Vorwort

Dies ist der Versuch, den klausrrelevanten Stoff der Vorlesung „Kommunikationssysteme“ von Herrn Bauer in kurzer, verständlicher Form zusammenzufassen. Dieses Dokument unterliegt keinen besonderen Beschränkungen und wird hiermit unter der GNU Lizenz für freie Dokumentation veröffentlicht.

Grundlagen

Digitalisierung

Unterschiede zwischen Digital und Analog

Ich nehme nicht an, dass dieses Thema direkt in der Klausur abgefragt wird, es ist aber hilfreich, diese Grundlagen verstanden zu haben und im Hinterkopf zu haben.

Ein digitales Signal unterscheidet sich grundlegend von einem analogen Signal. Während es in der digitalen Welt nur „an“ und „aus“ gibt, kennt die analoge Welt unendlich viele Zwischenschritte.

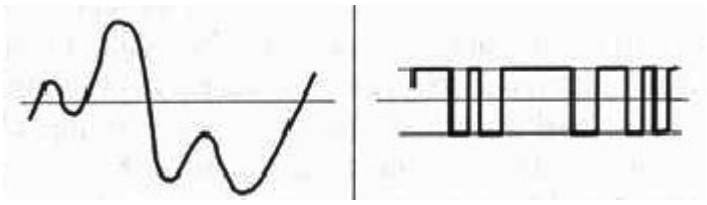


Abb. 1: Vgl. Analog-Signal mit Digital-Signal

Nun ist die Information „an oder aus“ freilich nicht sehr aussagekräftig. Darum fasst man einfach mehrere dieser „an oder aus“ zu einer Einheit zusammen, so dass nun verschiedene Kombinationen möglich sind. Und diese Kombinationen nummeriert man einfach durch. So ist es also möglich, mit ein paar Bit (ein paar „an oder aus“) ganze Zahlen von 0 bis $2^{\text{Anzahl der „an oder aus“}}$-1 darzustellen, was $2^{\text{Anzahl der „an oder aus“}}$ Möglichkeiten entspricht.

Die AD-Wandlung

Jetzt wo wir die Möglichkeit haben, einfache Zahlen digital darzustellen, können wir auch das analoge Signal ins Digitale umwandeln. Dazu machen wir nichts anderes, als in regelmäßigen Abständen das Signal zu messen (hier die Spannung messen, es gibt aber noch andere Möglichkeiten). Sind wir damit fertig, so haben wir das linke Signal aus Abb. 1 in das rechte Signal aus Abb. 1 umgewandelt.

Bei der Analog->Digital-Konvertierung (AD-Konvertierung) spielen nun zwei Größen eine bedeutende Rolle:

- Die Zeiten, zu denen wir das Signal messen
- Bis zu welcher Zahl wir zählen gelernt haben

Die Zeiten nennt man auch ganz professionell „Samplingrate“ oder „Abtastrate“ und wird unter echten Kennern nur in Hz oder kHz gehandelt. Also in „Anzahl der Abtastungen pro Sekunde“.

Der Zahlenraum wird von Profis auch „Samplingtiefe“ genannt und wird meistens in Bit angegeben. Beträgt die Samplingtiefe also 16 Bit, heißt das nichts anderes, dass bei der Abtastung nur Zahlen zwischen [0, 65535] verwendet werden. (0 und 2^{16}-1).

Erhöht man die Abtastrate, tastet man das Signal öfters ab und erhält somit ein zeitlich besser aufgelöstes Abbild des Signals. Dabei besagt das Nyquist-Theorem (übrigens aus den 1950ern), dass man die Abtastrate am besten doppelt so hoch wählt, wie die höchste zu erfassende Frequenz. Da der Mensch im Bestfall nur Frequenzen bis ca. 20 kHz hören kann, liegt die Samplingrate einer CD also bei 44.1 kHz. Über höhere Werte freut sich höchstens der Hund, für den Menschen ist es verschwendeter Speicherplatz!

Die Genauigkeit der Abtastung kann auch erhöht werden, indem man einfach mehr Bits zur Verfügung stellt. Verstärkt man das

abzutastende Signal also so sehr, dass seine größte Amplitude (vereinfacht „Lautstärke“) gerade alle Bits beansprucht, so stehen viel mehr Zahlen zur Verfügung, um die Messwerte des Signals zu erfassen. Und hier besteht sehr wohl ein Unterschied zwischen 16 Bit und 24 Bit!

Die DA-Wandlung

Irgendwann wird man Verlegenheit kommen, das digitale Signal wieder analog zu wandeln (DA-Wandlung). Hier ist es natürlich wünschenswert, dass das resultierende Analog-Signal möglichst genau dem ursprünglichen Signal entspricht. Allerdings wird man dies nie erreichen, egal wie hoch man die Samplingrate und Samplingtiefe wählt. Der Feind heißt hier „Quantisierung“ und ist einfach nicht tot zu kriegen. Wichtig ist allerdings: Digitalisierung nicht gleich Quantisierung! Die Quantisierung ist nur ein Nebeneffekt, der bei der Digitalisierung auftritt.

Dadurch, dass wir beim Digitalisieren immer nur zu diskreten Zeitabständen (z.B. alle 0,01 Sekunden) messen können, und dadurch, dass wir nur ganze Zahlen von 0 bis 2^n-1 verwenden können, schaffen wir es nicht, alle unendlich viele Zustände des Analog-Signals zu erfassen. Statt dessen findet schon bei der Digitalisierung eine Reduzierung der Datenmenge statt. Das Resultat nach der DA-Wandlung ist die berühmte „Treppenkurve“:

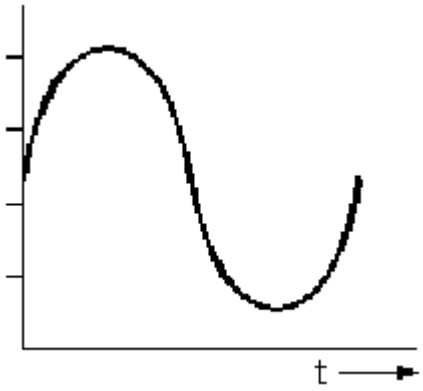


Abb. 2: Analoges Signal vor der ADA-Wandlung

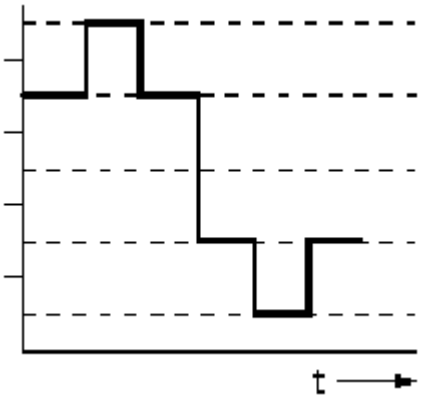


Abb. 3: Das selbe Signal nach der ADA-Wandlung

Vorteile von Digital gegenüber Analog

Obwohl also immer nur eine Annäherung an das ursprüngliche Signal (inklusive Qualitätsverlust) möglich ist, birgt die AD-Wandlung auch einige Vorteile. Da ein digitales Signal nur „an“ und „aus“ und nichts dazwischen kennt, ist es somit sehr unanfällig gegen Leitungsstörungen. Wird ein Signal rein analog übertragen, machen sich Phänomene wie Rauschen (Fremdspannungen) und Einstreuungen (nie vermeidbar!) direkt bemerkbar. Überträgt man aber nur ein digitales Signal und wandelt es beim Empfänger wieder in ein Analogsignal um, müssen solche Leitungseinbusen schon erheblich stärkerer Natur sein, um die Qualität zu beeinträchtigen.

Klassifizierung von Netzwerktechnologie

Neue Technologien werden oft anhand bekannter Eigenschaften klassifiziert. Das selbe gilt natürlich auch für die Netzwerktechnik. Hier haben sich die folgenden Begriffe (neben vielen anderen) eingebürgert:

Leitungsorientiert: *z.B. das gute, alte Telefon. Wird eine Verbindung zwischen zwei Teilnehmern aufgebaut, schalten mehrere Schaltstellen irgendwelche Schalter, so dass eine Direktverbindung (Leitung) zwischen den beiden Teilnehmern besteht. Diese Ändert sich während des ganzen Gespräches nicht.*

Oder zwischen zwei Firmengebäuden verläuft ein Glasfaserkabel, um die beiden Netzwerke miteinander zu koppeln. Zwischen den beiden Netzwerken besteht eine feste Leitung, die sich nie ändert.

Paketorientiert: *Es gibt mehrere Leitungen, welche beliebig miteinander verbunden sein können. Die Daten, welche zwischen zwei Stationen ausgetauscht werden sollen, werden nicht wie beim Telefon kontinuierlich über eine feste Leitung geschickt, sondern in kleine Pakete zerhackt. Jedes dieser Pakete kann einen beliebigen Weg durch das Netz nehmen, bis es beim Empfänger ankommt.*

Nachrichtenorientiert: *z.B. Funk. Es gibt keine festen Verbindungen zwischen den Stationen. Nachrichten werden als ganzes abgesetzt und von allen Stationen in Reichweite entweder empfangen oder nicht.*

Sendet man z.B. im LAN eine Nachricht über die Broadcast-Adresse des Netzes (höchste IP), ist dies damit vergleichbar. Die Nachricht durchläuft das gesamte Netz (keine einzelne Route) und wird von allen erreichbaren Hosts aufgefangen.

Verbindungslos / Verbindungsorientiert: *Hier darf „Verbindung" nicht mit „Leitung" verwechselt werden. Verbindung heißt nur, dass zwei oder mehrere Stationen sich absprechen (Handshake), dass sie miteinander kommunizieren wollen. Wie die Daten tatsächlich transportiert werden (über eine Leitung oder in Form von Paketen) spielt dabei keine Rolle.*

Verbindungslos bedeutet, dass einfach drauf los gesendet wird, egal ob die andere Station gerade empfangen will oder nicht. Hat man eine hohe Burstiness im LAN, kann dies bei Ethernet u.U. zum Ausfall des Netzes führen.

Verbindungsorientiert bedeutet: Das zwei Stationen sich erst absprechen (Handshake) und dann bei Bedarf senden.

Synchrone Übertragung: *Die Daten kommen in der Reihenfolge beim Empfänger an, wie sie gesendet wurden. z.B. Kabelfernsehen, Telefon, VoIP mit bestimmten QoS-Regeln ...*

Asynchrone Übertragung: *Die Daten kommen in beliebiger Reihenfolge beim Empfänger an und müssen bei Bedarf vom Empfänger wieder sortiert werden. Berühmtestes Beispiel: FTP – Es ist nicht wichtig, wie die Daten ankommen, Hauptsache sie sind am Schluss vollständig.*

Wichtig ist, jede dieser Eigenschaften, kann mit jeder anderen kombiniert werden!

Bitrate und Baudrate

- Bitrate = Anzahl der pro Sekunde übertragenen Bits
- Baudrate (z.B. bei Modems): Anzahl der Signalwechsel pro Sekunde
- Ein Baud = Je nach Technologie können pro Baud mehrere Bits übertragen werden!!! (Bitrate > Baudrate)
- Gibt es nur zwei Zustände für ein Baud: Bitrate = Baudrate

- Die Bitrate wird oft fälschlicherweise als Bandbreite bezeichnet.
- Formaler Unterschied zwischen kB, MB, GB ... (KiloByte, MegaByte, GigaByte) und kiB, MiB, GiB ... (KibiByte, MibiByte, GibiByte) wegen zu großer Verwirrung der Skalen. Erstere Rechnen mit dem Faktor 1000, letztere mit dem Faktor 1024. In der Praxis hat sich diese Trennung noch lange nicht durchgesetzt.
- 1 KBit = 1000 Bit, 1 MBit = 1000 KBit (bei Übertragungsraten)

RFCs – Request for comments

Beschreibung

Was für das Internet das W3-Consortium ist, ist für die Netzwerktechnologie der RFC-Editor. RFC steht für „Request for comment", also „Anfrage nach Kommentaren". Obwohl es sich also streng genommen nur um „Anfragen" oder „Vorschläge" handelt, haben diese RFCs durchaus verbindlichen Charakter.

Der RFC-Editor ist eine gewählte Person, welche alle eingereichten RFCs probeliest, bearbeitet veröffentlicht. Hin und wieder wird das Amt des RFC-Editors weitergegeben.

Aufbau des Kopfbereiches

Der Kopfbereich jedes vom RFC-Editor veröffentlichten RFCs ist genormt und beinhaltet u.a. die folgenden Angaben:

- Magic number: RFC
- Nummer der RFC
- Autor
- Titel / Überschrift
- Datum
- Kurzbeschreibung des Themas
- Copyright-Hinweis

RFCs sind über das Internet jedermann zugänglich und liegen in unterschiedlichen Formaten vor. Das traditionelle Format für RFCs ist jedoch das ASCII-Format, welches ebenso wie alle anderen Formate wie ein Buch nach Seiten und Kapiteln gegliedert ist.

Multiplexing-Techniken

Multiplexing was ist das?

Ein definierendes Beispiel

Informationen werden immer von einem Sender zu einem Empfänger übertragen. Das Medium, über das die Informationen übertragen werden, nennt man in der Informationstechnologie auch Kanal. Möchten nur zwei Stationen miteinander kommunizieren reicht ein Kanal völlig aus. Sollen mehrere Stationen eingebunden werden, benötigt man mehrere Kanäle. Was aber, wenn man nur eine Leitung z.B. zwischen Karlsruhe und Berlin hat, aber auf beiden Seiten je 1000 Stationen stehen? Ganz klar, die Leitung muss irgendwie geteilt werden.

Realisierung von Multiplexing

Multiplexing kann auf verschiedene Arten realisiert werden. Vier davon haben wir kennen gelernt:

Zeitscheiben-Multiplexing

Die Zeitscheibe ist eine der Metaphern schlechthin in der Informatik. Oft wird sie in Zusammenhang mit Begriffen wie „Time-Sharing“, „Multitasking“ oder „Round-Robin“ verwendet.

In unserem Fall bedeutet sie nichts anderes, als dass jede Station ein kurzes Zeitintervall zugeordnet bekommt, zu dem sie Senden und empfangen darf. Am Ende war jede Station einmal dran und das ganze beginnt von vorne.

Werden keine Prioritäten festgelegt, so dass sich die Reihenfolge und Zeitintervalle nie ändern

z.B. Station 1 (0,5 Sek.) -> Station 2 (0,5 Sek.) -> ... -> Station 1000 (0,5 Sek) ->

Station 1 (0,5 Sek.) -> Station 2 (0,5 Sek.) -> ... -> Station 1000 (0,5 Sek) ->

Station 1 (0,5 Sek.) -> Station 2 (0,5 Sek.) -> ... -> Station 1000 (0,5 Sek) ->

...

spricht man von „Round-Robin“.

Ausgefeiltere Systeme ermitteln, den Bedarf einer jeden Station und ändern so die Reihenfolge und die Dauer, die jede Station senden darf. Hier gibt es die unterschiedlichsten Ansätze, welche ganz besonders unter Designern von Multitasking-Kerneln heiß diskutiert werden.

Frequenzmultiplexing, Wellenlängenmultiplexing bei Glasfaser

Normalerweise wird über eine Glasfaserleitung nur ein Lichtstrahl von einer bestimmten Wellenlänge übertragen. Beim Wellenlängenmultiplexing werden einfach mehrere Lichtstrahlen unterschiedlicher Wellenlängen über die selbe Leitung geschickt. Da sich die Wellenlängen nicht überschneiden, können die einzelnen Lichtstrahlen durch einfache Filter von einander getrennt werden.

Etwas komplizierter, aber im Prinzip gleich, funktioniert das Frequenzmultiplexing (Frequenz = 1 / Wellenlänge), welches beim Funk angewendet wird, aber auch für Kupferkabel und viele weitere Übertragungsarten geeignet ist. Es wird ein Signal einer bestimmten Frequenz erzeugt (Sinuswelle). Dieses heißt „Trägerwelle“. Dieser Trägerwelle werden die zu übermittelnden Informationen durch Modulation aufgepresst. Mehrere solcher Wellen können über ein Medium übertragen werden und durch einfache Filter voneinander getrennt werden. (z.B. Kondensator-Widerstand-Filter für elektrische Signale).

Hauptsächlich werden zwei Modulationsarten unterscheiden, welche hier nur grob umrissen werden:

1. Amplitudenmodulation
2. Frequenzmodulation

<u>Amplitudenmodulation:</u>

Schaut man sich eine Sinuskurve in einem Graphen an, so ist ihre Höhe zu einem Zeitpunkt gleich die Amplitude zu dieser Zeit. Macht man die Welle hörbar, kann man die Amplitude (vereinfacht) auch als Lautstärke bezeichnen.

Eine Trägerwelle kann nun den Verlauf einer zweiten Welle übermitteln, in dem man die Trägerwelle einfach Lauter und leiser werden lässt:

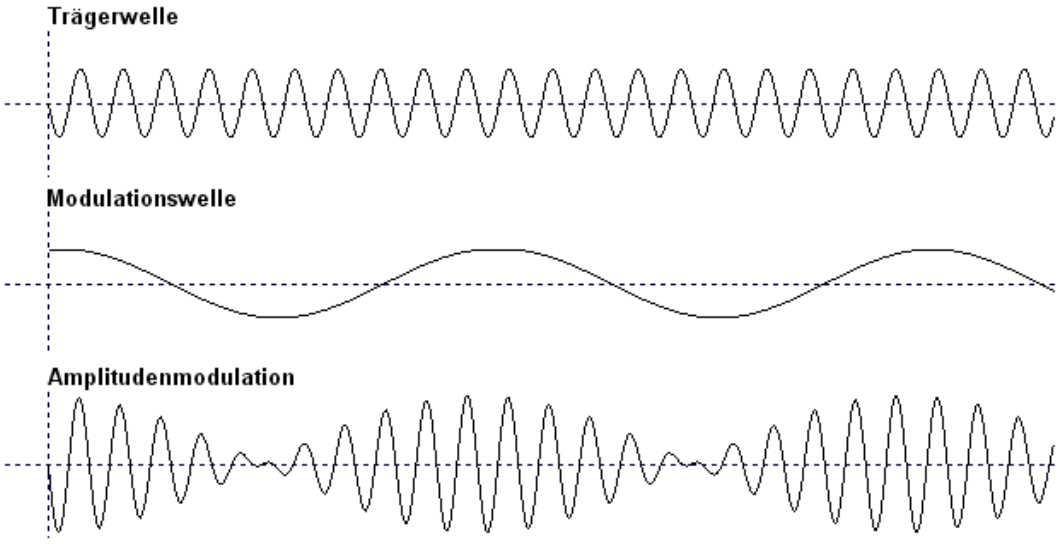


Abb. 4: Amplitudenmodulation

Frequenzmodulation:

Die Frequenzmodulation funktioniert genau so, wie die Amplitudenmodulation. Allerdings wird nicht die Höhe (Amplitude) der Trägerwelle geändert, sondern ihre (Breite) Frequenz. Die Breite der Kurve wird also enger und weiter, je nachdem wie sich die Modulationswelle verändert.

Dadurch, dass sich die Frequenz der Trägerwelle ändert, reicht es nicht mehr, genau diese eine Frequenz herauszufiltern. Vielmehr muss ein Frequenzband festgelegt werden, in dessen Bereich sich die Trägerwelle bewegt. (z.B. 100 – 101 MHz).

Statistisches Multiplexing (Port- / Paket- / Dienst-Multiplexing)

Die über einen Kanal übertragenen Daten werden in sog. Pakete aufgeteilt. Jedes dieser Datenpakete bekommt die Information, zu welchen Dienst es gehört (z.B. Port 80 HTTP, Port 21 FTP, Port 25 SMTP etc.). Der Empfänger leitet diese Pakete nun intern an die entsprechenden Adressaten weiter.

Code-Multiplexing mit der Walsh-Funktion

- Eine Walsh-Funktion kann zu jedem Zeitpunkt nur die Werte +1 oder -1 annehmen.
- Verschiedene Walsh-Funktionen sind vorgegeben und sowohl Sender als auch Empfänger gekannt.
- Alle Walsh-Funktionen müssen orthogonal zueinander sein (Vgl. Orthogonalität von Vektoren)
- Eine Walsh-Funktion kann vereinfacht (für uns) als n-Tupel aufgefasst werden, also als eine Sammlung von n-Werten.
- => Alle Walsh-Funktionen müssen die selbe Anzahl an Werten aufweisen
- Jeder Sender sendet genau 1 Bit pro Walsh-Funktion!
- Da Bits nur 0 und 1 kennen, die Walsh-Funktion aber nur +1 und -1, legen wir fest, dass der Bitwert 0 dem Walshwert -1 entspricht.

- $B \in \{+1; -1\}$ Das zu sendende Bit
- n Anzahl der Werte pro Walsh-Funktion

- $w_i \in \{+1;-1\}$ Ein Wert aus der Walsh-Funktion
- $W_i = (w_1 | w_2 | \cdots | w_n)$ Die i-te Walsh-Funktion als n-Tupel dargestellt
- W_{mi} Die modifizierte Walsh-Funktion i
- S Die Summenfunktion nach dem Codieren
- T Zwischenergebnis beim Decodieren
- w_{ti} Der i-te Wert aus T

Das sieht jetzt fürchterlich kompliziert aus, macht die Sache aber schön einfach, wenn man noch weiß, wie man mit Vektoren (oder n-Tupeln) rechnet:

Codieren:

- $W_{mi} = B \cdot W_i$

  Alle Werte der Walsh-Funktion mit dem zu sendenden Bit multiplizieren

- Die so modifizierte Walsh-Funktion wird vom Sender an eine zentrale Station geschickt, welche mehrere modifizierte Walsh-Funktionen einfach addiert:

  $$S = \sum_{i=1}^{n} W_{mi} = W_{m1} + W_{m2} + \cdots + W_{mn}$$

  Einfach den ersten Wert der ersten Funktion, mit dem ersten Wert der anderen Funktionen addieren. Dann das selbe mit dem zweiten, dritten, ... n-ten Wert machen.

  Daraus entsteht die Summenfunktion, welche allen Empfängern zugeschickt wird.

Decodieren:

- $T = S \cdot W_i$

  Die entsprechenden Werte der Summenfunktion, mit den entsprechenden Werten der ursprünglichen Walsh-Funktion multiplizieren.

- $B = \varnothing(T)$

  $$B = \frac{\sum_{i=i}^{n} w_{ti}}{n}$$

  $$B = \frac{w_{t1} + w_{t2} + \cdots + w_{tn}}{n}$$

  Um B zu decodieren, wird einfach der Querschnitt aus T gebildet: Man addiert alle Werte aus T und teilt sie durch ihre Anzahl.

Das OSI-Modell im Schnelldurchlauf

Das OSI-Modell zum Auswendig lernen

Schicht	Englisch	Deutsch	TCP/IP, UDP/IP	Dateneinheit
1	Physical layer	Bitübertragungsschicht	Verbindungsschicht	Kontinuierlicher Bitstream
2	Data link layer	Sicherungsschicht		Frame
3	Network layer	Verbindungsschicht	Internet- / IP-Schicht	Paket / Datagramm
4	Transport layer	Transportschicht	Transport (TCP, UDP)	Segment
5	Session layer	Kommunikations-Steuerungsschicht	Anwendungsschicht (HTTP, SMPT, FTP etc.)	Message
6	Presentation layer	Dartstellungsschicht		
7	Application layer	Verbindungsschicht		

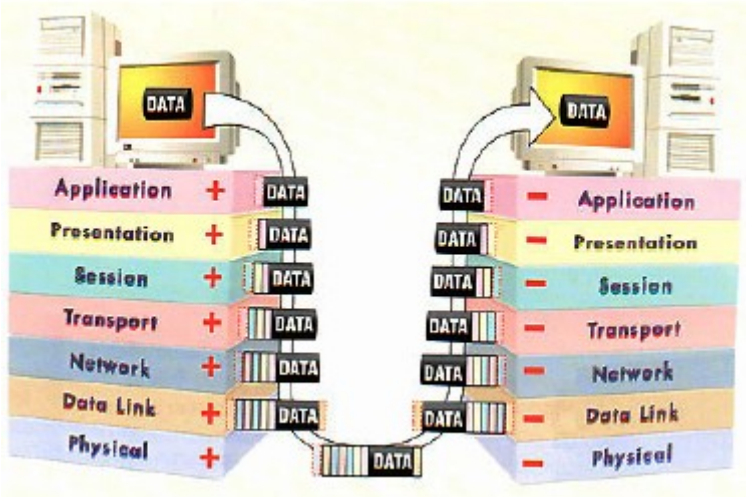


Abb. 5: Jede Schicht fügt dem Datenstrom ihren eigenen Header zu

Layer 1 – Physical layer

Definition eines physikalischen Mediums zur Übertragung von Bits (Hardware-Layer): Kabeltypen, Belegung von Pins, Spannungspegel, ...

Hier schickt man Nullen und Einsen.

Layer 2 – Data link layer

Bereitstellen der Möglichkeit, strukturierte Bitströme zwischen benachbarten Systemen sicher zu übertragen. Auf dieser Ebene operiert auch die Netzwerkkarte.

Hier überprüft man, ob Bits auch gut ankommen. Hier gibt es auch schon MAC-Adressen, welche eine Netzwerkkarte eindeutig identifizieren (sollten).

Layer 3 – Network layer (IP-Schicht)

Ab hier wird logisch gearbeitet, völlig unabhängig von der Hardware. Hier können Teilnetze verknüpft werden und Daten über Vermittlungsknoten weitergeleitet werden (Pfadschaltung / Routing / Wie kommt ein Paket von A nach B?)

Layer 4 – Transport layer

Hier wird nochmal geprüft, ob eine Verbindung steht. Ebenso werden hier die Ports definiert, um mehrere Dienste auf einem Rechner anzusprechen. (Statistisches Multiplexing anhand von Diensten). Auf dieser Ebene operieren hauptsächlich TCP und UDP.

Layer 5 – Session layer

Steuerung des Auf- und Abbaus einer Kommunikationsverbindung. Bei Abbruch einer Transportverbindung (Session existiert noch) wird die unterbrochene Übertragung ab einem Synchronisationspunkt wieder angefahren. Beispiele:

- Voice over IP stellt Verbindungen mit dem SIP-Protokoll her
- Druckjobs von Netzwerkdruckern
- Kostenpflichtige Angebote im Internet
- Telefonschach:
    - Partie = Session
    - Die Sitzung erstreckt sich über mehrere Telefongespräche (Verbindungen).

Layer 6 – Presentation layer

„For a long time it was a layer in search for a function". (Tannebaum)

- Globale Datendarstellung (Konvertierung zwischen verschiedenen Zeichensätzen)
- Ursprünglich Datenkompression
- Ursprünglich Kryptographie
- Heute nur noch wichtig für die Anbindung Mainframe<->Client

Kompression und Kryptographie wird heute auf allen Layern gemacht, nur nicht auf Layer 6.

Layer 7 – Applicaiton layer

Nicht die Anwendung selbst! Hier werden hochwertige Kommunikationssysteme für typische Anwendungen zur Verfügung gestellt: u.A. werden hier folgende Protokolle definiert:

- http
- pop, pop2, pop3
- imap
- shttp
- ftp
- ...

Typische Komponenten eines Netzwerkes

- Passive Komponenten
    - Kabel
    - Funk

    - …
- Aktive Komponenten
    - Kopplungseinheiten
    - Netzwerkkarte
- Netzwerkbetriebssystem
- Netzwerkperipherie
    - Drucker
    - Modem
- Server im Netz
    - Fileserver
    - eMail-Server
    - HTTP-Server
    - DHCP-Server
    - MASQ-Server

Layer 1 des OSI-Modells

Netzwerktopologie

Die Topologie ist ein Begriff aus der Mathematik und lässt sich umgangssprachlich am besten mit „Die Verbindungen, die ein Graph hat“ umschreiben. (z.B. Dreieckstopologie, Netztopologie ...). Dabei bleiben diese Verbindungen nach dem Falten einer Figur bestehen.

Physikalische Topologie

So liegen die Kabel

Logische Topologie

So funktioniert die Sache technisch.

Man kann also die Kabel anders miteinander verbinden, als sie liegen. z.B. könnten die Kabel in Form eines Sterns liegen, das Netz aber zu einem Kreis geschaltet sein.

Wichtig der Unterschied, wie die Kabel liegen und wie man sie nutzt

In beiden Fällen kann man u.a. zwischen den folgenden Topologien unterscheiden:

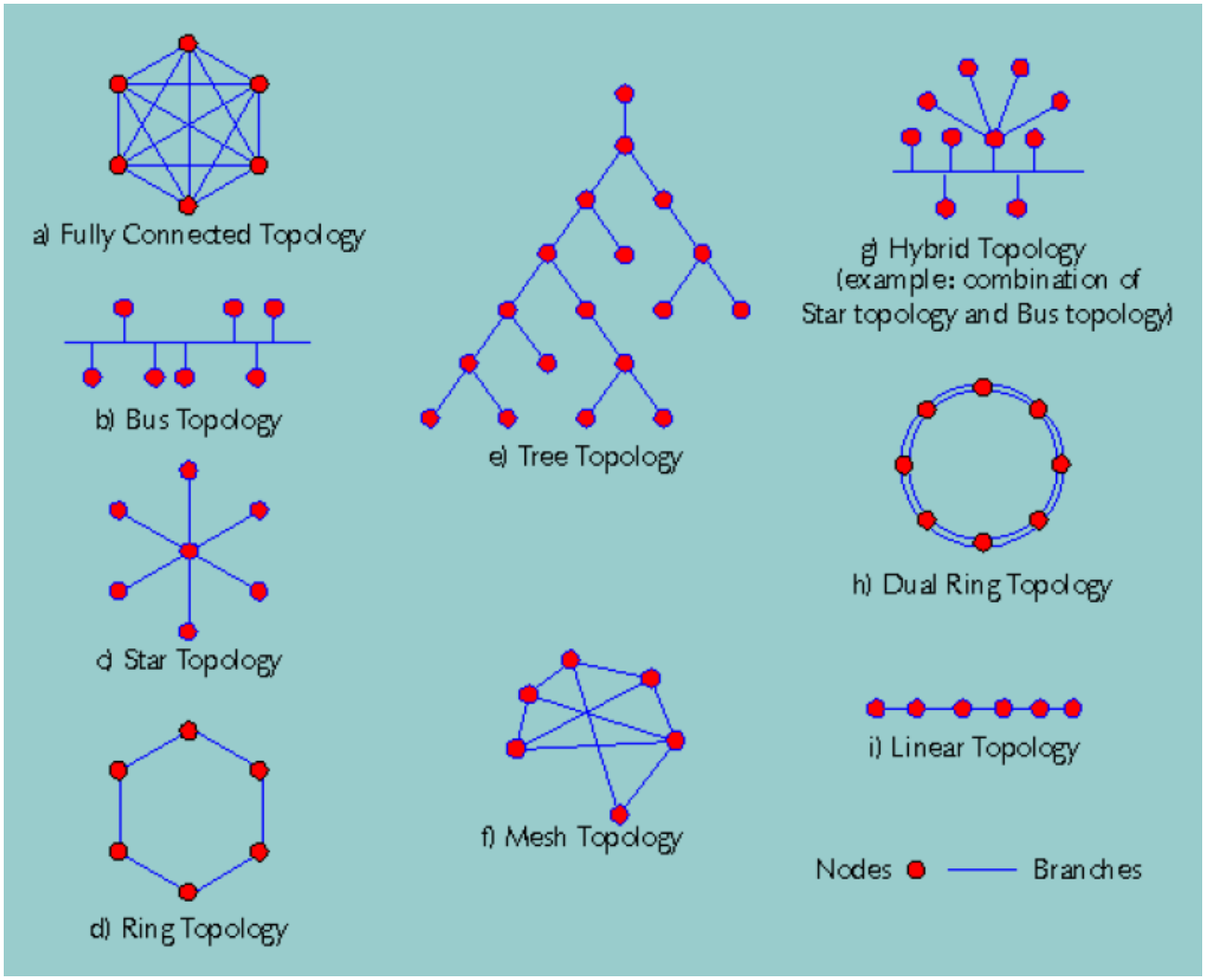


Abb. 6: Verschiedene Topologien

Namenskonvention bei Kabelbezeichnungen

- Technische Bezeichnung
  - <Datenrate in Mbps Megabit/Sek><Übertragunsverfahren><max. Länge in 100m><Art der Verkabelung>
  - Beispiele:
    - 10Base5
    - 10Base2
    - 10BaseTX (Twisted Pair)
    - 100BaseFX (Glasfaser)
- Klassifizierung durch Kategorien
  - CAT1 ~ CAT7

    - Alle 10m sollte eine Netzwerkdose eingeplant werden !!
    - CAT5 ist heute Standard
    - CAT7 ist zukunftssicherer
    - Über CAT kann man sogar (und sollte man!) mit dem Architekten eines Gebäudes reden
- Glassfaserleiter:
    - Multimode (Billig)
    - Gradientfaser (schlechter)
    - Monomode (teuer)

QoS (Quality of Service) / Leitungsparameter

- Kriterien, die die Güte eines Dienstes im Netz beschreiben bzgl. Übertragungsgeschwindigkeit und Zuverlässigkeit
    - Dämpfung (attenuation)
        - Signal wird schwächer (Energie wird kleiner)
        - Glas dämpft z.B. viel vom UV oder IR-Bereich
        - Bei Kupfer höher als bei Glasfaser
        - Dämpfungskurve zeigt die Dämpfung [db] für jede Frequenz [Hz]
        - Abhängig von Entfernung und Frequenz
        - Bei Funk $1/r^2$ da Kugelartige Ausbreitung
            - Oberfläche der Welle nimmt wie ein Kugel zu
            - Energiedichte auf der Oberfläche nimmt ab
            - Oberfläche: r^2
            - => Abnehmen des Signals von Funk, Gravitation, elektr. Feldern: $1/r^2$
    - Dispersion (Laufzeitverzerrung)
        - Ein Laserstrahl wird in einem Glasfaserkabel unterschiedlich reflektiert
        - Das Licht nimmt als unterschiedlich lange Wege
        - Das ursprünglich klare Signal verschmiert also in die Breite (Zeit)
        - Oder auch bei Langwellenfunk: Mehrere Wellen umrunden die Erdkugel in verschiedenen Richtungen und kommen darum zeitversetzt beim Empfänger an: z.B. Phantombilder im Fernsehen
    - Rauschen
        - Nicht bei Lichtwellenleitern
        - Häufigst durch Elektromagnetische Induktion verursacht
        - Verhältnis Nutzsignal / Hintergrundsignal = SNR (Signal/Raus-Abstand = Signal/Noise Ratio)

- Burstiness
    - Verhältnis zwischen Spitzenauslastung und der mittleren Auslastung: B = S/E
    - Sollte im Netzwerk gering und gleichmäßig sein
    - B = 1 => ganz gleichmäßig
    - Je höher B desto schlechter (kann dem Netz schaden, [Ethernet-]Netz kann kollabieren durch zu hohe Datenkollision)
- Latenz
    - Hauptsächlich durch aktive Komponenten
        - z.B. Switches die Daten umformatieren oder Routen berechnen
        - CODECs, die Daten umwandeln müssen
    - Durch Transferzeit nur bei exterrestrischer Kommunikation, da sonst zu kleine Wege
    - Zeitverzögerung zwischen Sender und Empfänger
    - Eine Verteilung ist messbar, da nicht alle Pakete die selbe Verzögerung haben
    - ca. 200 ~ 300 ms sind für VoIP noch in Ordnung
    - Latenz (Verzögerung) Kann man für jedes einzelne Gerät messen
- Jitter
    - **Halbwertsbreite** der Verteilung der Latenz (Varianz der Latenz)
    - z.B. wichtig bei Video (Film ruckelt durch Jitter nicht Latenz)
    - Puffert man um dem entgegenzuwirken, erhöht sich die Latenz
- Transferzeit
    - Benötigte Zeit von A nach B
    - Funk und Lichtwellen: Lichtgeschwindigkeit c
    - Kabel: ca. 0,7 * c
    - Bemerkbar bei exterrestrischer Kommunikation (Satelliten, Shuttles)
    - Auf der Erde zu geringe Abstände, um sich ernsthaft bemerkbar zu machen
        - Obwohl: USA, Europa: 30 ms
    - Transferzeit = Sendelatenz + Empfangslatenz + Übertragungszeit = Gesamtlatenz
- Bandbreitenbegrenzung
    - Durch Begrenzung der Bandbreite wird aus einem Rechtecksignal ein welligeres Signal
        - Umgekehrte Furieraddition zur Erzeugung von Rechteckwellen !!
        - Jedes Kabel hat eine Bandbreitenbegrenzung, darum werden Frequenzen abgeschnitten
- Die zwei (fast) wichtigsten sind heute Jitter und Latenz

- Aktive Komponenten müssen QoS-Parameter unterstützen:
    - Anforderung an eine Leitung werden vom Client gegeben
    - Aktive Komponenten müssen diese Anforderung möglichst einhalten

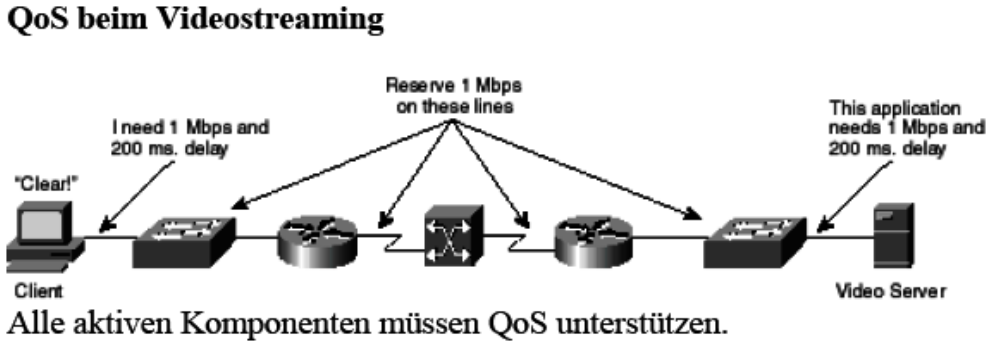


Abb. 7: Beispiel für QoS-Parameter

Layer 2: Data Link Layer

- **Schicht 2a: MAC - media access control**
    - Adressierung
    - Zugangsverfahren
- **Schicht 2b: LLC - Logical link control**
    - Fehlererkennung und Behebung
    - Flusssteuerung
    - Synchronisieren

Schicht 2a (MAC – Media Access control)

- Jede Netzwerkkarte (sollte) eine weltweit eindeutige MAC-Adresse haben
    - Nummernkreise der Hersteller
- Nur homogene Adressierung möglich (nur eine Adresse pro Rechner / Netzwerkkarte)
- Schnelle Verbindungsschaltung durch Pufferung der MACs in den Switches

Schicht 2b (LLC – Logical link control)

- Fehlererkennung z.B. durch Paritätsbits
    - Ein zusätzliches Prüfbit ist entweder 1 oder 0, je nach gerader oder ungerader Anzahl der Bits (Paritätsbit)
- Fehlererkennung durch Checksummen
    - Einzelne Bits können repariert werden

- Das können Paritätsbits nicht

- Flusssteuerung
  - Konkurrierendes (stochastisches) Zugangsverfahren
    - Bsp. CSMA/CD im Ethernet-LAN
      - Carrier Sense Multiple Access With Collission Detection (Kollisionserkennung im LAN, dazu später mehr)
  - Koordiniertes (deterministisches) Zugangsverfahren
    - Bsp. Token passing bei Token ring (überhaupt nichts zufälliges)
      - Token läuft immer im Kreis
      - Wer den Token hat, darf was anhängen
      - Teuer, langsam, sicher
    - Teurer aber zuverlässiger als stochastische Verfahren
  - **Wo beginnt und endet ein Paket im ganzen Nullen- und Einerwuscht? (Bitstaffing)**
    - Eingestopfte Bits (z.B. 111111) umklammern ein Frame
    - Kommt im Datenteil des Frames nun eine solche Kombination vor, wird einfach eine Null eingestopft: 1111101
    - Der Empfänger macht das rückgängig und kann an den sechs 1en Anfang und Ende erkennen
    - Was ist wenn man nun 1111101 übertragen will?
      - Bit staffing greift wieder: Aus 1111101 wird 11111001
  - Hier gibt es noch viel, viel mehr Protokolle und Definitionen
- Synchronisation

Restriktionen der Netzwerkgröße (in Metern) durch das Kollisionsverfahren)

Die minimale Framelänge liegt bei 64 KB für normales Ethernet (10 MBit/Sekunde). Das Senden dieses Pakets dauert daher genau 51,2 µSekunden.

Kommt aus auf einer Kollisionsdomäne (meist eine Busleitung) zur Kollision muss das Jam-Signal alle Empfänger erreichen, noch bevor die kollidierenden Frames zu Ende gesendet wurden.

Wir gehen dabei vom schlechtesten Fall aus, und zwar, dass zwei Stationen den größtmöglichen Abstand zu einander haben sollen und nur Pakete von 64 KB Größe verschicken.

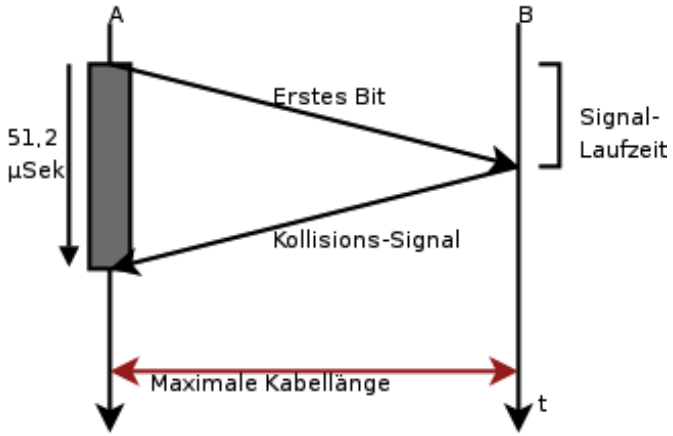


Abb. 8: Laufzeit des JAM-Signals

Das Kollisionssignal muss also noch während dem Senden des Frames A bei A ankommen. Da man von einer Übertragungsgeschwindigkeit in Kupferleitungen mit ca. 0,7*c ausgehen muss, kann die maximale Entfernung zwischen zwei Stationen maximal 2.000 m sein. Natürlich liegt die Praxis deutlich darunter. (10 MBit Ethernet)

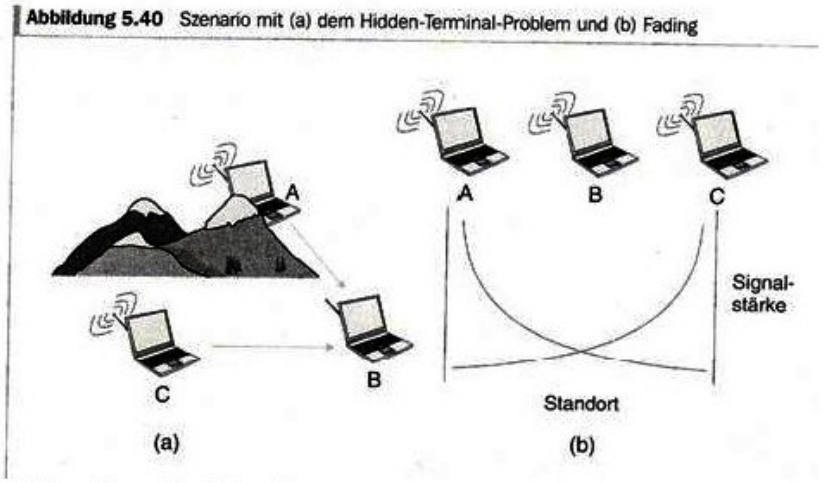
- Ethernet 10 MBit / Sek
- Fast Ethernet 100 MBit / Sek
- Gigabit Ethernet 1000 MBit / Sek
- 10-Gigabit Ethernet 10.000 MBit / Sek

Bei Fast Ethernet hat man also eine 10x höhere Sendefrequenz. d.h. Statt 51,2 µSek benötigt ein Paket nur noch 5,21 µSµSek. Darum schrumpft die maximale Entfernung zwischen zwei Stationen auf max. 200 m. Das ist auch der Grund, warum die Verteilerschränke möglichst in der Mitte von einem Gebäude stehen.

Ein Switch verwaltet pro Strahl (Anschluss am Switch) eine eigene Kollisionsdomäne.

Das heißt ab Gigabit Ethernet wird es unrentabel, was die Länge in Metern angeht. Darum erhöht man die minimale Framelänge auf 512 Byte, also ca. 10x größer, so dass es wieder ca. 5,12 µSek braucht, bis ein Paket von kleinster Größe verschickt wurde.

WLAN auf Layer 1 und Layer 2



Hidden-Terminal Problem:

Station C empfängt Station A durch ein Funkhindernis nicht. Beide haben aber Empfang zu B. Bei B könnte also eine unerkannte Kollision entstehen.

Fading:

Die Stärke des Funksignals nimmt quadratisch mit der Entfernung ab. Auch hier könnten bei einer Station in der Mitte (B) Kollisionen durch A und C entstehen, die nicht erkannt werden, da A und C gegenseitig „außer Reichweite“ sind.

Da Kollision im WLAN also nicht immer erkannt werden kann, muss sie vermieden werden!! (CSMA/CA -> Carrier-Sense Multiple Access with Collision Avoidance)

Für diese Kollisionsvermeidung gibt es zwei Wege:

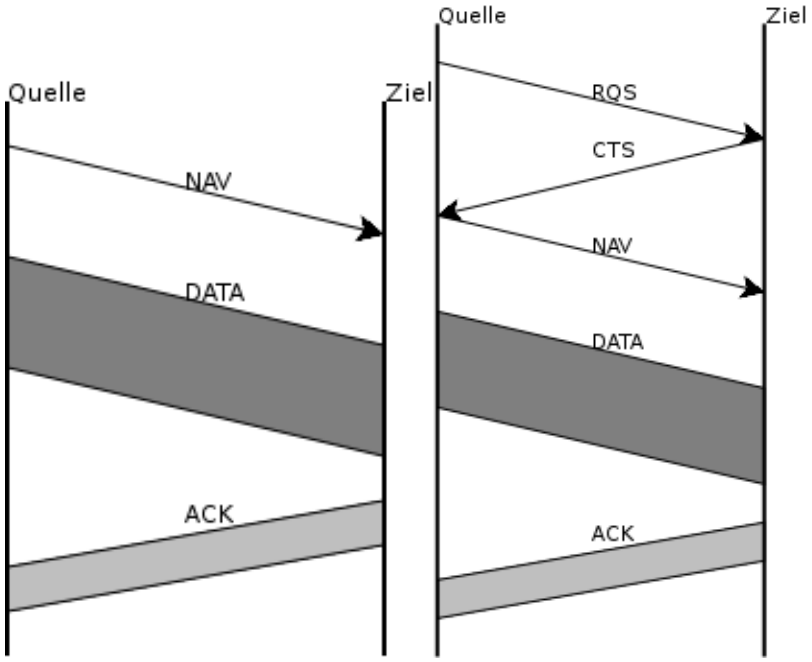


Abb. 9: a) NAV – Network access vector (links)

b) RTS/CTS – Request to send, Clear to send (rechts)

a) Sendezeit den anderen Stationen mitteilen und nach dem Senden auf Bestätigung warten

NAV = Network Allocation Vector (Das Zeitsignal)

==> Unsicheres Verfahren, da bei Fading und Hidden-Terminal nicht alle Stationen alle Daten (inkl. Zeitsignal) mitbekommen.

Arbeitet man mit einem zentralen Access Point, hat man da weniger Probleme.

b) RTS/CTS-Funktion noch mit einschalten: Handshake mit Empfänger vor Senden des NAVs. (Request to send / Clear to send)

Darum etwas langsamer, aber natürlich sicherer

Überlappungsfreie Kanäle

Die Frequenzbänder des IEEE 802.11 Standards überlappen sich teilweise (in Europa, Asien und USA unterschiedlich!). Keine Frequenzüberlappung haben Sie bei folgenden Kanalkombinationen:

a. 1 – 6 – 11
b. 2 – 7 – 12
c. 3 – 8 – 13
d. 4 -9
e. 5- 10

Layer 3 des OSI-Modells: Network layer (Transportschicht)

Grundlagen

Über das Internet Protocol (IP)

- Das ARP-Protokoll dient als Vermittler zischen Layer 2 und 3 (Konvertierung IP-Adresse <--> MAC-Adresse)
- Wir behandeln hier nur das IP-Protokoll
- Der IP-Header beinhaltet mehrere Steuerdaten:
    - Quell-IP
    - Ziel-IP
    - Prüfziffer für Header
    - TTL (Time to live = Anzahl der Hops = Anzahl der Stationen, die dieses Paket passieren darf)

Logische Adressen / Domain names

Logische Adressen (Domain names) sind einfach nur synonyme für IP-Adressen in Form von fast beliebigen Zeichenfolgen (Wörtern). Als Trennzeichen dient der Punkt. Jeder Teil einer Domain muss mit einem Punkt enden, nur am Schluss darf er ausgelassen werden (www.windows3.de ist genauso gültig wie www.windows3.de.).

Von hinten gelesen bringt jedes Segment den Besucher einen Schritt weiter an den Zielrechner bzw. dessen Dienst:

z.B.: ncc-1701a.homelinux.net.

- net Top-Level-Domain für im Internet erreichbare Netze
- homelinux Ein bestimmter Server von DynDns irgendwo in den USA (Second level domain)
- ncc-1701a Sub-Domain zum DynDns-Server. Wird aufgelöst in die IP des Rechners

Top-Level-Domains sind meist Ländercodes oder Codes, welche die Art die Organisation beschreiben (z.B. DE für Deutschland oder GOV für Regierungsapparate).

Sub-Level-Domains können beliebig benannt werden. Je nach Verwalter der TLDs gelten hier gewissen Einschränkungen wie z.B. Mindestlänge von 3 Zeichen (DENIC) oder, dass die SLD nicht mit eine Zahl beginnen darf (ebenso DENIC).

Domain-Anfragen werden durch das DNS-Protokoll an Domain-Name-Server weitergeleitet.

Subneting durch Klassen (Nummernkreise)

- IPv4-Adressen sind 32 Bit breit
- Jede IP-Adresse beschreibt weltweit eindeutig einen Rechner
- Je nach Umfang des ans Internet anzubindenden Netzes können verschiedene IP-Klassen gekauft werden. (Man kauft das Recht, gewisse IPs im freien Internet zu verwenden / für sich zu beanspruchen)
- Es handelt sich hierbei einfach um Nummernkreise, welche durch den Aufbau der IP-Adressen definiert werden.
- Dabei ist der Netzteil vorgegeben, der Host-Teil steht zur freien Verfügung

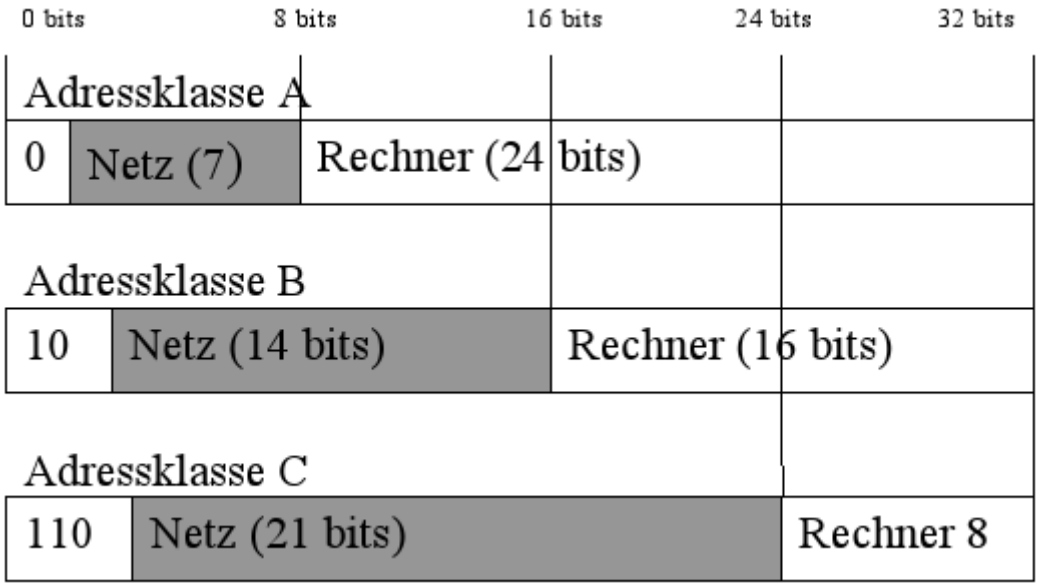


Abb. 10: Aufbau von IPs der Klassen A, B und C

- Da mit Klasse A nur sehr wenige Netze mit vielen Hosts reserviert werden können, sind diese für die interne Verwendung vorgesehen. IP-Adressen der Klasse A werden häufig nicht im Internet verfügbar gemacht.
- Umgekehrt ist es bei IPs der Klasse C.
- Da es mehr Rechner wie IPs auf dieser Welt gibt, kann nicht mehr jeder Rechner mit seiner eigenen IP im Internet erscheinen.
- Stattdessen hängt man mehrere Rechner an einen NAT-Server, welcher den ganzen Internetverkehr regelt. Von außen

sieht es so aus, als würde nur der NAT-Server im Internet surfen, da nur seine IP-Adresse im Internet verwendet wird. Anfragen an seine IP wandelt er um in Anfragen an interne IPs um (Network Access Translation).

Klassenloses Subnetting

NAT-Techniken helfen, mehrere Rechner zu einer öffentlichen IP zusammenzufassen. Dies wirkt dem Problem der knappen IPv4-Adressen aber nicht ausreichend entgegen. Es musste also ein Weg gefunden werden, mit den knappen IP-Adressen noch besser zu haushalten. Die Lösung waren Subnet-Masken.

Streng genommen handelt es sich hierbei nur um eine Weiterentwicklung der alten Netzklassen. Bisher war durch die Angabe einer Klasse fest angegeben, welche Bits einer IP das Netz beschreiben und welche Bits innerhalb dieses Netzes den einzelnen Rechner beschreiben. Durch Subnet-Masken kann diese Einteilung nun selbst vorgenommen werden.

Somit gehört zu jeder IP auch die Angabe einer Subnet-Maske um festzustellen, um welchen Rechner es sich in welchem Netzwerk handelt. Dies ist z.B. wichtig für Router und Switches welche darauf basierend entscheiden ob ein IP-Paket an den Rechner weitergereicht wird oder nicht. Es ist nicht wichtig, um den Rechner weltweit eindeutig zu identifizieren, da es ja keine IP-Adresse doppelt geben darf.

Das Protokoll, welches Subnet-Masken definiert nennt sich CIDR (Classless Inter Domain Routing) und hat sich im gesamten Internet durchgesetzt. Kauft man also heute IP-Adressen so kauft man diese von einem Provider, welche seinen eigenen IP-Bereiche mit Hilfe von Subnet-Masken in Teilnetze aufteilt.

IP $\wedge$ subnetmask	=	subnetz
IP $\wedge\neg$ subnetmask	=	host

Abb. 11: Berechnen von Netz- und Host-Anteil einer IP-Adresse

Für jede Subnet-Maske gilt, wenn man sie binär betrachtet: eine 1 steht für der Netz-Anteil, eine 0 für den Host-Anteil der IP, auf der die Maske angewendet wird. Es dürfen 0 und 1 nicht gemischt werden. Oder anders: Auf eine 0 darf keine 1 folgen!

Gültige Subnetmasken sind also (binär):

- $11111111.11111111.11110000.00000000_2$
- $11111111.11111111.11111111.10000000_2$
- $11111111.00000000.00000000.00000000_2$

Ungültige Masken sind:

- $11111111.11110011.11000000.00000000_2$
- $11111111.11111111.11111111.10100000_2$

Durch diese Vereinfachung lässt sich eine Subnet-Maske auch einfach durch die Anzahl der 1en darstellen (Präfix-Notation):

- z.B. /24 für die Subnet-Maske 11111111.11111111.11111111.00000000 (255.255.255.0)
- oder /26 für 11111111.11111111.11111111.11000000 (255.255.255.192)

Statt 192.168.1.254 255.255.255.0 kann also auch 192.168.1.254/24 geschrieben werden. Der letzte Teil ist dabei die Subnet-Maske.

In jedem Teilnetz gilt:

- Die niedrigste Adresse (Host-Bits alle 0!) beschreibt das Netz selbst
- Die höchste Adresse (Host-Bits alle 1!) bildet die Broadcast-Adresse um alle Rechner gleichzeitig zu erreichen

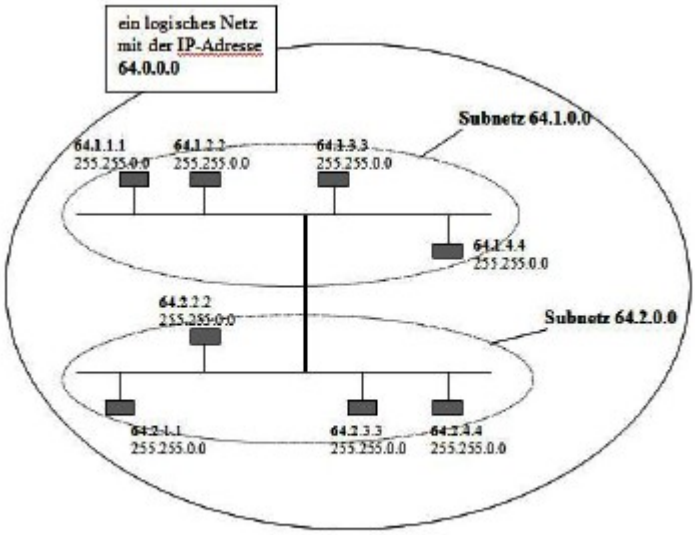


Abb. 12: 64.0.0.0/8 nach außen, 64.0.0.0/16 innen

IPv6 (IPng – Internet Protocol next generation)

Wie kommunizieren zwei IPv6-Inseln über ein IPv4-Netzwerk (z.B. das bisherige Internet)

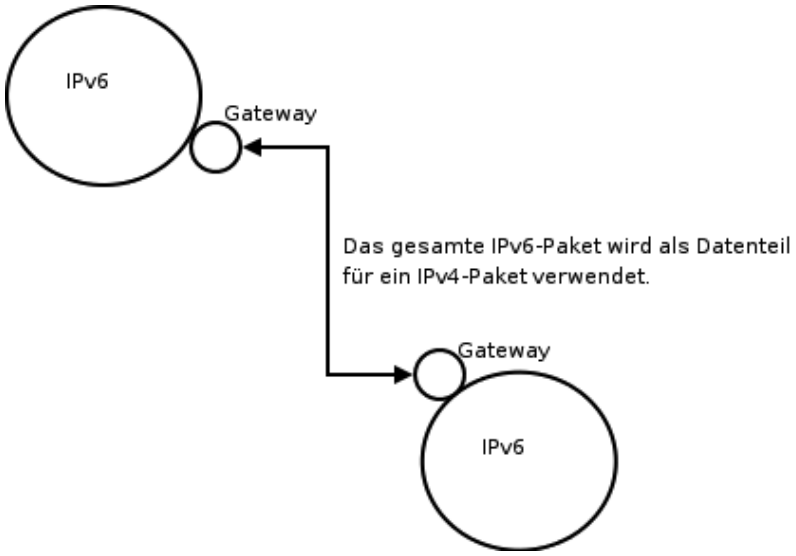


Abb. 13: Verbindung zwischen zwei IPv6-Inseln über ein IPv4-Netz

Um dem Mangel an IPv4-Adressen zu entgegnen, führte man zuerst das NAT-Protokoll ein. Statt wie bisher (auch an der BA) jeden Rechner direkt mit seiner eigenen IP mit dem Internet zu verbinden, verbindet man diese nur noch mit einem NAT-Server, welcher den Rechnern interne (private) IPs zuordnet. Da andere Institutionen die selben internen IPs verwenden könnten, muss den NAT-Server diese IPs gegen seine eigenen Internet-IP austauschen / maskieren.

Aufbau von IPv6 / IPng (IP next generation)

$F_{16} = 1111_2 = 16_{10}$

$FFFF_{16} = 1111\ 1111\ 1111\ 1111_2 = 65535_{10}$

$FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF = 65535^8_{10} = (2^{16})^8_{10} = 2^{128} = 3*4*10^{38}_{10}$ = Mehr IP-Adressen als Atome auf der Erdoberfläche!!

Problematisch die Trennung durch Doppelpunkte, da wir den schon für verschiedene Sachen verwenden, z.B. die Abtrennung des Ports oder in diversen Kommandos. Darum muss eine IPv6-Adresse bei der Verwendung immer in eckige Klammern eingepackt werden.

http://[xx:xx...:xx]:8080

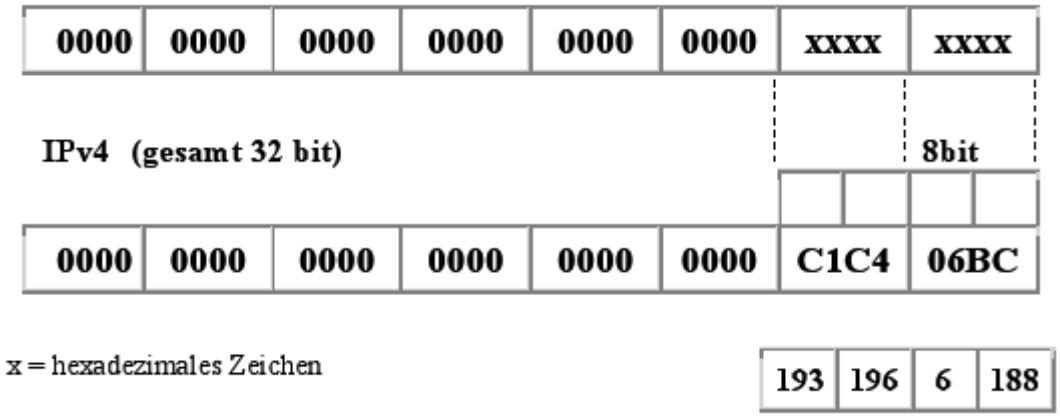


Abb. 14: Aufbau von IPv6-Adressen

- Man sieht auch gut, wie eine IPv4-Adresse in eine IPv6-Adresse verwandelt wird. Man füllt von Links einfach mit Nullen
- Groß- /Kleinbuchstaben egal
- Nuller-Blöcke dürfen einmal mit :: abgekürzt werden
- MAC-Adressen bleiben erhalten (s. Autokonfiguration)

Adresspräfixe:

Adresspräfixe werden wie bei IPv4 mit einem Schrägstrich und der Anzahl der relevanten bits (von links!) in Dezimalnotation angegeben, z.B:
`fa79::/64` bedeutet folgenden Adressbereich: `fa79:0:0:0:*.*.*.*`

Aufbau von IPv6-Adressen

Bei IPv6 gibt es anders wie bei IPv4 keine flexiblen Netzmasken mehr! Die IPv6-Adressen haben immer 64 Bit „Netzanteil" und 64 Bit „Interface-Anteil".

16 bit	16 bit	16 bit	16 bit	16 bit	16 bit	16 bit	16 bit
48 bit			16 bit	64 bit			
Netzwerkadresse			Subnet	Interface ID			

Durch diesen immer gleichen Aufbau wird das Routing effizienter, da das Auswerten von Netzmasken entfällt.

Der Netzwerk-Anteil ist fest vom Provider vorgegeben.
Mit den 16 bit „Subnet ID" kann man ca. 65 000 Subnetze einrichten.
Mit den 64 bit „Interface ID" kann man in jedem Subnetz 18 x 10^{18} (18 Exa) Interfaces (Rechner, Mobiltelefone, Kühlschränke, ...) addressieren.

- Netzwerkadresse von IPv6 nur bei Kommunikation mit anderen Netzen nötig, im internen Netz braucht man diesen Teil nicht.

Vergabe von IP-Adressen, SAC (Stateless Autoconfiguration)

Bei IPv6 können zwar Adressen auch statisch vergeben werden, die dynamische Vergabe von IP-Adressen ist aber Standard! Hierbei wird der „Interface"-Teil der IP-Adresse aus der MAC-Adresse generiert:
Im ersten Byte der MAC-Adresse wird eine 2 addiert, und zwischen Byte 3 und 4 wird fffe eingefügt:

MAC-Adresse:	`00:04:e2:09:da:ad`
Generierte Interface ID:	`02`**`04:e2ff:fe`**`09:daad`

Die MAC-Adressen werden mit einem Protokoll (amer.) „Neighbor Detection" (brit.) „Neighbour Detection" das mit Multicasts arbeitet ermittelt. Dies ist der Nachfolger von arp. (s. auch Kapitel Neue IPv6 Kommandos). Mit der generierten Interface ID wird mit dem Präfix fe80::/64 eine „Link Local-Adresse" gebildet. Mit dieser wird der nächste Router nach Subnetpräfixen abgefragt. Dann wird aus Interface ID und Network ID die endgültige IP-Adresse zusammengebastelt. IPv6 kennt auch das DAD (Duplicate Adress Detection), ein Protokoll, das über Multicastadressen abfrägt, ob eine IP-Adresse doch schon zufällig belegt ist.

- *DHCP ist also häufig überflüssig*
- Keine statischen Adressen mehr, nur noch weltweit eindeutige, dynamische Adressen

- Generierte Adressen werden per Broadcast dem Router mitgeteilt
- Jetzt neu: DHCPv6 um Probleme (experimentell) mit sich wechselnden IPs (bei Austausch von Netzwerkkarten) zu umgehen. Man denke nur an einen großen Provider, der eine Netzwerkkarte austauscht und dass es gut einen Tag gehen kann, bis weltweit alle DNS-Server die daraus folgende IP-Änderung mitbekommen.

Neues Headerkonzept

IPv6 hat ein sehr ausgefeiltes header-Konzept mit mehreren optionalen Headern, u.a. sind auch header für kryptographische Verfahren vorgesehen.

Durch diese mehrstufige Headerschichten erreicht man folgendes:

- Nur die allerwichtigsten Daten (z.B. Quell- und Zieladresse) stehen im ersten Header
- Die aktiven Komponenten können daher IPv6-Pakete sehr schnell analysieren und weiterleiten.
- Alle anderen Informationen sind in Folgeheadern abgelegt. Eine Applikation (z.B. Kryptoverfahren) die diese header benötigt, muss dann entsprechend mehr untergeordnete header analysieren.

Integrierte Security-Standards (IPsec)

Authentication-Header:

Überprüfung der Authentizität des Senders und der Unveränderheit der Daten

ESP-Header (encapsulation security protocol):

Verschlüsselung der Daten selbst

IPv4 Tunneling:

Zwei IPv6 Netzwerke können heute schon über das Internet mit IPv4 gekoppelt werden. Hierzu werden die IPv6 Daten mitsamt IPv6 Header als „Nutzdaten" in IPv4 verpackt. 6 zu 4 Tunneladressen haben das Format: `2002::/16`

QoS

Durch QoS-Angaben können Pakete priorisiert werden. Pakete mit gleichem IP-Header können zu sogenannten „Flows" zusammen gefasst werden. Router werten die untergeordneten Header eines Flows nur einmal aus, alle nachfolgenden Pakete des Flows werden dann so behandelt.

Rooming

IPv6 wurde Roomingfähig entwickelt. Man kann durch unterschiedliche Funknetze wandern, ohne die Kommunikation zu unterbrechen.

Fazit:

IPv6 ist eine völlig neue Technologie und hat mit IPv4 nur noch den Namen IP gemeinsam. Viele Konzepte, die in IPv4 in Zusatzprotokollen definiert werden mussten, sind jetzt fester Bestandteil von IPv6. Ein Umstieg auf IPv6 sollte man so lange es geht hinauszögern, bis alle Anfangsschwierigkeiten bereinigt sind. Neuanschaffungen im Netzwerkbereich sollten aber in jedem Fall IPv6-fähig sein! Vor allem im asiatischen Raum existieren derzeit schon IPv6-Inseln im Internet.

IPv4	BSD	Linux	Windows
ping	ping6 –I „if"	ping6 –I „if"	ping6
arp –a	ndp –a	ip neigh show	???
traceroute	Traceroute6	traceroute6	tracert6 ???

Routing

Routing im LAN (Routing nach Graphen)

Allgemein:

- Ermittlung des Pfades vom sendenden Host zum empfangenden Host
- Der Weg kann über mehrere Teilnetze erfolgen
- Es kann mehrere mögliche Wege geben
- Wahl des günstigsten Weges nach folgenden Kriterien
    - Kürzeste Wartezeit
    - Beste Leitungsauslastung
    - Geringe Zwischenknoten
    - Geringe Kosten
    - Geringe Fehlerrate
    - Policy-Aspekte

Abstraktion des Netzes auf einen Graphen:

Knoten	= Router
Kanten	= physikalische Verbindung
Wert der Kante	= Kosten

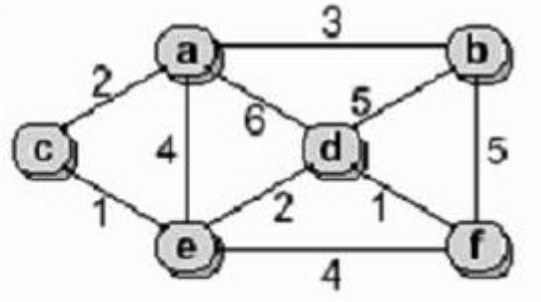


Abb. 15: Abstraktion eines LANs auf einen Graphen

- Kanten = Edge E
- Knoten = Vertex V
- Irgendein Netz (z.B. Straßenbahn, Straßen, Netzwerk, etc. pp.)

Dijkstra-Algorithmus:

1. Ausgehend von einem Punkt A: Ermitteln der Kosten, die direkten Nachbarn zu erreichen
2. Der Punk selbst hat immer unendlich hohe Kosten.
3. Alle nicht direkt erreichbaren Punkt werden auch mit „unendlich“ bewertet.
4. Von allen unbesuchten Punkten: Suche den billigsten Punkt und besuche ihn.
5. Untersuche seine Nachbarn und rechne aus, was es kostet, diese von A aus zu besuchen.

6. Wenn eine sich somit eine billigere Verbindung ergibt als bisher, schreib dir das auf. (Alte Werte löschen)
7. Gehe zurück nach A
8. Zurück zu 4, bis alle Punkte von A aus besucht wurden!

Prim-Algorithmus:

1. Fange z.B. bei Punkt A an
2. Laufe immer die billigste Kante entlang
3. Wird ein Knoten so nicht erreicht, probiere ihn auf billigste Weise von A zu erreichen (Rekursives Absuchen des Graphen)

Kryskal-Algorithmus:

Informall:

1. Sortiere alle Kanten nach Billigkeit aufsteigend
2. Pass auf, dass dabei keine Schleifen entstehen

Formaler:

1. Suche die billigste Kante
2. Erstelle einen Teilgraphen
3. Suche die nächste billigste Kante
4. Füge sie dem Teilgraphen hinzu, aber nur, wenn sich dadurch keine Schleife bildet.
5. Usw.

Routing in großen Netzen (z.B. Internet)

- Pflege von Tabellen, in denen die bekannten Hosts und Wege gespeichert werden
- Unter Windows ist das „Standard-Gateway“ (falscher Begriff!!) der Standard-DNS-Server / Router der gefragt wird, wenn Windows nicht mehr weiter weiß.
- Routing nach Graphen nicht passend, da viel zu rechenintensiv! Bei n Knoten: $O(n^2)$ und mehr!
- => Zum Teil sehr ausgefeilte Algorithmen um auch Ausfälle von Strecken zu erkennen und zu umgehen
- => Zentrale Router des Internets sind Hochleistungsrechner mit mehreren Millionen Einträgen in einer Routing-Tabelle.

Layer 4 des OSI-Modells: Transport layer (TCP und UDP)

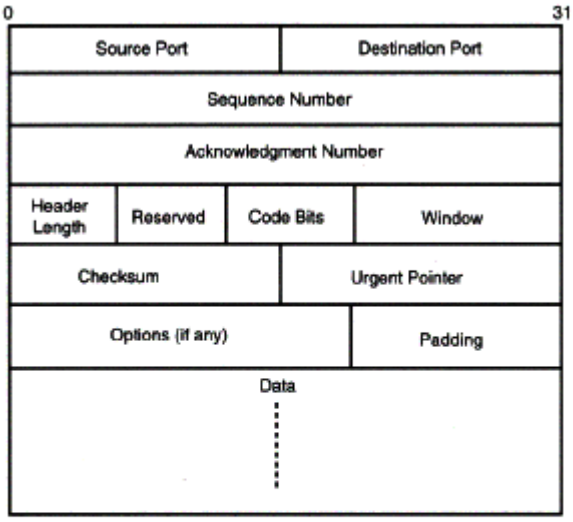
Sockets und TCP/IP-Protokolle

- Hier werden Sockets definiert, um auf einem Rechner (mit einer IP-Adresse) mehrere Dienste ansprechen zu können. (Statistisches Multiplexing und Demultiplexing von Anwendungsprozessen!)
- Es wird aber nicht definiert, welche Dienste an welchem Port hängen. Das ist Aufgabe von Layer 7!
- Socket = IP-Adresse + Portnummer (Servicenummer) = Weltweit eindeutige Identifizierung eines Dienstes auf einem Rechner
- Dazu zwei Protokolle: TCP und UDP
- Gleiche Funktionalität

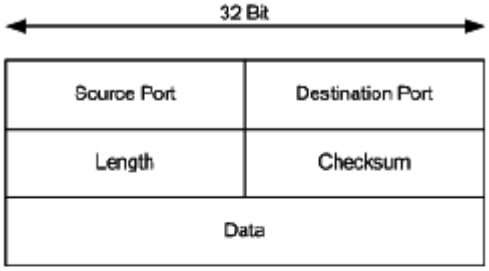
- TCP kontrolliert alles (Überprüfung, Bestätigung, Sortieren der Empfangspakete etc.) - Langsam und sicher (Verbindungsorientiert)
- UDP kontrolliert nichts => schneller (VoIP, VideoStreams bis zu 5% Paketverlust als Toleranz) – Schnell und unsicher (Verbindungslos)

Anwendung	Anwendungsprotokoll (Layer 7)	Zugrunde liegendes Transportprotokoll (Layer 4)
E-Mail Versand	SMTP	TCP
E-Mail Empfang	POP / IMAP	TCP
Web	HTTP	TCP
Filetransfer	FTP	TCP
Remote file Server	NFS	UDP
Streaming Multimedia	diverse	UDP
Internet-Telefonie	diverse	UDP
Netzwerkmanagement	SNMP	UDP
Routing-Protokoll	RIP	UDP
Namensauflösung	DNS	UDP

TCP-Header



UDP-Header



Im TCP-Header sind u.a. enthalten:

- Absender-Port
- Empfänger-Port
- Sequenznummer
- Quittungsnummer
- Fenstergrösse
- Prüfsumme

Abb. 16: Vergleich TCP-Header mit UDP-Header

Socket-Programmierung

Mit connect(), close(), read() und write() kann man nun zwischen zwei Endpunkten (Rechner + Dienst) kommunizieren (Strings austauschen).

Dabei präsentiert sich das Netzwerk beiden Endpunkten wie eine Datei (die berühmte „Everything is a file“-Abstraktion aus UNIX). Die Vorgänge sind exakt die Gleichen: Öffnen, Lesen / Schreiben, Schließen.

Aktive Komponenten

Begriffe

- Internetworking = Netzwerkübergreifende Kommunikation
- Homogenes Internetworking = selbe Protokolle etc. in den verschiedenen Netzen
- Heterogenes Internetworking = Verschiedene Protokolle der Teilnetze

Die Aktiven Komponenten der einzelnen Layer:

Layer 4-7:	Gateways, Firewall-Technologien
Layer 3:	Router/ L3-Switches
Layer 2:	Switches
Layer 1:	Repeater/Hub (homogene Netzwerke)

Abb. 17: Die aktiven Komponenten der einzelnen Layer

Layer-1-Kopplung (Repeater / Hubs)

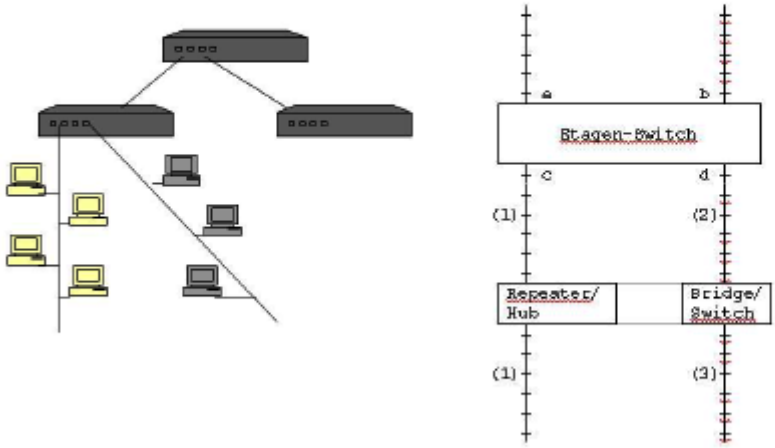
- Repeater = Verstärker
- Hub = Repeater mit mehreren Ausgängen
- Layer-1-Kopplung zunehmend unwichtiger, da nur physikalische Weiterleitung (Verstärkung, Auffrischung)
- Auf Layer 1 sitzen auch: Modems (Analog, ISDN, DSL) sowie passive Komponenten wie Patchdosen etc.

Layer 2 - Switches

- Ein Switch koppelt mehrere LANs zu einem logischen Netz (Verschiedene Übertragungsmedien und -Geschwindigkeiten)
- Hierzu zählen auch AccesPoints für WLAN
- Lastentrennung (Funktioniert wie eine Telefonvermittlung)
- => getrennte Kollisionsdomänen => größere Entfernungen möglich, indem man den Switch einfach in die Mitte setzt.
- Da Layer 2: Nur Arbeit mit MAC-Adressen im ARP-Puffer
- **Cut through:** So schnell wie möglich von einem Strang zum Anderen (Keine Prüfungen)
- **Store and Forward:** Zwischenpuffer, Prüfung der Pakete auf Konsistenz, Verwerfen von kaputten Paketen
- Teurere Switches können beides und lassen sich konfigurieren, wann was eingesetzt wird

Netzaufbau mit Switches

Jeder Port eines Switches bildet ein eigenes Netzsegment; die Segmente können gekoppelt werden indem die Ports durchgeschaltet werden.
Sternförmige Verkabelung bei Ethernet: „Zwei Strahlen" des Sterns bilden dann jeweils einen Ethernet-„Bus"



Kollisionsdomänen: (1) und (1) bilden eine Kollisionsdomäne
(2) und (3) sind jeweils eigene Kollisionsdomänen

Abb. 18: Netzaufbau und Kollisionsdomänen mit Switches

Layer 3 - Router

- Ein Router koppelt heterogene Netzwerksegmente (überbrückt verschiedene Protokolle auf Layer 1 und 2)
- => Routing zwischen heterogenen Netzwerken
- Zwei Aufgaben: Routing und Sicherheit (IP-Filter, Inhaltsfilter etc.)
- Höhere Isolation (Broadcast-Meldungen werden i.d.R. nicht weitergeleitet)
- Eigentlich ein Rechner mit zwei (oder mehr) Netzwerkkarten (je nach Hersteller mit proprietären Betriebssystemen oder einem Unix)
- Bei großen Routern: Konfiguration per Terminal (Laptop) oder Webinterface
- Am Besten komplette Abkopplung des Gerätes (kein Webinterface), so dass es nur seine eigene Aufgabe erfüllt, mehr nicht! (Sicherheitsrisiken minimieren)

Switches auf höheren Ebenen

- Abgespeckte Router, welche nicht Routen, sondern nur Sicherheitsaspekte bedienen
- Deutlich schneller als herkömmliche Switches (L3-Switches können z.B. keine verschiedenen Netzarten miteinander verbinden)
- Für homogene Router
    - 1. Auf Layer 3: Wer darf mit wem?
    - 2. Auf Layer 2: Weiterleitung der gültigen Pakete

- L4-Switches können auch Ports (und damit Dienste) sperren
- Je höhere der Switch im OSI-Modell sitzt, desto besser kennt er die Datenpakete, desto mehr Filteroptionen bietet er. (Ab Layer 4 sagt man auch Gateway, da der Switch meist zwischen zwei Netzen sitzt und hier Filter- Konverterfunktionen auf höheren Ebenen übernimmt).

Typischer Aufbau eines Netzwerkes mit Firewall

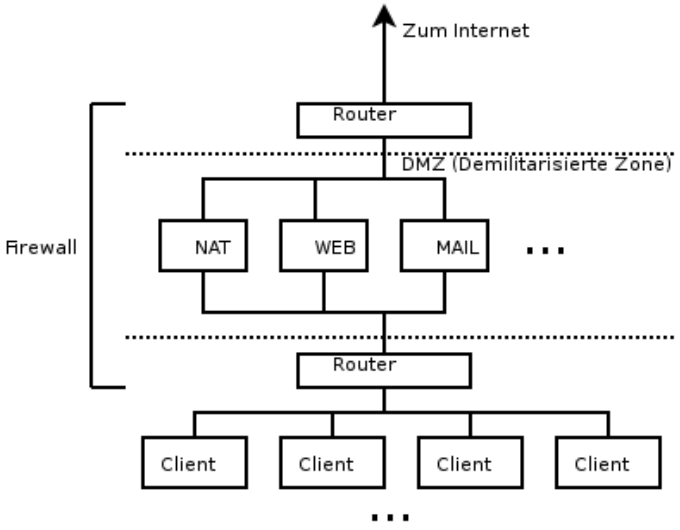


Abb. 19: Typische Serverlandschaft.

In der DMZ gelten weniger strenge Bedingungen, so dass die Server auch öffentlich zugänglich sind.

Virtuelle LANs

Durch den Einsatz von VLAN-Switches macht man den logischen Aufbau des Netzes vom physikalischen Aufbau unabhängig. Also egal, in welche Netzwerkdose man seinen Rechner schickt, er gehört immer noch dem selben „Teilnetz“ an. Ein Verwaltungsrechner bleibt ein Verwaltungsrechner, auch wenn er in der Produktionshalle ans Netz geht.

Bisher wurden Netzsegmente auf Layer 2 definiert, indem man festlegte, das alles was einem bestimmten Port des Switches folgt (ein Strahl) zu einem Teilnetz gehört.

Jetzt arbeiten wir auch auf Layer 3 und bilden die Teilnetze allein durch die IP-Vergabe. Dazu stehen uns alle Mittel, die wir aus Layer 3 kennen zur Verfügung.

VPN – Virtual Private Network

Bei VPN handelt es sich um Technologien auf den Ebenen 2 bis 4, welche dazu gedacht sind, verschlüsselte „Tunnel“ in öffentlichen Netzen einzurichten. z.B. kann man eine Niederlassung mit der Zentrale über einen VPN-Tunnel im Internet verbinden. Die gesamte Kommunikation zwischen den beiden stellen erfolgt im öffentlichen Raum (Internet) und könnte so theoretisch von jedermann abgehört werden. Allerdings werden die Daten auf mehreren Ebenen verschlüsselt, so dass diese abgehörten Informationen wertlos sind.