

Region Nordjylland

Teknisk kursus, dag 2

Agenda for dag 2



Grundlæggende Netværk

Netværkskomponenter

Databaser

Sikkerhed/Authentication

Grundlæggende Netværk

- Datakommunikation
- Netværksmodeller
- Netværkets virkemåde
- Netværkskomponenter

Datakommunikation

- Datakommunikation er en proces, hvor digital information overføres mellem to eller flere enheder eller systemer. Det kan inkludere overførsel af data som tekst, billeder, lyd eller video og kan foregå over forskellige typer netværk som lokale netværk (LAN), større netværk (WAN) eller internettet.
- Processen med datakommunikation involverer typisk, at data sendes og modtages ved hjælp af en kommunikationsprotokol, som sikrer, at data overføres på en pålidelig og sikker måde. Protokollerne kan variere afhængigt af hvilken type netværk, der anvendes, og hvilken type data, der overføres.
- Datakommunikation spiller en vigtig rolle i moderne netværks teknologier, og det er vigtigt at forstå, hvordan data bliver overført og behandlet, for at kunne udnytte teknologien på bedst mulig måde.

Begreber og forkortelser

- (Kommunikations)Protokol er en sæt af regler og standarder, der definerer, hvordan information udveksles mellem enheder eller systemer i en kommunikationskanal. Det fungerer som en slags "sprog" eller "vejledning" for kommunikationen mellem enheder, så de kan forstå hinanden og effektivt udveksle data.
- Interface er en defineret grænseflade eller et sæt regler og metoder, der specificerer, hvordan to enheder, komponenter eller systemer kan interagere eller kommunikere med hinanden. Det fungerer som en kontrakt, der angiver, hvilke operationer der kan udføres, og hvilke data der kan udveksles mellem de involverede parter.
- Kommunikationskanal er den vej, gennem hvilken information eller data overføres mellem en sender og en modtager i en kommunikationsproces. Kanalen fungerer som forbindelsen mellem senderen og modtageren og muliggør transmission af information.

Begreber og forkortelser

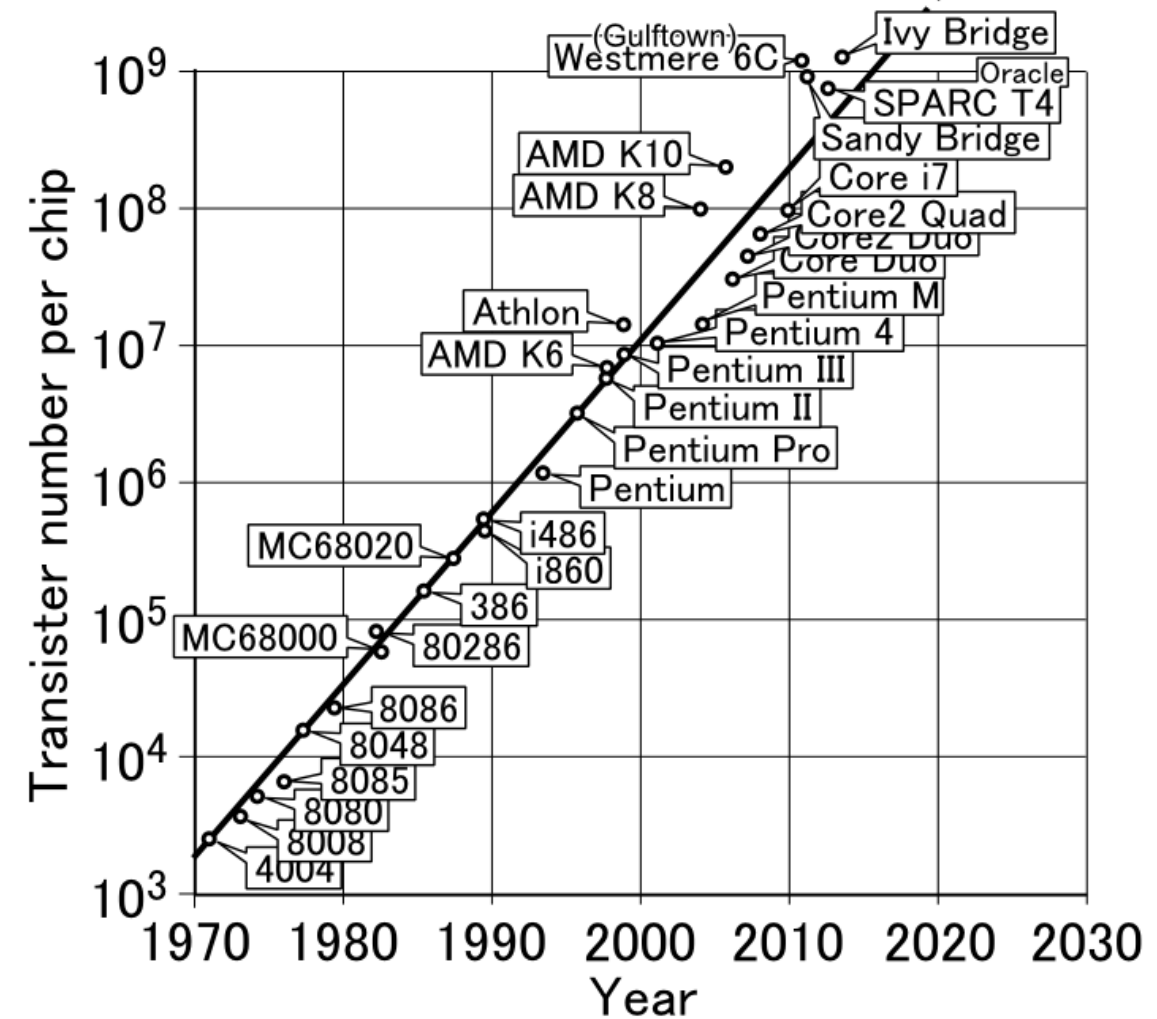
- Bit repræsenterer en logisk værdi der kan antage to tilstande 0 eller 1 (sand eller falsk)
- Byte en størrelse på 8 bit kan f.eks anvendes til at repræsentere tal med
- Binær talsystem anvendes til at repræsentere værdier via strenge af binære værdier (0 og 1) eksempel 11101001
- Hexa decimal talsystem anvender symbolerne 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F eksempel 0xFF

Udvikling af teknologien

- *Computerens udvikling*

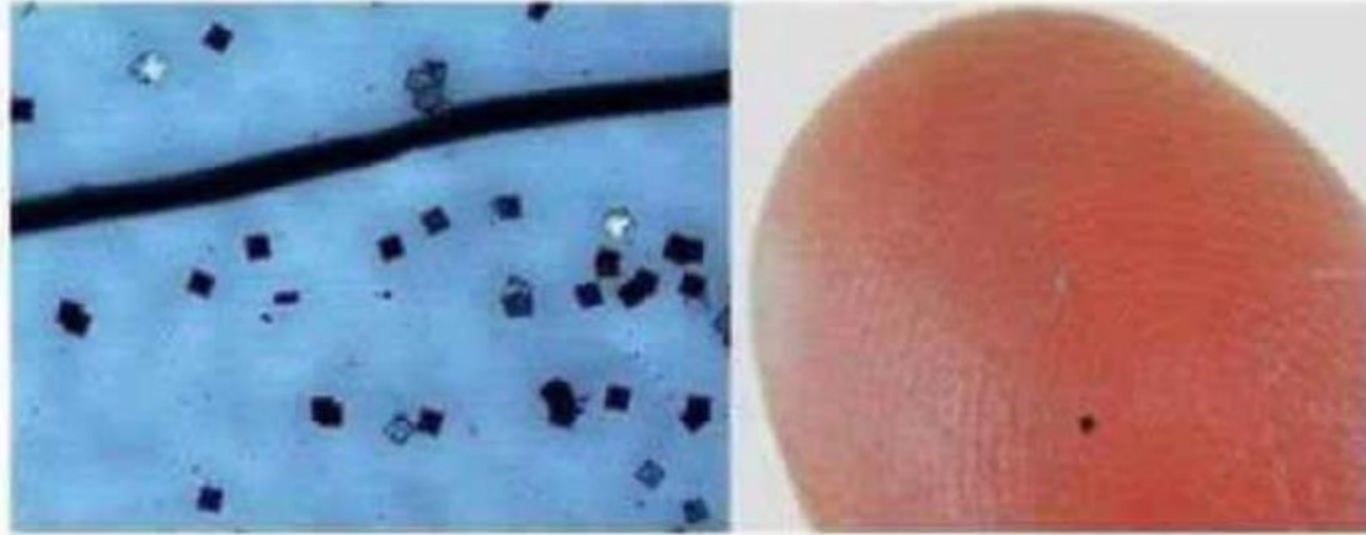


Moore's law



Hvor småt kan det blive ?

- Smart dust



"These are made by Hitachi. They measure only .15X.15 mm each and have GPS capabilities! Sometimes called 'smartdust' as they can be sprayed on us and absorbed or taken in foods, drinks and even injected."

Elite Revolt

Historisk

Telegrafen opfundet 1837



Semaforkommunikation: 1793



1876

1920

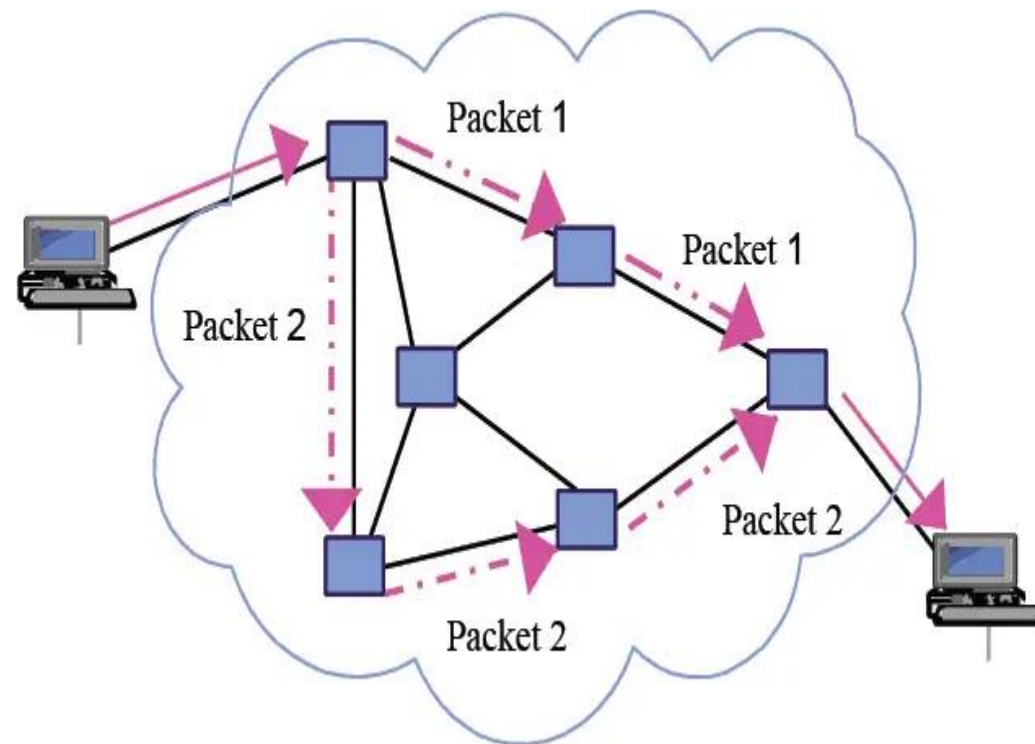
1954

2016



Packet Switching

- Packet switching er en metode til at sende data over et netværk, hvor dataene opdeles i mindre pakker og sendes individuelt gennem netværket. Hver pakke indeholder både data og metadata, der indeholder informationer om, hvor pakken skal sendes hen i netværket, samt andre relevante oplysninger.
- Når en pakke sendes gennem netværket, følger den forskellige ruter og kan passere gennem flere forskellige netværksenheder, såsom routere eller switche. Hver af disse enheder undersøger metadataen i pakken for at afgøre, hvor pakken skal sendes hen næste gang, og sender derefter pakken videre til den næste enhed i ruten.



Packet Switching

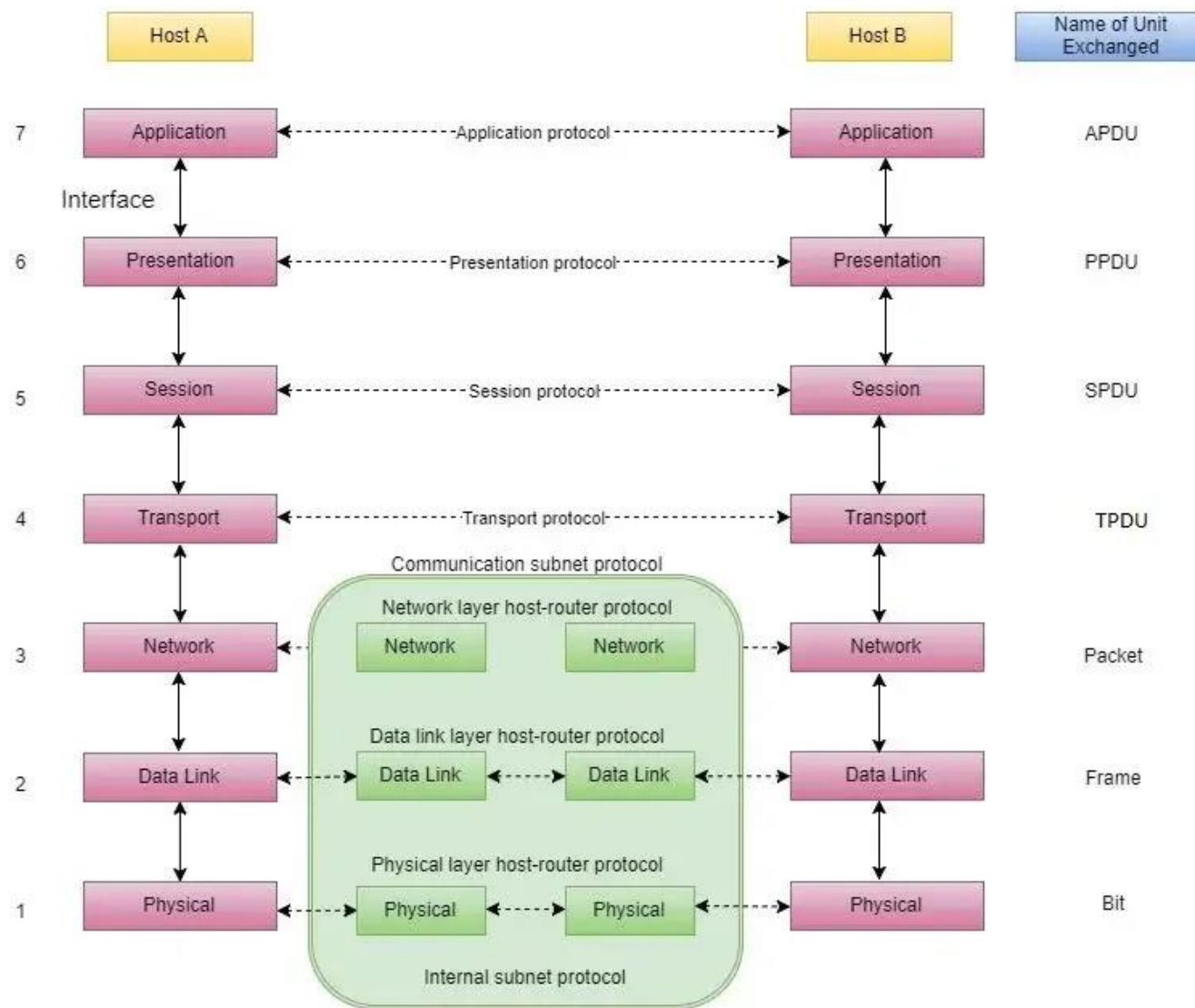
- På denne måde kan flere pakker sendes samtidigt gennem netværket, og de kan bruge forskellige ruter for at nå deres destination. Dette gør det muligt for netværket at optimere brugen af båndbredde og at undgå overbelastning af enkelte ruter.
- Packet switching er en effektiv måde at sende data over et netværk på, da det gør det muligt for flere brugere at dele netværksressourcerne og sikrer, at dataene kan komme frem, selv hvis der er problemer i netværket, f.eks. hvis en rute er blokeret.
- TCP/IP-protokolsuiten, som er grundlaget for internettet, er et eksempel på en protokol, der bruger packet switching.

Netværksmodeller

- En netværksmodel er en abstrakt beskrivelse af, hvordan data kommunikeres i et netværk. Det er en standardiseret måde at organisere og beskrive de forskellige lag og funktioner, der er involveret i datakommunikation, og den angiver de regler og protokoller, der skal følges for at sikre en effektiv og pålidelig kommunikation mellem forskellige enheder i netværket.
- Der er forskellige typer af netværksmodeller, men den mest kendte og anvendte er OSI-modellen (Open Systems Interconnection), som er udviklet af International Organization for Standardization (ISO). OSI-modellen består af syv lag, som hver især har en specifik opgave og ansvar i forhold til at håndtere datakommunikation. Disse lag spænder fra den fysiske forbindelse mellem enhederne til applikationslaget, som er ansvarlig for at håndtere selve applikationerne og brugerdataene.

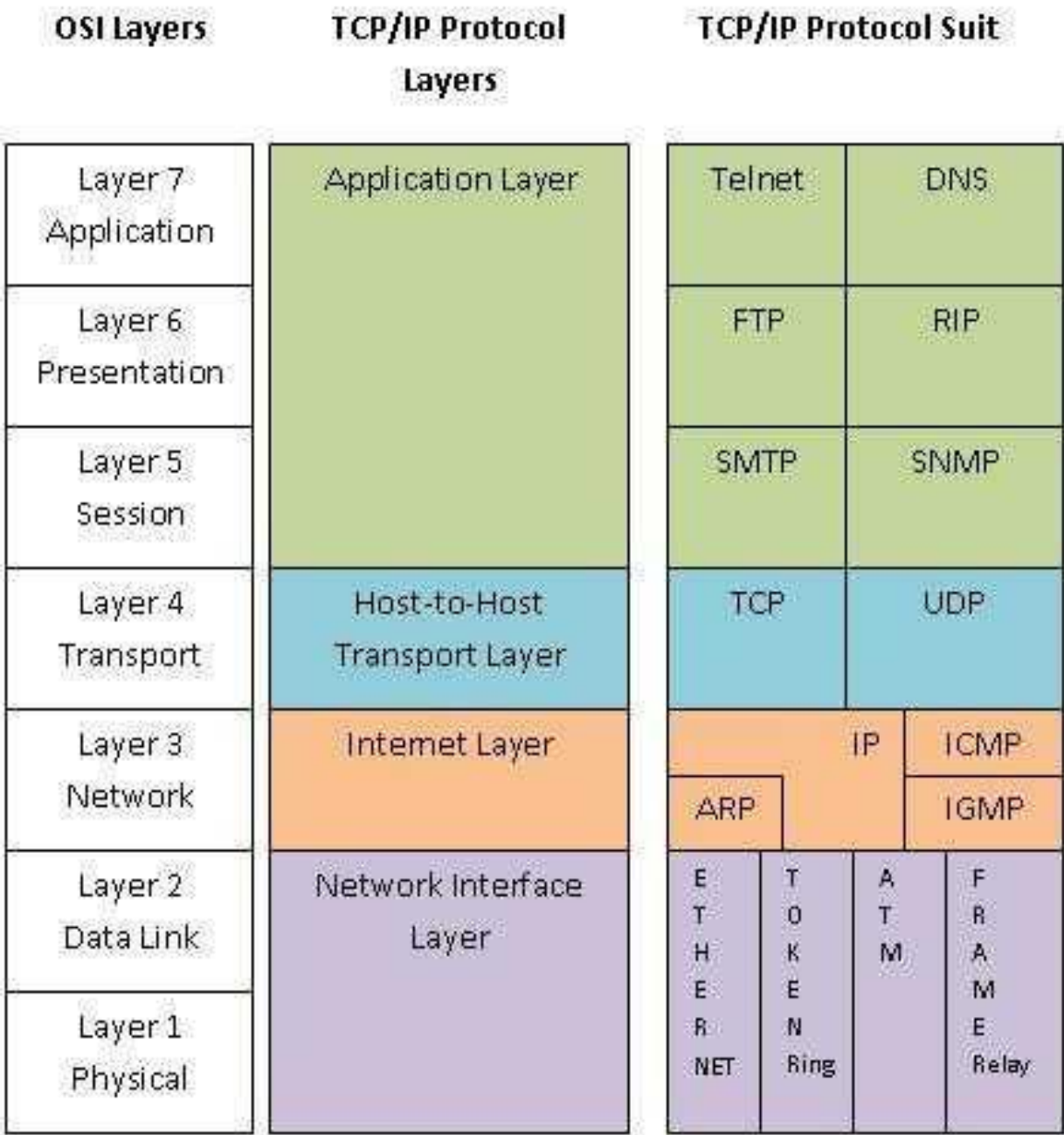
ISO's OSI model

- Netværksmodeller anvendes til at sikre, at forskellige enheder i et netværk kan kommunikere med hinanden på en pålidelig og sikker måde.
- De bruges også til at opbygge og vedligeholde netværksinfrastruktur og til at udvikle og implementere netværksprotokoller og standarder, der sikrer, at netværksenheder og applikationer kan fungere sammen på tværs af forskellige leverandører og platforme.



OSI vs TCP/IP

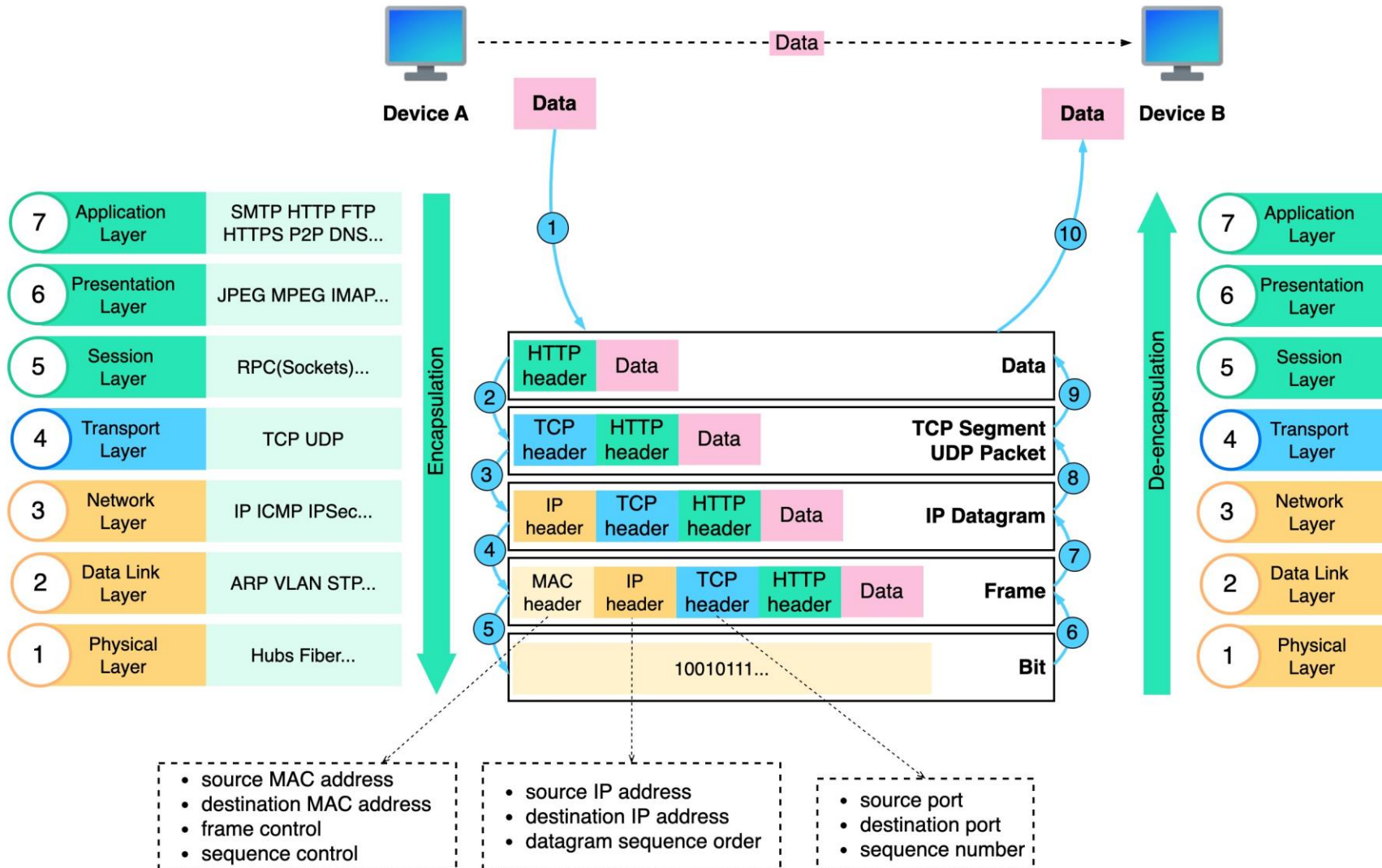
OSI Model	TCP-IP Model
7 layers	4 layers
Model was initially defined before the implementation of the stack.	Model was defined after protocol stack was implemented.
OSI does not support internet working.	TCP-IP supports internet working.
Strict layered	Lossely layered
Support connectionless and connection oriented communication in the network layer.	Support only connection oriented communication in the transport layer.
Horizontal layer	Vertical approach
Separate session layer and presentation layer exist.	There are no session and presentation layers. Characteristics of session layer are provided by transport layer where as characteristics of presentation layer are provided by application layer.



Netværksmodel, indkapsling

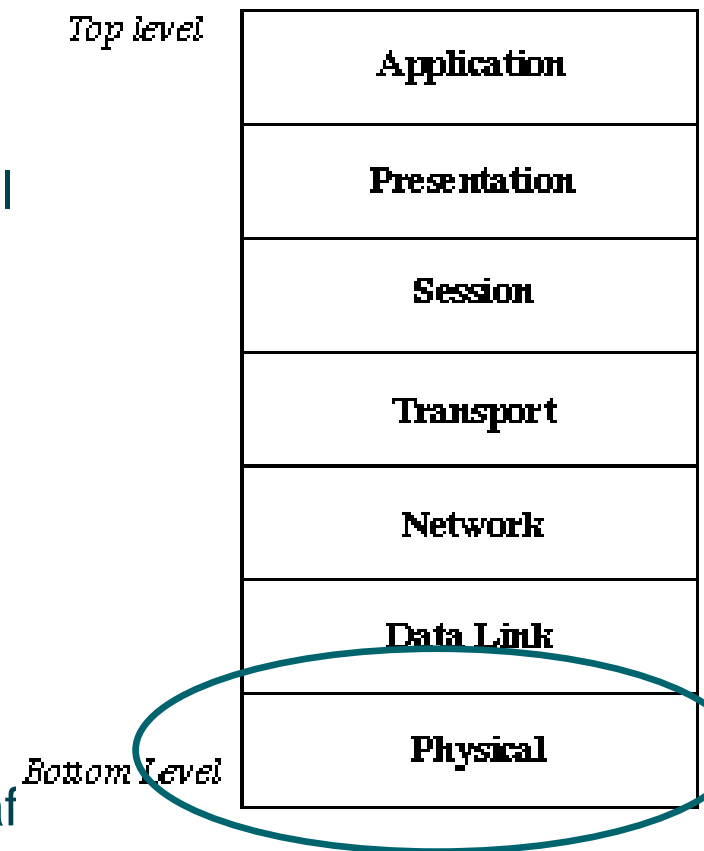
TCP/IP Encapsulation

 blog.bytebytego.com



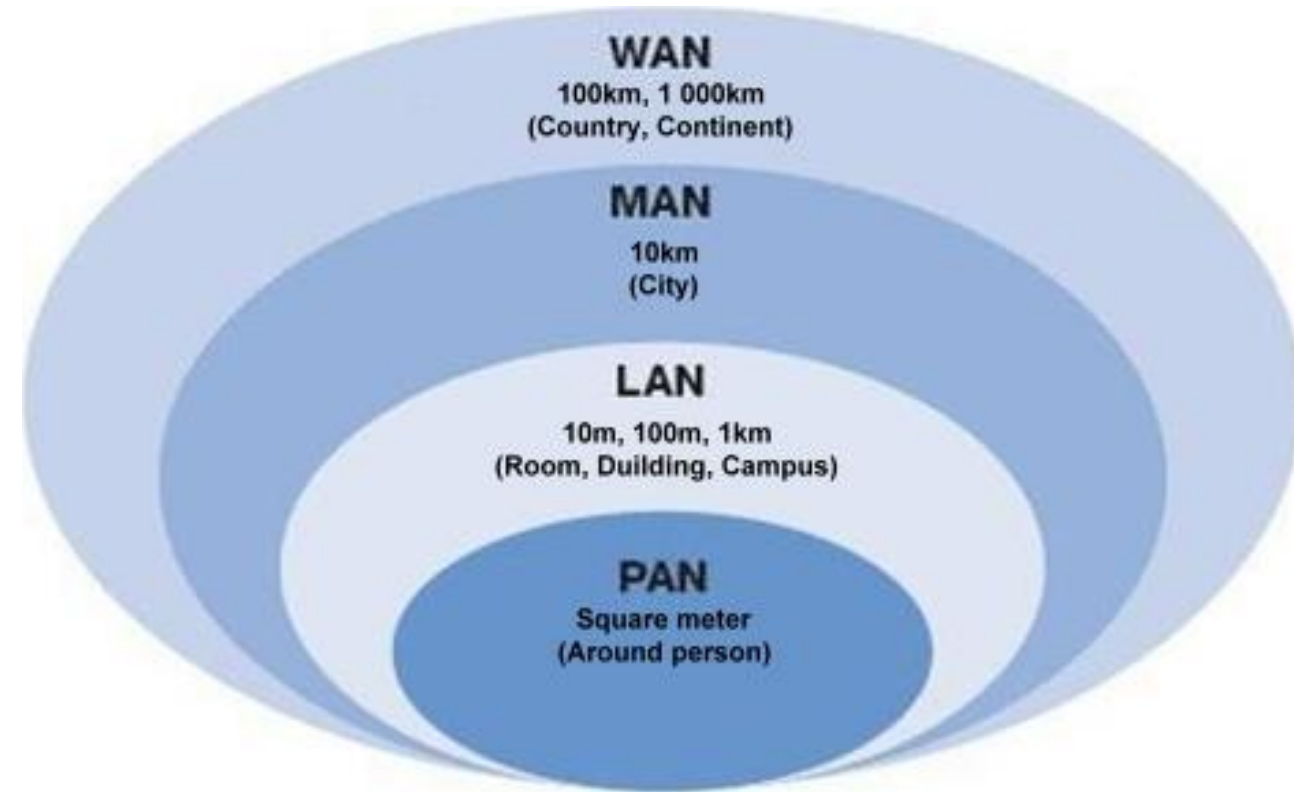
ISO's OSI Model Lag 1

- ISO's OSI-model (Open Systems Interconnection) består af 7 lag, hvoraf lag 1 er det nederste lag i hierarkiet.
- **Lag 1** i OSI-modellen kaldes også for det fysiske lag eller "Physical Layer" på engelsk. Dette lag er ansvarligt for at håndtere den fysiske transmission af data over netværket. Det omfatter hardware-komponenter som kabler, stik, sendere og modtagere, der er nødvendige for at sende og modtage data.
- Det fysiske lag definerer også de elektriske, optiske og mekaniske specifikationer for den fysiske forbindelse mellem enheder, såsom kablets længde, stiktypen og signalfrekvensen.
- Lag 1 har til formål at sikre, at data kan overføres på en pålidelig måde mellem enheder ved at minimere støj, forvrængning og tab af data under transmission.

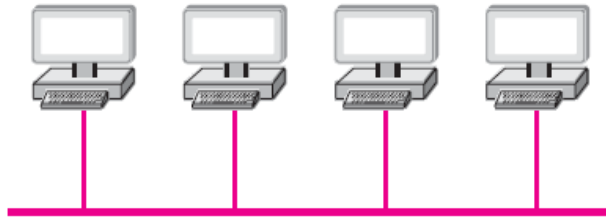


Network type (by size)

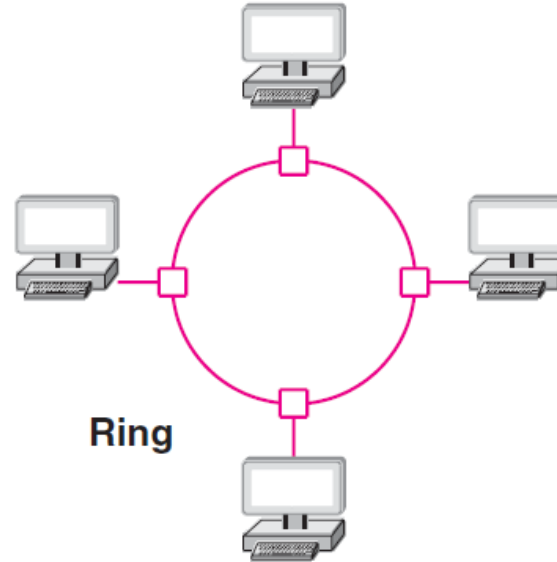
- Personal Area Networks (PAN)
- Local Area Networks (LAN)
- Metropolitan Area Networks (MAN)
- Wide Area Networks (WAN)
- Global Area Networks (GAN)



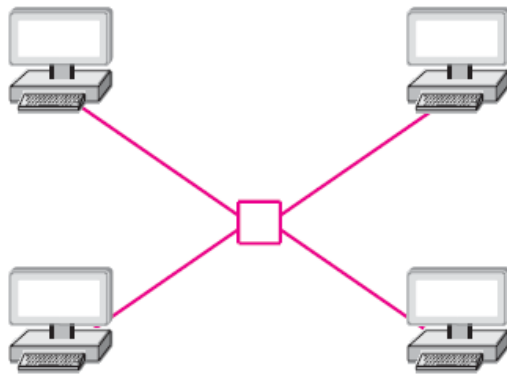
Network topologies



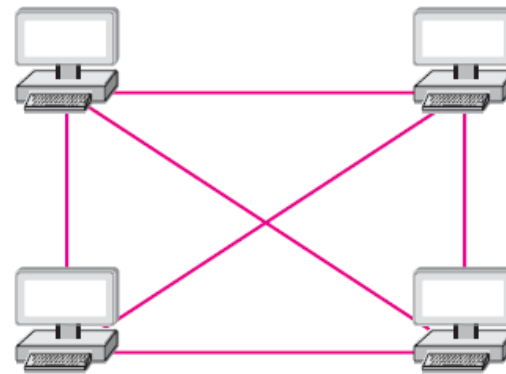
Bus



Ring



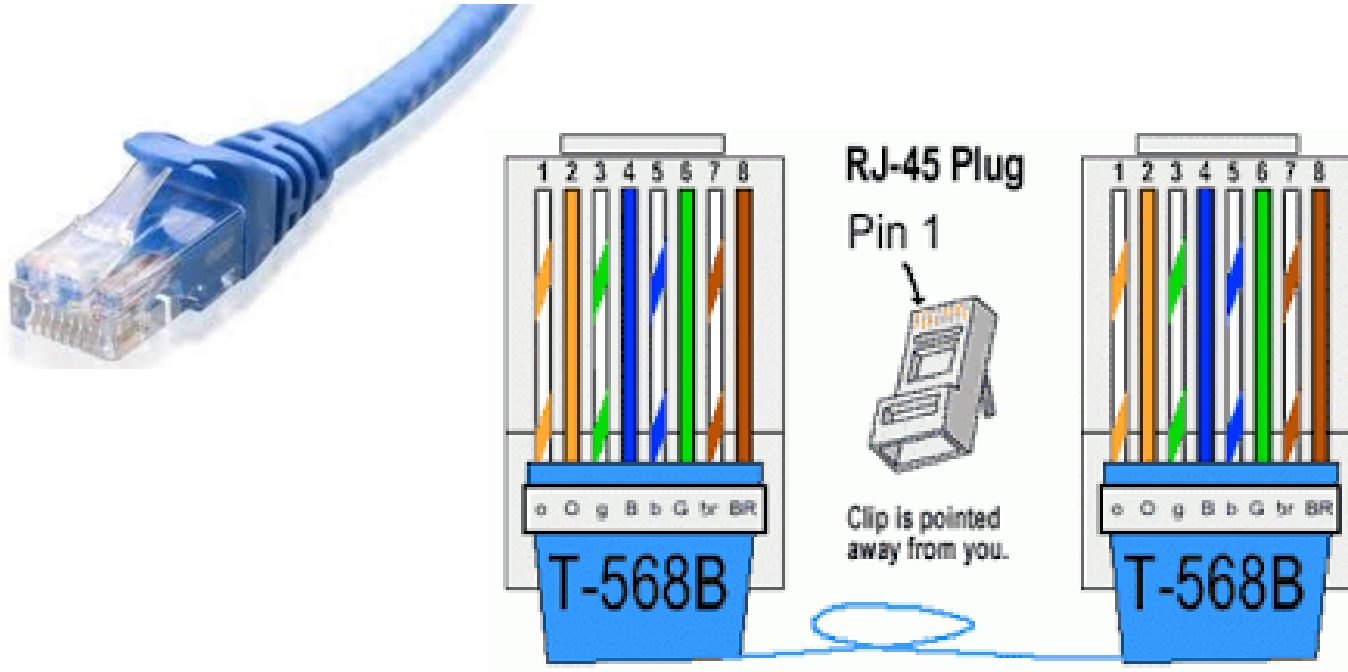
Star



Mesh

connections in a mesh network = $\frac{n!}{(n-2)! 2!} = \frac{n^2 - n}{2}$

Ethernet fysiske stik og sokler



Optiske forbindelser



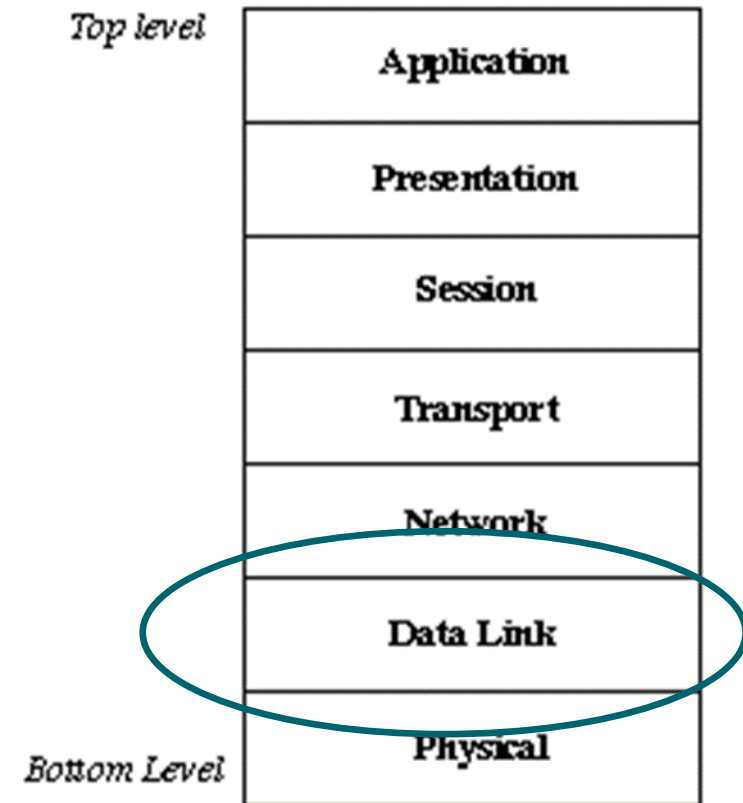
Fiber

SFP

Small Form-factor
Pluggable

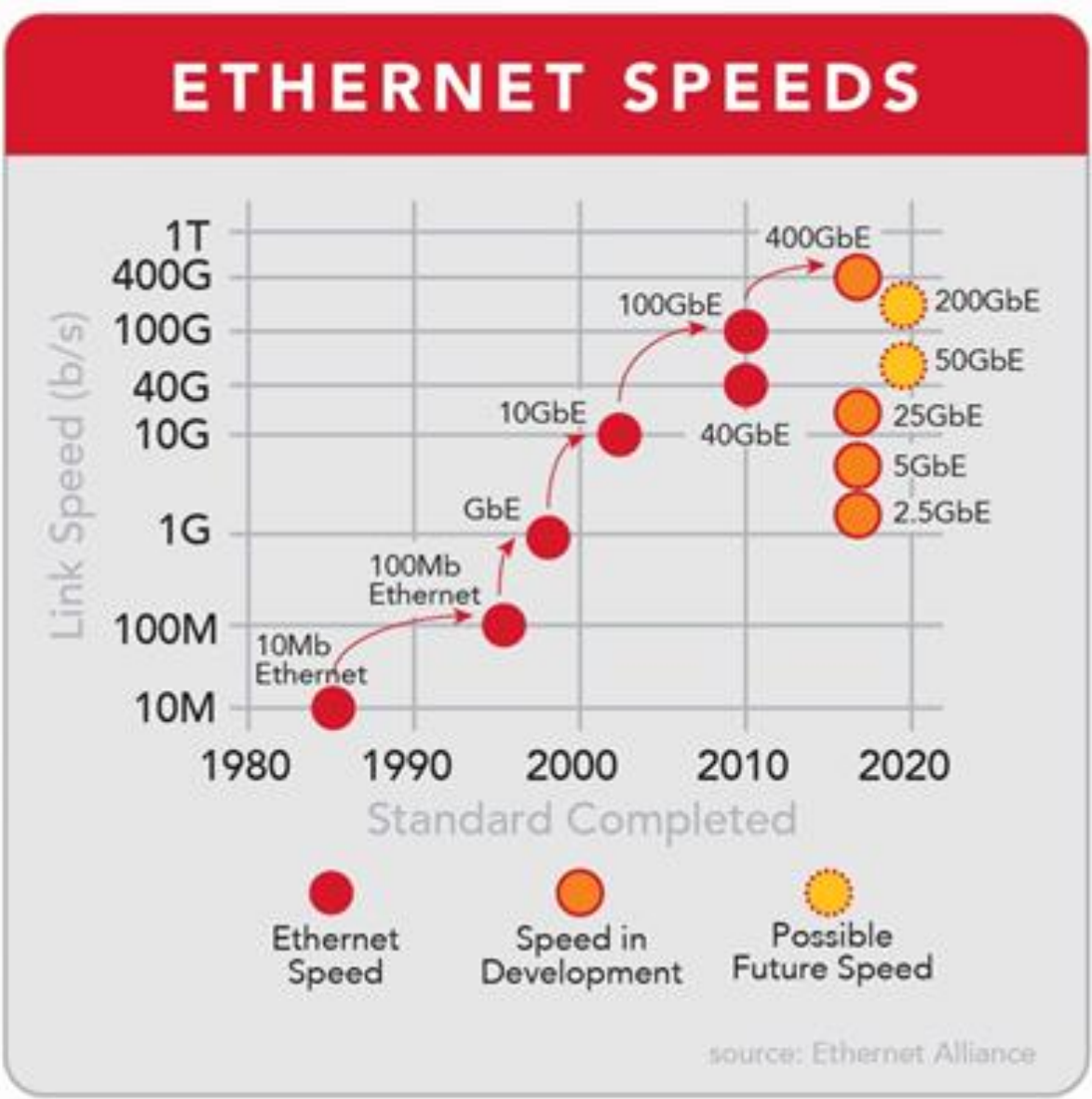
ISO's OSI Model Lag 2

- **Lag 2** i OSI-modellen kaldes også for "Data Link Layer" på engelsk. Dette lag har til opgave at sikre, at dataene kan overføres pålideligt mellem to enheder, der er direkte forbundet til hinanden, som f.eks. en computer og en switch.
- Data Link Layer er ansvarligt for at dele de rå data fra Lag 1 opdeles i logiske enheder kaldet "frames". Hvert frame indeholder en adresse, der identificerer den enhed, som dataene sendes til, og en kontrolsum, som bruges til at detektere eventuelle transmissionfejl.
- Data Link Layer har også ansvaret for at kontrollere adgangen til mediet, så kun en enhed kan sende data ad gangen. Dette kan ske enten ved hjælp af en central koordinator, som f.eks. en switch, eller ved at enhederne på netværket anvender en distribueret algoritme, som f.eks. CSMA/CD (Carrier Sense Multiple Access with Collision Detection).
- Data Link Layer spiller en vigtig rolle i at sikre, at dataene overføres pålideligt, hurtigt og effektivt mellem de direkte forbundne enheder.

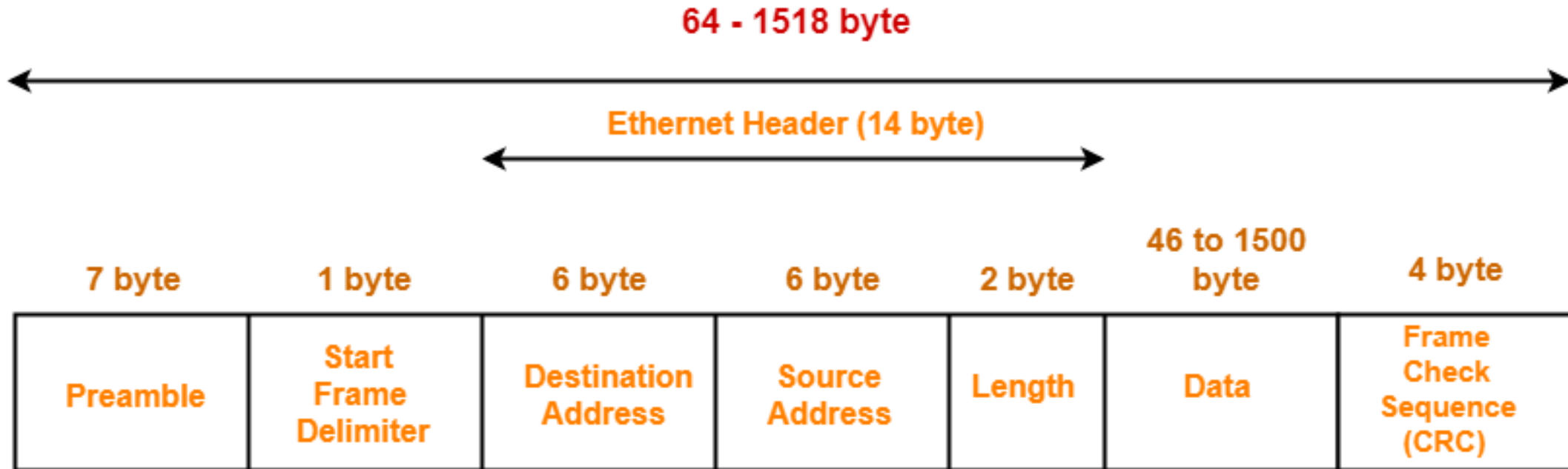


Ethernet

Type	Transmission rate	Frequency bandwidth	Distance	Standard
Cat.8	40Gbps	2000MHz	30m	SSTP
Cat.7	10Gbps	600MHz	100m	SSTP
Cat.6A	10Gbps	500MHz	100m	STP/UTP
Cat.6	1000Mbps	250MHz	100m	STP/UTP
Cat.5e	1000Mbps	155MHz	100m	STP/UTP
Cat.5	100Mbps	100MHz	100m	STP/UTP



Ethernet frame



IEEE 802.3 Ethernet Frame Format

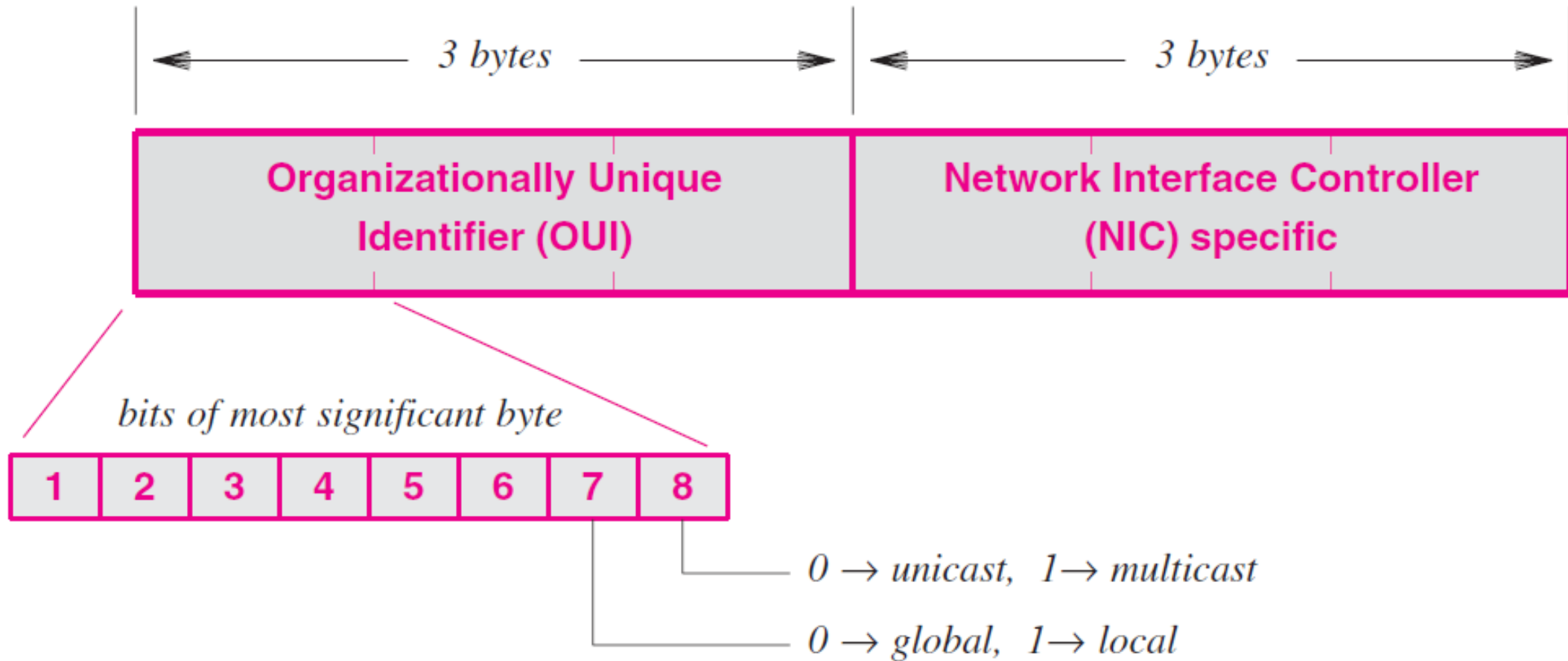
Ethernet frame

- Ethernet frame strukturen er defineret i IEEE 802.3 standarden

Preamble	SFD	Destination MAC	Source MAC	Type	Data and Pad	FCS
7 Bytes	1 Byte	6 Bytes	6 Bytes	2 Bytes	46-1500 Bytes	4 Bytes

- **Preamble** – informs the receiving system that a frame is starting and enables synchronisation.
- **SFD** (Start Frame Delimiter) – signifies that the Destination MAC Address field begins with the next byte.
- **Destination MAC** – identifies the receiving system.
- **Source MAC** – identifies the sending system.
- **Type** – defines the type of protocol inside the frame, for example IPv4 or IPv6.
- **Data and Pad** – contains the payload data. Padding data is added to meet the minimum length requirement for this field (46 bytes).
- **FCS** (Frame Check Sequence) – contains a 32-bit Cyclic Redundancy Check (CRC) which allows detection of corrupted data.

Packet Identification, MAC addresses



Antal MAC adresser ?

- Hvad er det teoretiske antal unikke MAC adresser der er muligt med den valgte adresse størrelse i MAC laget ?

Unicast, Broadcast, and Multicast Addresses

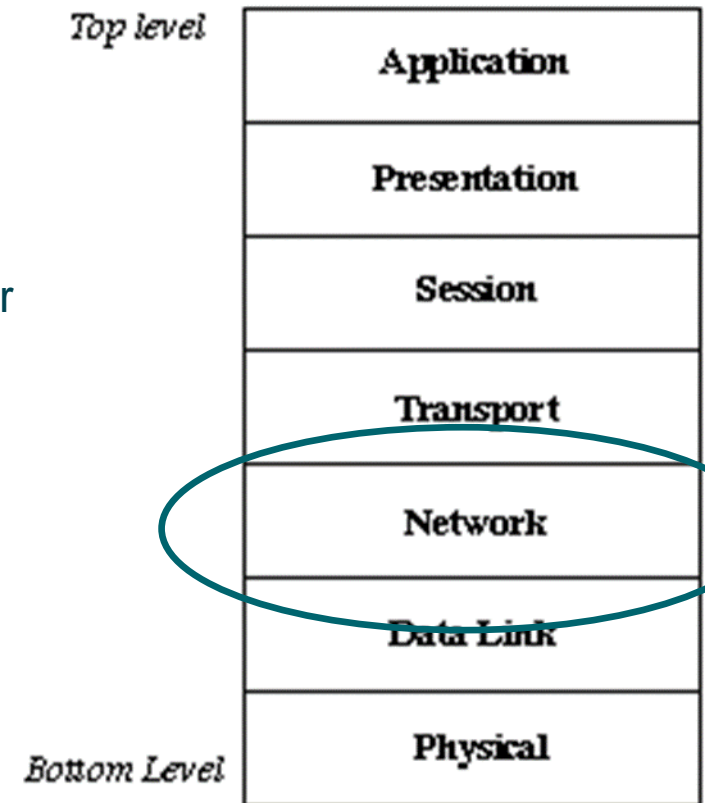
Address Type	Meaning And Packet Delivery
unicast	Uniquely identifies a single computer, and specifies that only the identified computer should receive a copy of the packet
broadcast	Corresponds to all computers, and specifies that each computer on the network should receive a copy of the packet
multicast	Identifies a subset of the computers on a given network, and specifies that each computer in the subset should receive a copy of the packet

Lookup af OUI del af MAC adresse (5 min)





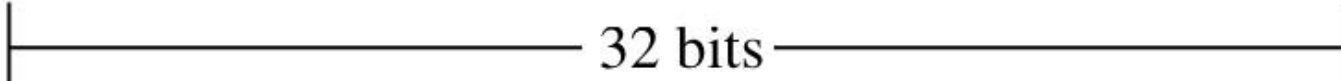
- Hvem er producent af dit netkort (wifi eller ethernet)
- Brug denne url til at slå OUI op med:
- <http://www.whatsmyip.org/mac-address-lookup/>
- Hint:
- Windows brug kommandoen:
- **ipconfig /all**
- Linux brug kommandoen:
- **ip address**
- til at finde mac adresse (fysisk adresse)

ISO's OSI Model Lag 3

- **Lag 3** i OSI-modellen kaldes også for "Network Layer" på engelsk. Dette lag har til opgave at sikre, at data kan overføres på tværs af netværk og routere i et distribueret netværksmiljø.
- Network Layer bruger en logisk adresse, som kaldes en IP-adresse (Internet Protocol), til at identificere de forskellige enheder på netværket. Når data skal sendes til en enhed på et andet netværk, bruger Network Layer routingprotokoller, som f.eks. OSPF (Open Shortest Path First) eller BGP (Border Gateway Protocol), til at bestemme den bedste vej gennem netværket.
- Network Layer er også ansvarligt for at fragmentere og reassemblere data, når det er nødvendigt for at sikre, at dataene kan sendes og modtages på tværs af netværket. Dette sker, når datastørrelsen overstiger den maksimale størrelse, som kan overføres på en enkelt netværksforbindelse, og det kræver, at dataene opdeles i mindre stykker.
- På dette lag er der også nogle protokoller, som er designet til at tage sig af netværkssikkerhed, herunder IPsec (Internet Protocol Security) og VPN (Virtual Private Network), som kan kryptere og beskytte dataene mod uautoriseret adgang og modifikation.
- Network Layer spiller en vigtig rolle i at sikre, at data kan overføres på tværs af netværk og routere på en pålidelig og effektiv måde.



Struktur på en 32-bit IPv4 adresse

decimal	10	•	10	•	20	•	1
binary	00001010		00001010		00010100		00000001
							
	octet-4		octet-3		octet-2		octet-1
							

IPv4 IPv4 Address Classes og Address Range

Class	IP Address Range
A	0.0.0.0 to 127.255.255.255
B	128.0.0.0 to 191.255.255.255
C	192.0.0.0 to 223.255.255.255
D	224.0.0.0 to 239.255.255.255
E	240.0.0.0 to 254.255.255.255

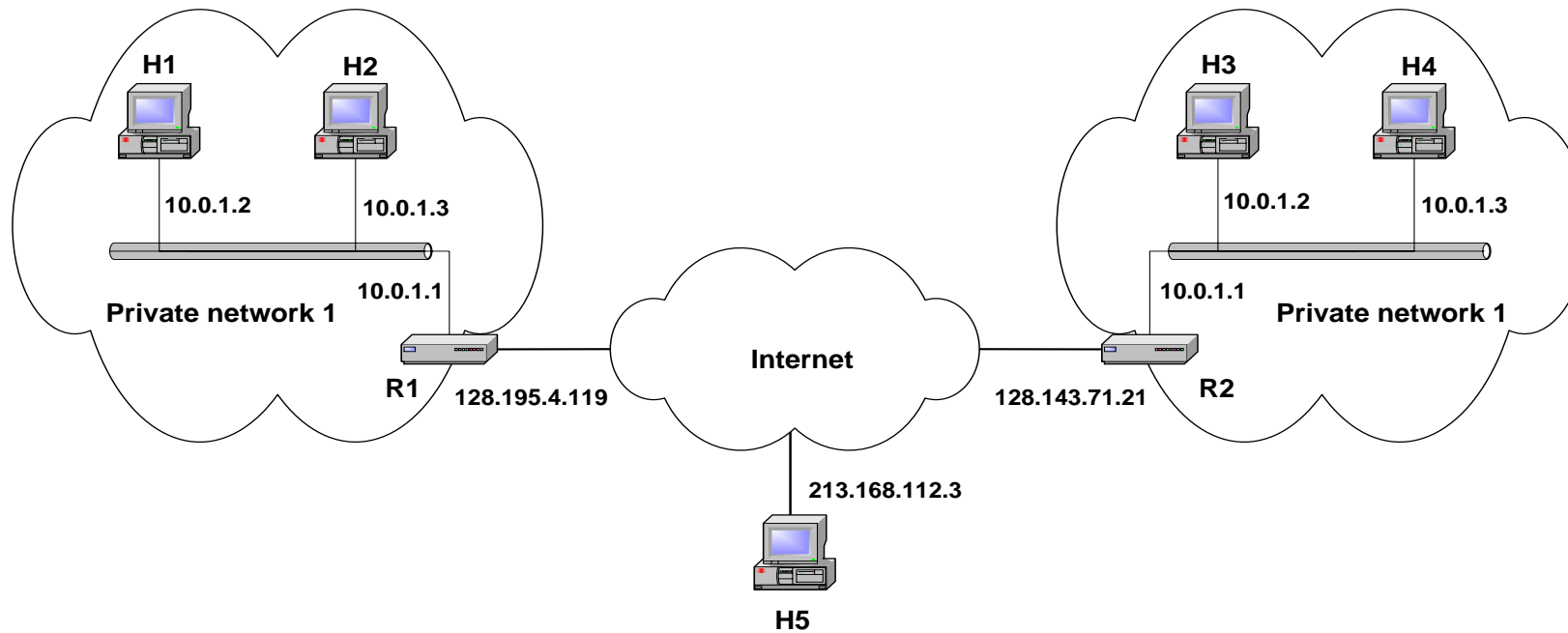
Private Netværk

- *Private IP* netværk er et IP network der ikke har direkte forbindelse til Internet
- IP adresser i et privat netværk kan tildelse enheder som man ønsker.
 - De skal ikke registreres og behøves IKKE at være global unikke, dog skal det være såkaldte private adresser

Private IPv4 adresser

Class	Adresseområde (range)
A	10.0.0.0 til 10.255.255.255
B	172.16.0.0 til 172.31.255.255
C	192.168.0.0 til 192.168.255.255

Private netværk



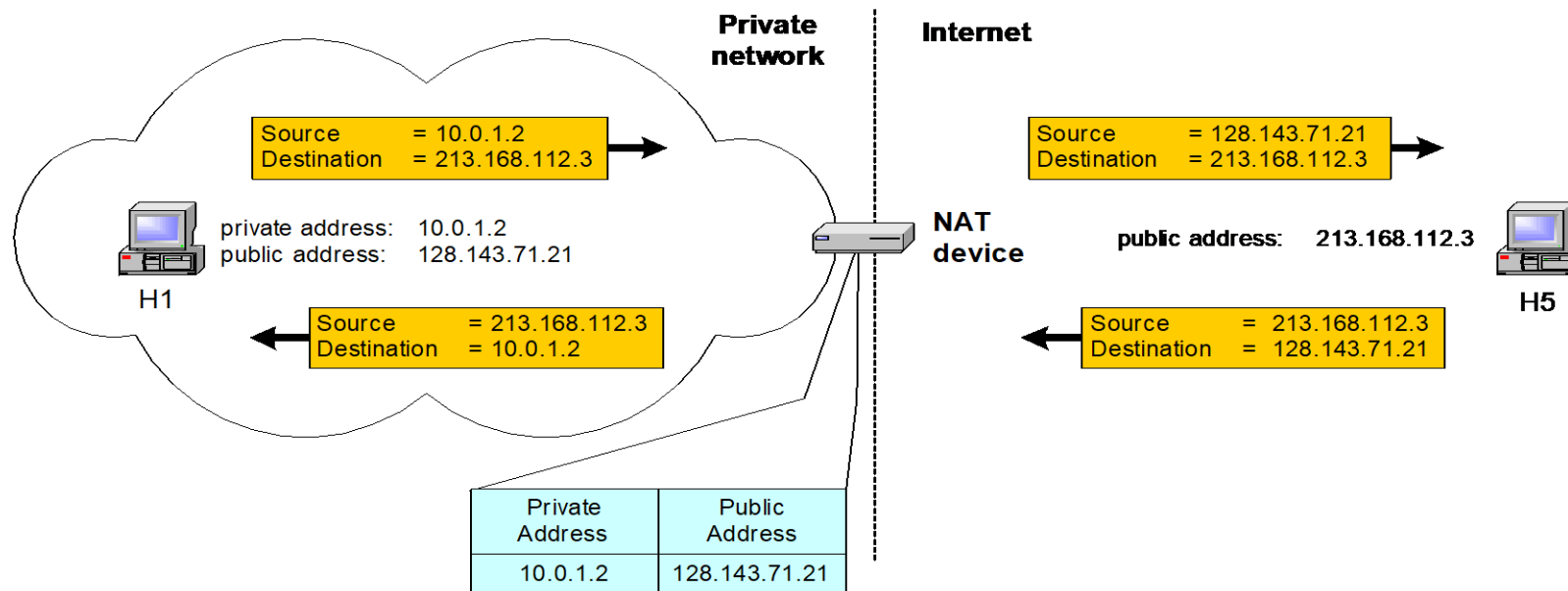
Network Address Translation (NAT)

- RFC 1631 (Udklip)
- A short term solution to the problem of the depletion of IP addresses
 - Long term solution is IP v6
 - CIDR (Classless InterDomain Routing) is a possible short term solution
NAT is another
- NAT is a way to conserve IP addresses
 - Can be used to hide a number of hosts behind a single IP address
 - Also have some safety properties

Network Address Translation (NAT)

- NAT is a router function where IP addresses (and possibly port numbers) of IP datagrams are replaced at the boundary of a private network
- NAT is a method that enables hosts on private networks to communicate with hosts on the Internet
- NAT is run on routers that connect private networks to the public Internet, to replace the IP address-port pair of an IP packet with another IP address-port pair.

Basic Operation of NAT



- NAT device has address translation table
- One to one address translation

Pooling of IP Addresses

- **Scenario:** Corporate network has many hosts but only a small number of public IP addresses
- **NAT solution:**
 - Corporate network is managed with a private address space
 - NAT device, located at the boundary between the corporate network and the public Internet, manages a pool of public IP addresses
 - When a host from the corporate network sends an IP datagram to a host in the public Internet, the NAT device picks a public IP address from the address pool, and binds this address to the private address of the host

ARIN (www.arin.net)

- IP addresses are assigned by **ARIN**, the American Registry for Internet Numbers.
- ARIN assigns IP address space to Internet Service Providers (ISP) and end users. ARIN only assigns IP address space to ISPs and end users if they qualify.
- This requires that the ISP or end user be large enough to merit a block of addresses. In the case where blocks of addresses are allocated by ARIN to the ISPs, the ISPs issue addresses to their customers.
- For example, a Telco could be the ISP that has a large block of IP addresses and issues an IP address to a user. A local ISP could also be assigned a block of IP addresses from ARIN, but the local ISP must have a large number of users.

Local host IPv4 adresse (mig selv 😊)

- 127.x.x.1

Tools for network, Linux og windows

Linux	Windows	Purpos
ip address ping tracert dig netstat -rn (1)	ipconfig /all ping tracert nslookup route print -4	Network addresser Testing connection Trace Route to dest Lookup DNS entry Show routing
wireshark	wireshark	For sniffing the network

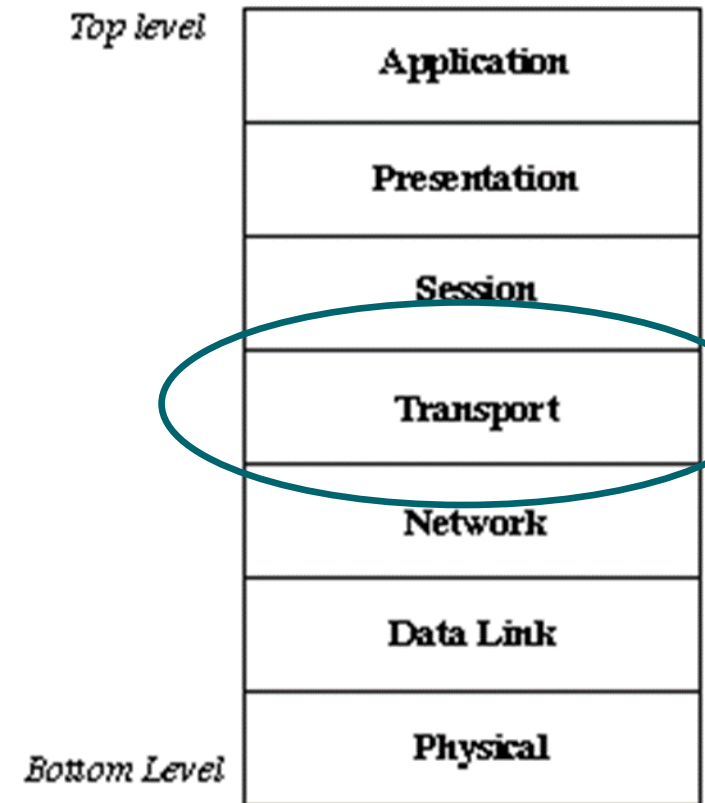
(1) Del af pakken "net-tools", installeres med kommandoen:
sudo apt-get install net-tools

Hvilke IP adresse bruger dit WiFi netkort ? (5min)

- Undersøg om der på det net du er tilkoblet anvendes NATning.
- Brug kommandoen `ipconfig /all` til at finde dit WiFi netkorts IP adresse
- Du kan herefter med hjemmesiden <https://www.showmyip.com/> se hvilken IP adresse du anvender når du går på nettet.
- Hvad fortæller de IP adresser du får ved hjælp af `ipconfig` og hjemmesiden `showmyip`??

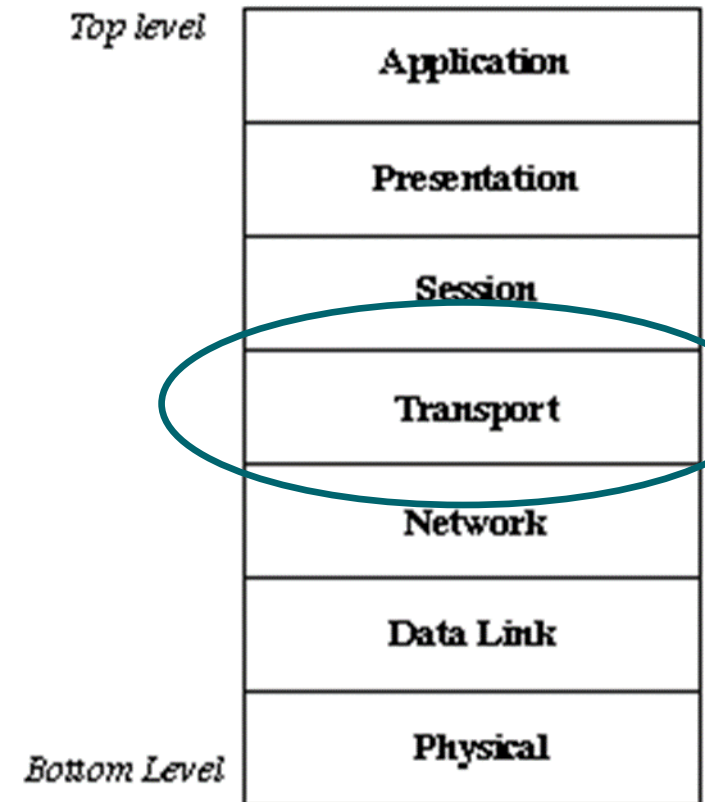
ISO's OSI Model Lag 4

- **Lag 4** i OSI-modellen kaldes også for "Transport Layer" på engelsk. Dette lag har til opgave at sikre, at dataene kan overføres pålideligt og effektivt mellem applikationer, der kører på forskellige enheder i netværket.
- Transport Layer tilbyder forskellige protokoller, som f.eks. TCP (Transmission Control Protocol) og UDP (User Datagram Protocol), som er designet til at styre dataoverførslen mellem applikationer.
- TCP er en forbindelsesorienteret protokol, der opretter en pålidelig forbindelse mellem to enheder, før dataene sendes. Det betyder, at TCP sikrer, at alle data ankommer i den rigtige rækkefølge og uden fejl. Hvis der opstår en fejl under transmissionen, vil TCP genoverføre de tabte data, indtil alle data er modtaget korrekt.

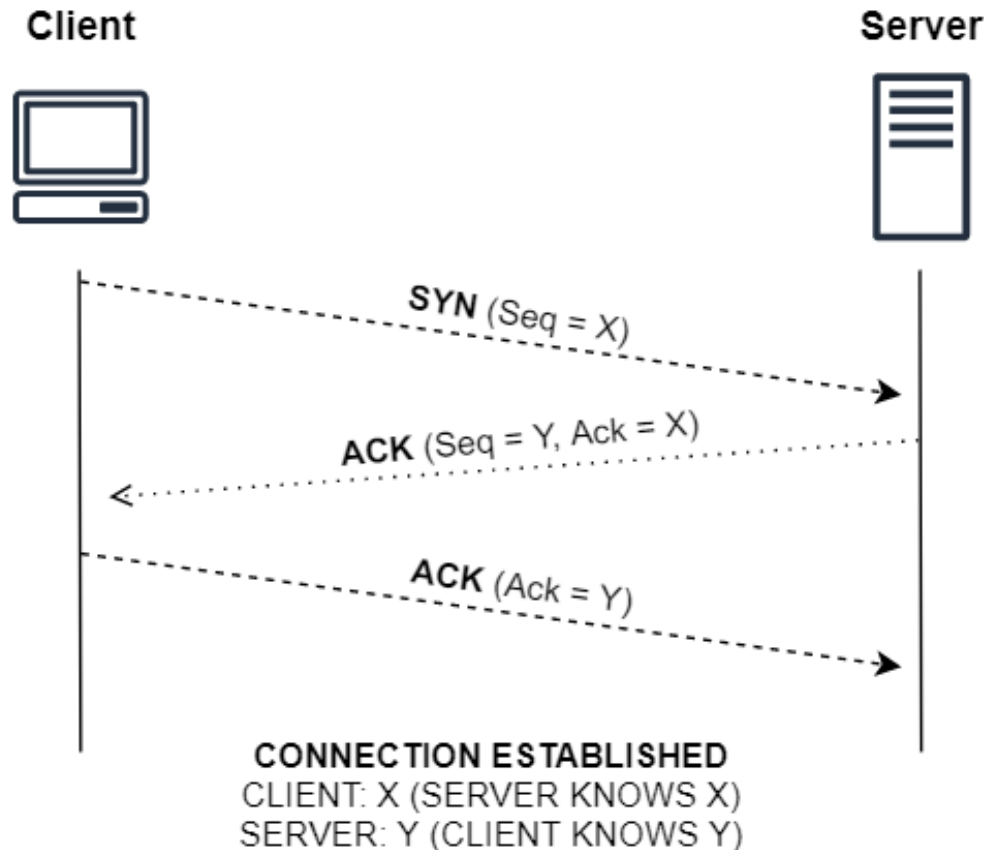


ISO's OSI Model Lag 4

- UDP er en forbindelsesløs protokol, som ikke etablerer en pålidelig forbindelse mellem enhederne, før dataene sendes. Det betyder, at UDP ikke sikrer, at alle data ankommer i den rigtige rækkefølge eller uden fejl. UDP er dog ofte hurtigere end TCP, da den ikke har det ekstra overhead, som TCP har.
- Transport Layer spiller en vigtig rolle i at sikre, at data kan overføres pålideligt og effektivt mellem applikationer på forskellige enheder i netværket.

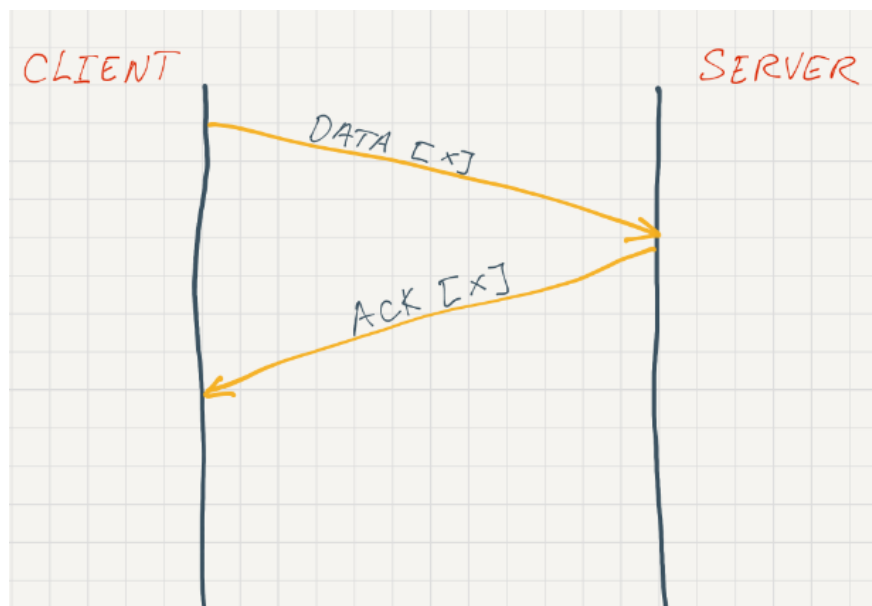


Three-Way Handshake



- Three-way handshake er en metode, der bruges i TCP-protokollen (Transmission Control Protocol) til at etablere en pålidelig forbindelse mellem to enheder, f.eks. en klient og en server, før de begynder at udveksle data. Det består af tre trin, som gør det muligt for begge parter at bekræfte hinandens tilgængelighed og etablere en sikker og pålidelig kommunikationskanal.
- Trinene i en typisk three-way handshake er som følger:
- Trin 1 - SYN: Den klient, der ønsker at etablere forbindelse, sender en SYN-pakke (Synchronize) til serveren. Denne pakke indeholder en sekvensnummerværdi, der bruges til at identificere datastrømmen.
- Trin 2 - SYN-ACK: Serveren modtager SYN-pakken og svarer tilbage med en SYN-ACK-pakke (Synchronize-Acknowledge). Denne pakke bekræfter modtagelsen af SYN-pakken og inkluderer både et bekræftelsesnummer (ACK) og en sekvensnummerværdi til serveren.
- Trin 3 - ACK: Endelig modtager klienten SYN-ACK-pakken og sender en ACK-pakke (Acknowledge) tilbage til serveren. Denne pakke bekræfter modtagelsen af SYN-ACK-pakken og inkluderer det næste forventede sekvensnummer.
- Når serveren modtager ACK-pakken, anses forbindelsen for etableret, og de to enheder kan begynde at udveksle data på en pålidelig og synkroniseret måde.
- Three-way handshake er vigtig for at sikre, at både klient og server er klar til at kommunikere, og det giver mulighed for fejlkontrol og pålidelig dataoverførsel. Det sikrer også, at begge parter er klar over hinandens status og tilgængelighed, inden de påbegynder datatransmission.

Sådan sikres at data når frem til modtager



- Efter der er oprettet en forbindelse via Three-way handshake, sendes der for hver data pakke en kvitering (ACK) tilbage til modtager der bekræfter at data er modtaget korrekt.
- Modtager afsender IKKE efter et vist stykke tid en kvitering sendes data pakken igen.
- Hvis modtageren modtager same data pakke to eller flere gange smides disse væk, men der sendes en kvitering for hver datapakke der modtages.

Hjernevrider (10 min)

- Jeg har lejet en satellit baseret dataforbindelse med en båndbredde på 20 Mbit/sek, jeg sender data i pakker med en størrelse på 1500byte (1 byte = 8 bit) og sender 1 data pakke, hvorefter jeg venter på en acknowledgement pakke fra modtager.
- Jeg forudsætter at de satellitter der anvendes er geostationære satellitter der har en afstand til jordoverfladen på 36.000km, endvidere forudsætter jeg at radiobølgerne bevæger sig med en hastighed på 300.000km/sek.
- Hvilken kommunikationshastighed (bit/sek) kan jeg forvente mellem afsender og modtager ?
- Forslag til hvordan kan kommunikationshastighed gøres bedre ?

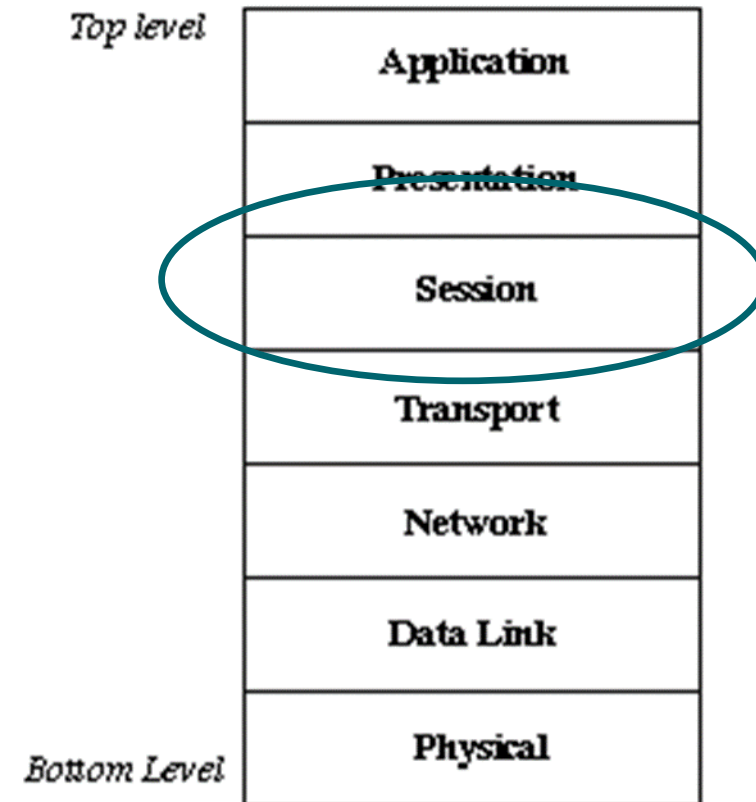
Eksempel på udbyder af satellit baseret Internet: <http://gea-sat.com/>

Andre funktioner i lag 4

- Ordning af pakker i korrekt rækkefølge
- Congestion control (flaskehalse i kommunikation skal ikke oversvømmes)
- UDP ("boilerplate" sender og modtager that is it 😊, men man kan selv bygge mere på)

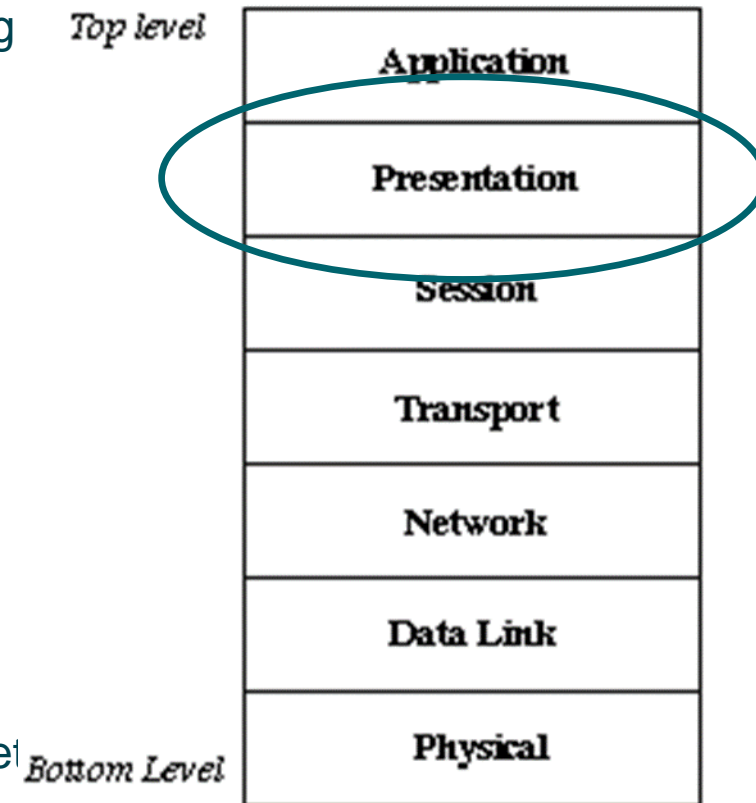
ISO's OSI Model Lag 5

- **Lag 5** i OSI-modellen kaldes også for "Session Layer" på engelsk. Dette lag har til opgave at etablere, opretholde og afslutte sessioner mellem applikationer på forskellige enheder i netværket.
- Session Layer bruger protokoller til at styre kommunikationen mellem applikationerne.
- Session Layer kan også etablere og administrere flere sessioner på samme tid mellem forskellige applikationer. Dette giver brugeren mulighed for at arbejde med flere applikationer på samme tid og skifte mellem dem uden at miste data eller forbindelsen.



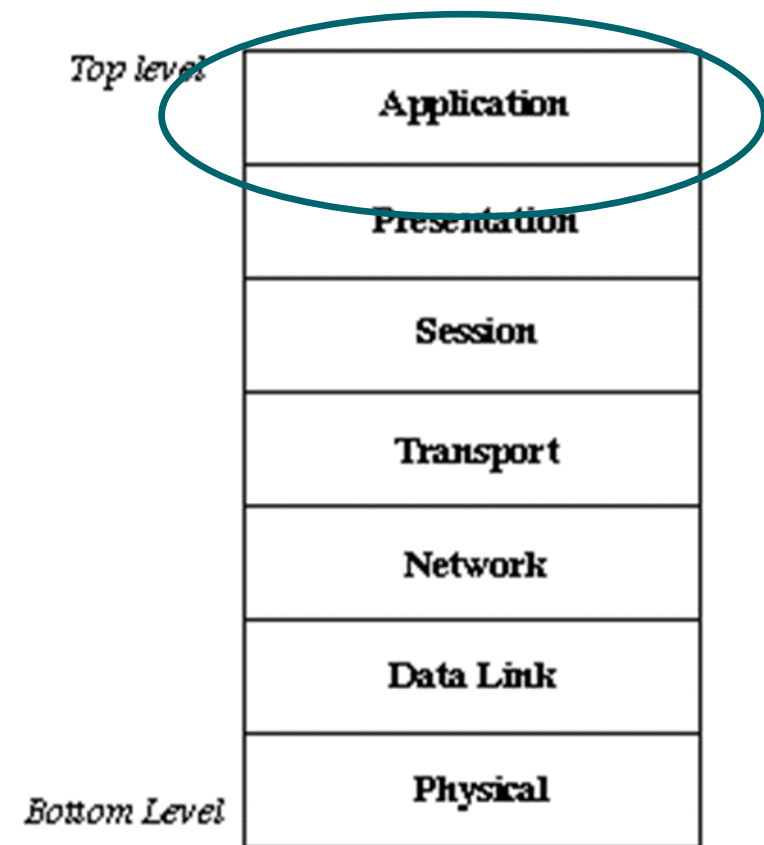
ISO's OSI Model Lag 6

- **Lag 6** i OSI-modellen kaldes også for "Presentation Layer" på engelsk. Dette lag har til opgave at håndtere datarepræsentationen og datakodningen, således at applikationer kan forstå og behandle dataene korrekt.
- Presentation Layer kan konvertere data fra en form til en anden form, f.eks. fra tekst til binær eller fra ASCII-kode til Unicode-kode. Det kan også komprimere dataene, så de kan sendes mere effektivt over netværket.
- Protokoller på Presentation Layer inkluderer f.eks. MIME (Multipurpose Internet Mail Extensions), som er en standard til at udveksle forskellige typer data over internettet, og SSL (Secure Sockets Layer), som er en krypteringsprotokol, der sikrer fortrolighed og integritet af data over internettet.
- Presentation Layer spiller en vigtig rolle i at sikre, at applikationer kan kommunikere på tværs af forskellige platforme og operativsystemer ved at konvertere dataene til et standardformat, der kan forstås af alle applikationer. Det bidrager til interoperabilitet og fleksibilitet i netværkssystemet.



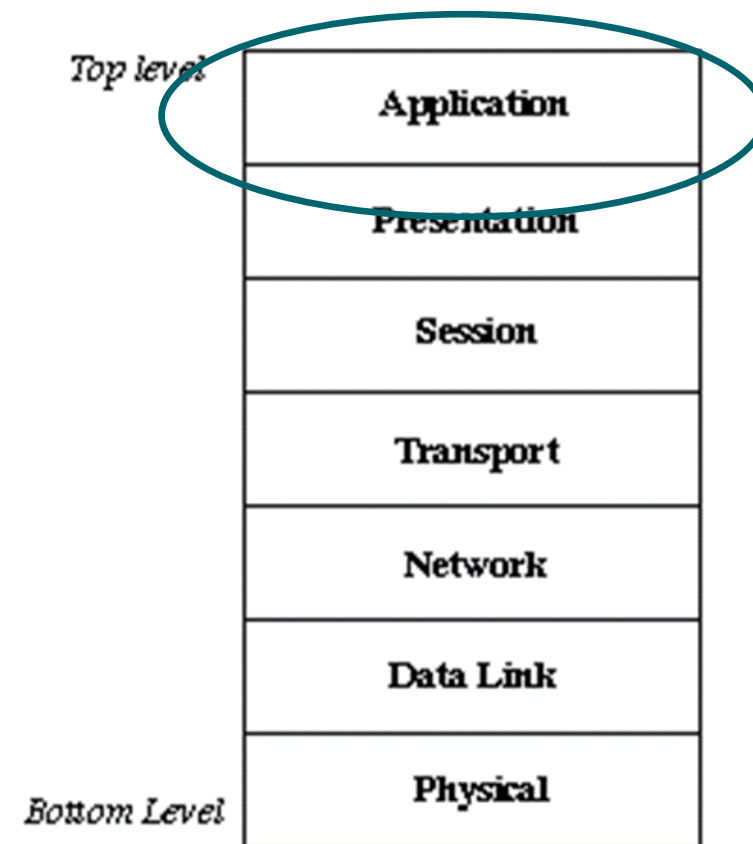
ISO's OSI Model Lag 7

- **Lag 7** i OSI-modellen kaldes også for "Application Layer" på engelsk. Dette lag er det øverste lag i hierarkiet og det lag, hvor applikationerne interagerer direkte med netværket.
- Application Layer indeholder en bred vifte af protokoller og standarder, der understøtter forskellige applikationer og tjenester. Eksempler på protokoller omfatter HTTP (Hypertext Transfer Protocol), FTP (File Transfer Protocol), SMTP (Simple Mail Transfer Protocol) og DNS (Domain Name System).
- HTTP bruges til at overføre websider og andre webressourcer over internettet, FTP bruges til at overføre filer mellem computere i netværket, SMTP bruges til at sende og modtage e-mails, mens DNS bruges til at oversætte domænenavne til IP-adresser, således at computere kan finde hinanden på internettet.



ISO's OSI Model Lag 7

- Application Layer-prokollerne er designet til at være nemme at bruge for applikationerne og kan fungere på tværs af forskellige operativsystemer og platforme.
- Application Layer spiller en afgørende rolle i at sikre, at applikationer og tjenester kan kommunikere på tværs af forskellige netværk og operativsystemer, og at de kan gøre det på en sikker og pålidelig måde. Det er det lag, der giver brugerne mulighed for at kommunikere med og bruge forskellige tjenester og applikationer i netværket.



Lagdeling, encapsulation og standardisering, gør det "let" at skrive applikationer (Server)

```
host = "192.168.2.3"           # Set the server address to your PC's IP address
port = 7913                    # Sets port to 7913 ( > 1023)
from socket import *          # Imports socket module
s = socket(AF_INET, SOCK_STREAM)
s.bind((host, port))          # Binds the socket. Note that the input to
s.listen(1)                   # Sets socket to listening state (simple chat app, only
1 connection !!)
print("Listening for connections.. ")
q, addr = s.accept()          # Accepts incoming request from client print("Client
from {0}".format(addr))      # Client address
msg = q.recv(1024)            # Receives upto 1024 bytes from client and stores it in
variables msg
print("Revieved from client : "+ msg.strip().decode('ascii'))
data = "Hello Client !!"      # Data to be send is stored in variable data from user
q.send(data.encode('utf-8'))  # Encode data to utf-8 and sends data to client
s.close()
```

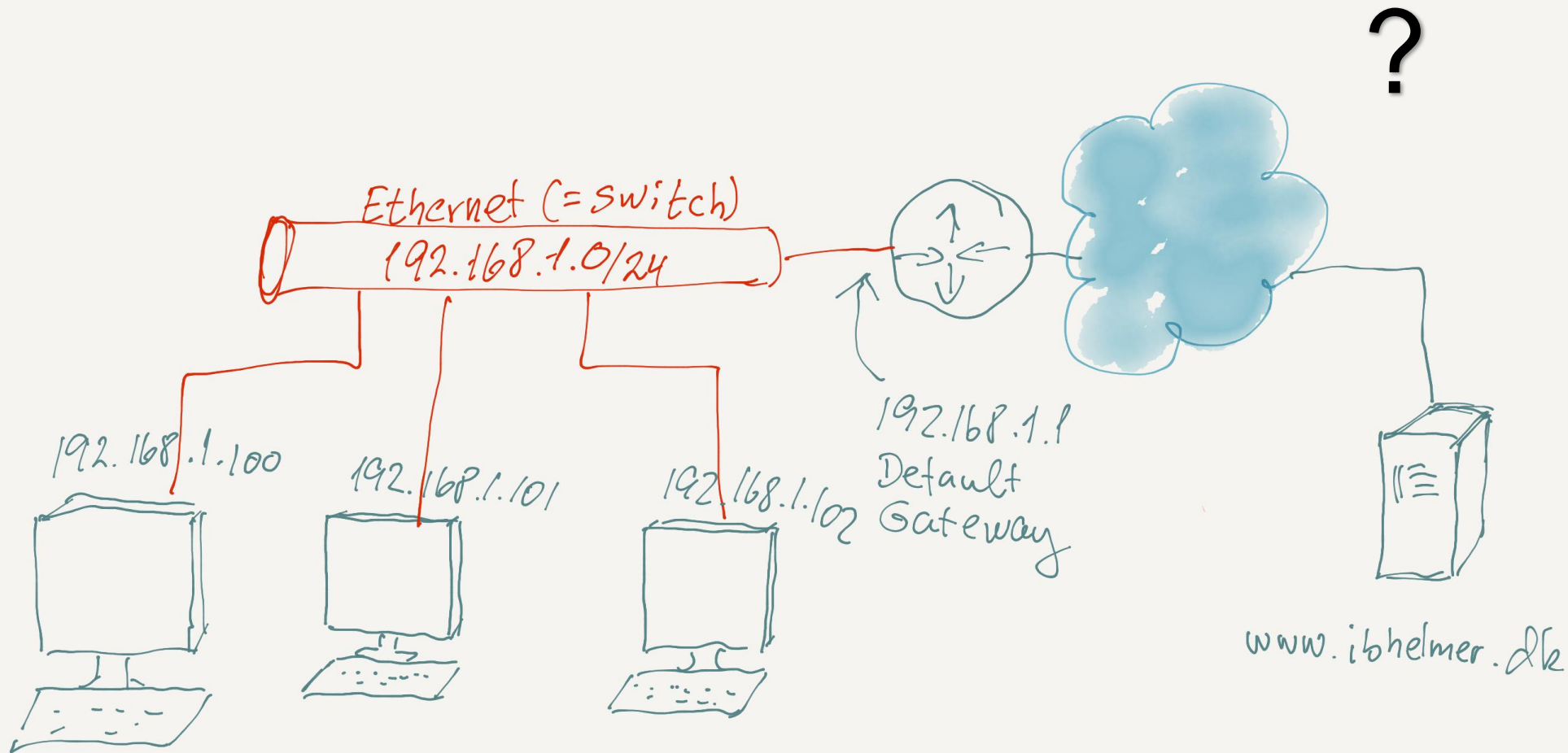
Lagdeling, encapsulation og standardisering, gør det "let" at skrive applikationer

Client:

```
host = "www.ibhelmer.dk"          # Set to address of server
port = 7913                       # Set to port used by server
from socket import *             # Imports socket module
s = socket(AF_INET, SOCK_STREAM)  # Creates a socket
print("Connection to server...")
s.connect((host, port))          # Connect to server address
print("Send msg Hello server!!")
data = "Hello server!!"
s.send(data.encode('utf-8'))      # Encode data to utf-8 and sends data to server
msg = s.recv(1024)               # Receives upto 1024 bytes from server and
stores it in variables msg
print("Back from server : " + msg.strip().decode('ascii'))
s.close()
```

Vigtige applikationer og protokoller

- DNS (Domain Name System) “Oversætter” mellem host adresser og ip adresser.
- http/https (hyper text transfere protocol) Protokol der b.la anvendes ved overførsel af html og anden data mellem web browser og web server.
- DHCP (Dynamic Host Configuration Protocol)



Jeg vil tilgå web-serveren

www.ibhelmer.dk

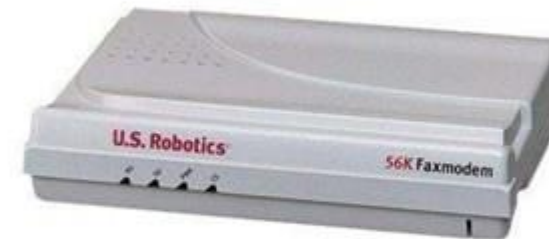
fra PC med ip
192.168.1.100
hvordan foregår
det ?

Geolocation (virker måske på din IPv4 adresse 😊)

- <https://www.iplocation.net/>
- <http://www.infosniper.net/>
- <https://www.ipfingerprints.com/>
- <http://www.ipvoid.com/ip-to-google-maps/>

Netværkskomponenter

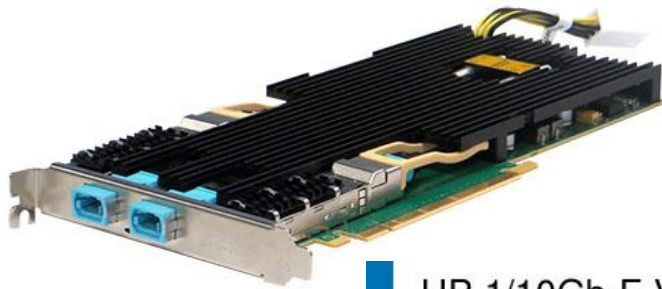
Modem (Modulator/DEModulator)



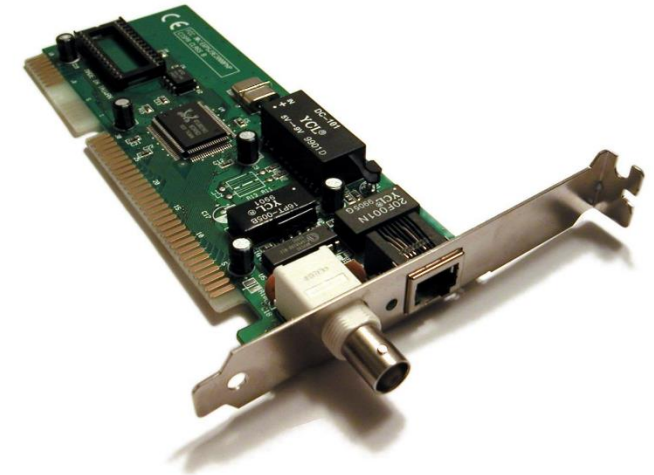
NIC (Network Interface Controller)



- The NIC is what enables a network node to communicate over a computer network



Network Card



HP 1/10Gb-F Virtual Connect Ethernet Module

MidPlane 16x 1Gb Ethernet – Connects to one NIC in each HH blade server bay
1x 10Gb – Cross-link between adjacent VC-Enet modules
Management Interfaces to Onboard Administrator



1x 10Gb Ethernet (CX4)

Usable as data center links or as stacking links.

2x 10Gb Ethernet (XFP)

Support for copper or fiber 10Gb connectivity to the data center

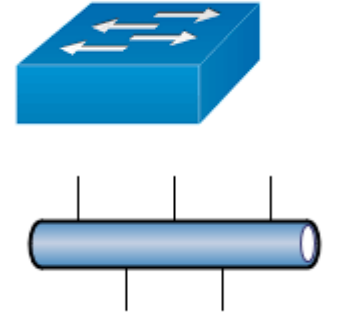
2x 1Gb Ethernet (SFP)

Support for copper or fiber 1Gb connectivity to the data center

4x 1Gb Ethernet (RJ45)



Switch (Layer 2)



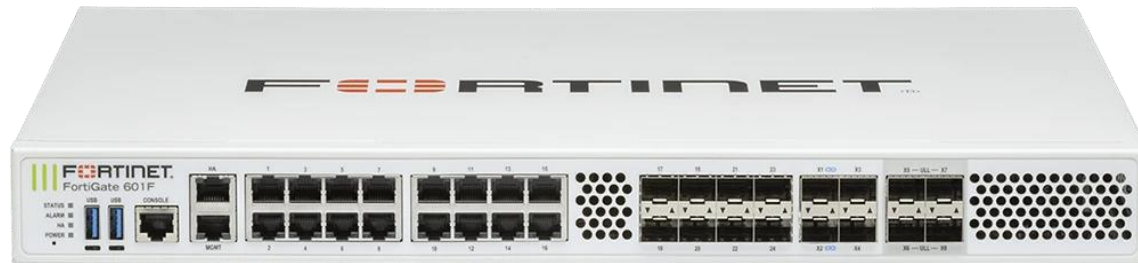
Access point



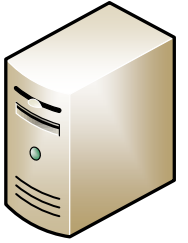
Router



Firewall

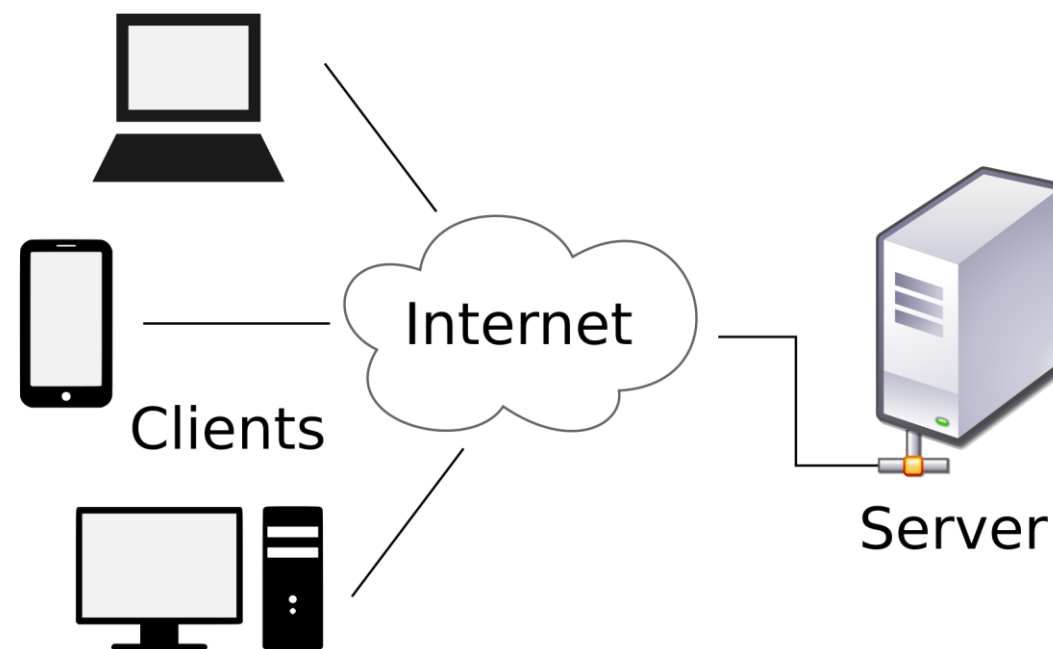


Servere (Fysiske servere !!)

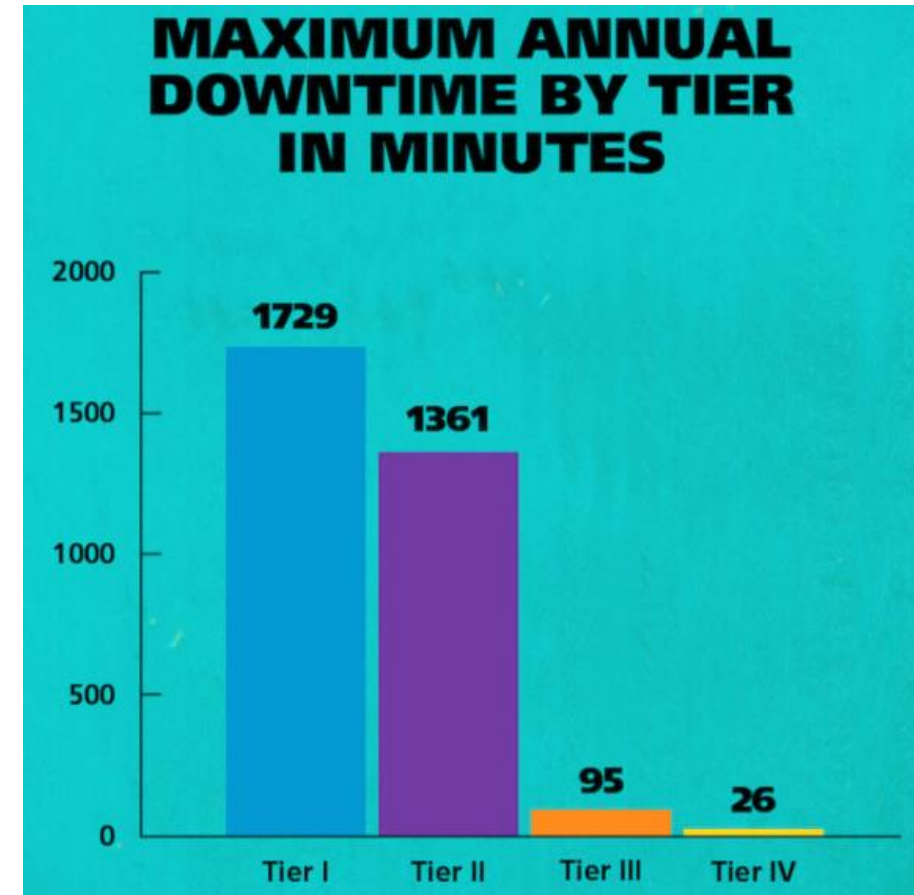
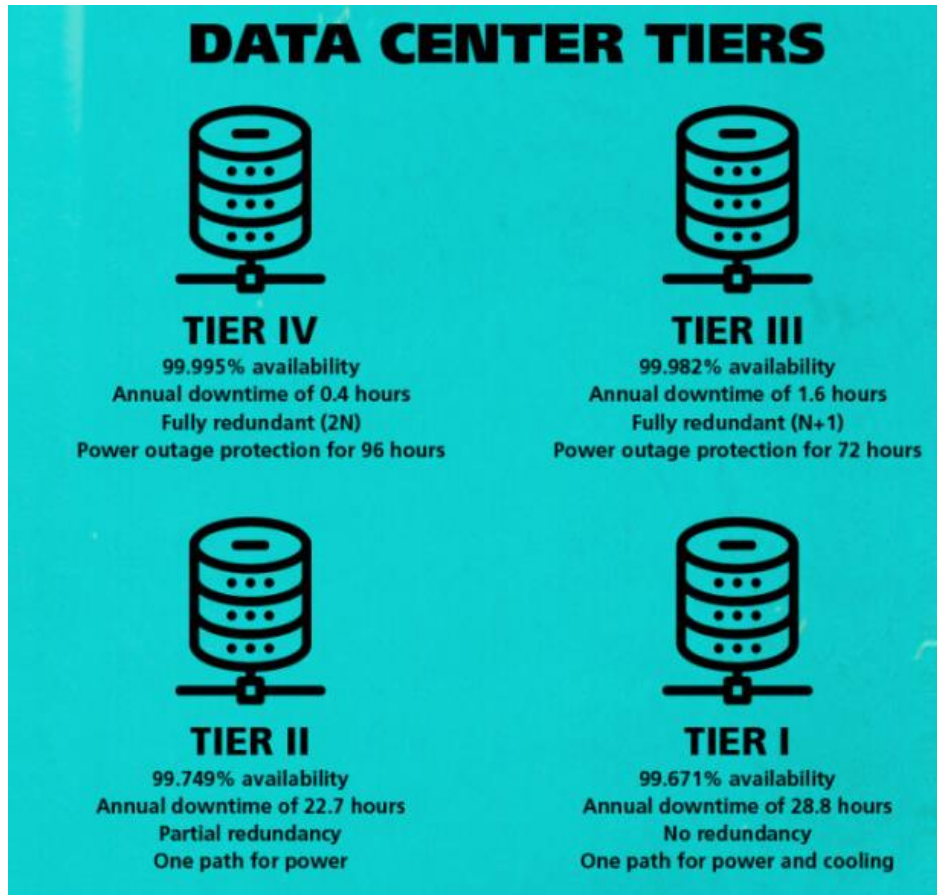


Client/Server paradigm

- Client-server-paradigmet er en model, hvor klienter anmoder om tjenester eller data fra servere via et netværk. Klienten er brugerens enhed, der sender anmodninger, og serveren er en dedikeret enhed, der leverer de ønskede tjenester eller data til klienten. Paradigmet muliggør opdeling af ansvar og fleksibel skalérbarhed.



Fysiske server er computer med særlig god pålidelighed



Region Nordjyllands datacenter er første hospitalsorganisation i verden til at få eftertragtet certificering

Region Nordjyllands datacenter er blevet Tier IV certificeret af det globalt anerkendte Uptime Institute. Certificeringen er bevis på høj forsyningssikkerhed, hvilket kommer patienter til gavn i form af en stabil hospitalsdrift og sikring af patienternes data.



KONTAKT

Klaus Larsen

DIREKTØR, DIGITALISERING OG IT
Digitalisering og IT

Telefon

97 64 97 00

Mail

kl@rn.dk

Michael Lundsgaard Sørensen

KONTORCHEF
Sikker og Stabil Drift

Telefon

22 64 06 16

Mail

mls@rn.dk

FORFATTER

Carsten Randers
PRESSEKONSULENT
Regionssekretariat

Telefon

29 26 02 81

Mail

c.randers@rn.dk



Hvilke lag i ISO's OSI model virker disse netværkskomponenter på ?

- Modem
- Switch
- Router
- Firewall
- Server