Schichten Modell

Physical-Layer

Bitübertragungsschicht: versendet und empfängt einzelne Bits. Übertragung durch Kabel, Stecker, spezielle Codierung, ...

Data-Link-Layer

- Ver- und entpackt Datenblöcke aus der Physical Layer (Framing)
- Fehlererkennung und Korrektur
- Flow-Controll: stellt sicher, dass Empfänger oder Sender nicht überlastet werden.
- Adressierung und Accesskoordinierung

Network-Layer

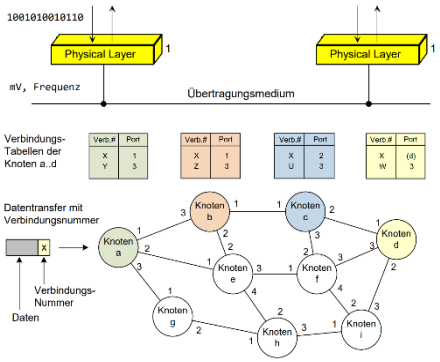
- Vermittlungsschicht: erstellt Verbindungstabelle der einzelnen Knoten
- Paketvermittlung und Leitungsvermittlung

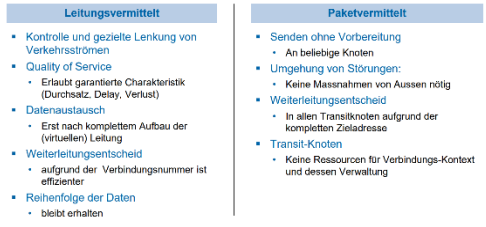
Transport-Layer

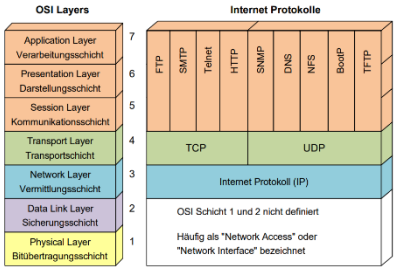
- User Data Protocol (UDP): verbindungslos und unzuverlässig, send & forget, keine Infos über Empfänger, einfach umzusetzen
- Transmission Control Protocol (TCP): verbindungsorientiert und zuverlässig, Infos über Empfänger, Kommunikationsphasen, ...

Kritik

- zu komplex, teuer und ineffizient







Übertragungsmedien

Dämpfung

- Je grösser die Bandbreite (Hz), desto grösser die Bitrate (Bit/sek)
- Je kleiner die Bitrate (bei gleich bleibender Dämpfung), desto grössere Distanzen können zurückgelegt werden

Die Dämpfung wird in **dB (Dezibel)** angegeben.

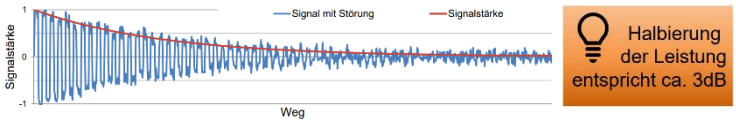
Dämpfung $A = 10 * \log (P_1 / P_2)$

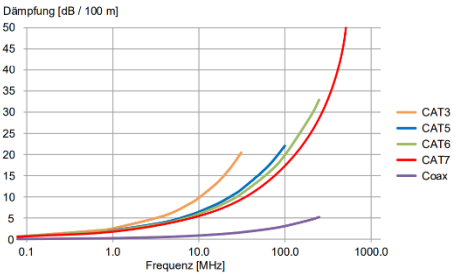
Mit $P_1/P_2 = (U_1/U_2)^2$ ergibt das $A = 10 * \log((U_1/U_2)^2) = 20 * \log (U_1/U_2)$

Obiges Beispiel:

$U_1 / U_2 = 1 / 0.5 = 2$ entspricht einer Leistungsabnahme um den Faktor 4: $20 * \log (2) \approx 6$ dB

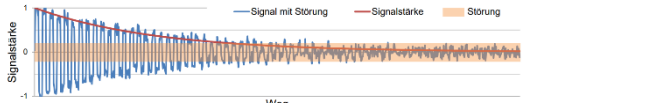
- Je grösser die Frequenz, desto höher die Dämpfung

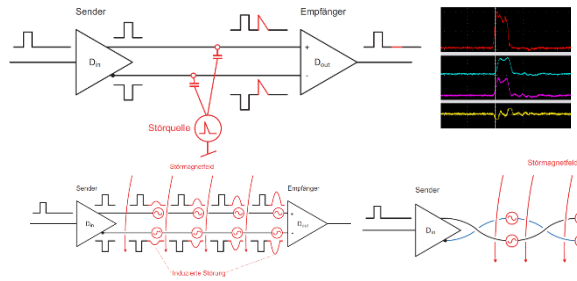




Störung/Rauschen

- Signal-Noise-Ratio (SNR): $SNR = 10 * \log (P_{Signal} / P_{Störung})$ dB
- Störungsbehebung durch komplementäres Signal
- Störungsbehebung durch Erdung
- Induktiv eingekoppelte Störungen: lassen sich nicht durch Erdung oder Abschirmung verhindern → verdrehte Kabel





Kabel

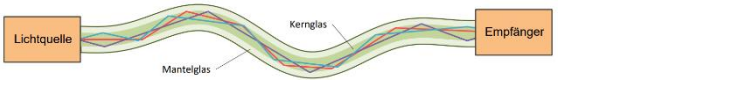
- Koaxialkabel: hochfrequente Signale, unempfindlich gegenüber elektromagnetische Störungen
- Paarsymmetrische Kabel (Twisted Pair/TP): auch für breitbandige Übertragung
- Shielded Twisted Pair besser gegen Störung geschützt

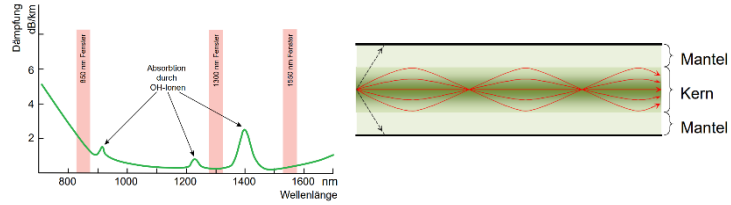
Lichtwellenleiter

- Hohe Bandbreite, geringe Dämpfung, resistent
- Zentrum aus Kernglas mit hoher optischer Dichte, Mantel geringere Optische dichte. Das Signal ist im Kernglas und wird von der Innenseite total reflektiert

$$\alpha > \sin^{-1}\left(\frac{n_{Luft}}{n_{Wasser}}\right) = 48.75°$$

- Totalreflexion:
- Multimode-Glasfaser: dicker Kern, kleine Datenrate und Weg
- Multimode-Gradientenfaser: dichte des Kernglases nimmt zum Mantel hin ab (macht quasi einen Bogen zum Kern)
- Singlemode-Glasfaser: sehr dünner Kern, ein Lichtstrahl hohe Datenrate und lange Strecken, sehr teuer





Physical-Layer

Serielle asynchrone Übertragung

- Das Least Significant Bit (ganz rechts) kommt als erstes, das Most Significant Bit als letztes (ganz links)
- Bevor jedoch die Daten kommen, kommt zu erst das Startbit, eine 0. Die letzten zwei Bits nach den Datenbits ist das Paritybit (1 oder 0, sorgt dafür, dass die Anzahl 1en gerade oder ungerade ist, je nach Codierung) und das Stoppbit (eine 1).
- Damit der «Takt» der Bits stimmt, kommt ein zweites Signal, das Taktsignal (oder auch Clocksignal). Immer, wenn das Signal von 1 auf 0 wechselt, wird das Datensignal gelesen.

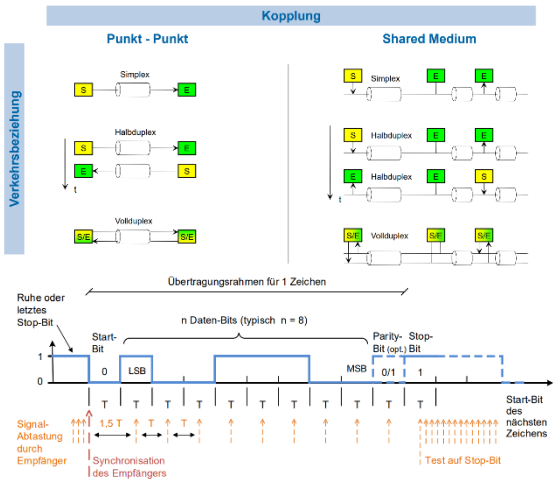
Leitungscodes

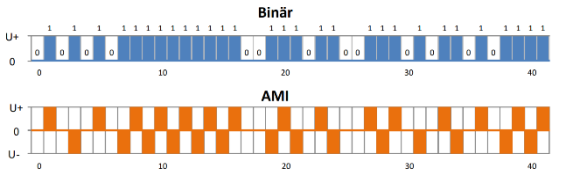
- Muss möglichst gelichspannungsfrei sein, um Sender und Empfänger mit Übertragern galvanisch trennen zu können.
- Bsp: AMI Codierung → Eine 1 wird entweder als positive (+V) oder negative (-V) dargestellt. Jede 1 wird als Polarität der vorherigen 1 dargestellt, also die umgekehrte Spannung. Eine 0 wird als Nullspannung (0V) dargestellt. So hebt sich die positiven und negativen Spannungen im Durchschnitt auf und der Gleichstromanteil ist eliminiert.

Nachteil: Bei einer längeren Folge von 0en gibt es keinen Spannungswechsel, was die Synchronisation deutlich erschwert.

Kanalkapazität und Bandbreite

- Einflussfaktoren sind: Art der Leitungscodierung, Bandbreite der Datenleitung, Stöungen/Rauschen
- Nutzung der Bandbreite: Die maximale Baudrate (Symbol/Sekunde) ist gleich die doppelte Bandbreite (in Hz); gilt für Ideale Übertragungen ohne Störung.
- Maximale Bit-Rate (Hartley-Gesetz) ist gleich die Bandbreite des Kanals (Hz) multipliziert mit dem Zweier-Logarithmus von 1 plus dem Signal-Noise-Ratio.





$$f_s = 2B$$

$$C = B \log_2(1 + \frac{S}{N})$$

- Bandbreite B – Einheit Hertz (Hz)
  - Eigenschaft des Übertragungskanals und durch das Medium begrenzt
- Symbolrate f_s – Einheit Baud (Bd)
  - Anzahl der Symbole pro Zeit. Limitiert durch die Bandbreite ($\leq 2B$) (Nyquist)
- Bitrate R – Einheit Bit/s (bps)
  - Produkt von Symbolrate und mittlerem Informationsgehalt der Symbole (Hartley)
- Kanalkapazität C – Einheit Bit/s (bps)
  - Berücksichtigt für einen realen Kanal das Signal-zu-Rausch Leistungsverhältnis S/N (Shannon)

Data-Link-Layer

Framing

- **Asynchron**: Besteht meistens aus einem **Header** bestehend aus Informationen über die gesendeten Daten, die **Daten** selbst, und einem **Fehlererkennungsblock** am Ende.
- **Synchron**: Frames werden ohne Unterbruch gesendet. Wenn keine Daten gesendet werden, werden Flags gesendet.
- **Bitstopfen**: Für Start- Endflags wird ein bestimmtes Bitmuster verwendet. Um zu verhindern, dass dieses Muster auch in den Daten vorkommen, wird die Bitfolge abgeändert. (siehe Bild)

Fehlerwahrscheinlichkeit

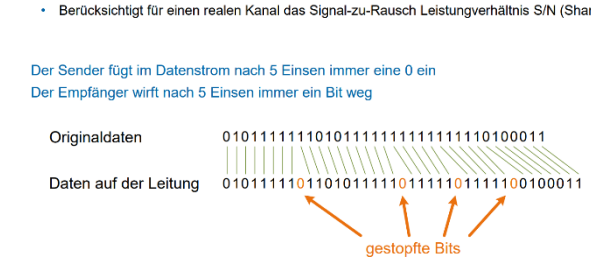
- Bitfehlerwahrscheinlichkeit (Bit-Error-Ratio: BER)
- Je länger der Frame (inkl. Header), desto grösser die Chance, dass ein Bitfehler auftritt. Lange Frames sind gut für die Netobitrate

Fehlererkennung

- Es kommt der CRC-32 zum Einsatz; Minimale Hamming-Distanz bei 12000 Bits ist 4, somit können 3 Fehler erkannt werden
- Längs- und Querparity
- Prüfsumme

Fehlerkorrektur

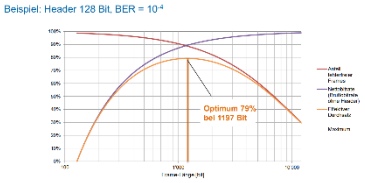
- **Backward Error Correction**: Für jede erhaltene Nachricht sendet der Empfänger eine Quittung an den Sender. Falls der Sender keine Quittung erhält, sendet er die Nachricht nochmals.
- **Forward Error Correction**: Empfänger korrigiert die empfangene Nachricht selbst zur der am wahrscheinlichsten gesendete Nachricht.

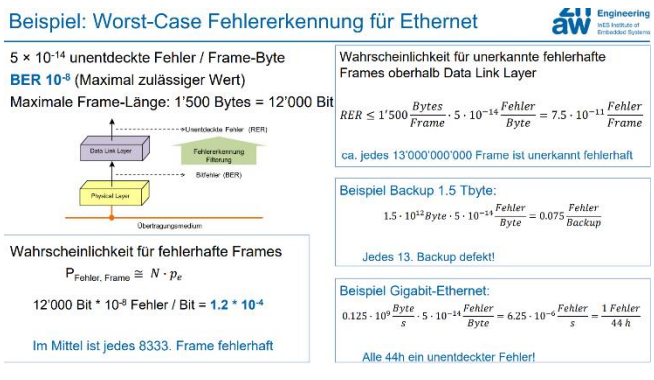


Die Betrachtung der Erfolgswahrscheinlichkeit ist einfacher als die der Fehlerwahrscheinlichkeit:

- Um N Bit fehlerfrei zu empfangen, muss jedes einzelne Bit fehlerfrei empfangen werden
- Erfolgswahrscheinlichkeit für 1 Bit: $P_{Erfolg} = (1 - p_e)$
- Erfolgswahrscheinlichkeit für den ganzen Frame (N Bit): $P_{Erfolg,\ Frame} = (1 - p_e)^N$
- Fehler-wahrscheinlichkeit für den ganzen Frame: $P_{Fehler,\ Frame} = 1 - (1 - p_e)^N$

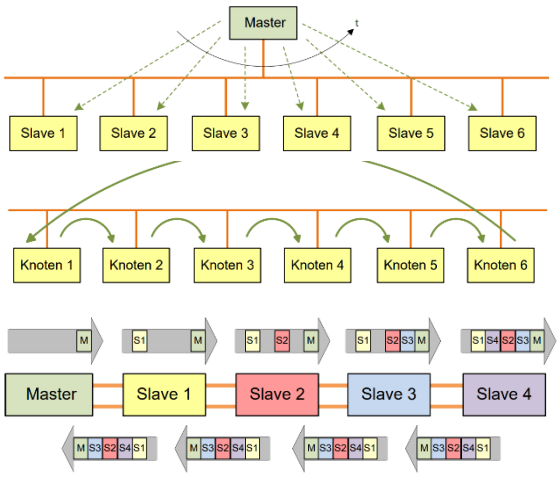
Für $p_e \ll 1$ gilt folgende Näherung: $(1 - p_e)^N \cong (1 - N \cdot p_e)$, also: $P_{Fehler,\ Frame} \cong N \cdot p_e$ (=FER)





Zugriffsmechanismen auf das Medium

- **Master-Slave Verfahren**: Master kontrolliert alle Zugriffe auf das Medium, so entstehen keine Konflikte. Wenn Master ausfällt, dann alles futsch.
- **Token Verfahren**: Knoten senden nur, wenn sie einen Token haben, sonst werden sie gesperrt. Reihenfolge ist fix. Aufwändige Implementation.
- **Token Verfahren Variante**: Anstelle eines Tokens werden Frames versendet. Die einzelnen Knoten fügen dann ihre Daten an vorbestimmten Stellen im Frame ein und schicken es zum nächsten Knoten. Hat der letzte Knoten seine Daten eingefügt, bekommt der Master das Frame zurück.
- **Zeitgesteuerter Zugriff**: Optimierung möglich, jedoch kann ungeplanter Verkehr zu Konflikten führen.
- **Carrier Sense Multiple Access**: Jeder Knoten ist gleichberechtigt und hat Zugang zum Übertragungsmedium. Bevor ein Knoten sendet, hört es die Leitung ab und stellt sicher, dass gesendet werden kann, sonst wird gewartet.

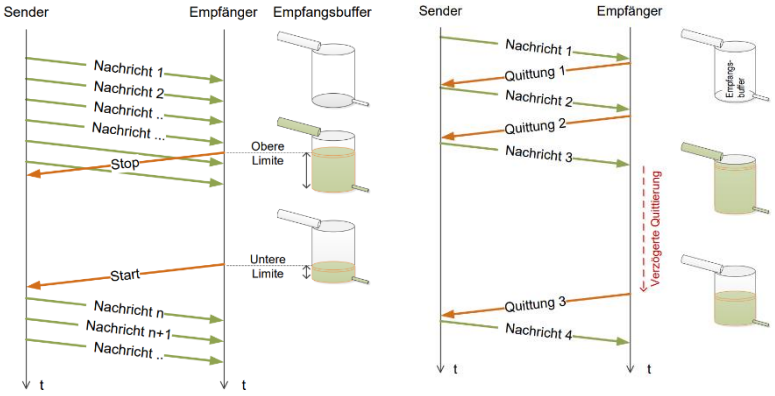


WLAN

- **WLAN** kommuniziert über ein geteiltes Medium und muss daher mit Kollisionen umgehen können. Ein Sender kann eine Kollision nicht selbst erkennen. Daher ist eine Kollision nur über das Ausbleiben einer Quittung (ACK/Acknowledgement) erkennbar.
- **Request-to-Send; Clear-to-Send**: (Siehe Bild)

RTS/CTS Mechanismus:

1. Sender verschickt «Request-to-Send» mit geplanter Sendedauer
2. Designierter Empfänger schickt «Clear-to-Send» mit geplanter Sendedauer
3. Eigentlicher Versand der Daten



Flow Control

- **Explizit**: Der Sender sendet so lange, bis er Empfänger ein Stop-Signal sendet. Wenn der Empfänger wieder bereit ist, neue Daten zu empfangen (zB. wenn Buffer wieder leer ist), sendet er ein Start-Signal.
- **Implizit**: Der Sender darf nur senden, wenn er nach jeder Nachricht eine Quittung erhalten hat.

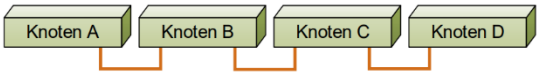
Ethernet

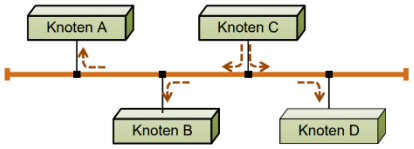
Lokale Netzwerke

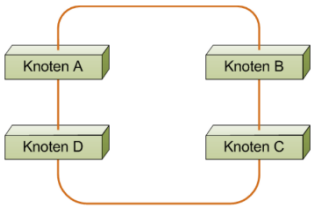
- Local Area Network (LAN): Reichweite wenige Kilometer, 1 Gbit/s. Für die Verbindung von PCs, Drucker, Server, ...

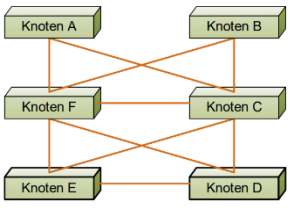
Topologie

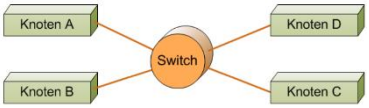
- **Linie**: Jeder Knoten ist mit dem nächsten Knoten (in beide Richtungen) verbunden. Fällt ein Knoten aus, ist das LAN zweigeteilt.
- **Bus**: Alle Knoten sind miteinander verbunden. Empfänger erkennt anhand Adresse, ob die Daten für ihn sind.
- **Ring**: Wie die Linie, aber der letzte Knoten ist mit dem ersten verbunden (und bildet somit ein Ring). Wenn eine Station ausfällt, ist immer noch jede erreichbar.
- **Vermascht**: Hohe Redundanz, somit stabil gegen Ausfälle. Grosser Aufwand.
- **Stern**: Ein zentraler Switch im Zentrum, der alle Knoten miteinander verbindet. Regelt den Datenverkehr und macht das LAN somit weniger Störungsanfällig.
- **Baum**: Erweiterung der Stern-Topologie. Einzelne Zweige sind unabhängig.

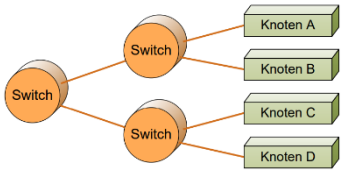




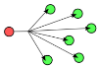






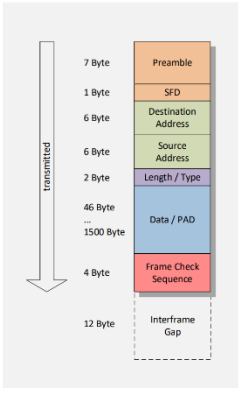


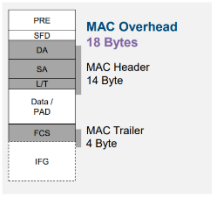
Übertragungsarten

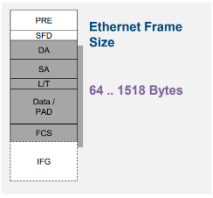
Unicast:  Multicast:  Broadcast: 

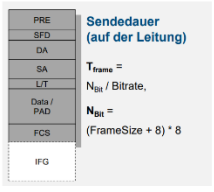
Frame Format

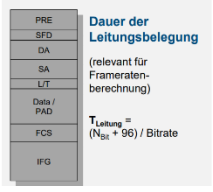
- Das Least Significant Bit (LSB) wird zuerst, das Most Significant Bit (MSB) zuletzt übertragen.
- Der Datenblock muss mind. 46 Bytes lang sein. Sind die Daten kürzer, wird der Block mit Padding (0en) aufgefüllt.





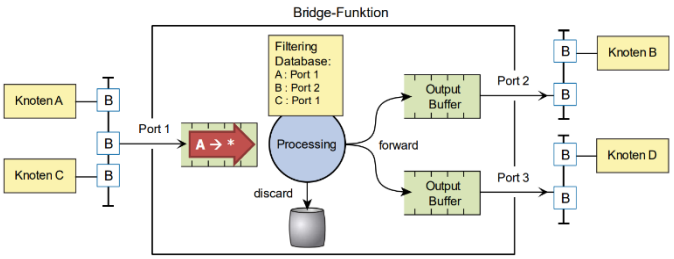






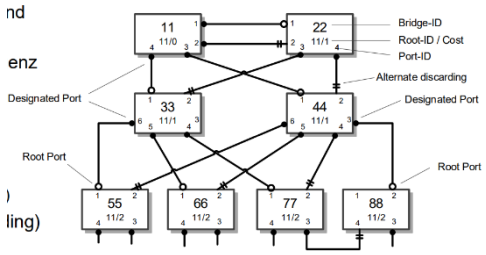
Switches und Bridges

- Arbeitet auf der Schicht 2 (Data Link Layer)
- Switches sind für Endgeräte unsichtbar. Das bedeutet, dass die Switches wissen müssen, welcher Port zu welcher MAC Adresse gehört → **Filtering Database**; eine Tabelle, in der alle Zieladressen und die dazugehörigen Ports aufgelistet sind.



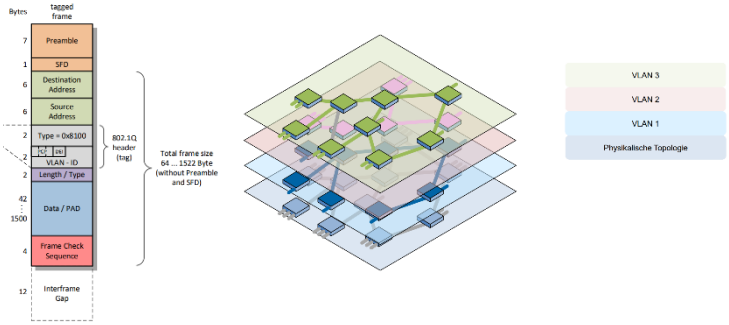
Spanning Tree

- Ein LAN darf ohne weitere Massnahmen keine Loops enthalten. Darum müssen alle (Swicht-) Segmente loopfrei sein.
- Algorithmus in drei Schritten: 1. Root auswählen (meistens kleinste ID) 2. Vom Root aus alle Nachbarn verbinden 3. Schritt wiederholen. Ein Router darf nur verbunden werden, wenn er noch nicht Teil vom Netzwerk ist. Am Schluss werden alle übrigen Verbindungen gesperrt.



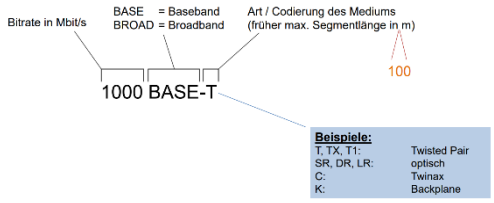
VLAN

- Im gleichen Netzwerk kann es mehrere LANs haben, die untereinander nicht kommunizieren dürfen und somit nach aussen als unabhängiges Netzwerk erscheint.
- Der Ethernet Header wird um einen **VLAN-Tag** erweitert, der Informationen über das zugehörige Netzwerk enthält.
- Der Frame wird beim Ein- resp. Austritt ins Neztwerk getagged. Ist für Endgeräte unsichtbar.
- Je nach Tag kann der Frame unterschiedliche Prio haben.



Twisted Pair Varianten

- Aufbau: siehe Bild
- **Auto-Negotiation**: Wie wird die effektive verwendete Konfiguration zwischen zwei Stationen vereinbart? Mittels **Fast-Link-Pulses** (FLP)
- FLP: Beim Vernindungsaufbau sendet das Gerät eine Seuquenz von Signalen an das Endgerät, um ihre Fähigkeiten auszuhandeln.
- **Auto-Polarity**: Polarität der Signale werden automatisch erkannt und korrigiert.
- Auto-Crossover: Erkennt, ob ein Straight-Through-Kabel oder ein Crossoverkabel verwendet wird.

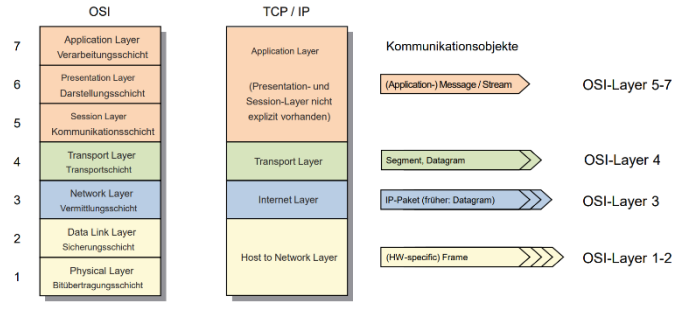


100BASE-TX

Internet Protokolle

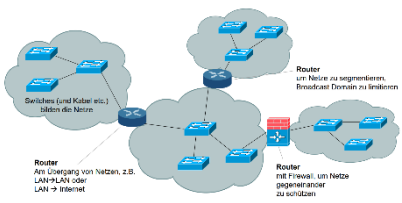
Network Layer

- Das gesamte Internet gehört zum Network Layer, bestehend aus Endgeräten und Routern.
- Die Basis bildet ein verbindungsloser Network Layer
- Kümmert sich ausschliesslich um den Transport der Pakete
- Es gibt keine zentrale Funktionssteuerung



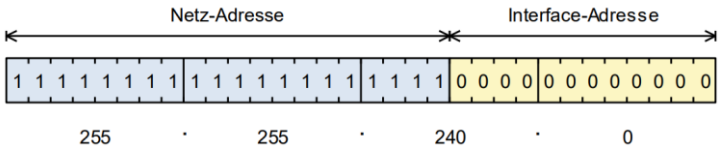
Router

- Ein Router verbindet verschiedene Netze. Dabei muss die Verwendung von unterschiedliechen Technologien in den Netzen berücksichtigt werden.
- Die Hauptaufgabe von Routern ist das **Routing** (die Wegfindung) und das **Forwarding** (das Weiterschicken der Pakete)



IPv4

- Besteht aus Netz-Adresse + Interface-Adresse. Die Subnetzmaske bestimmt die Grenze der Netz-Bits und den Interface-Bits.
- **Netzadresse**: Ist reserviert und darf nicht für Interfaces verwendet werden. Ist die tiefste Adresse im Subnetz, da alle Interface-Bits 0en sind.
- **Broadcastadresse**: Ist die höchste Adresse im Subnetz, da alle Interface-Bits 1en sind.



Routing (und Forwarding)

- **Routing**: Aufbau und Update der Routingtabellen
- **Forwarding**: Weiterleiten von Paketen
- Die **Routing-Tabelle** gibt an, über welche Interfaces die Netze erreichbar sind. Für die Skalierbarkeit ist die Netzadresse wichtig.
- Beim Routing werden nur die Netzadressen verglichen und dann das erst beste genommen.
- In einem **flachen Routing** kennt der Router jeder eyplizite Weg zum Ziel -> sehr grosse Routing Tabellen, Pakete mit unbekannten Adressen werden verworfen.
- In einem **hierarchischen Routing** kennt nur jeder nächste Router der nächste Schritt -> kleine Routing Tabellen mit eine Default Eintrag, das Paket wird immer weitergereicht.

Festlegung der Adresse

- **Klassen** (Dino): Die ersten Bits (das Prefix) bestimmt die Klasse -> Klasse A: 0000 (Netz 8 Bits), Klasse B: 1010 (Netz 16 Bits), Klasse C: 1100 (Netz 24 Bits), Broadcast: 1110, Klasse E: 1111
- **Subnetz**: Das Netz kann von vielen Unternetzen zusammengefügt werden. So kann zum Beispiel jeder Router sein eigenes Subnetz haben.
- Wie können wir ein Netz der Klasse B (160.85.0.0) in 8 Subnetze aufteilen? -> siehe Bild

Spezielle Adressen

- **localhost**: Das gesamte A-Netz (127.0.0.0) ist reserviert
- 192.168.0.0 - 192.168.255.0 (Klasse C) ist zum Beispiel ein privater Adressbereich und wird im Internet nicht weitergeleitet.

IPv4 im Detail

Der Header besteht aus:

- **Version**: 4 oder 6
- **Internet Header Length** (IHL): Länge des Headers
- **Differentiated Services**: Priorisierung, Verkehrsklassen
- **Total Length**: Header + Daten, bis zu 65535 Bytes. Wenn das Paket zu gross ist, wird es vom Router fragmentiert.
- **ID Number/Flags/Fragment Offset**: Relevant für fragmentierte Pakete (siehe unten)
- **Time to Live**: Verbleibende Lebenszeit (in Hops) für ein Paket. Verhindert, dass ein Paket in einer Endlosschleife stecken bleibt. Pfad zwischen zwei Hosts meist nie mehr als 25 Hops.
- **Protocol**: Übergeordnetes Protokoll der Nutzdaten TCP, UDP, ...
- **Header Checksum**: 16 Bits, wird in jedem Router neu berechnet. Gilt NUR für den Header.
- **Sender- und Empfänger-Adresse**

Fragmentierung

- Die **Identification Number** erlaubt eindeutige Kennung des ursprünglichen IP Pakets und zusammengehörige Fragmente
- **Flags** kontrollieren Fragmentierung

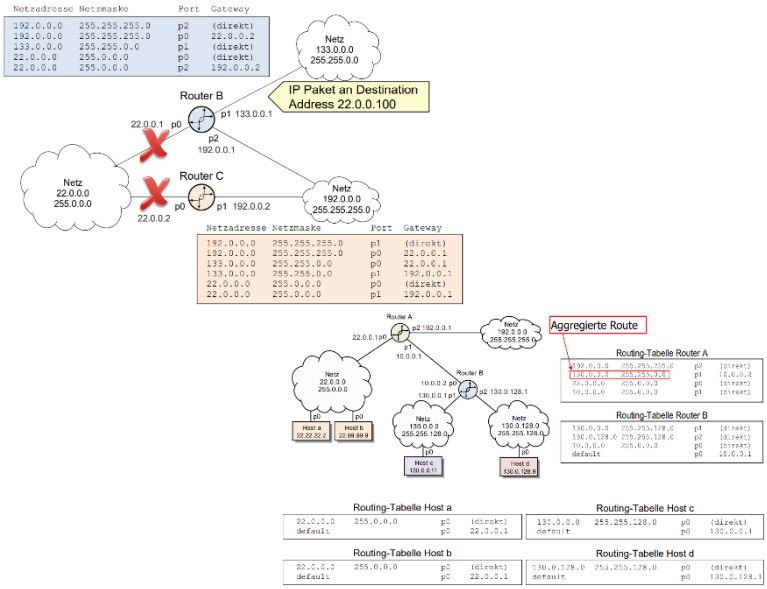
0	1	2
0	DF	MF

- **Fragment Offset**: Gibt die Position des Fragments im gesamten Paket an.
- Fragmentierung geschieht dann, wenn der **Maximum Transfer Unit** (MTU) überschritten wird.
- Fragmentierung findet im SENDER statt. Dafür muss der Sender die MTU über den gesamten Pfad kennen.
- Erst der Empfänger fügt die einzelnen Fragmente wieder zusammen.

Kapselung und Adressauflösung

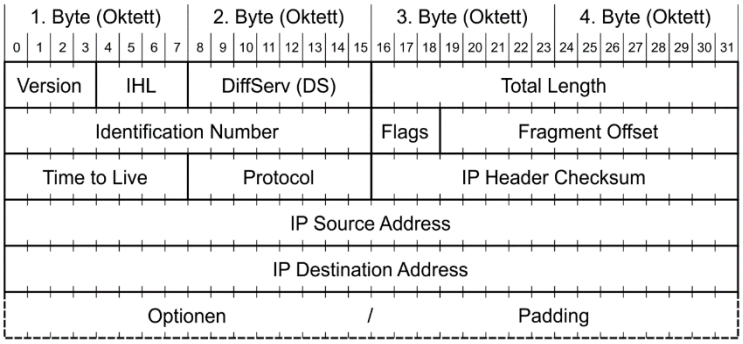
Übertragung eines IP-Pakets mit Kapselung (siehe Bild)
Problem: wie findet **a** die Hardware-Adresse von S heraus?
-> Adressauflösung

- **Adressauflösung**: Ein Knoten muss die Hardwareadresse des Empfängers. Ist diese nicht bekannt, so wird ein Broadcast Signal per **Address Resolution Protocol** (ARP) gesendet.
- Der Request und Response ist ein normaler Ethernet Frame

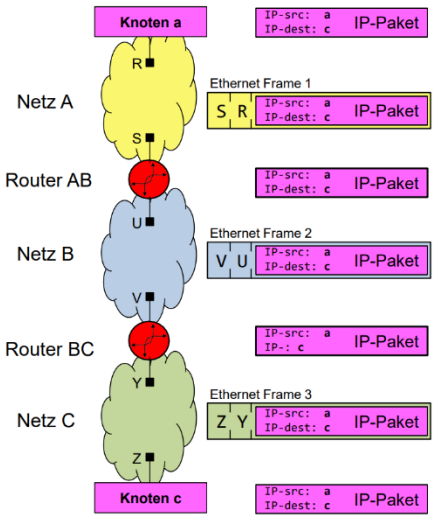


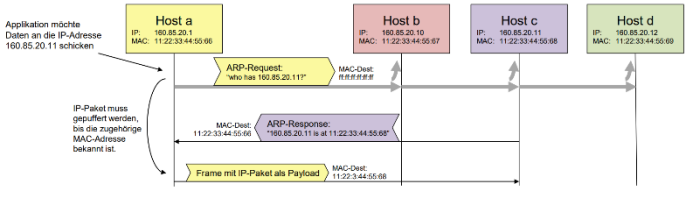
Damit haben wir neu 8 Subnetze mit den folgenden Netzadressen:

- 160.85.0000 0000 0000 0000 = 160.85.0.0
- 160.85.0010 0000 0000 0000 = 160.85.32.0
- 160.85.0100 0000 0000 0000 = 160.85.64.0
- 169.85.0110 0000 0000 0000 = 160.85.96.0
- ...
- 160.85.1110 0000 0000 0000 = 160.85.224.0



Feld	Position	Werte	Funktion
	0	0	Reserved, must be Zero
DF	1	0 / 1	May / Don't Fragment
MF	2	0 / 1	Last / More Fragments





- ARP wird für die Erkennung von **Adresskonflikten** und die **Erneuerung des ARP-Caches** verwendet.

Internet Control Message Protocol (ICMP)

- ICMP ist für die Übertragung von Fehlermeldungen und Informationen verantwortlich.
- Gehört auch zur Kapselung und werden auch in IP Paketen gepackt (gekapselt)
- ICMP ist ein unzuverlässiger Dienst und gehört zum Network-Layer, auch wenn es von aussen icht so aussieht.

Meldungstypen:
- Fehler: Destination Unreachable, Time Exceeded, Bad IP Header
- Optimierung: Redirect
- Information: Echo Request/Reply, Timestamp Request/Reply
- **Destination Unreachable** mit Code 4 (**Fragmentation Needed and DF set**) -> Es muss ein neuer Pfad gefunden werden, der die mindeste MTU unterstützt

Feld	Wert/Semantik
Type	3
Code	`0 = net unreachable, 1 = host unreachable, 2 = protocol unreachable, 3 = port unreachable, 4 = fragmentation needed and DF set, 13 = communication administratively prohibited`
Checksum	Prüfsumme über die ICMP Meldung
IP Header + 64 Bits of Original Datagram	Information für den Empfänger zur Zuordnung der Meldung zu einem gesendeten IP Paket

- **Time Exceeded**: Time to Live wird von 1 auf 0 gesetzt (Code 0) oder Zielhost kann fragmentiertes Paket nicht rechtzeitig zusammensetzen (Code 1).
- Der Linuxbefehl *traceroute* kann der gegangene Weg mit allen IP Adressen nachverfolgen.

IPv6

- Moderne Betriebsysteme und Netzwerk-Knoten sind über IPv4 und über IPv6 erreichbar. Die Router können beides.
-IoT verwendet sehr häufig IPv6
- Adressbereiche werden duch Prefixes gekennzeichnet.
- In der Praxis wird diese Aufteilung verwendet: (siehe Bild)
- Statt ARP gibt es **Neighbor Discovery Protocol (NDP)**.
- Vor dem Senden wird die MTU des gesamten Pfades ermittelt.
- **Header**: Kann mehrere Source und Destination Addresses enthalten. Meist hat es mind. eine **Link Local Adresse** und eine **Global Unicast Adresse**.
- **Extension Headers**: Verfrachtet alle optionalen Erweiterungen in separate Bausteine. Reihenfolge ist festgelegt.

Transport Layer

Transport Layer im Internet

- Schliesst die Lücke zwischen IP und Applikation.
- Ist als End-zu-End Dienst zu verstehen.
- Bildet die **Schnittstelle** zwischen **Betriebsystem** (Kernel) und unprivilegierten **Anwendungen** (User-Space).
- Die Nutzdaten werden vom Transportlayer in ein IP-Paket gekapselt. Das verwendete Protocol wird im Paket im Header angegeben.

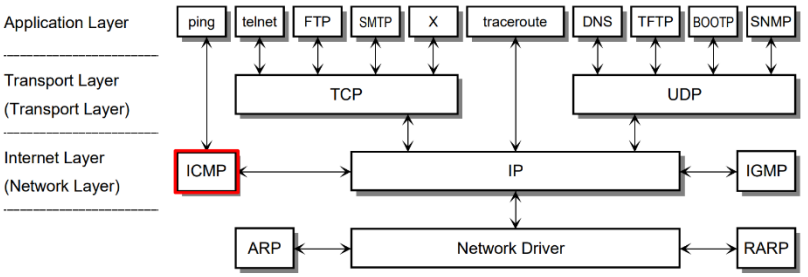
Port Nummern werden in drei Bereiche unterteilt:
- Well-Known Ports: für UDP und TCP reserviert, sollten nicht für andere als die vorhergesehene Anwendungen gebraucht werden.
- Registered Ports: reserviert für spezifische Anwendungen
- Dynamic/Private Ports: Für beliebige Anwendungen

Well-Known und Registered Ports kennen das Betriebsystem
-> Den Port braucht es, damit der Empfänger weiss, welche Applikation gemeint ist.

User Datagram Protocol (UDP)

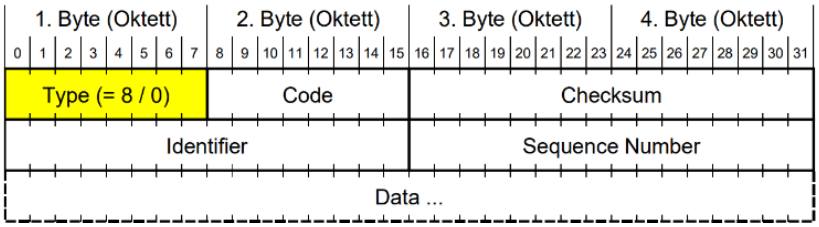
UDP dient dem Multipexing, resp. Demultiplexing der Datagramme einer Applikation, also das Zusammenfügen und Auseinadernehmen von mehreren Datenströmen zu einem Einzelnen.

Verbindungslos: Applikationsdaten werden nur in Datagramme eingefügt und dann gesendet.

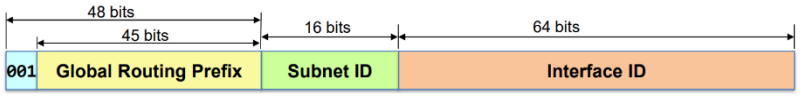


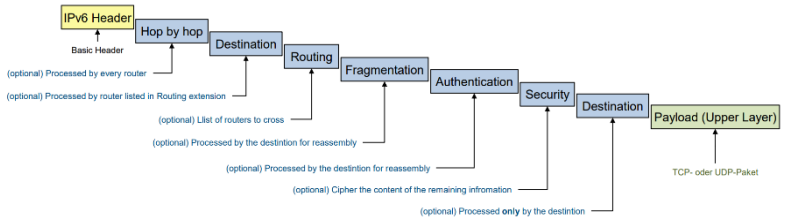
ICMP-Typ	Bedeutung (Fehler)
3	Destination Unreachable
5	Redirect
11	Time Exceeded
12	Parameter Problem: Bad IP Header

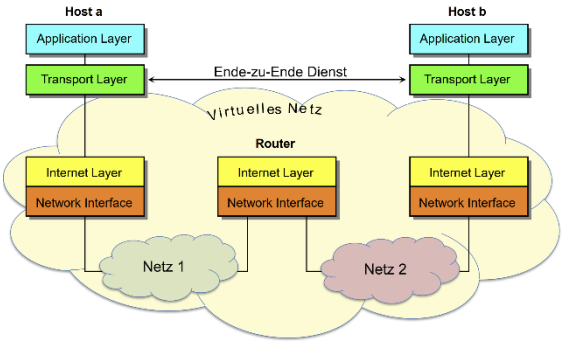
ICMP-Typ	Bedeutung (Information)
0	Echo Reply
8	Echo
13	Timestamp
14	Timestamp Reply

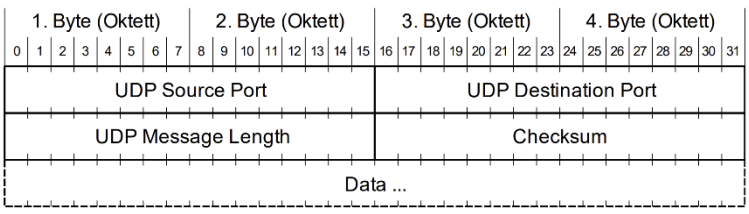


- In Zahlen können links stehenden Nullen weggelassen werden:
  `2001:0620:0000:0004:0A00:20FF:FE9C:7E4A` und
  `2001:620:0:4:A00:20FF:FE9C:7E4A` sind gleichwertig
- **Eine** Nullfolge kann durch 2 Doppelpunkte dargestellt werden
  `1023:0000:0000:0000:1736:a673:88a0:a620` und
  `1023::1736:a673:88a0:a620` sind gleichwertig









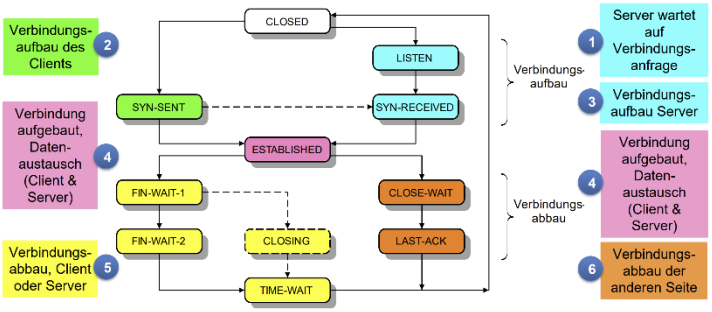
Unzuverlässig: Keine Massnahmen gegen Verlust oder Fehler

Transmission Control Protocol TCP

Verbindungsorientiert: Vor dem Senden wird eine Verbindung von Sender und Empfänger aufgebaut.
Zuverlässig: Verbindung wird von Sender und Empfänger bestätigt. Massnahmen gegen Verlust und Fehler.
Vollduplexübertragung: Gleichzeitige Kommunikation in beide Richtungen möglich.
Stream: Sendet und empfängt unstrukturierter Byte-Stream.
Graceful Termination: Verbindung wird kontrolliert abgebaut.
Punkt-zu-Punkt: Multicast oder Broadcast existiert nicht.

TCP Verkehrssteuerung

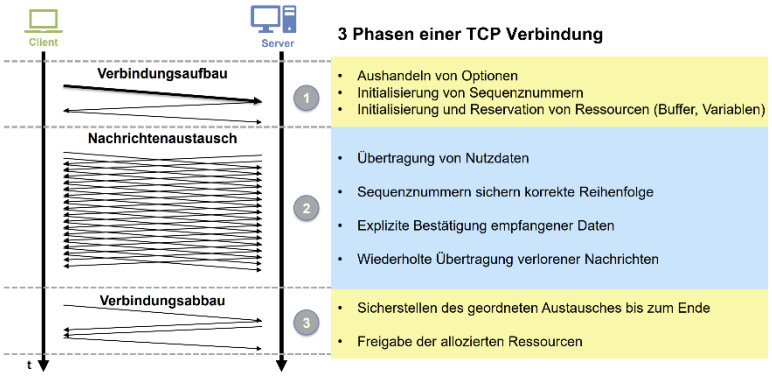
- Client fragt Daten für eine Applikation an. Server wartet, bis sie von einem Client angefragt wird.
Eine TCP-Kommunikation erfolgt in drei Schritten:
- **Verbindungsaufbau**: Server horcht (LISTEN) auf eine bestimmte Portnummer.
Client sendet Paket mit SYN
Server bestätigt mit SYN und ACK
Client bestätigt mit ACK
- **Nachrichtenaustauch**: Ein Paket ist folgenermassen zu interpretieren.
«Ich stehe hier (Seq), du stehst hier (Ack), ich sende x Byte Daten»
Antwort: «OK, ich stehe hier (Seq), du stehst hier (Ack + Daten), ich sende x Byte Daten»
- **Verbindungsabbau**: Die Verbindung muss in beide Richtungen geschlossen werden.
Der Empänger des FIN-Flages muss dies mit einem ACK bestätigen.

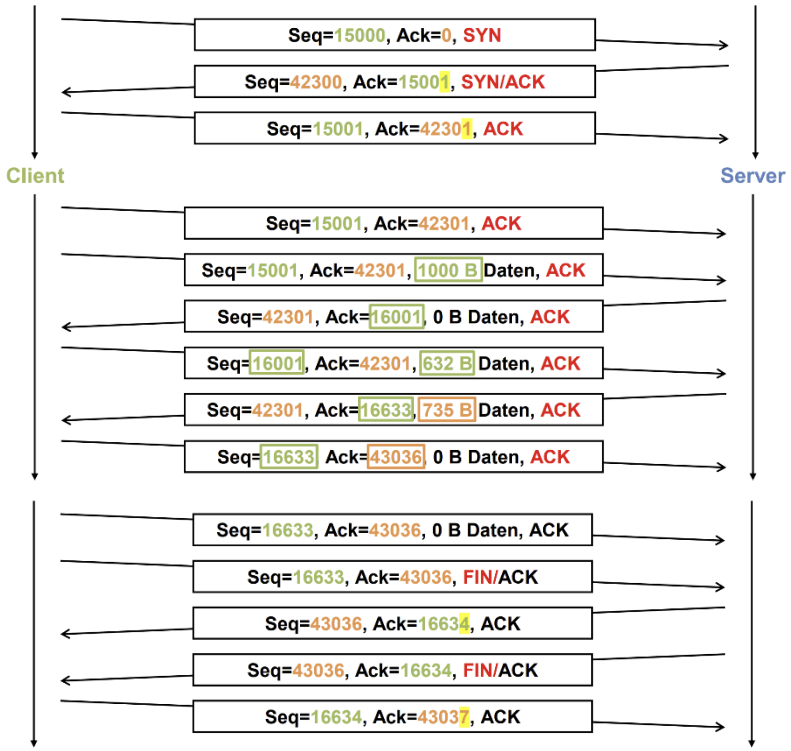


TCP Adaptive Elemente

Erkennen verloren gegangener Elemente: Wird kein ACK empfangen, bevor die Zeit abläuft, sendet der Sender das Paket erneut. Die Pakete werden nummeriert, da der Sender nicht weiss, ob das Paket auf dem hinweg verloren ging, oder das ACK auf dem Rückweg verloren ging.
Berechnung des **Retransmission Time-Out**: (siehe Bild)

Übelastung des Empfängers: Um das erneute Senden von Paketen aufgrund einer Überlastung beim Empfänger zu verhindern, muss die Senderate angepasst werden (Flow Control)
- **Stop & Wait**: Nach jedem Senden wird explizit auf das ACK gewartet -> ineffizient
- Lösung: **Sliding Window**: Mehrere Pakete nacheinander senden und auf die ACKs von allen Paketen warten. Mit jedem ACK wird der verfügbare Pufferplatz beim Empfänger mitgeteilt und der Empfänger kann dynamisch darauf reagieren. Wird ein Pufferplatz von 0 Bytes mitgeteilt, so wartet der Sender auf ein erneutes ACK.
Berechnung des Bandwith-Delay-Product:
BDP (bits) = RTT (sec) X Bandbreite (bps)





Gewichteter Mittelwert ***SRTT*** (Smoothed Round-Trip Time):

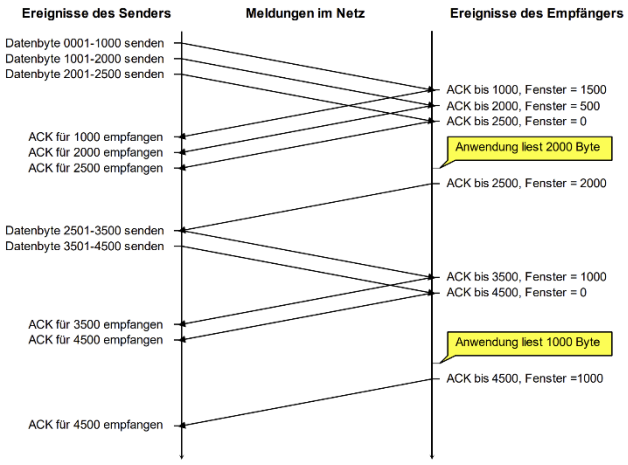
$$SRTT_{neu} = (1 - \alpha) * SRTT_{alt} + \alpha * RTT \qquad \text{mit } \alpha = 0.125$$

Streuung ***RTTVAR*** ist gewichteten Mittelwert der Abweichungen:

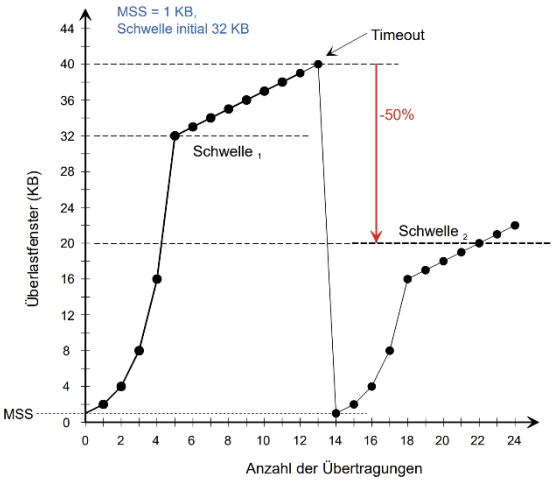
$$RTTVAR_{neu} = (1 - \beta) * RTTVAR_{alt} + \beta * |SRTT - RTT| \qquad \text{mit } \beta = 0.25$$

Retransmission Time-Out ***RTO***:

$$RTO = SRTT + 4 * RTTVAR$$



Überlastung des Netzes (Congestion Control)
Schützt das Netz, ist eine Funktion des Senders
Der Sender beginnt mit kleinen Sendefester und vergrössert diese mit jedem gesendeten Paket exponentiell, bis zur Schwelle, danach vergrössert der Sender die Pakete linear.
Wenn ein Timeout geschieht, reduziert er das nächste Sendefenster wieder auf das minimum und das Spiel fängt wieder von vorne an. Diesmal mit einer geringeren Schwelle.

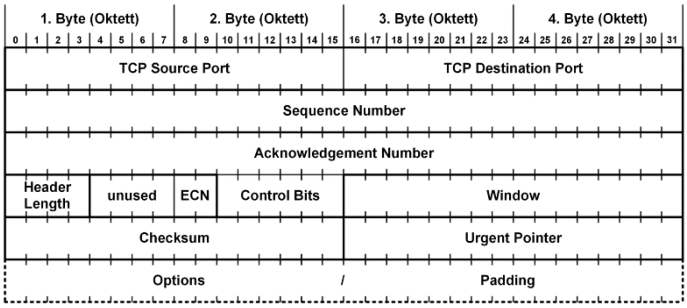


TCP Header

- **TCP Source und Destination Port**: Beim Verbindungsaufbau ist dies der gewünschte Dienst auf der Serverseite.
- **Seq. Number**: Erlaubt es dem Empfänger, die Daten richtig zu ordnen.
- **Ack. Number**: Teilt dem Sender mit, dass alle Bytes bis und mit n angekommen sind.
- **ECN (Explicit Congestion Notification)**: CWR (Bit 8) oder ECE (Bit 9)
- **Control Bits**:

10	11	12	13	14	15
URG	ACK	PSH	RST	SYN	FIN

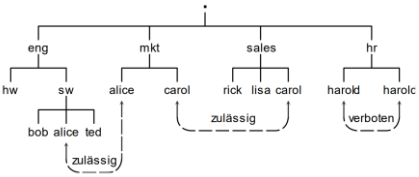
- **Window**: Verfügbare Puffergrösse
- **Urgent Pointer**: Falls URG-Flag gesetzt, bezeichnet Position von wichtigen Daten im Paket (zB. Ctrl-C)



Application Layer

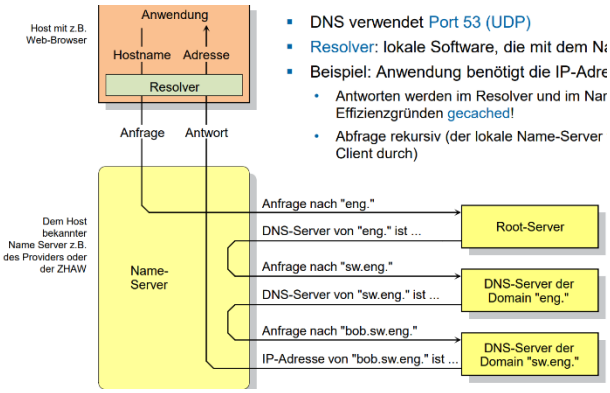
Domain Name System

- Dient als «Übersetzung», sodass Menschen nicht die IP Adresse auswendig lernen müssen, sonder «www.google.com»
- **Domain Name Space**: Im Subdomänen dürfen zwei Knoten nicht denselben Namen haben. Es wird einer Baumstruktur gefolgt. Die höchsten Knoten (.com, .ch, .de, ...) nennt man Top Level Domain.



DNS läuft über Name Server und kennt:
- IP-Adressen zu Hostnamen in seiner Zone
- IP-Adressen zu weiteren Name Servern
- IP-Adressen zu höhren Knoten (wie Root oder Top Level Domain)
Subdomains sind für andere Name Servern (nach aussen) unsichtbar.
- **DNS** verwendet **UDP**
- Lokaler Namer Server führt Abfragen für den Clienten aus und speichert die Antworten temporär in seinem Cache
Wichtige Typen sind: **A** (IPv4) und **AAAA** (IPv6), MX (Mail), NS (Name Server)
Top Level Domains lassen sich in folgende Kategorien unterteilen
- **Generische, weltweite Domains**: .com (grosse Unternehmen), .edu (Bildungseinrichtungen, .net (Internet Providers), ...
- Generische UDA Domains: .gov, .mil
- **Landesspezifische Domains**: .ch, .de, .it
- **Neue Top Level Domains**: .info, .wtf, .gg
Reverse DNS Lookup: Ein Server identifiziert einen Client anhand des Namens, nicht anhand der IP Adresse -> in-addr.arpa enthält ein Baum, der den kompletten IPv4 Adressbaum abbildet; eine Ebene entspricht jeweils einem Byte.

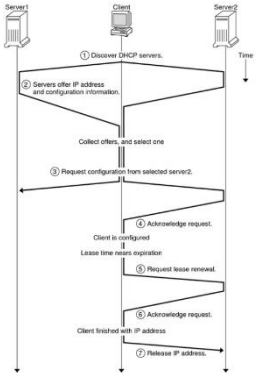
Problem	Schicht 2	Schicht 4	Massnahmen bei TCP
Nachrichtenverlust	$p_{Verlust}$ = FER	$p_{Verlust}$ >> FER	Positives ACK
Telegramm-Reihenfolge	fix	kann variieren	Sequenznummern
Round Trip Time	konstant, µs .. ms	variabel, ms .. s	Adaptiver Retransmission Timeout
Überlast des Empfängers	kommt vor	kommt vor	Sliding Window mit dynamischer Fenstergrösse
Überlast des Netzwerks	direkt beobachtbar (Medium)	nur indirekt beobachtbar	Slow Start (Congestion Window)
Neustart von Hosts	direkt beobachtbar	nur indirekt beobachtbar	3 Weg Handshake, Initialisierung Sequenznr.



- DNS verwendet Port 53 (UDP)
- Resolver: lokale Software, die mit dem Na
- Beispiel: Anwendung benötigt die IP-Adre
  - Antworten werden im Resolver und im Nar Effizienzgründen gecached!
  - Abfrage rekursiv (der lokale Name-Server Client durch)

Dynamic Host Configuration Protocol (DHCP)

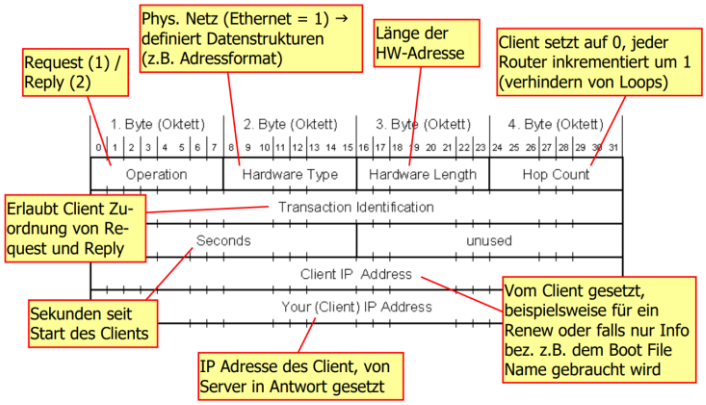
Erlaupt dynamische Zuteilung von IP Adressen an Geräten, beispielsweise für Heimnetzwerk, die Handys, Computer, SmartTV, .. verwalten müssen. Wenn ein Client das Netz verlässt, so wird die Adresse wieder frei.
- Basiert auf **UDP**

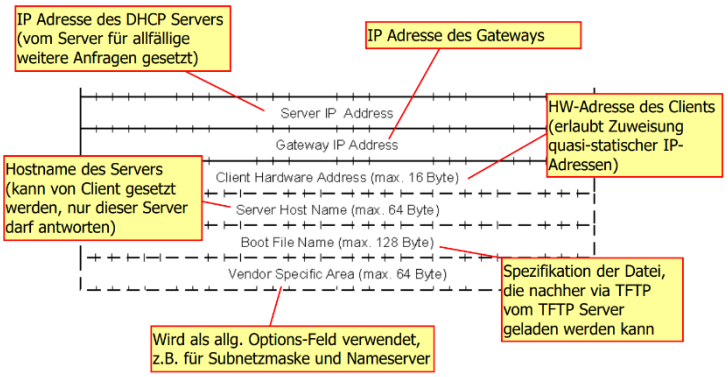


Ablauf gemäss RFC 2132

1. Ein Client sucht einen DHCP-Server mittels Broadcast (DHCP discover)
2. Ein oder mehrere DHCP-Server antworten (DHCP offer)
3. Der Client wähl einen Server und fordert eine Auswahl der angebotenen Parameter an (DHCP request)
4. Der Server bestätigt (DCP acknowledge) mit einer Message, welche die endgültigen Parameter enthält
5. Vor Ablauf der Lease Time erneuert der Client mittels Unicast die Adresse

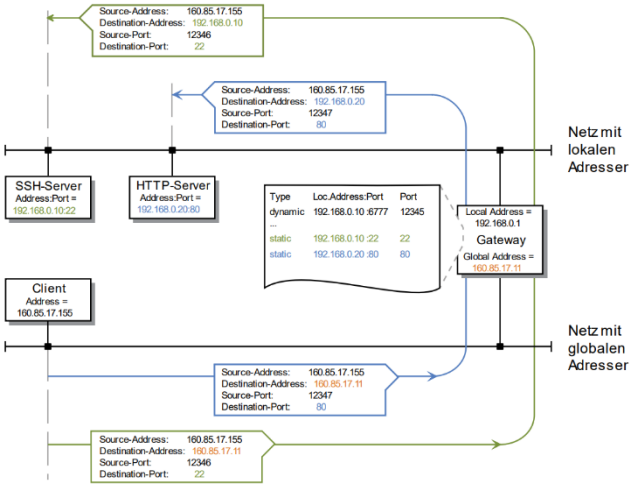
- Ablauf: (siehe Bild oben)
Paketformat: (siehe Bild)

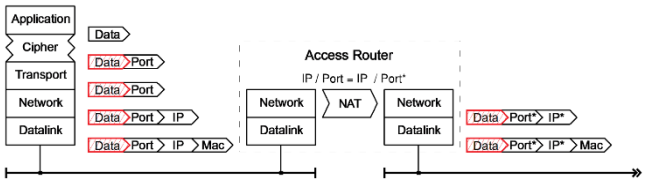




Network Address Translation (NAT)

Ausgangslage: **es gibt keine freie IPv4 Adressen mehr**
- **Port basiertes NAT** (NAPT): Alles Hosts verwenden im privaten Netz (192.168.0.0) verwenden 192.168.0.1 als Default Gateway. Der NAPT ersetzt im IP-Header der ausgehenden Pakete die lokale Source-Adresse 192.168.0.10 durch die globale GatewayAdresse 160.85.17.11.
Ausserdem ersetzt er im Transport-Layer-Header der ausgehenden Pakete das Source-Port 6777 durch eine eindeutige / freie PortNummer.
Wen ein Paket reinkommt, ersetzt er die ursprüngliche IP Adresse und Port wieder zurück.
- Statisches Port basiertes NAT (Port Mapping) im Bild rechts: Internet bekommt Zugriff auf lokale Hosts/Dienste. Ein Port hat eine fix zugeordnete IP-Adresse. Der extern verwendetet Port kann frei gewählt werden.
Probleme sind: verletzt das Konzept der OSI Layer; greift auf Transport Header zu und IP Adressen werden verändert (Checksum muss angepasst werden)





Hypertext Transfer Protocol (http)

HTTP überträgt HTML, Bilder, (Ressourcen) über die **Uniform Ressource Locator** (URL)
- **HTTP** verwendet **TCP** (Port 80)
Requestmethoden: GET, POST, HEAD (Gibt nur Header der Response zurück)