tilgjengelig kapasitet
- F.ex: De trådløse Access Pointene på skolen har en max kapasitet på 300Mbps
 - medfører max ca 150 Mps for nedlasting
 - Max antall brukere er 30 => Max 5 Mps pr bruker, men lavere pga pakketap og fordi administrativt overhead for AP øker ved flere brukere.



LOCAL AREA NETWORK

LAN-teknologier

Linklaget til nå:

- tjenester, feildeteksjon/-korreksjon, multippel aksess

Neste: LAN-teknologier

- adressering
- Ethernet
- hub, switch
- PPP

LAN-adresser (= MAC-adresser)

32-bit IP-adresse:

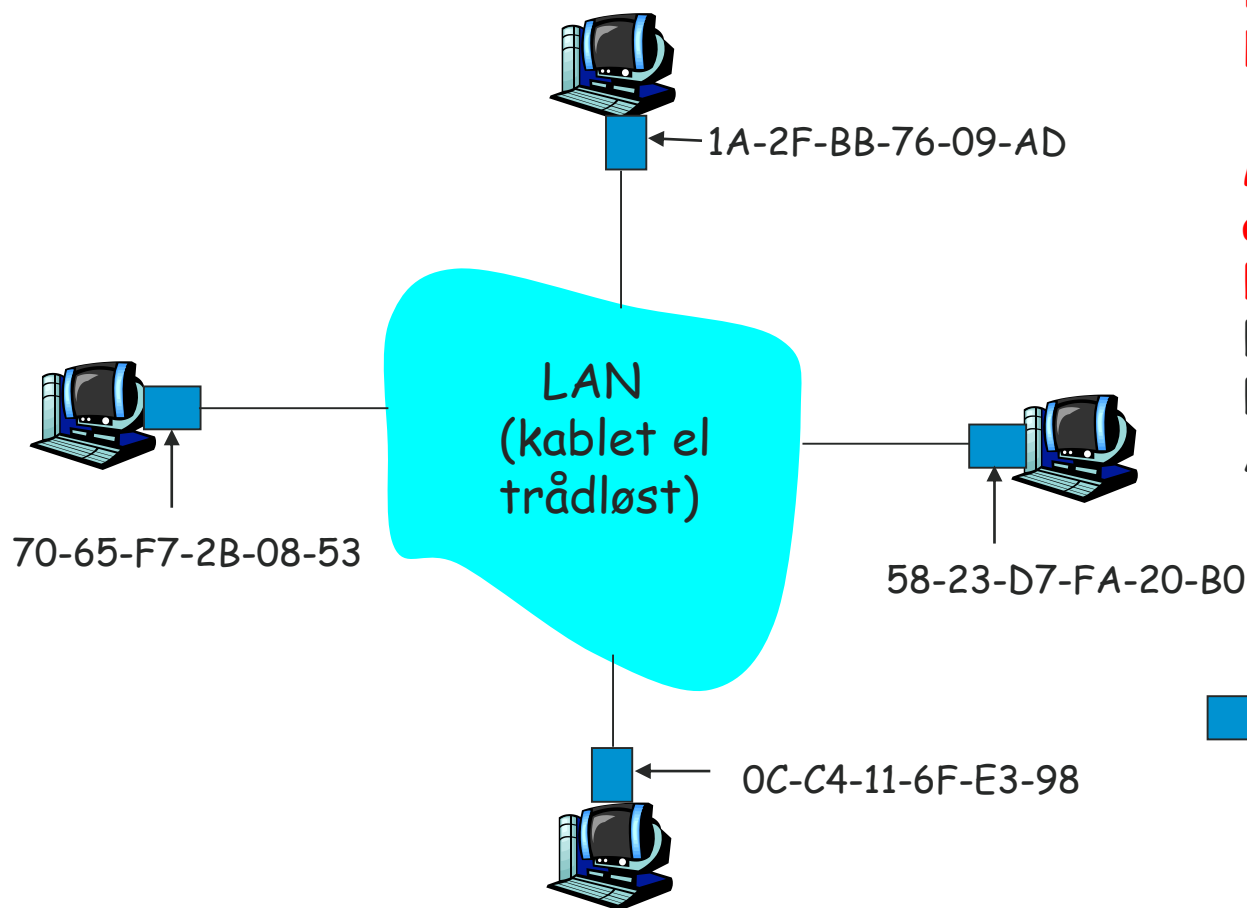
- *nettlags*-adresse
- benyttes for å få datagrammet fra ditt IP-nett fram til mottagerens IP-nett

MAC- (eller “LAN-” eller “fysisk-” eller “ethernet-”) adresse:

- benyttes for å få levert en ramme fra et interface til et annet interface **på samme nettet**
- **48 bit** MAC-adresse (for de fleste lokalnett) brent i nettverkskortets ROM

MAC-adresser

Hvert nettverkskort har en "unik" MAC-adresse



Broadcast-adressen er
FF-FF-FF-FF-FF-FF

Multicast-adresse har en
avsluttende 1'er i første
byte av MAC-adressen, og
behandles vanligvis som
broadcast, f.ex.
A3-05-77-31-27-BA

 = nettverkskort

MAC-adresser

- Tildeling av MAC-adresser administreres av IEEE
- produsent kjøper del av MAC-adresserommet
 - Første **24 bit** forteller hvem som er **produsent** av nettverkskortet
- Analogi:
 - (a) MAC-adresse: personnummer
 - (b) IP-adresse: postadresse
- MAC har “flat” adressestruktur → portabilitet
 - kan uproblematisk flytte et nettverkskort fra et lokalnett til et annet, forutsatt at det ikke er satt opp filtrering av MAC-adresser i AP eller switch
- IP-adresser er hierarkiske og derfor ikke portable
 - Prefix-delen av IP-adressen angir hvilket nett maskinen henger på

```
Frame 1: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits) on 0
Ethernet II, Src: AewinTec_0a:ab:c8 (00:0d:48:0a:ab:c8), Dst: Hewlett-_77:a0:3f (d8:d3:85:77:a0:3f)
  Destination: Hewlett-_77:a0:3f (d8:d3:85:77:a0:3f)
  Source: 158.36.131.1 (00:0d:48:0a:ab:c8)
  Type: IP (0x0800)
Internet Protocol Version 4, Src: 162.159.241.165 (162.159.241.165), Dst: 158.36.131.51 (158.36.131.51)
Transmission Control Protocol, Src Port: https (443), Dst Port: 55862 (55862), Seq: 2037027459, Ack: 26020
Secure Sockets Layer
```

ARP: Address Resolution Protocol

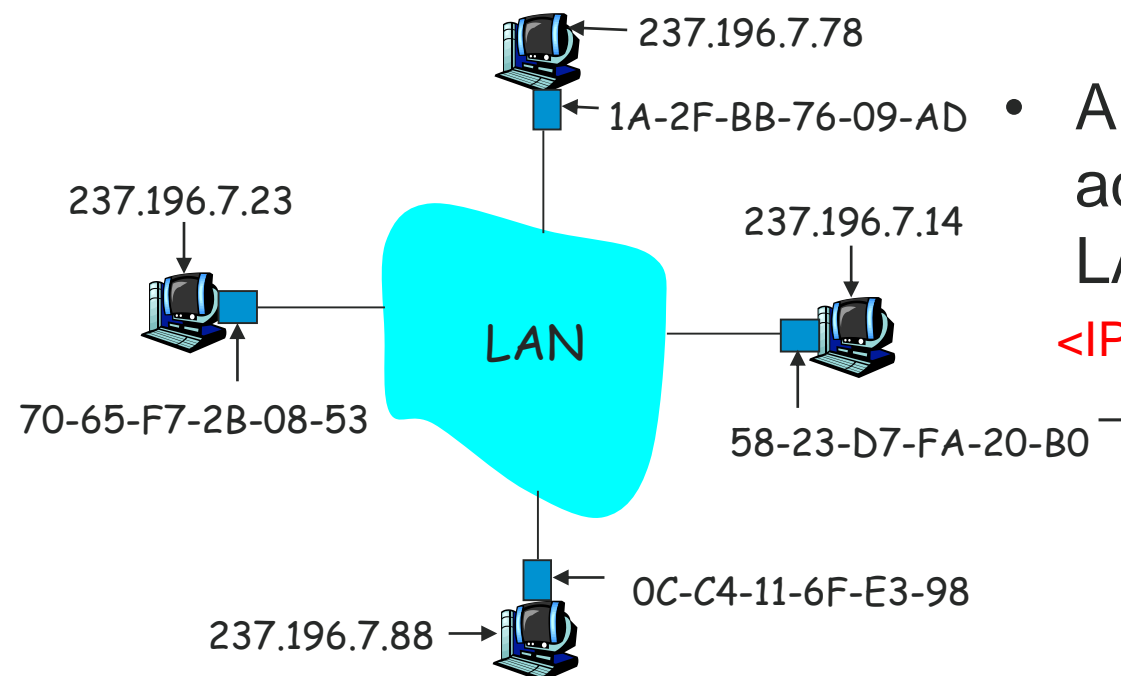
Hvordan finne MAC-adressen til en node man kjenner IP-adressen til?

- Hver IP-node (maskin og ruter) på et LAN har en **ARP**-tabell/cache

- ARP-tabell: IP/MAC adressemapping for noen LAN-noder

<IP-adresse; MAC-adresse; TTL>

TTL (Time To Live): tiden mappingen skal ligge i ARP-tabellen (typisk 20 min)



ARP (Address Resolution Protocol)

- A ønsker å sende et datagram til B og kjenner Bs IP-adresse
 - Anta at Bs MAC-adresse ikke er i As ARP-tabell
- A **kringkaster** en ARP forespørsel som inneholder Bs IP-adresse
 - alle maskiner på LAN mottar ARP-forespørselen
- B mottar også ARP-pakken og svarer A med sin MAC-adresse
 - ramme sendes direkte til As MAC-adresse
- A cacher (lagrer) IP-til-MAC adresseparet i sin ARP-tabell inntil informasjonen blir foreldet
 - “soft state”: informasjon som forsvinner dersom den ikke oppfriskes
- ARP er “plug-and-play”:
 - en node lager sin ARP-tabell uten hjelp fra noen

arp

- `arp`-kommandoer kan brukes til å
 - vise frem ARP-cache (`-a`)
 - tømme ARP-cache (`-d`)
 - Legge til faste IP-MAC koplinger (`-s`)

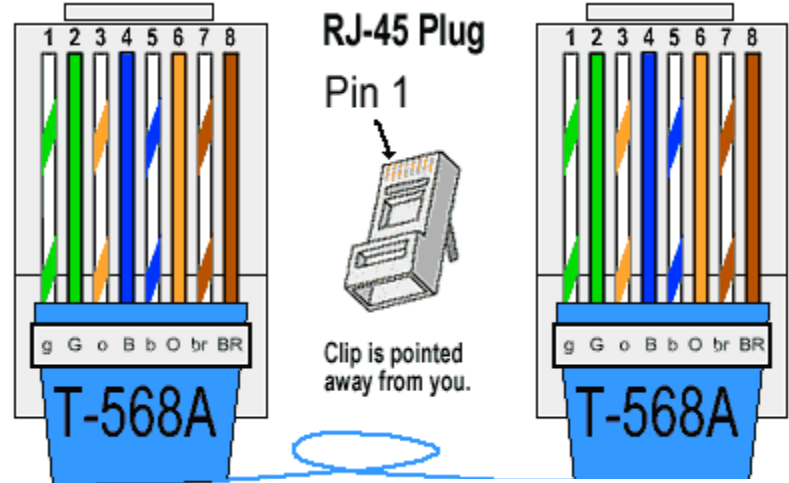
```
C:\>arp -a
```

```
Interface: 158.36.131.51 --- 0xc
Internet Address      Physical Address      Type
158.36.131.1          00-22-55-3e-da-ba    dynamic
158.36.131.127        ff-ff-ff-ff-ff-ff    static
255.255.255.255       ff-ff-ff-ff-ff-ff    static
```

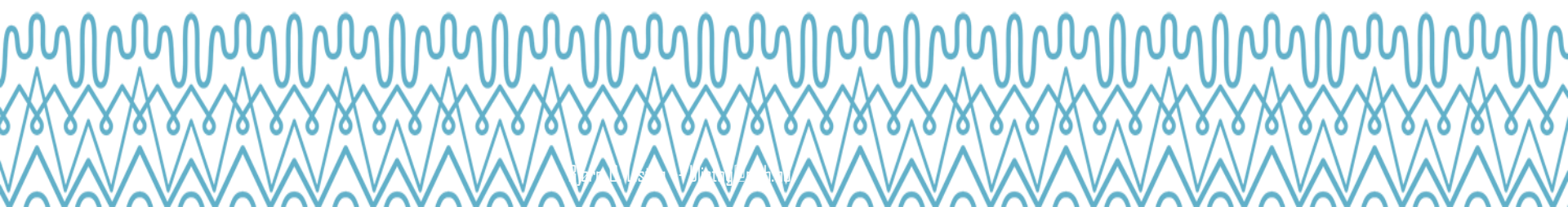


Terminal — bash — 80x24

```
NITHs-MacBook-Pro:~ foreleser$ arp -a -n
? (10.21.24.1) at 0:22:55:3e:da:ba on en1 ifscope [ethernet]
? (10.21.26.33) at 0:1b:77:76:6c:c6 on en1 ifscope [ethernet]
? (10.21.26.102) at 20:7c:8f:5:d2:cc on en1 ifscope [ethernet]
? (10.21.27.255) at ff:ff:ff:ff:ff:ff on en1 ifscope [ethernet]
NITHs-MacBook-Pro:~ foreleser$
```



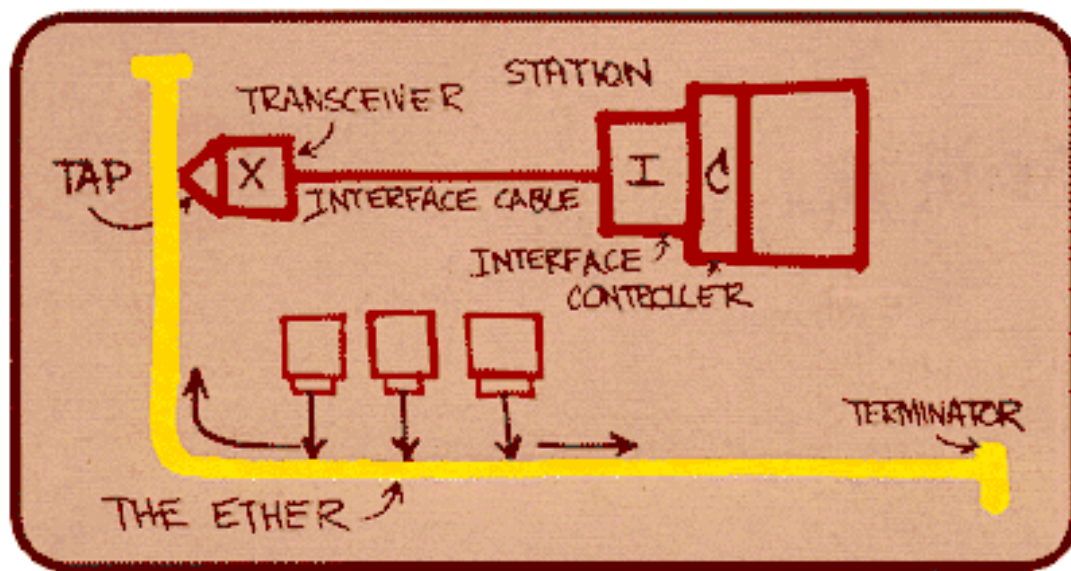
ETHERNET



Ethernet

Dominerende lokalnetsteknologi:

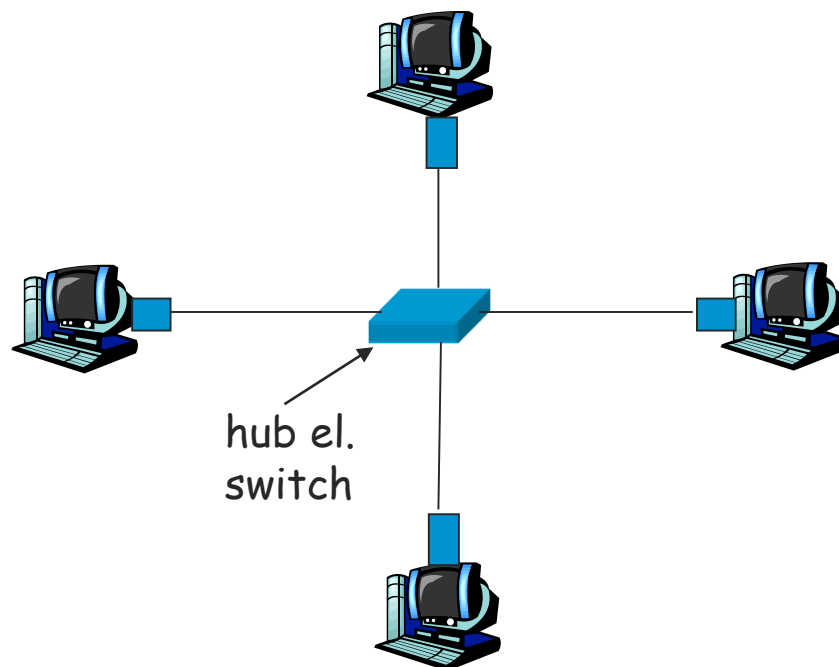
- billig
- første LAN-teknologi i utstrakt bruk
- Enklere og billigere enn token passing LAN
- Har holdt tritt i hastighetskappløpet: 10, 100, 1000 Mb/s



Bob Metcalfes
Ethernetskisse

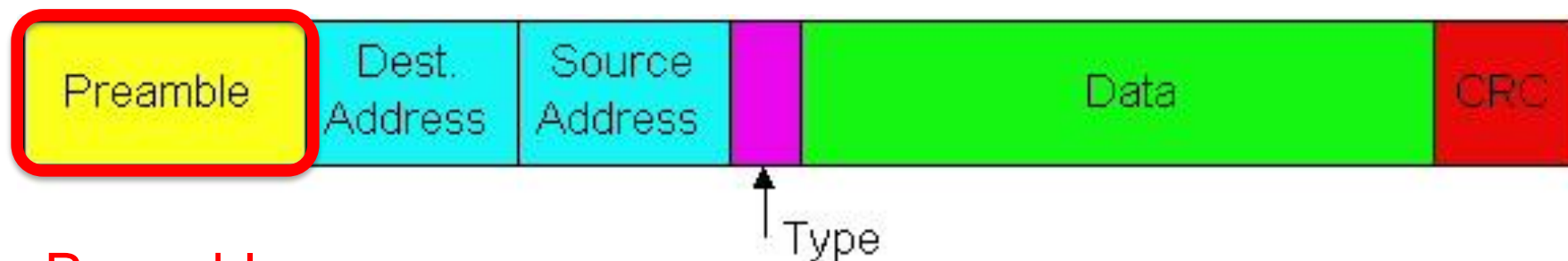
Stjernetopologi

- Busstopologi var populær til midten av 90-tallet
 - Ethernet er fremdeles definert med forutsetning om buss-topologi og hvordan løse kollisjoner.
- **Nå** er det stjernetopologi som “går og gjelder”
- Valgmulighet: (hub eller) **switch** (mer senere)



Ethernets rammestruktur

Nettverkskort (NIC, adapter) legger IP-datagrammet (eller annen nettlags-PDU) i en **Ethernetramme**

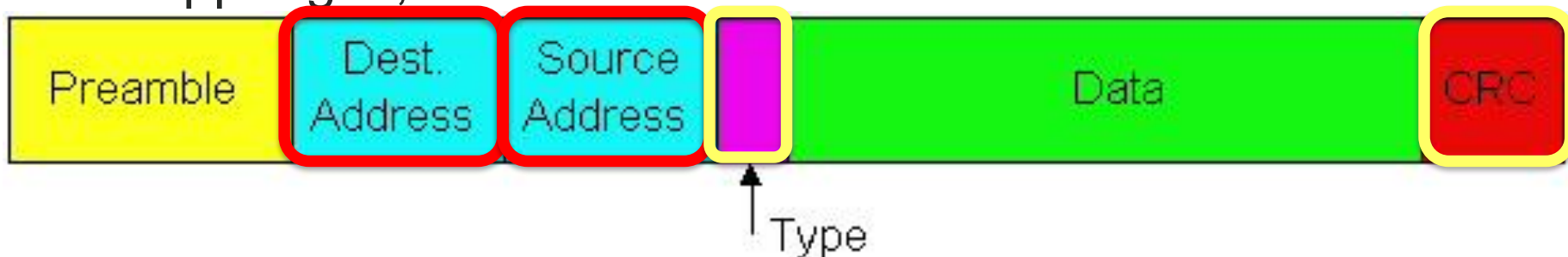


Preamble:

- 7 oktetter (7 byte) med bitmønster 10101010 fulgt av én oktett med bitmønster 10101011
- benyttes for å synkronisere mottagers “klokke” med senderens

Ethernets rammestruktur (forts)

- **Adresser:** 6 oktetter (48 bit)
 - hvis NIC mottar ramme med egen adresse som destinasjonsadresse eller en kringkastingsramme (f eks ARP-pakke), leverer den data i rammen til nettlags-protokollen
 - ellers kaster den rammen
- **Type:** indikerer hvilken nettlagsprotokoll data tilhører (normalt IP, men også andre muligheter, f. eks. Novell IPX eller AppleTalk)
- **CRC:** feildeteksjon (cyclic redundancy check) – hvis feil oppdages, kastes rammen



Upålitelig, forbindelsesløs tjeneste

- **Forbindelsesløs:** Ingen håndhissing mellom sender og mottager
 - Derimot så fremforhandler nettverkskortene hvilken IEEE 802-versjon og bitrate de skal benytte første gang de er i forbindelse
- **Upålitelig:** mottager sender ikke ACK eller NAK tilbake til senderen
 - strømmen av datagrammer som leveres til nettlaget kan ha gap
 - dersom TCP benyttes, sørger denne for å fylle eventuelle gap
 - ellers vil/må applikasjonen se gapene i datastrømmen

Ethernet benytter CSMA/CD

- Ingen tidsluker
- nettkort lytter på nettet før den skal sende (**carrier sense**)
 - sender ikke dersom noen andre allerede sender
- senderen fortsetter å lytte mens den sender og avbryter sendingen dersom den merker at en annen også sender (**collision detection**)
- Før senderen forsøker en retransmisjon, venter den en tilfeldig valgt tid (**random access**)

Ethernet CSMA/CD algoritmen

1. Nettkort får datagram fra nettlag og lager en ramme
2. Sender lytter på mediet for å se om det er ledig. Hvis ingen andre er å høre, vil nettkortet starte sendingen. Hvis mediet er opptatt, venter den til det blir ledig og sender deretter
3. Hvis hele rammen er sendt uten kollisjon, er nettkortet ferdig med rammen
4. Hvis senderen oppdager at en annen sender samtidig med den selv, avbryter den sendingen og sender i stedet et jamme-signal
5. Etter avbruddet vil senderen foreta en “**exponential backoff**”: etter kollisjon nr m , velger senderen tilfeldig en K fra mengden $\{0, 1, 2, \dots, 2^m - 1\}$. Så venter den $K \cdot 512$ bit-tider og returnerer til trinn 2.

Ethernets CSMA/CD (forts)

Jammesignal: for å forsikre seg om at alle er oppmerksom på kollisjonen; 48 bit

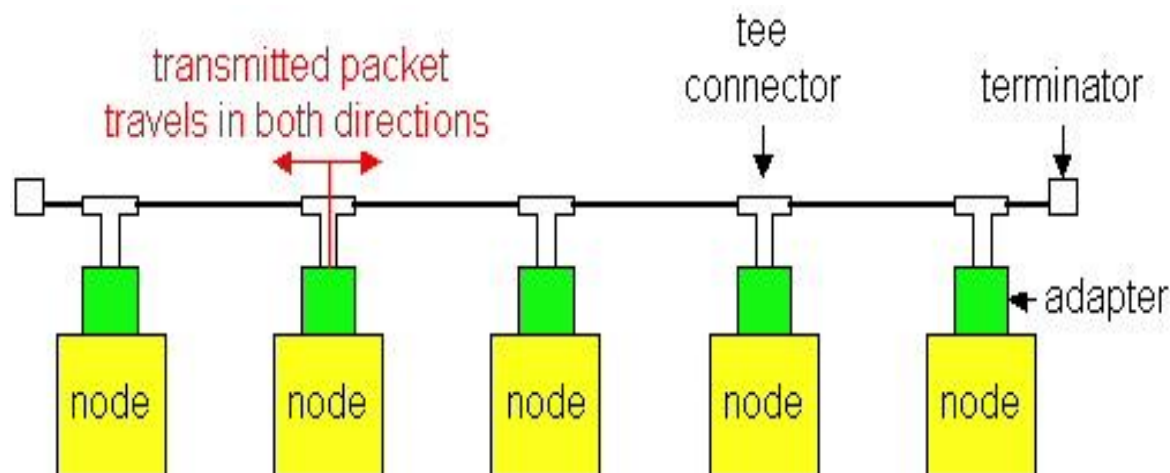
Bit-tid: 10 ns for 100 Mb/s Ethernet;
for $K = 1023$ vil følgelig ventetiden være omkring 5 ms

Eksponential Backoff:

- **Mål:** tilpasser forsøk på retransmisjon etter estimert last for øyeblikket
 - stor belastning: tilfeldig ventetid ofte lenger
- første kollisjon: velg K fra $\{0, 1\}$; ventetid er $K \cdot 512$ bit-tider
- etter andre kollisjon: velg K fra $\{0, 1, 2, 3\}$
- etter ti kollisjoner: velg K fra $\{0, 1, 2, 3, 4, \dots, 1023\}$
- Dersom fremdeles ikke sendetid: Gi opp..

Ethernet-teknologier: 10Base2

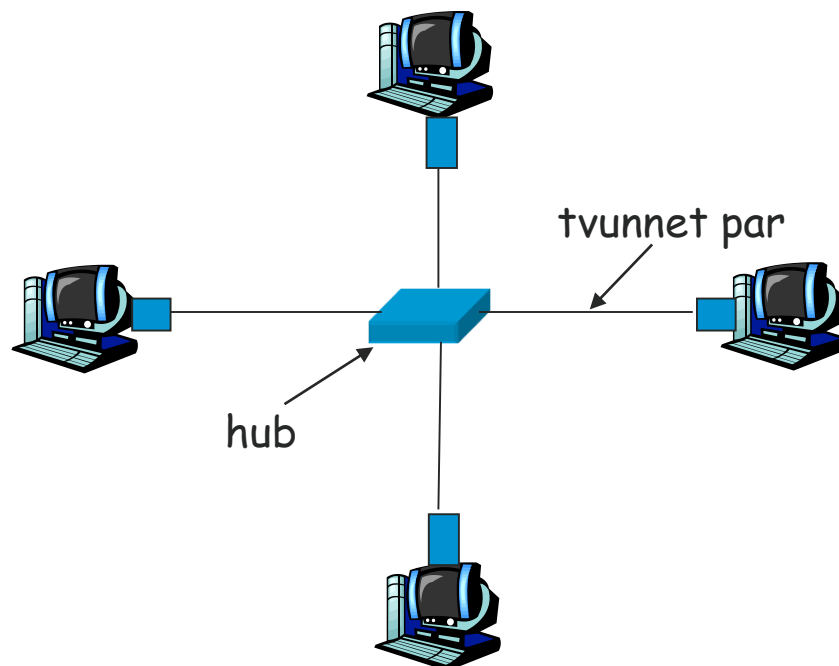
- **10**: 10 Mb/s; **Base**: basisbånd; **2**: maks 200 meters kabel
- tynn koaksialkabel i en busstopologi



- maks 30 noder pr *segment*
- repeatere brukes for å knytte sammen flere segmenter
- repeater gjentar bit den hører på ett interface på sine andre interface: opererer på fysisk lag!

10BaseT og 100BaseT

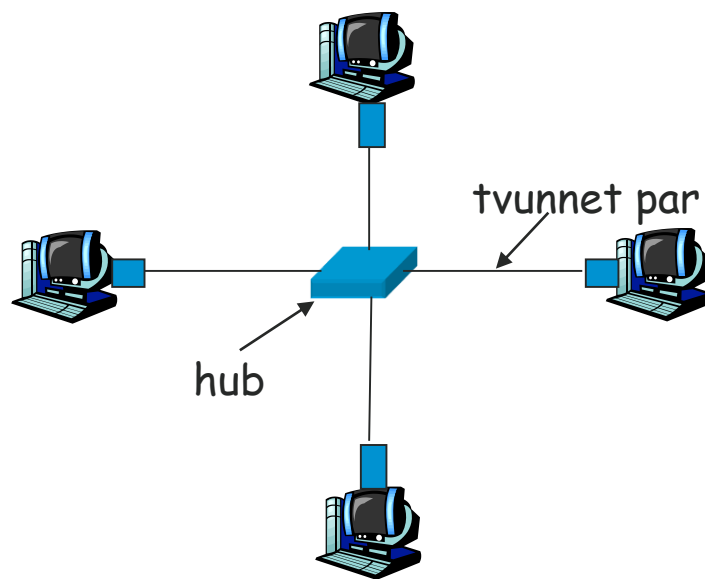
- 10/100 Mbps rater; sistnevnte kalles “fast ethernet”
- 1 GbE (1000BASE-T)
 - Bruker alle trådparrene, og komplisert koding
- **T** står for “twisted pair” (tvunnet par)
- Noder forbundet med en “hub”(eller switch): stjernetopologi; maks avstand fra node til hub er ca 100 m



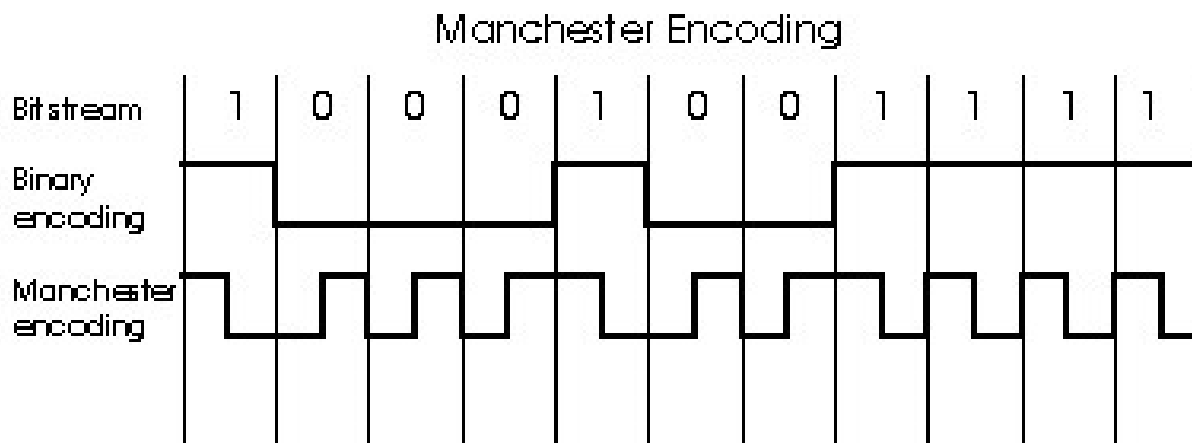
Huber

Huber er multiport repeatere (fysisk lag):

- bit som kommer inn på en link sendes ut på alle andre linker
- ingen buffring av rammer
- ingen CSMA/CD på huben: NIC detekterer eventuelle kollisjoner
- gir visse network management funksjoner



Manchester-koding



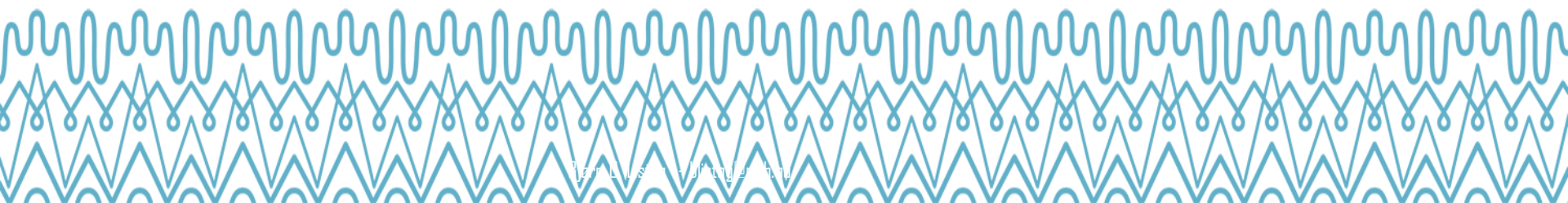
- Brukes i 10BaseT og 10Base2
- Hvert bit har en overgang (lav-til-høy el. høy-til-lav)
- Muliggjør synkronisering av sender og mottaker
 - mottager må vite hvor hvert bit starter
 - ikke behov for sentralisert, global klokke
- (Dette tilhører **fysisk** lag – ikke linklaget)

Gigabit ethernet (1000Base-T)

- benytter standard ethernet rammeformat
- Benytter alle **fire** tråddparrene i UTP-kabelen
- ved delte kanaler brukes CSMA/CD; bør ha korte avstander mellom noder for topp ytelse, men kan brukes opp til 100 m
- Benytter flere avanserte kodingsteknikker: 5 nivå puls-amplitude-modulering m.fl.
- Full-dupleks ved 1 Gb/s for punkt-til-punkt linker
- Finnes også for fiber mm
- 10 Gbps finnes og vinner stadig terreng
 - Mange ulike fysiske standarder (PHY)
 - 10GBASE-T for Cat 6, 6A eller 7 UTP med RJ45

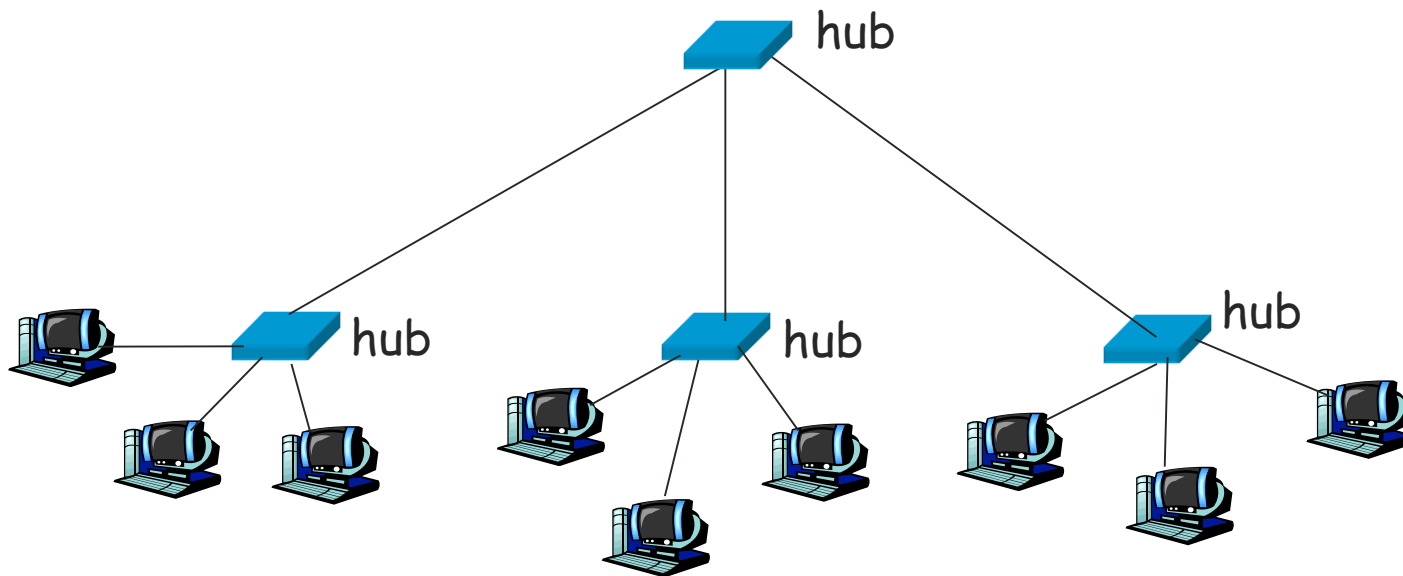


SWITCH



Sammenkopling med huber

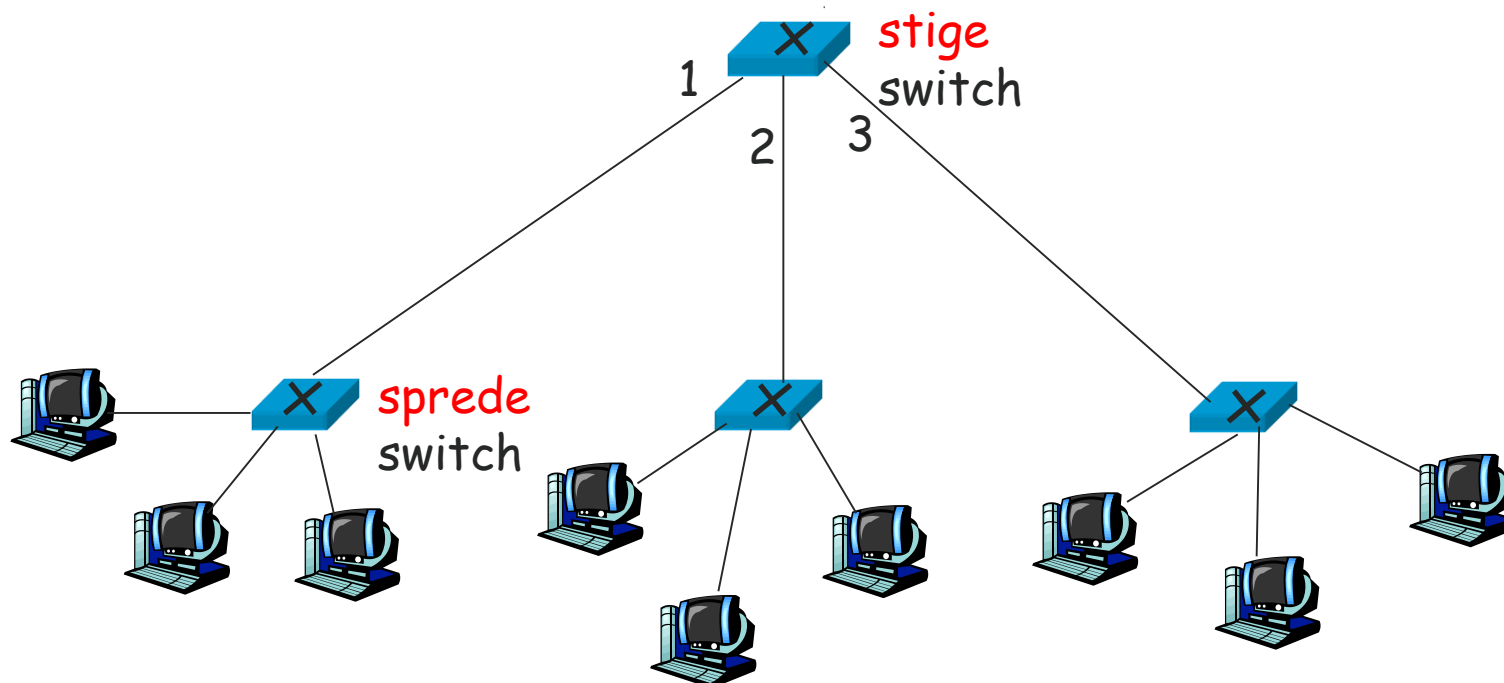
- Rygggradshub sammenkobler LAN-segmenter
- Utvider maks avstand mellom noder
- Gir ett stort **kollisjonsdomene** (ulempe)
- Kan ikke koble sammen 10BaseT og 100BaseT
 - For å kople sammen ulike fysiske og lag2-teknologier trenger man en **bridge**



Switch/Svitsj

- Linklags-boks
 - mellomlagerer og videresender ethernetrammer
 - sjekker rammeheader og videresender **når det er behov for det** – basert på mottagers MAC-adresse
 - når rammen skal videresendes på et segment, benyttes CSMA/CD
- transparent
 - maskiner er uvitende om svitsjens tilstedeværelse
- plug-and-play, selvlæring
 - svitsjer trenger man ikke å konfigurere

Videresending



- Hvordan bestemme hvilket LAN-segment rammen skal sendes ut på?
- Ser ut som et routingproblem...

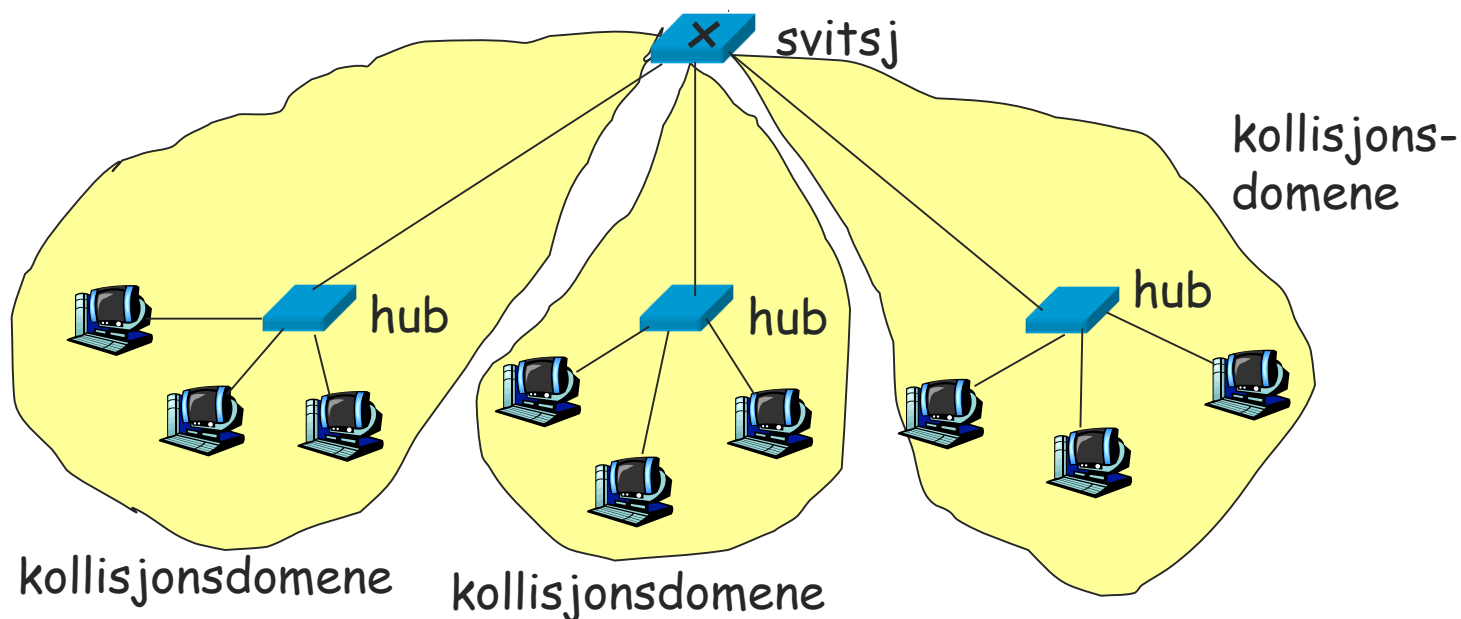
Selvlæring

Destination Address	Address Type	VLAN	Destination Port
0000.001e.2a52	Dynamic	1	FA1/1
0000.001e.345e	Dynamic	1	FA1/1
0000.001e.bb3a	Dynamic	1	FA1/1
0000.001e.eba3	Dynamic	1	FA1/2
0000.001e.face	Dynamic	1	FA1/3
0000.001e.3519	Dynamic	1	FA1/4
0000.001e.2dc1	Dynamic	1	FA1/5
0000.001e.8465	Dynamic	1	FA1/5
0000.001e.1532	Dynamic	1	FA1/5
0000.001e.8ab2	Dynamic	1	FA1/6
0000.001e.15b1	Dynamic	1	FA1/6
0000.005a.1b01	Dynamic	1	FA1/6
0000.005a.4214	Dynamic	1	FA1/7
0000.005a.5129	Dynamic	1	FA1/8
0000.00cc.bbe2	Dynamic	1	FA1/9
0000.00cc.2291	Dynamic	1	FA1/10

- En switch har en **switch-tabell**
- rader i svitsjetabellen:
 - (MAC-adresse, interface, tidsstempel)
 - foreldede innslag i tabellen slettes (TTL kan være 60 min)
- Switch'en **lærer** hvilke maskiner som kan nås på de ulike interface
 - når den **mottar** en ramme, vil svitsjen “huske” hvilket segment rammen kom fra kombinert med avsenders MAC-adresse
 - lagrer dette i sin svitsjetabell

Svitsj: trafikkisolasjon

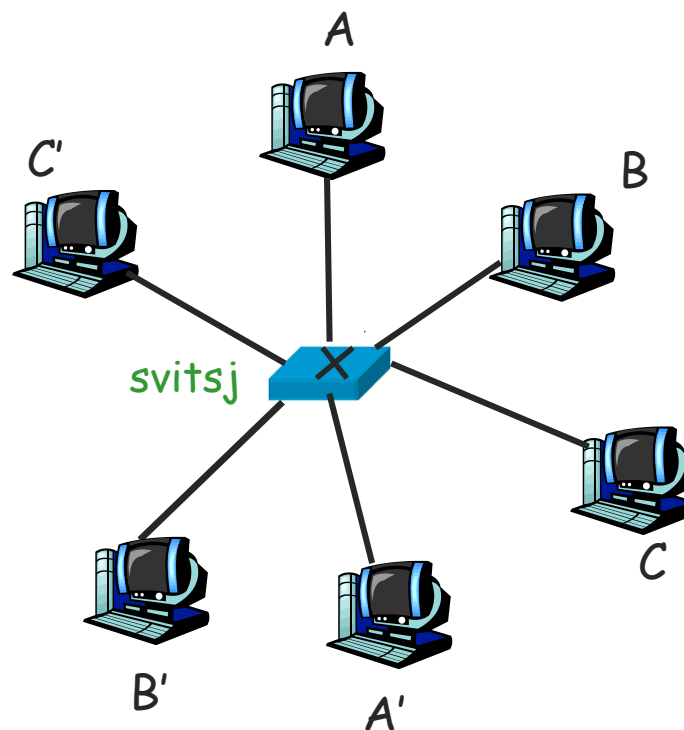
- installering av en svitsj vil dele lokalnettet i segmenter
- svitsjen **filtrerer** rammer:
 - rammer som skal til maskin på samme segment vil normalt ikke bli sendt til andre segmenter
 - segmentene blir separate **kollisjonsdomener**



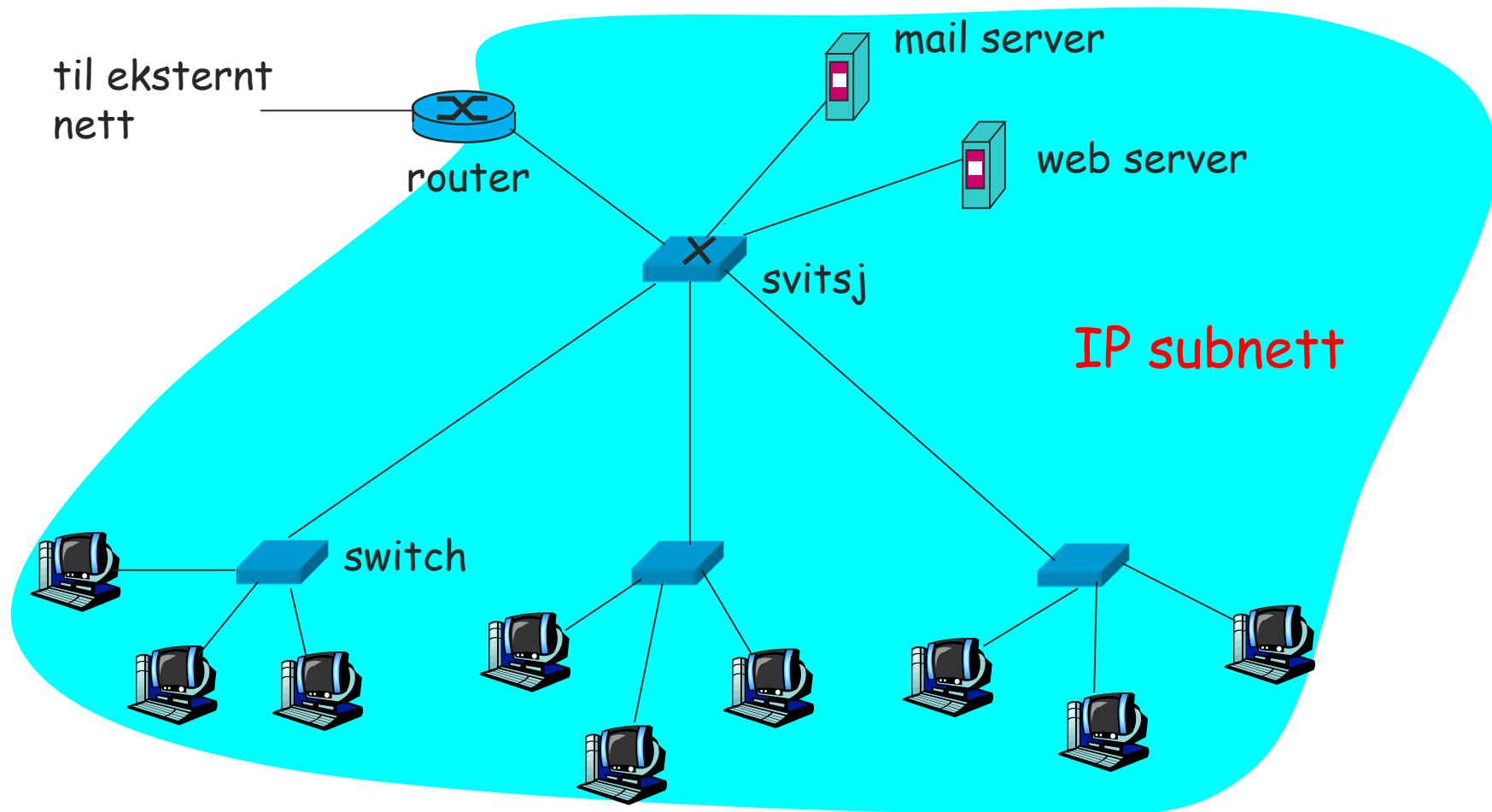
Svitsjer: dedikert aksess

- Svitsj med mange interface
- Maskiner har direkte forbindelse til svitsjen
- Ingen kollisjoner, full dupleks

Svitsjing: A-til-A' og B-til-B'
samtidig, ingen kollisjoner

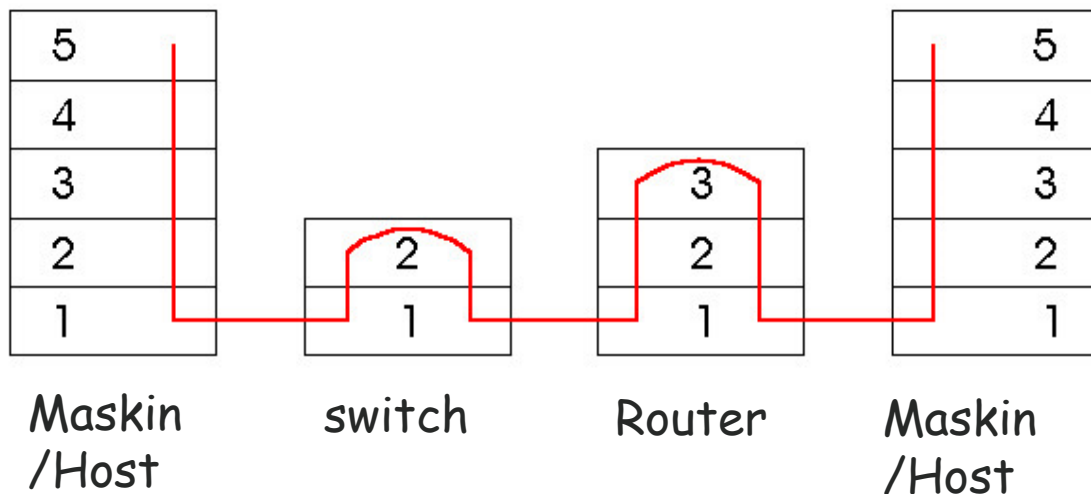


Nettverk for en institusjon



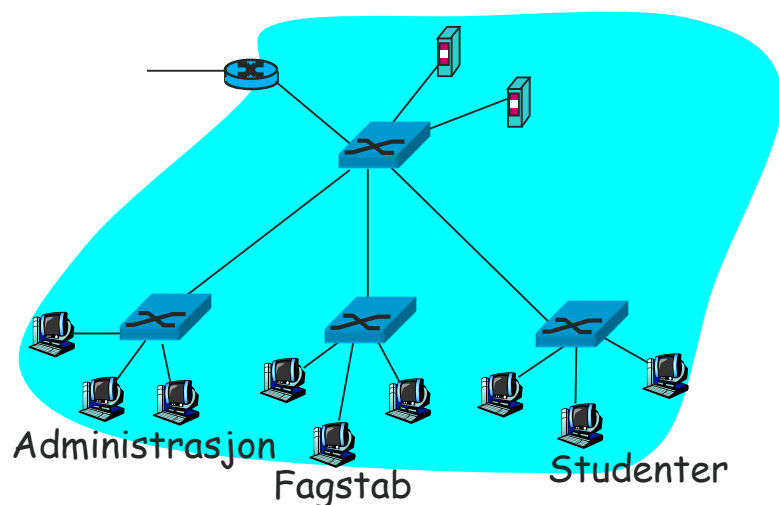
Svitsjer vs. routere

- begge er “store-and-forward” enheter
 - routere: nettlagsenheter (ser på nettlagsheadere)
 - svitsjer er linklagsenheter
- routere benytter routingtabeller og implementerer routingalgoritmer
- svitsjer benytter svitsjetabeller, gjør filtrering, og har selvlæring



VLAN: motivasjon

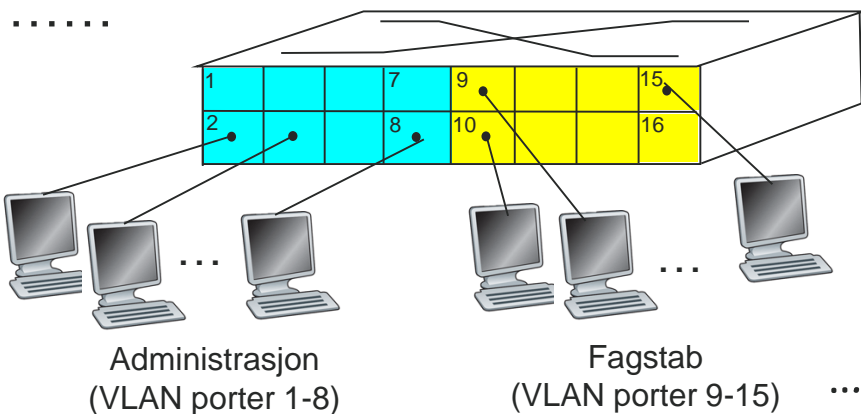
Hva er feil med dette LANet?



- Hva skjer dersom: Adm-bruker bytter kontor til Fagstab-gangen, men vil fortsette å henge på Adm-switch?
- Ett enkelt **broadcast** domene:
 - all lag-2 broadcast trafikk (ARP, DHCP) sendes til hele LANet (sikkerhet/privatliv, ineffektivt)
- Sprede-switchene bruker bare et fåtall av portene sine..

Virtuelle LAN

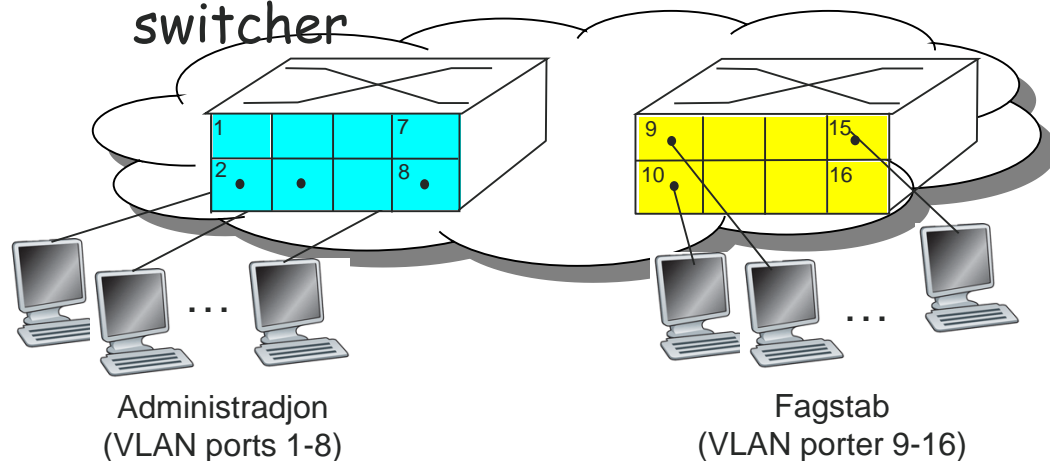
Port-basert VLAN: switch porter grupperes (med switch management software) slik en *enkelt* fysisk switch



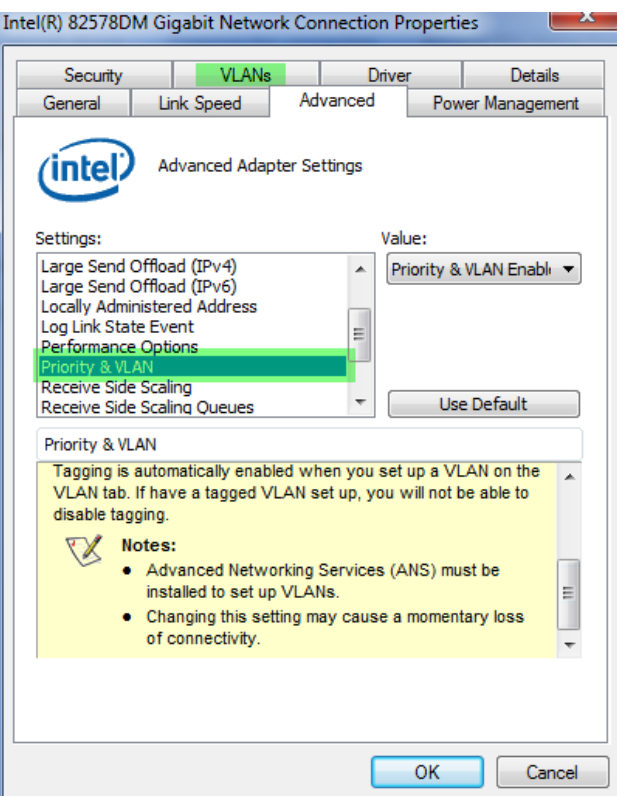
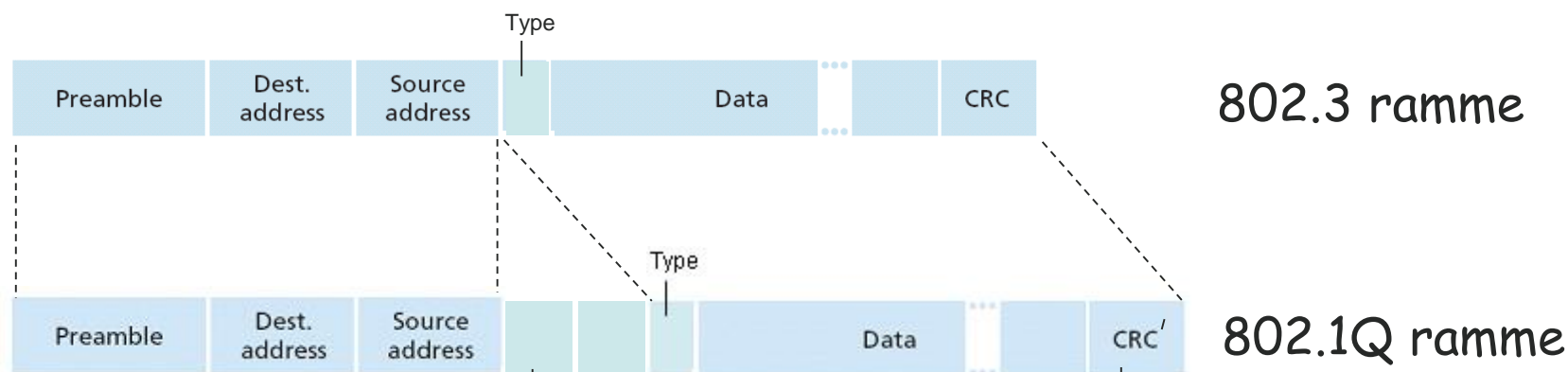
Virtual Local Area Network

Switch(er) som støtter VLAN kan konfigureres til å definere flere virtuelle LAN i ett enkelt fysisk LAN.

... fungerer som *flere* virtuelle switcher



802.1Q VLAN ramme format



2-byte Tag Protokoll ID
(value: 81-00)

Omregnet
CRC

Tag Control Information (12 bit VLAN ID felt,
3 bit prioritets felt, jf IP TOS)

Sammenligning - oppsummering

hub

ruter

svitsj

trafikk-
isolasjon

nei

ja

ja

plug & play

ja

nei

ja

optimal
ruting

nei

ja

nei

cut
through

ja

nei

ja

Access Point



Desktop Adapter
(PCI)



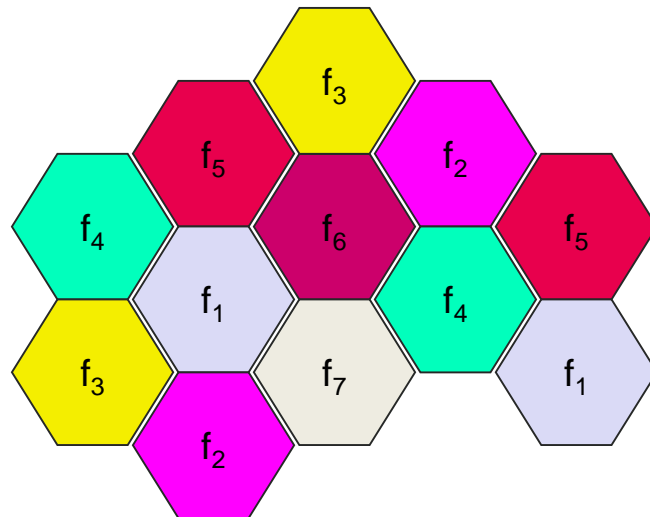
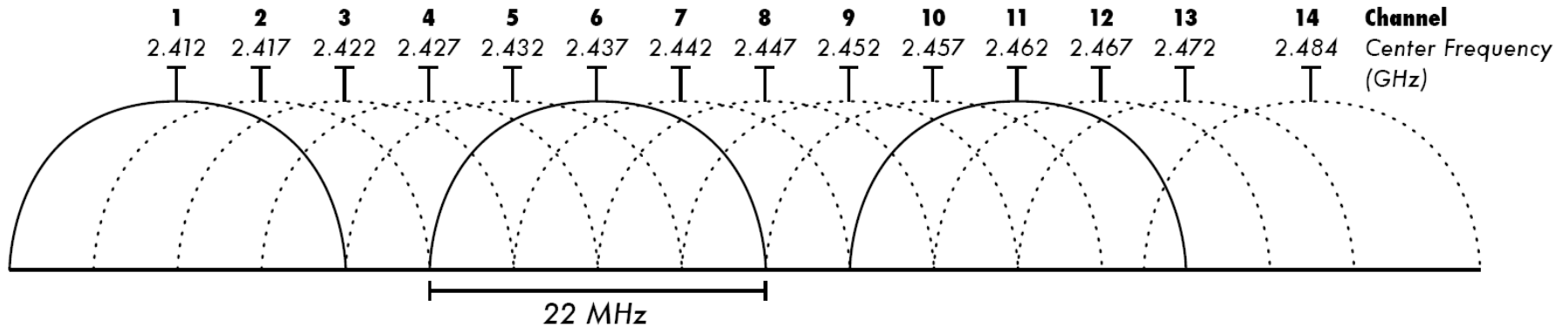
Laptop Adapter
(PC Card)

IEEE 802.11

IEEE 802.11 Wireless LAN (Wi-Fi)

- Alle bruker **CSMA/CA** for tilgang
- Alle tilbyr både base-stasjon (AP) og ad-hoc nettverk versjoner
- **802.11b**
 - 2.4 GHz lisensfritt radiobølgområde
 - opp til 11 Mbps
 - direct sequence spread spectrum (DSSS) i fysisk lag
 - Ligner CDMA, men alle vertsmaskiner bruker samme “chipping code”
 - Rekkevidde 38 / 140 m
 - Begynner å fases ut til fordel for **g** og **n**; men de fleste trådløse kort støtter den fremdeles.
- **802.11a**
 - 5-6 GHz
 - opp til 54 Mbps
 - OFDM
- **802.11g**
 - 2.4 GHz området
 - Opp til 54 Mbps
- **802.11n**
 - 2.4 og/eller 5 GHz området
 - Opp til 150 Mbps
 - Rekkevidde 70 / 250 m
 - Flere (4) antenner (**MIMO**)
 - *Forward Error Correction*

Kanaler: 1-13, 802.11 b/g/n



Signalforplantning

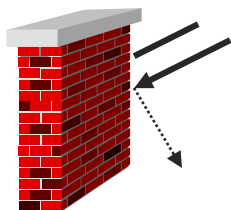
Signalforplantning i fritt rom samme som for lys (rett linje)

Mottatt effekt proporsjonal med $1/d^2$

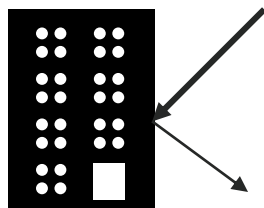
(d = avstand mellom sender og mottager)

Mottatt effekt/signal også påvirket av

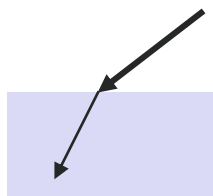
- **demping** (frekvensavhengig)
- hinder, ting og tang
- **refleksjon** fra store hindringer
- **refraksjon** avhengig av tettheten på mediet
- **spredning** fra små hindringer
- **diffraksjon** på kanter



hinder



refleksjon



refraksjon

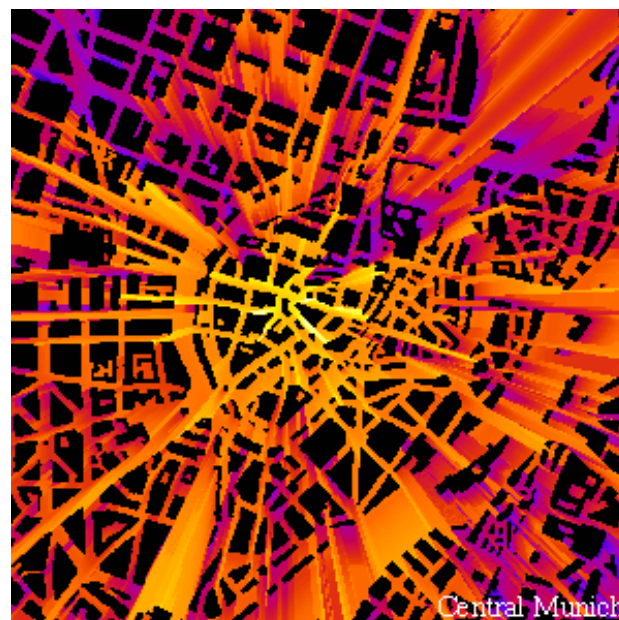
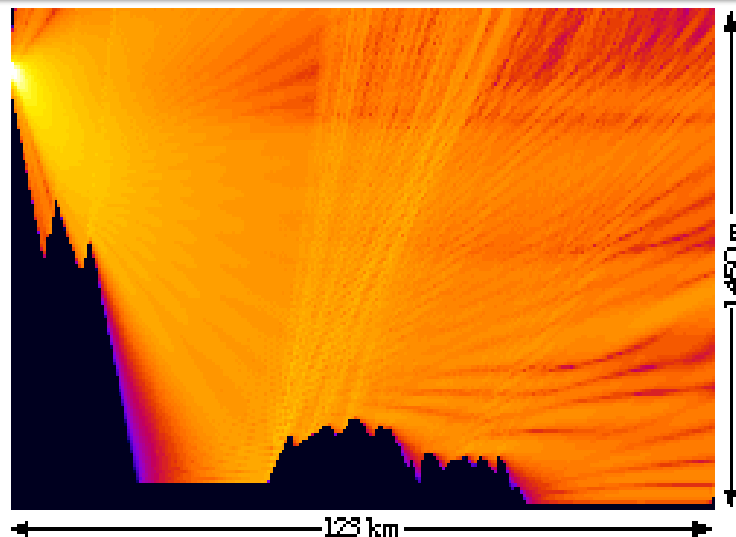
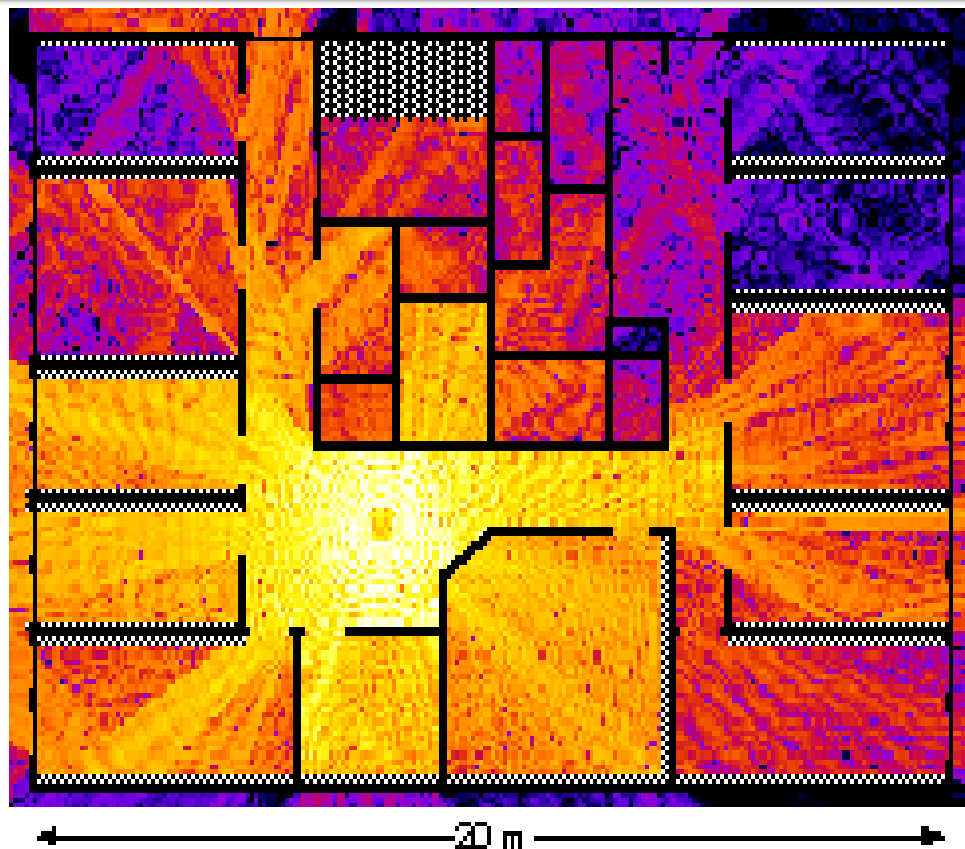


spredning



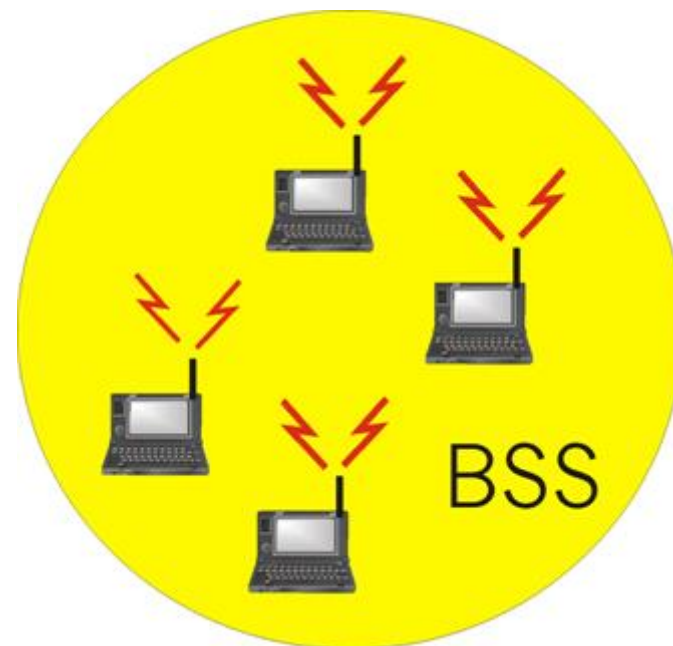
diffraksjon

Eksempel: radiostråling / intensitet



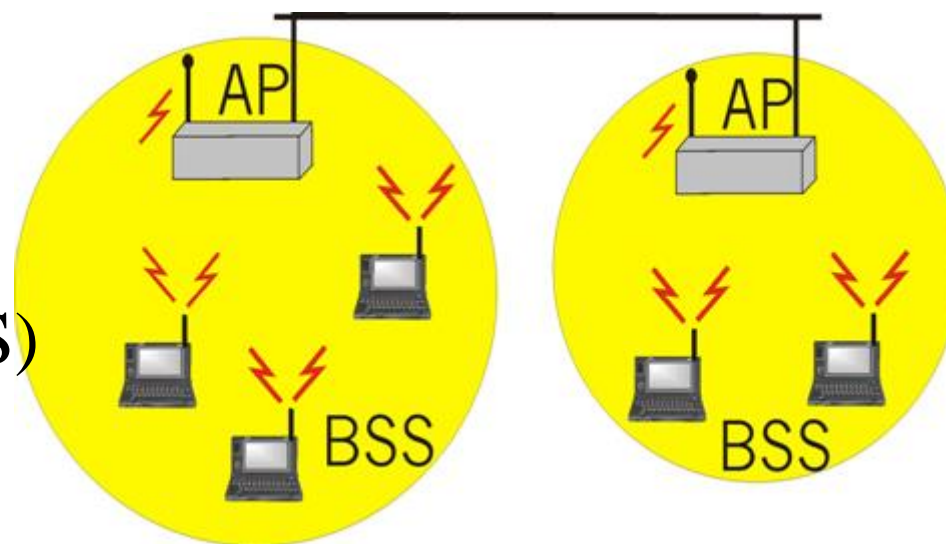
Ad Hoc Nettverk

- Intet AP (i.e., ingen basestasjon)
- Trådløse vertsmaskiner kommuniserer direkte og fungerer som switcher/routere for hverandre.
- Anvendelser
 - “laptop” møter i bilen, på hytta e.l.
 - slagmarken



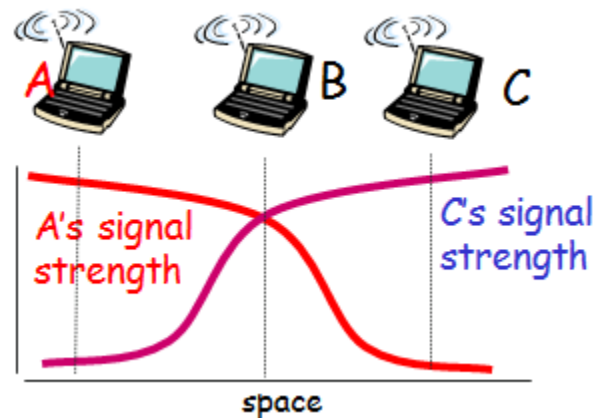
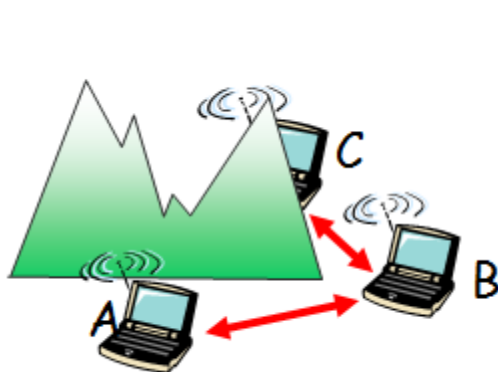
Nettverk med AP

- Vertsmaskinene kommuniserer via en basestasjon
 - basestasjon = **access point** (AP)
- **Basic Service Set (BSS)** (tilsvarer “celle” i mobiltelefonsammenheng) består av:
 - Trådløse vertsmaskiner
 - Access Point (AP)
- BSS'er kombineres til et DistribusjonsSystem (DS) (WLAN)
- BSS har en SSID



Trådløst LAN: Problem -> CA

- Hvis to terminaler er "gjemt" for hverandre kan det oppstå kollisjoner



- Collision Detection er dermed ikke tilstrekkelig
- Bruker CSMA/CA (Collision Avoidance)
- Kan benytte Request-To-Send og Clear-To-Send signaler for å reservere forbindelse

Collision Avoidance

- Problem:
 - To noder, som er skjult for hverandre, sender ut komplette pakker til basestasjonen der de interferer og går tapt.
 - Bortkastet båndbredde for alle!
- Løsning:
 - Små reservasjonspakker!
 - Nodene holder selv orden på reservert tid med egen “network allocation vector” (NAV)

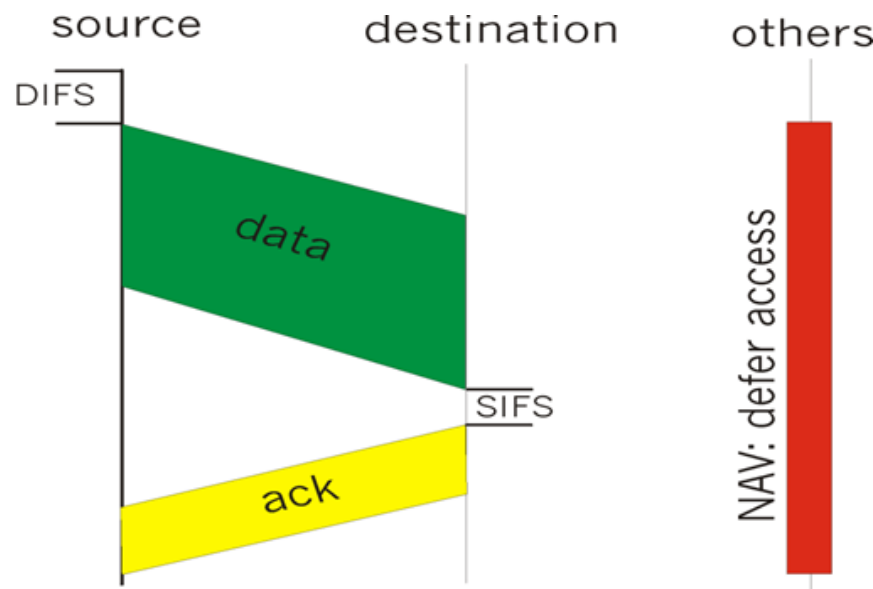
IEEE 802.11 MAC Protocol: CSMA/CA

802.11 CSMA: sender

- if sense channel ledig for **DIFS** sekunder.
then transmit entire frame (ingen kollisjonsdeteksjon)
- if sense channel busy
then binary backoff

802.11 CSMA receiver

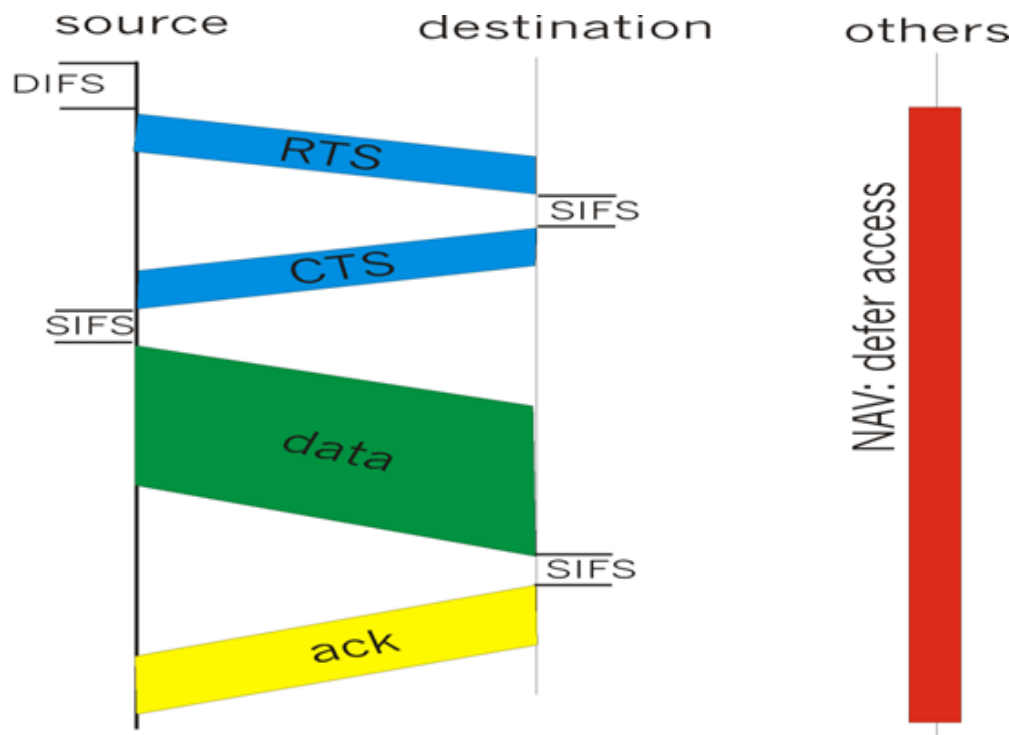
- if received OK
send ACK etter **SIFS**
(ACK nødvendig pga “skjulte noder problemet”)



DIFS = Distributed Inter Frame Spacing
SIFS = Short Inter Frame Spacing

RTS-CTS utveksling

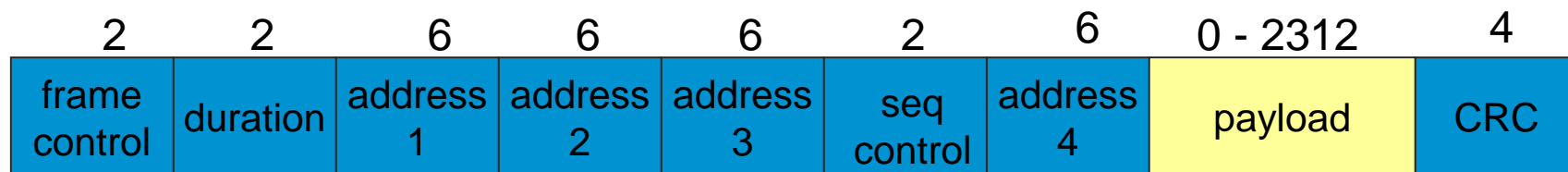
- Sender overfører kort RTS (request to send) pakke: antyder varigheten på overføringen
- Mottager svarer med kort CTS (clear to send) pakke
 - varsler (muligvis skjulte) andre noder
- Skjulte noder avstår da fra å sende i det reserverte tidsrommet: NAV



Ulike CA'er

- IEEE 802.11 tillater altså:
 - “ren” CSMA
 - CSMA/CA
 - Reservasjoner
 - Polling fra AP
 - AP tildeler tidsrom til hver node på rundgang (RR)

802.11 ramme: addressing



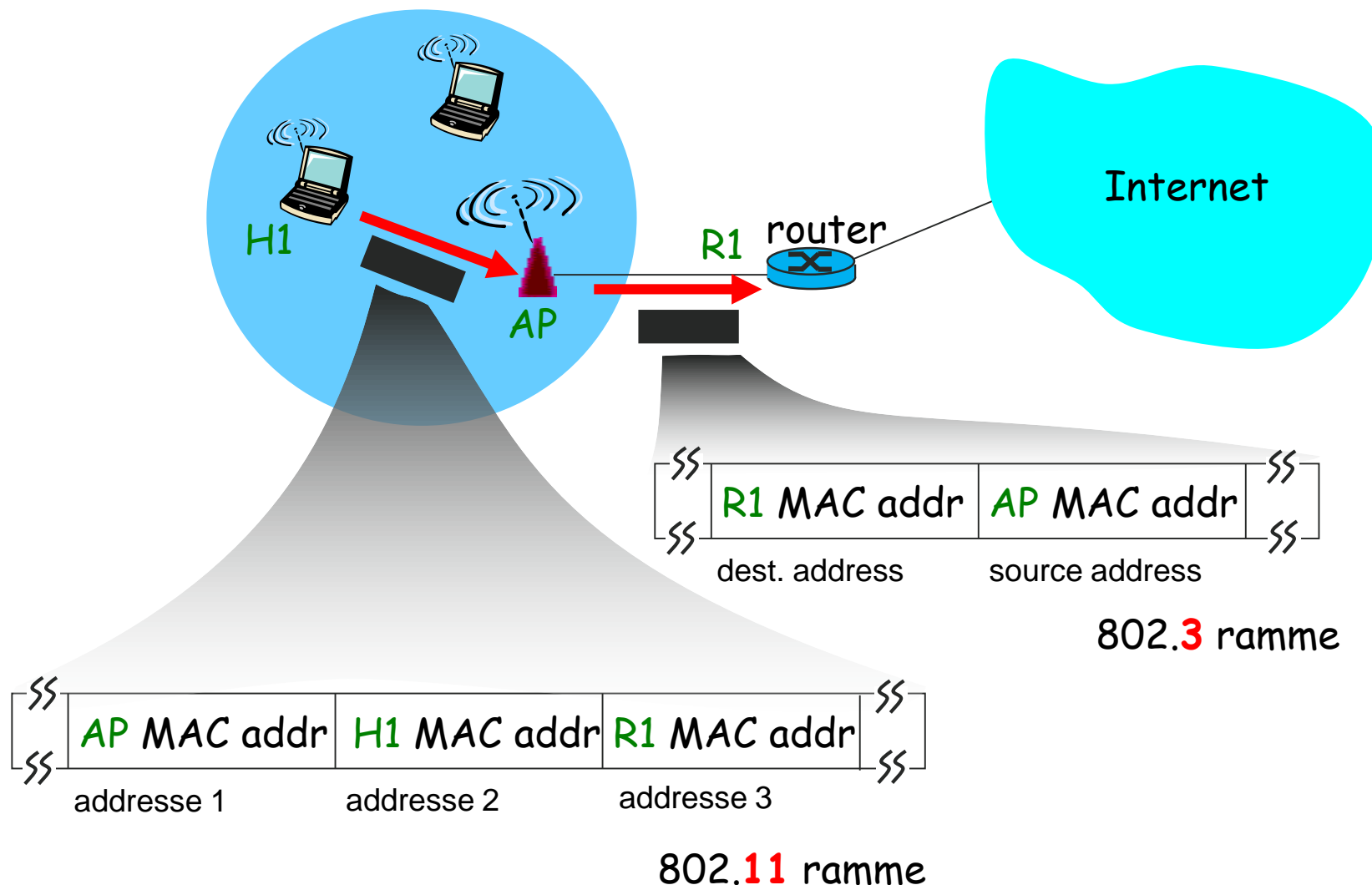
Address 1: MAC adresse til vert eller AP som skal motta rammen

Address 2: MAC adresse til vert eller AP som sender rammen

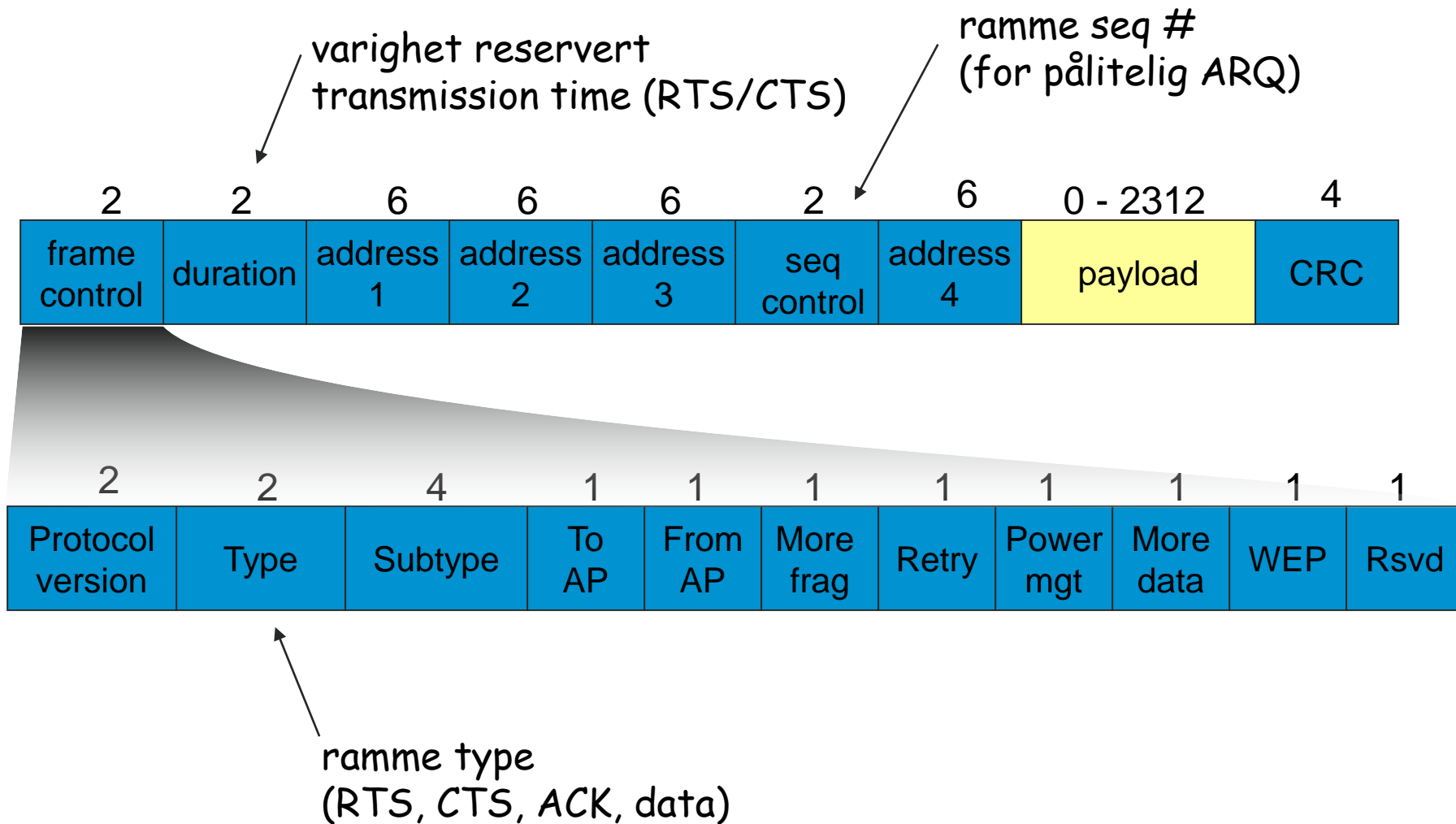
Address 3: MAC adresse til router interface som AP er tilknyttet

Address 3: brukes bare i ad hoc modus

802.11 ramme: adressering



802.11 ramme: mer



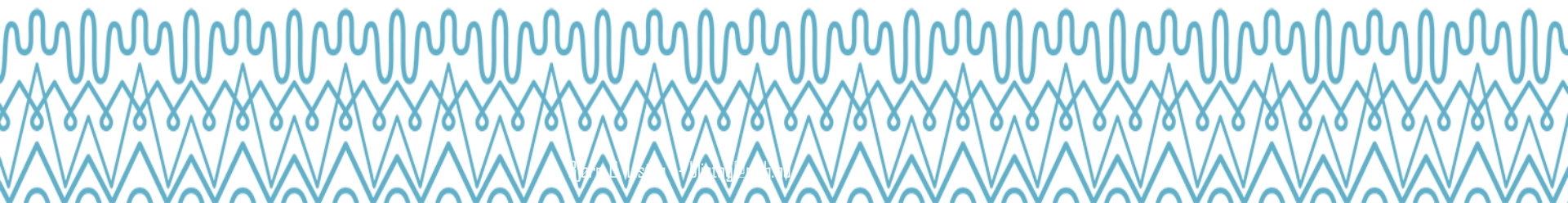
802.11 Sikkerhet & WiFi

- Mål
 - Tilgangskontroll
 - Kun den som blir gitt tilgang skal få det
 - DataIntegritet
 - Ingen skal kunne endre innholdet i datapakkene under overføring («man in the middle attacks»)
 - Konfidensialitet
 - Forhindre avlytting og sesjons-kapring
- Metode
 - Kryptering av trafikken mellom AP og brukermaskin
- Teknikker
 - **WEP**
 - Lett å knekke pga repeterende 40 bit nøkler mm; bedre med 128 bit nøkkel men fremdeles lett å knekke
 - Statisk nøkkel
 - **WPA 1 & 2**
 - Ny krypteringsnøkkel for hver enkelt pakke = bedre kryptering
 - WPA 1 lar seg cracke,
 - WPA 2 lar seg også cracke
 - WPA 2 lar deg også benytte autentisering-server (sikrere en PSK = felles passord))
 - CCMP kryptering er sterkere en TKIP
 - **EAP** (Extensible Authentication Protocol)
 - Skal gjøre ulike WPA-Enterprise metoder interoperable.

Hvordan sikre eget trådløst nett

- Ikke bruk default-innstillinger inn mot eget WLAN/ISP
- Oppdater firmware på AP og drivere for WNIC (minimum årlig)
- Slå av «Remote Administration» på AP
 - Velg komplisert/sikkert Admin-passord
- Velg «uvanlig» IP-nett
 - gjerne fra 172.16.0.0/12 området
 - begrenns DHCP-pool
- Filtrer tilgang på MAC-adresser
- Slå av SSID-broadcast
- Sørg for å sjekke rekkevidden på signalet!
- Bruk WPA2 med langt og komplisert passord

AVSLUTTNING



Hva skal vi kunne?

- Hvilke **oppgaver** løses på linklaget?
- Hvilke mekanismer for feildeteksjon finnes og benyttes?
 - Beregne paritetsbit og bruke 2D-paritet, beregne **CRC**
- Hvilke måter finnes for å dele medium (MA)?
 - Hvordan unngå, eller håndtere **kollisjoner**?
- **MAC-adresser**
 - Oppbygging, multicast, broadcast
- IEEE 802.3 / Ethernet 2
 - **Oppbygging av rammen**
 - Kjenne til de ulike typene.

Hva skal vi kunne?

- Hvorfor **ARP** trengs og hvordan den virker
 - Herunder bruk av `arp`-kommandoen
- Kjenne til de forskjellige LAN-topologiene
 - Buss, stjerne, ring
- Vite forskjellen på **hub** (nav), **bridge** (bro), **switch** (svitsj) og **router** (ruter).
- Kunne forklare hvordan en **switch** fungerer
 - Hvordan bygges switche-tabellen opp?
- Kunne forklare hvilke nye **problemer** som må løses i trådløse nett (IEEE **802.11**)
 - Forklare rollen til AP, hva er SSID

Dagens øving

- Øvingsoppgaver på Canvas
- Skriv ned 3 ting du føler du sliter med, og som du må jobbe mer med før eksamen
- Gjennomfør en tidligere eksamen
- OBS: Vi fortsetter 15.15 😊