

ToC

- 1. Codierung von Information
- 2. Netzwerktopologien
- 3. Netzwerkprotokolle
- 4. Das ISO/OSI-Referenzmodell
- 5. Der TCP/IP-Stack

I. Codierung von Information

Digitale Information ist im *Binärsystem* kodiert, in diesem lassen sich mit k Stellen ("bits") Zahlen von 0 bis 2^k-1 darstellen.

1001₂ = 1(2³) + 0(2²) + 0(2¹) + 1(2⁰) = 2³+2⁰ = 8+1 = 9

I.1: Addition im Binärsystem

1001 |
0011 | +

0110
1100

I.2: Umwandlung zwischen Dezimal, Hex und Binär

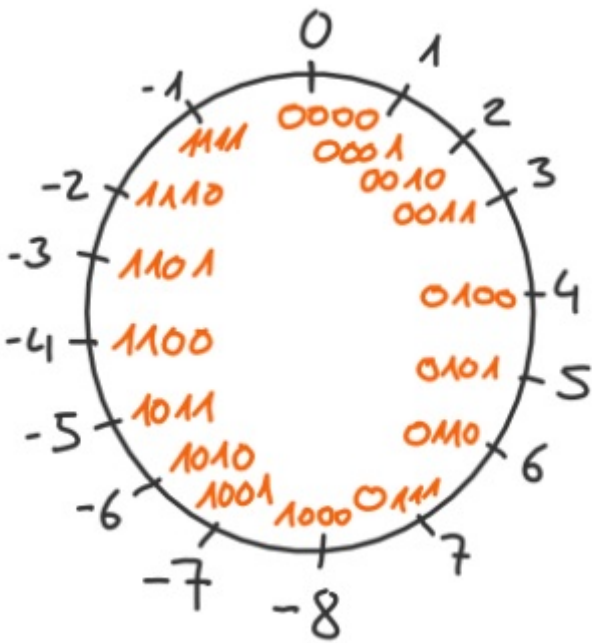
Merke: Eine Stelle Hexadezimal ist equivalent zu vier Stellen Binär.

Bin	Hex	Dec	Bin	Hex	Dec
0000	0	0	1000	8	8
0001	1	1	1001	9	9
0010	2	2	1010	A	10
0011	3	3	1011	B	11
0100	4	4	1100	C	12
0101	5	5	1101	D	13
0110	6	6	1110	E	14
0111	7	7	1111	F	15

1100 0001 0001 0101 0111 0101 0101 1101 1111 0010 1100 0011 0110 =
000C 1157 55DF 2C36 =
06 × 16⁰⁰ + 03 × 16⁰¹ + 12 × 16⁰² + 02 × 16⁰³ +
15 × 16⁰⁴ + 13 × 16⁰⁵ + 05 × 16⁰⁶ + 05 × 16⁰⁷ +
07 × 16⁰⁸ + 05 × 16⁰⁹ + 01 × 16¹⁰ + 01 × 16¹¹ +
12 × 16¹² + 00 × 16¹³ + 00 × 16¹⁴ + 00 × 16¹⁵ =
3396766521044022.

I.3: Negative Zahlen

Mit k bits können aber auch Zahlen von -2^{k-1} bis +2^{k-1}-1 dargestellt werden. Diese Zahlen werden *signed* genannt, während die anderen (von 0 bis 2^k-1) als *unsigned* bezeichnet werden.



II. Netzwerktopologien

II.1 Point-2-Point

Vor- & Nachteile

Vorteile:

- Bei jeder Verbindung wird die maximale bandbreite des Datenträgers ausgenutzt.
- Leicht Erweiterbar
- Leicht Verständlich
- Leichtes Traceback von Fehlern
- Kein Routing benötigt

Nachteile:

- Max. 2 Nodes (Knotenpunkte)
 - Mit mehr als 2 Nodes sind es andere Topologien (-> Routing wird Benötigt).
- Wird meist nur bei geringer Entfernung der Nodes Genutzt.
- Wenn die Verbindung unterbrochen wird ist das Netzwerk kaputt.

Unterschiede zur Peer-to-Peer (P2P) topologie

P2P: Mehrere gleichberechtigte Nodes verbunden; Datentransfer aber Point-to-Point Point-to-Point: 2 Nodes.

Kommunikationsweisen

Simplex

Ein Sender der nur Sendet und ein Empfänger der nur Empfängt

Half-Duplex

Beide Nodes können Senden (time-Multiplex) während die Andere Node Empfängt, und sind nicht an Feste Sender/Empfänger-Rollen Gebunden.

Bsp:

- USB-A -> USB-B
- Walkie-Talkie.

Full-Duplex

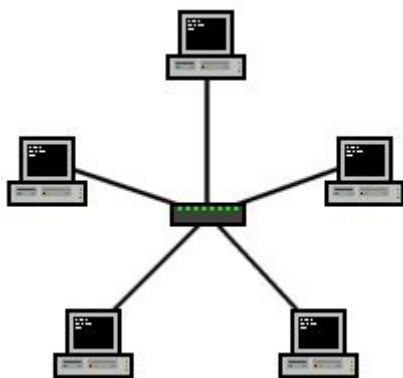
Beide Nodes können Gleichzeitig Senden & Empfangen. Bsp.

- USB-C -> USB-C
- Telefon

II.2 Stern-Topologie

Besteht aus Einer Zentralen Verbindungsstelle (Hub/Switch) mit der Sich alle anderen Teilnehmer (Nodes/Knoten) sternförmig direkt verbinden. Diese Technologie ist sehr weit Verbreitet. z.B.

- (W-)LAN (Local Area Network; Lokales Netz)



Wenn man Die Hubs mit Einem Zentralem Hub verbindet, entsteht eine sog. *Baum Topologie*

II.3 Bus-Topologie

Alle Teilnehmer sind mit *einem* Kabel verbunden, das Zentrale medium ist der Bus. Der Bus ist bei kleinen Systemen auf einer Leiterplatte (PCB).

Der Anschluss zwischen den Geräten und Segmenten des Kabels erfolgt über T-Stücke. (Veranschaulichung: Unter Unix-artigen (z.B. Linux) Betriebssystemen kann man den Befehl `tee` verwenden, um Datenausgaben sowohl auszugeben, als auch in eine Datei umzuleiten).

Anwendung des 'tee' Befehls auf Ubuntu (Linux); Das '|' leitet den Output eines Programmes (in diesem Fall `java Hiragana.java`) weiter.

```
akb1152@akb-lnx-ubuntu: ~/Desktop
akb1152@akb-lnx-ubuntu:~/Desktop$ java Hiragana.java | tee File.out

====
ひらがな
Hiragana
====

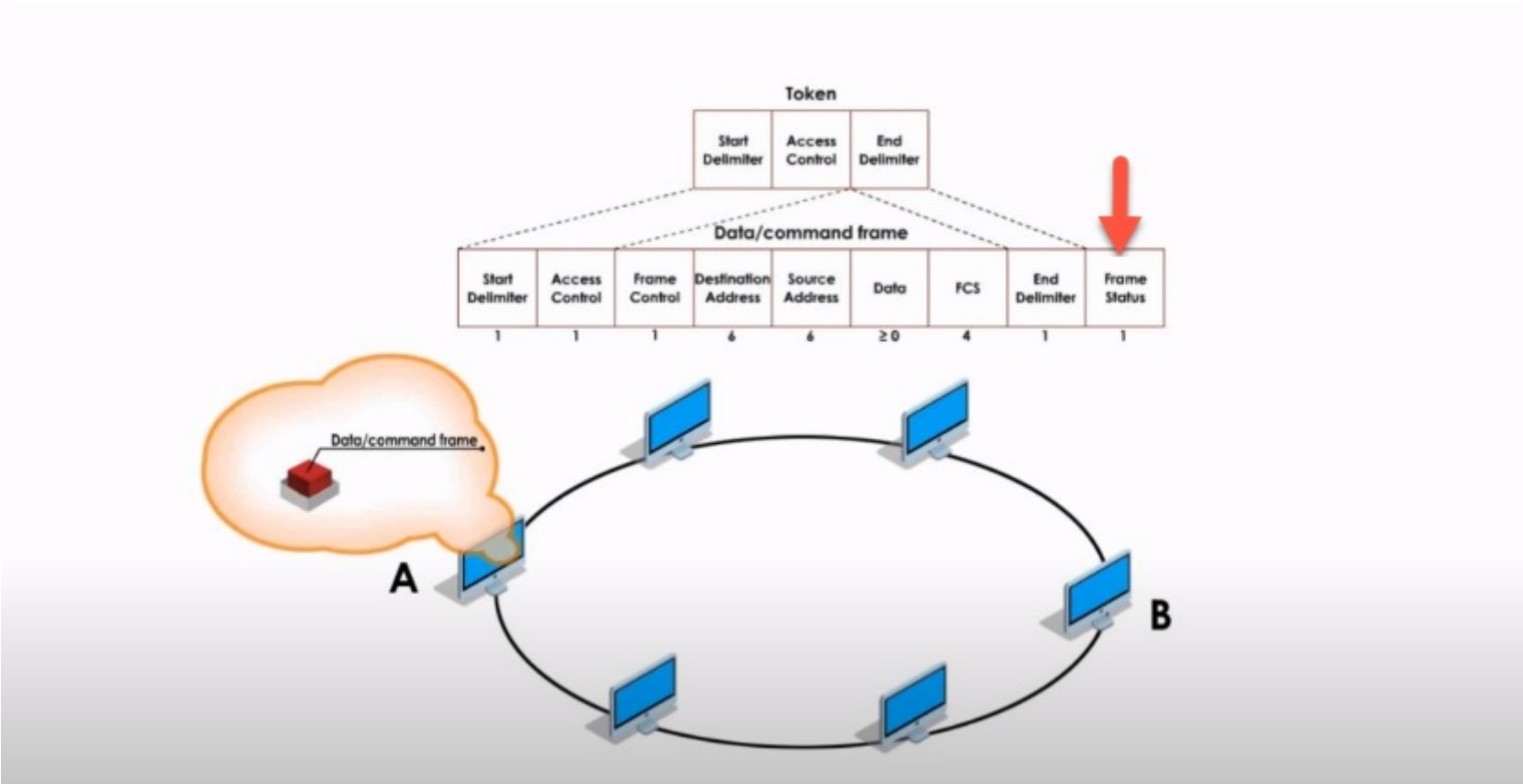
[3041]: あ
[3042]: あ
[3043]: い
[3044]: い
[3045]: う
[3046]: う
[3047]: え
[3048]: え
[3049]: お
[304a]: お
[304b]: か
[304c]: が
[304d]: き
[304e]: ぎ
[304f]: く
```

Bevor ein Gerät senden darf muss es, über eine separate Leitung, eine Anfrage an den *Bus Arbiter* (ein Richter der schnell entscheidet wer richtig und falsch liegt) stellen.

II.4 Token-Ring

Der Token Ring ist eine Vernetzungstechnik, die in den frühen 1980ern von IBM¹ als direkter Konkurrent zu Ethernet für Local Area Networks (LAN) entwickelt, allerdings hat er sich, durch einige Hürden in seinem grundlegenden Aufbau, nicht umgesetzt, aber wurde in 1989 as **IEEE 802.5** standardisiert.

¹IBM: International Business Machines Corp. (est. 1911)



Wenn ein Rechner (A) einem anderem (B), etwas senden möchte, muss er darauf warten, dass der Token ihn erreicht. Anschließend sendet A einen sog. *Data/Command Frame* auf den Ring, Dieser hat u.a. eine *Destination Address* und einen *Frame Status*. Jeder Rechner zwischen A und B auf Dem Ring überprüft ob die *Destination Address* seiner eigenen gleicht, wenn ja dann kopiert er die Daten auf die Festplatte und setzt die Flag² *Frame Status* auf 1, was signalisiert, das das Paket bei B angekommen ist und dieser die Anfrage nun bearbeitet; A sendet dann einen Leeren Token wieder Auf den Ring.

²Flag: Bit der etwas aussagt je nach dem Wert den er Hat. (in diesem Fall Byte-Flag: Byte statt Bit)

Aufbau des Tokens:

Start-delimiter	Access Control	End-delimiter
1 byte (8-bits)	1 byte (8-bits)	1 byte (8-bits)

- Start-delimiter: Signalisiert den Anfang einer Nachricht
- Access Control: Durch den Aufbau Festgelegt
- End-delimiter: Signalisiert das Ende einer Nachricht

Aufbau des Command/Data Frames

Start-delimiter	Access Control	Frame Control	Destination Address	Source Address	Data	CRC	End-delimiter	Frame Status
1 byte	1 byte	1 byte	6 bytes	6 bytes	bis zu 4,5 kilobytes	4 bytes	1 byte	1 byte

- Start-delimiter: Signalisiert den Anfang einer Nachricht
- Access Control: Durch den Aufbau Festgelegt
- Frame Control: Durch den Aufbau Festgelegt
- Destination Address: Adresse des Empfängers
- Source Address: Adresse des Senders
- Data: Daten
- Cyclic Redundancy Check (CRC): Zahl die dafür zuständig ist Fehler in den Daten zu erkennen
- End-delimiter: Signalisiert das Ende einer Nachricht
- Frame Status: Signalisiert ob das Paket angekommen ist und ob es Kopiert worden ist.

III. Netzwerkprotokolle

Der Ablauf des Datenaustausches zw. Rechnern, die über ein Netzwerk miteinander Verbunden sind, wird in einer gemeinsamen Sprache/ Vereinbarung (=Protokoll) geregelt.

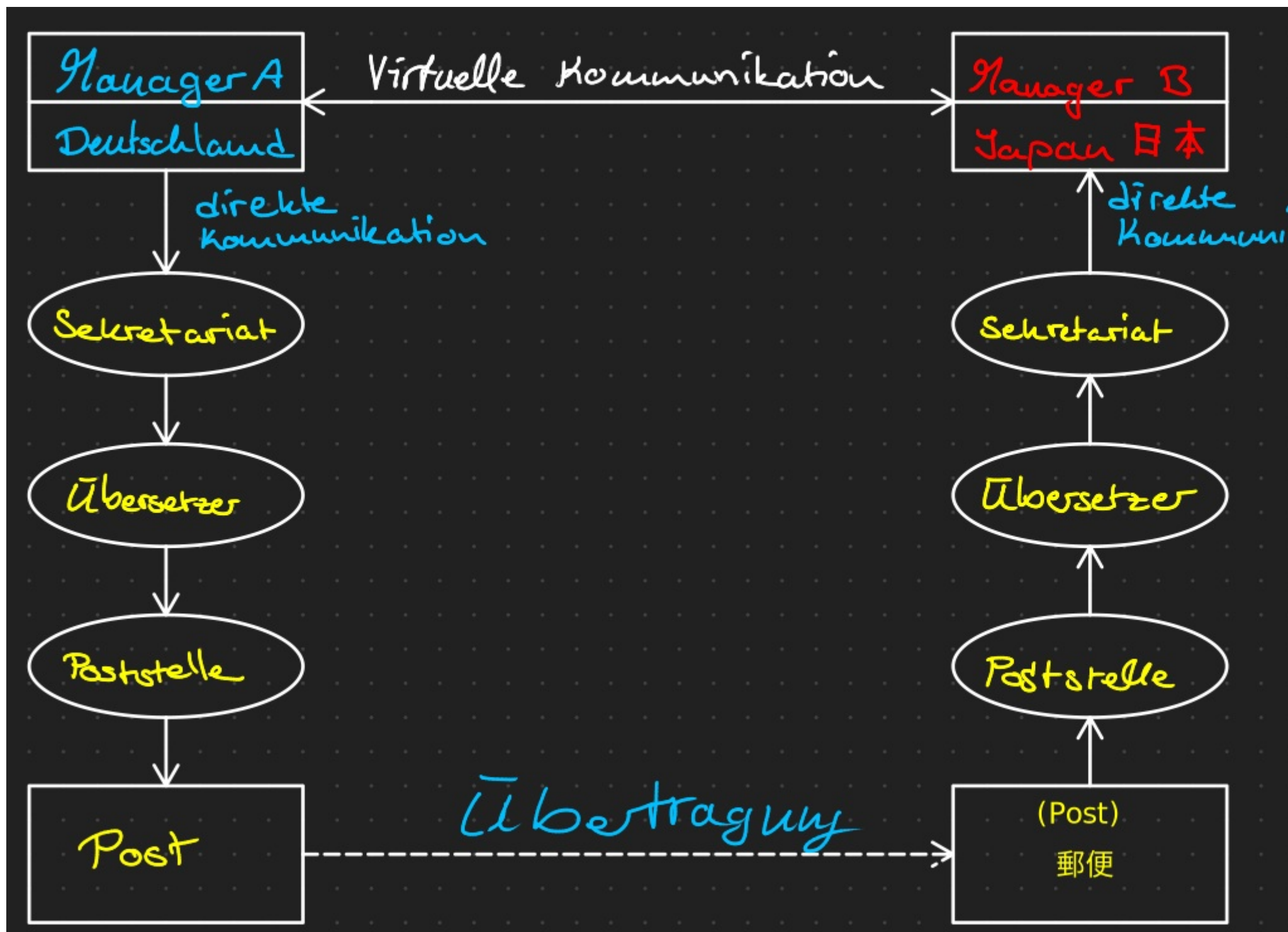
definition: "Protokoll":

Ein Protokoll ist die Gesamtheit aller Vereinbarungen, die Kommunikationsteilnehmer zur abwicklung der Kommunikationsvorgänge treffen.

Ein Netzwerkprotokoll legt u.a. fest:

- Schlüsselbegriffe (keywords) und deren Bedeutung(en).
- korrekte Nachrichtensyntax.
- korrekte Nachrichtenfolge.
- Wie man Verbindungen Auf- & Abbaut (+ Verbindungskonfirmation)
- Wie man mit Fehlern und Seltenen Szenarien (Edge-cases) umgeht.

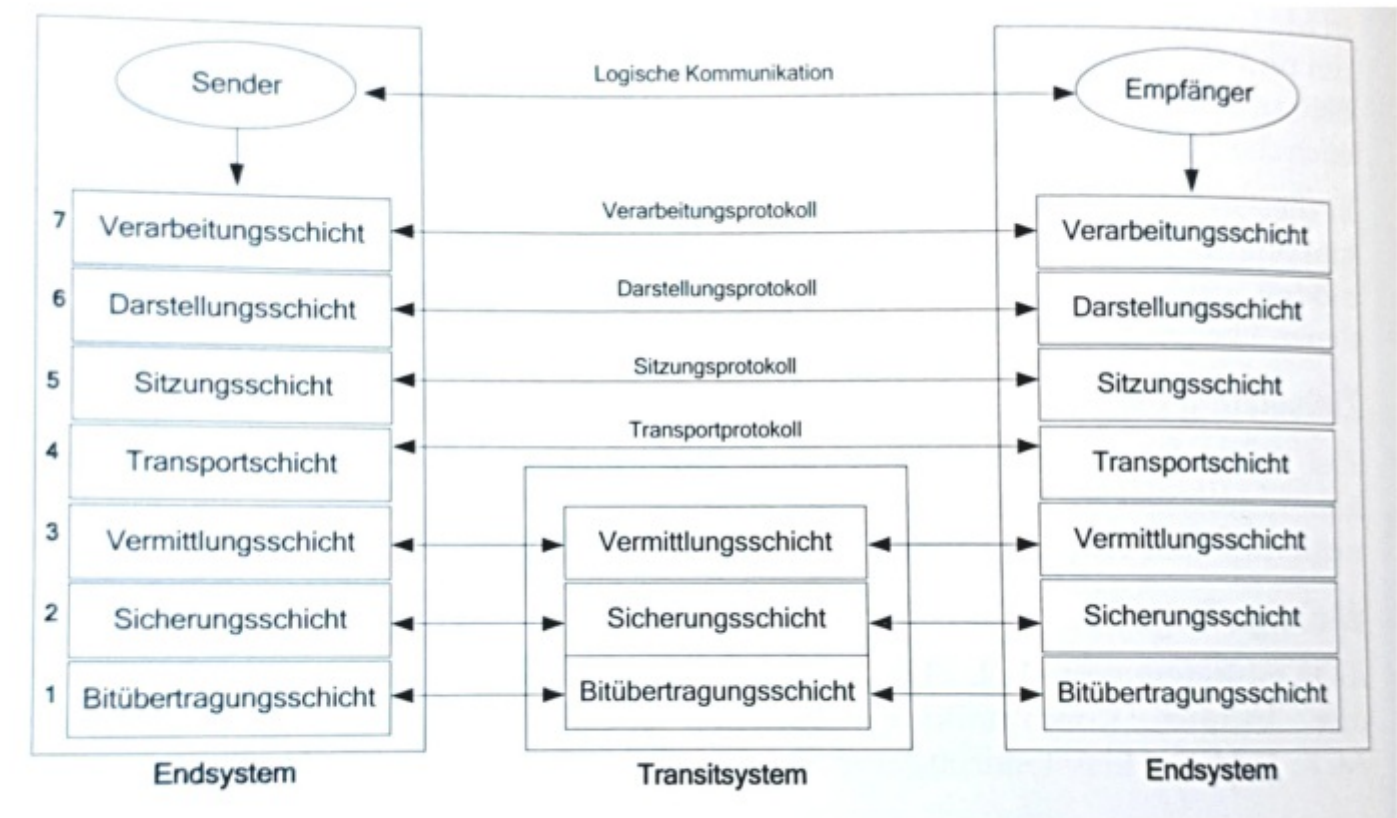
IV. Das ISO/OSI Referenzmodell



Die Kommunikation in Rechnernetzen kann mit einem Schichtenmodell effizient organisiert werden. Jede Schicht führt eine virtuelle Kommunikation mit dem Partner der Gleichen Schicht gegenüber. Jede Schicht implementiert eine *Schnittstelle*, so dass die Schichten unabhängig von einander sind.

Vorteile:

- Weniger Fehleranfällig
- Wiederverwendbarkeit
- Austauschbarkeit.



Bildquelle: Grundkurs Datenkommunikation, Mandl/Bakomenko/Weiß, Vieweg+Teubner-Verlag, 1. Auflage 2008

Schicht	Aufgabe	Veranschaulichung	
Anwendungsschicht	Schnittstelle zur eigentlichen Benutzeranwendung	Nachricht	anwendungsorientiert
Darstellungsschicht	Wandelt die systemabhängige Datendarstellung in eine unabhängige Form um (Sonderzeichen, Bilder, ...); Datenkompression; Verschlüsselung	Dolmetscher übersetzt in Firmensprache	
Sitzungsschicht	Synchronisiert die Abläufe auf den beteiligten Rechnern; behandelt Sitzungsabbrüche	Sekretärin vermerkt Postausgang	
Transportschicht	Zerlegung in Datenpaketen; Stauvermeidung; Reihenfolge der Pakete	Hauspostmitarbeiter klärt Versandweg	transportorientiert
Vermittlungsschicht	Wegesuche (Routing); Netzadressen	Kennzeichnung bei der Briefpost	
Sicherungsschicht	Fehlererkennung; Zugriff auf das Übertragungsmedium	Verteilstelle der Briefpost	
Bitübertragungsschicht	Bewerkstelligt digitale Bitübertragung in Form elektrischer Signale, optischer Signale oder elektromagnetischer Wellen	Transporter / Flugzeug	

V. Das TCP/IP Referenzmodell

Das *TCP/IP* modell stellt einen Defacto-Standard in der Rechkommunikation dar.

Schichten:

Schicht	OSI	Name	Erklärung	Bsp.
4	5-7	Verarbeitungsschicht	Umfasst protokolle die mit Anwendungen zusammenarbeiten	HTTP (Webseiten), IMAP (EMail), POP3 (EMail)
3	4	Transportschicht	Ermöglicht Peer-to-Peer Kommunikation ³	TCP, UDP
2	3	Internetschicht	Vermittlung von Paketen, Routing	IPv4, IPv6
1	1-2	Netzzugangsschicht	Nicht wirklich Spezifiziert	Ethernet (LAN), IEEE 802.11 (WLAN), FDDI (Glasfaser)

³ -> Verbindungsorientiert (zuverlässig): *TCP* (Dateitransfer)
-> Verbindungslos (unzuverlässig): *UDP* (Live-Streams, VoIP)