

# Referenzmodelle für Netzwerkprotokolle

OSI- und TCP/IP-Referenzmodelle



# Beweggründe für ein Referenzmodell

- Übergreifende Netzwerkkommunikation ist alles andere als trivial!
  - Verschiedenste **Hardware-Hersteller**
  - Verschiedenste **Geräte** (PC, Smartphone, Drucker, Smart Devices, ...)
  - Verschiedenste **Netzwerktypen** (kabelgebunden/drahtlos, stationär/mobil ...)
  - Verschiedenste **Anforderungen** an die Kommunikation (echtzeitfähig, zuverlässig, fehlertolerant, vertraulich, ...)
- Alle nutzen aber die **gleiche Netzwerkinfrastruktur!**
- Konsequenz: **Einheitlicher Standard/Referenzrahmen** für die geordnete Kommunikation erforderlich => **OSI-Referenzmodell!**

# Entwurfsaspekte einer einheitlichen Netzsoftware

- **Zuverlässigkeit** (Unsicherheiten durch Störungen, Ausfälle, ...)
  - Fehlererkennung (error detection)
  - Fehlerbehebung (error correction)
- **Weiterentwicklung**
  - Wachsende Netze => Adressierung / Namensgebung
  - Unterschiedliche Teilnetze => Internetworking
- **Ressourcenzuteilung**
  - Multiplexen (dynamisches Aufteilen von Bandbreiten)
  - Flusskontrolle (schneller Sender vs. langsamer Empfänger)
  - Netzüberlastung (Anfragen reduzieren)
  - Dienstgüte (konkurrierende Ansprüche – Echtzeitdaten vs. Durchsatz – ausgleichen)
- **Sicherheit**
  - Vertraulichkeit
  - Integrität
  - Verfügbarkeit
  - Authentizität

# Verbindungsorientierung und Zuverlässigkeit

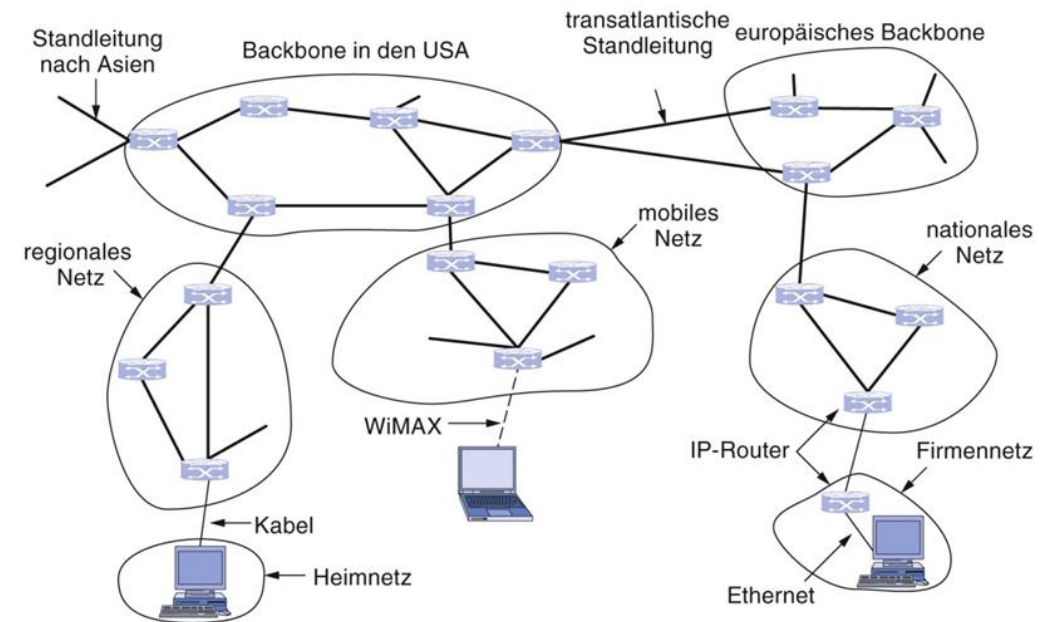
- Zwei Aspekte, stellvertretend:
  - **Verbindungsorientierung**
  - **Zuverlässigkeit**

# Verbindungsorientiert vs. verbindungslos

- **verbindungsorientiert**  
= analog zu Telefondienst
  - Verbindungsaufbau
  - Datenaustausch
  - Verbindungsabbau
  - Dienst funktioniert wie Rohr
- **verbindungslos**  
= analog zu Post
  - Jede Nachricht (Paket) enthält vollständige Adresse
  - Pakete werden unabhängig voneinander zugestellt



verbindungsorientierter Dienst



verbindungsloser Dienst (Datagrammdienst) im Internet

# Zuverlässigkeit

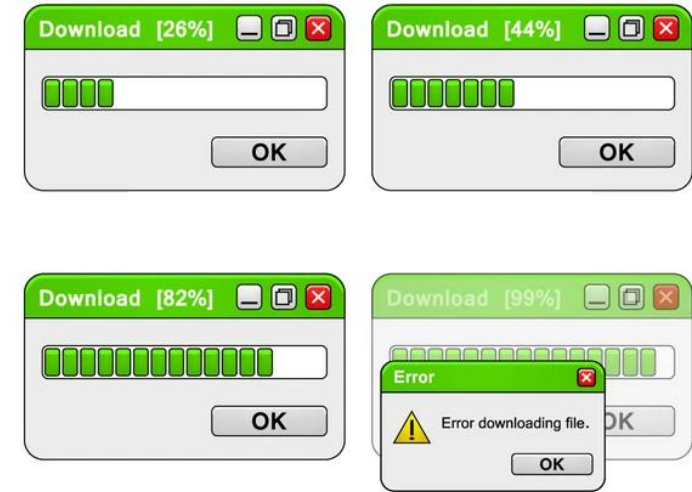
- Daten und Teile davon dürfen **nicht verlorengehen**
- Dienst versucht immer **korrekt zu übertragen**
- Anwendungsbeispiel: **Dateitransfer**
- Einbau von **geeigneten Kontrollmechanismen** die Zuverlässigkeit bei der Übertragung gewährleisten
- Nachteile:
  - Zeitverzögerung
  - Overhead

# Einteilung und Beispiele

	<b>verbindungsorientiert</b>	<b>verbindungslos</b>
<b>zuverlässig</b>	= Nachrichtensequenzen, Byteströme z. B. Dateitransfer	= bestätigter Datagrammdienst z. B. SMS
<b>unzuverlässig</b>	= unzuverlässige Verbindung z. B. Voice over IP (VoIP)	= unzuverlässiger Datagrammdienst z. B. Junk-Mail

# Weitere Aufgaben/Probleme

- **Übertragung** der Daten im **lokalen Netz** (Erzeugung der Signale, Zugriffskontrolle auf das Medium, ...)
- **Wegfindung** in großen Netzen (Internet)
- **Portionsweise Übertragung großer Datenmengen** (Beispiel: Videodatei mit mehreren GB)
- **Ausbalancierte Übertragungsgeschwindigkeit** („Datenflusssteuerung“: nicht zu schnell, nicht unnötig langsam)
- **Zuordnung** der Nachrichtepakete zu den **Diensten/Prozessen** eines Rechners („virtuelles Multiplexen“)
- Und vieles, vieles mehr!



www.shutterstock.com · 175211042



# Zusammenfassung

- Motivation:
  - Vielzahl von Systemen
  - Vielzahl von Anforderungen (Zuverlässigkeit, Sicherheit, Effizienz)
  - Vielzahl von Aufgaben und Problemen
- Enorme Komplexität der Netzsoftware!
- => Aufteilung von Aufgaben in Ebenen (= Schichten)
- Auf jeder Ebene: separate Umsetzung der Anforderungen

# Referenzmodelle

Zwei relevante Modelle:

- **OSI-Referenzmodell**

- der große, systematische Standard
- detaillierter als TCP/IP-Modell

- **TCP/IP-Referenzmodell**

- Grundlage für die Internetprotokollfamilie
- Vereinfachtes Modell
- Schichten 5 – 7 des OSI-Modells zusammengefasst zur Anwendungsschicht
- Schichten 1 – 2 des OSI-Modells zusammengefasst zur Netzzugangsschicht

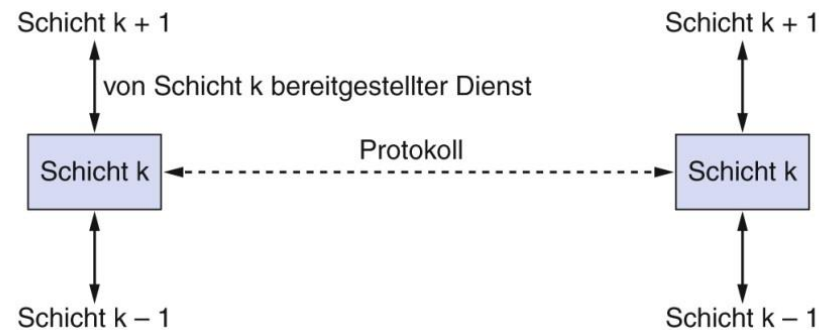
OSI-Schicht	TCP/IP-Schicht	Beispiel
Anwendungen (7)	Anwendungen	HTTP, UDS, FTP, SMTP, POP, Telnet, DHCP, OPC UA, SOCKS
Darstellung (6)		
Sitzung (5)		
Transport (4)	Transport	TCP, UDP, SCTP
Vermittlung (3)	Internet	IP (IPv4, IPv6), ICMP (über IP)
Sicherung (2)	Netzzugang	Ethernet, Token Bus, Token Ring, FDDI
Bitübertragung (1)		

# OSI-Referenzmodell

- *Reference Model for **Open System Interconnection*** (OSI, Referenzmodell für offene Kommunikationssysteme)
- 1983 von John Day und Hubert Zimmermann unter der ISO vorgestellt, 1995 überarbeitet
- **Offener Standard** (Grundgedanke bei Standardisierung):
  - Alle erforderlichen Protokolle und Informationen offengelegt (RFCs)
  - jeder Hersteller kann ein System gemäß dem Standard implementieren
- **Sieben Schichten**, jede erfüllt eine genau definierte Funktion
- Modell ist **keine** fixfertige Netzarchitektur
- Aussage: welche Aufgaben welche Schicht ausführt
- Wichtiger Grundsatz: **Austauschbarkeit von Protokollen**

OSI-Schicht
Anwendungen (7)
Darstellung (6)
Sitzung (5)
Transport (4)
Vermittlung (3)
Sicherung (2)
Bitübertragung (1)

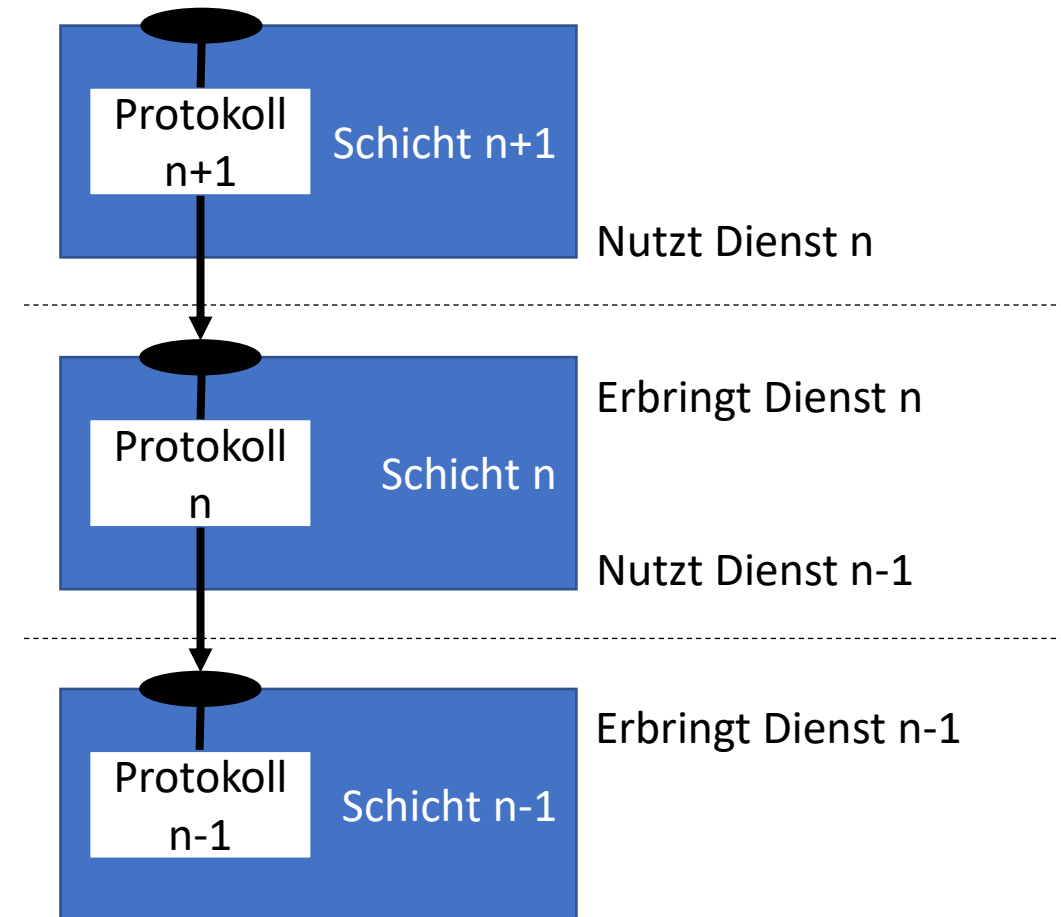
# Beziehung zwischen Diensten und Protokollen



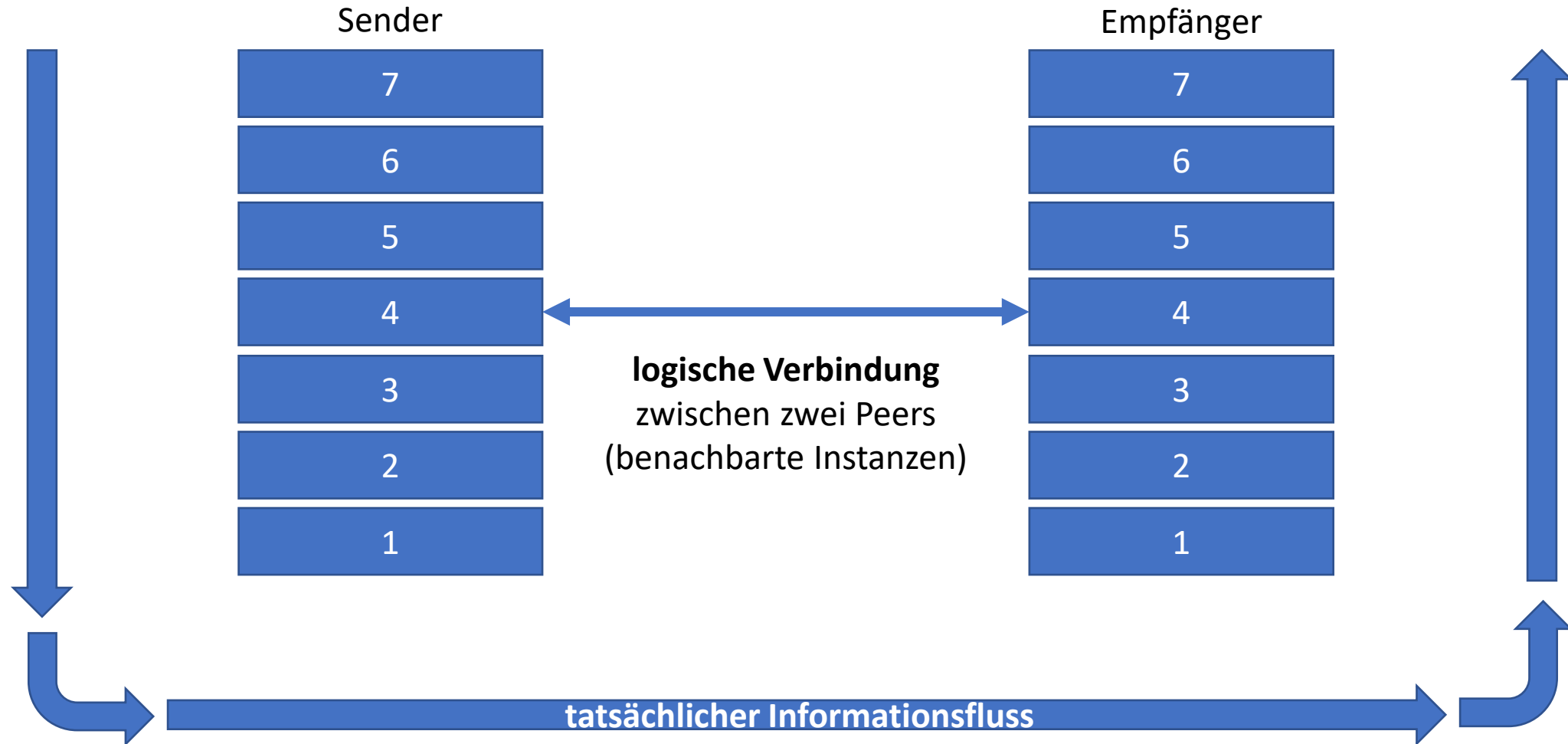
- **Definition Dienst:**  
Gruppe von Basisoperationen, die eine Schicht der über ihr liegenden Schicht zur Verfügung stellt – „**welche Operationen, aber nicht wie**“
- **Definition Protokoll:**  
Menge von Regeln die Format und Bedeutung der Pakete festlegt, welche von Peers ausgetauscht werden – „**das Wie: konkrete Implementierung der Dienste**“
- **Protokolle sind austauschbar**, solange die gleichen Dienste bereitgestellt werden!
- Wichtig: **Sender und Empfänger (Peers)** müssen nach den selben Protokollen arbeiten!

# Dienstbringer/Dienstbenutzer

- Im Schichtenmodell findet **Kommunikation** nur **über benachbarte Schichten** statt
- **Dienstbringer:**  
Schicht  $n$  **bietet** Schicht  $n+1$  einen **Dienst an**: Realisation eines Protokolls mit bestimmtem Funktionsumfang
- **Dienstbenutzer:**  
Schicht  $n$  **nutzt den Dienst** der darunter liegenden Schicht ( $n-1$ )
- Dienst wächst von Stufe zu Stufe
- Vergleich: Fertigungskette Fabrik

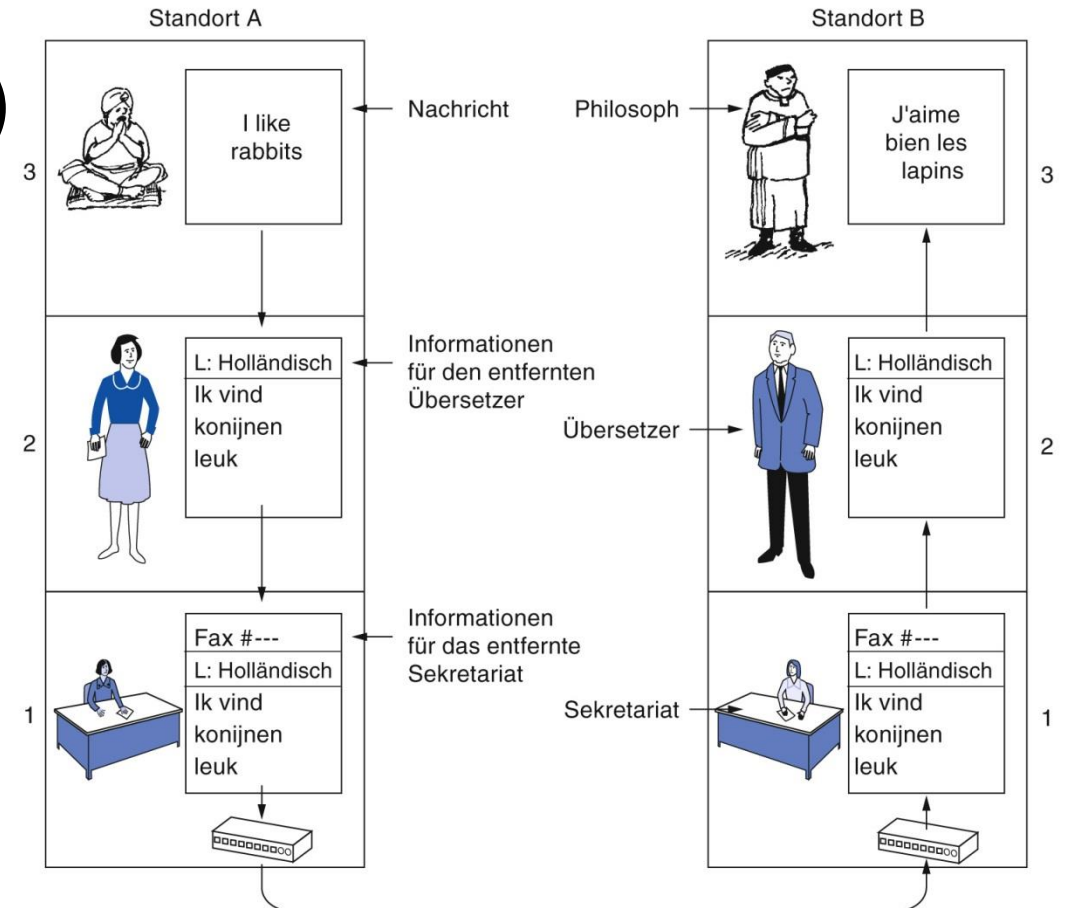


# Informationsfluss und Verbindung



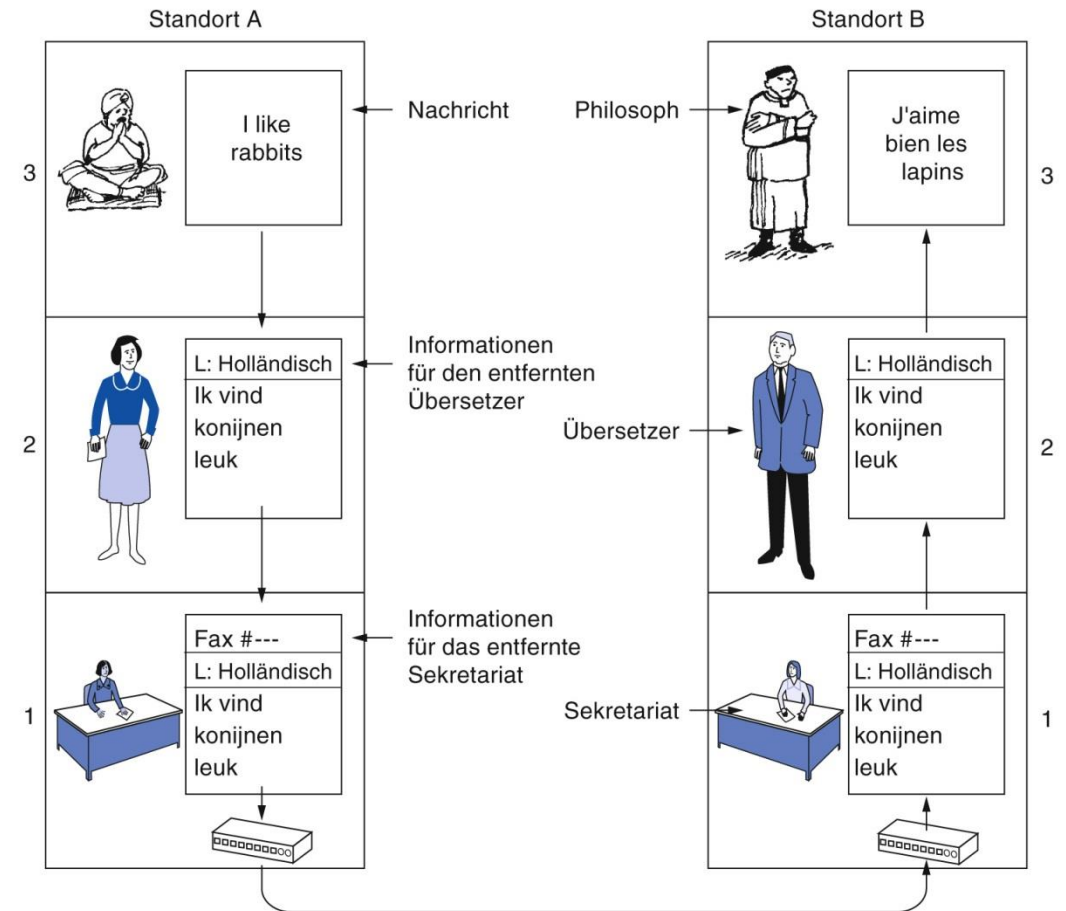
# Analogie I

- zwei Philosophen (Standorte A und B)
  - Philosoph A spricht Urdu und Englisch
  - Philosoph B spricht Chinesisch und Französisch
- jeder benötigt einen Dolmetscher, Festlegung auf "neutrale" Sprache:
  - A: Englisch > *Holländisch*
  - B: *Holländisch* > Französisch
- tatsächliche Übermittlung über Sekretariat per E-Mail



# Analogie II

- Was entspricht in dem analogen Modell einem **Dienst**?
- Was entspricht in dem analogen Modell einem **Protokoll**?
- Wie könnten die **Schnittstellen** definiert werden?
- Was könnte man hier **austauschen**?
- Welche **Informationen** fügen die Peers bei ihren Prozessen hinzu?





# Vorteile des Schichtenkonzepts

Sehr abstraktes Modell, trotzdem etliche Vorteile:

- Unabhängigkeit der Schichten voneinander
- Flexibilität (Austauschbarkeit der Protokolle)
- Physikalische Trennung der Schichten
- Vereinfachte Standardisierung
- Einfache Wartung und Implementation

# Nachteile

- Immenser Aufwand an Steuerinformationen (jede Schicht schreibt ihren eigenen Header)
- Übertragung der Daten dauert dadurch länger
  - Netto- vs. Brutto-Daten
  - Verarbeitungszeit

# Die Schichten des OSI-Modells

Nummer	Englische Bezeichnung	Deutsche Bezeichnung
7	Application Layer	Anwendungsschicht
6	Presentation Layer	Darstellungsschicht
5	Session Layer	Sitzungsschicht
4	Transport Layer	Transportschicht
3	Network Layer	Vermittlungsschicht
2	Data Link Layer	Sicherungsschicht
1	Physical Layer	Bitübertragungsschicht

# 1. Schicht – Bitübertragung (*Physical Layer*)

- Physikalische Eigenschaften:  
(elektrisch/optisch/akustisch/elektromechanisch)
  - Beschaffenheit (Kupfer-) Kabel
  - Steckerformen
  - Pin-Belegungen
- Zugeordnete Geräte:
  - Antennen, Verstärker
  - Stecker und Buchse
  - Repeater, Hub, Transceiver
- Dienst: Austausch einzelner Bits eines Bitstroms zwischen Rechnern

## 2. Schicht – Sicherung (*Data Link Layer*)

- Kommunikation zwischen 2 phys. verbundenen Geräten vereinfachen
- Wichtigste Aufgaben:
  - Rahmen (Frames) aus Datenstrom der Vermittlungsschicht erzeugen = SEGMENTIERUNG
  - Hinzufügen von Prüfsummen im Rahmen
- Protokolle legen fest:
  - Wie groß sind die Rahmen
  - Wie werden Fehler erkannt/korrigiert (aber nicht neu angefordert!)
  - Datenflusskontrolle: Empfänger steuert Geschw. des Senders dynamisch
- Zugeordnete Hardware: Bridge, Layer-2-Switch

# 3. Schicht – Vermittlung (*Network Layer*)

- Schichten 1 und 2 regeln Kommunikation direkt verbundener Geräte
- Vernetzung mit mehreren Geräten (Endsystem, Knoten) => Schicht 3!
  - Leitungsorientiert: Schalten von Verbindungen (Telefonvermittlung)
  - Paketorientiert: Weitervermittlung von Datenpaketen (Post)
- Wichtigste Aufgaben:
  - Einführung von Benennungen (= logische Adressen)
  - Routing (Wegfindung, Verwaltung von Routingtabellen)
  - Fragmentierung von Datenpaketen
- Protokolle: IP (v4, v6)
- Zugeordnete Hardware: Router, Layer-3-Switch

## 4. Schicht – Transport (*Transport Layer*)

- Vermittlungsschicht (3) erlaubte schon Adressierung von Endsystemen
- Transportschicht (4) erlaubt nun auch Adressierung von Anwendungen
- Beispiel: Server bietet
  - Web-Server
  - Mail-Server
- Unterscheidung aufgrund der Transportadresse = Port
- Transportschicht bietet anwendungsorientierten Schichten (5–7) einheitlichen Zugriff auf Kommunikationsnetz
- Protokolle: TCP, UDP

## 5. Schicht – Sitzung/Kommunikationssteuerung (*Session Layer*)

- Bis hier: transportorientiert, ab hier: anwendungsorientiert
- Wichtigste Aufgabe:
  - Wiederherstellungspunkte setzen, um Fortsetzung der Kommunikation nach Zusammenbruch der Sitzung zu ermöglichen (= Synchronisation)
  - Aushandeln der Zugriffsberechtigungen (Token)
- Als separate Schicht: Heute keine praktische Bedeutung



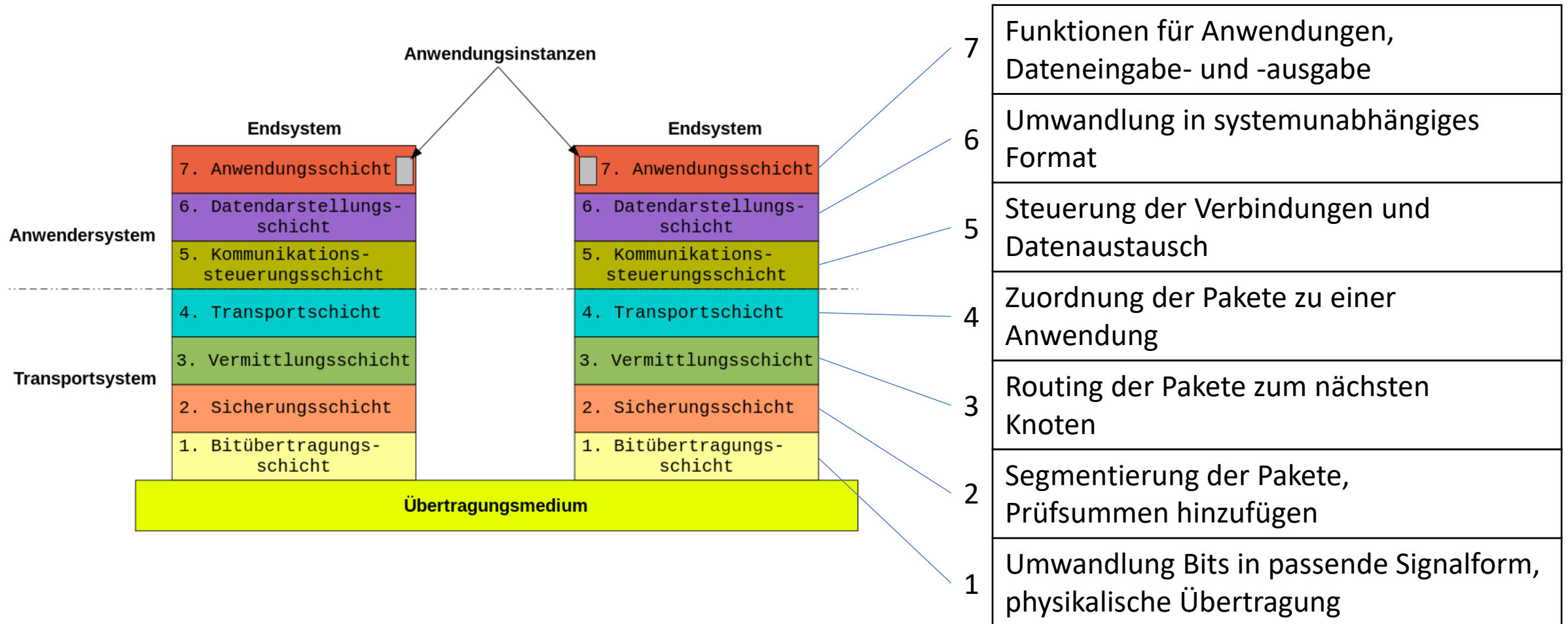
## 6. Schicht – Darstellung (*Presentation Layer*)

- Verschiedenste Systeme (Prozessoren, Betriebssysteme, Programmiersprachen) => Unterschiede in Repräsentation von Daten (ganze Zahlen, Gleitkommazahlen, Text...)
- Darstellungsschicht setzt systemabhängige Darstellung von Daten (z. B. ASCII) um in **unabhängige** Form
- Beispiel: ASCII enthält keine Umlaute, HTML verwendet u. a. symbolische Bezeichner => ö = „&ouml;“
- Weitere Aufgaben:
  - Datenkompression
  - Verschlüsselung

## 7. Schicht – Anwendung (*Application Layer*)

- Stellt Funktionen für die Anwendungen zur Verfügung
- Dateneingabe und –ausgabe
- Anwendungen selbst gehören aber nicht zur Schicht
- Anwendungen wären dann z. B.:
  - Webbrowser
  - E-Mail-Programm
  - Messenger

# Überblick und Zusammenfassung I



# Überblick und Zusammenfassung II

OSI-Schicht		Einordnung	<a href="#">DoD-Schicht</a>	Einordnung	Protokollbeispiele	Einheiten	Kopplungselemente
7	Anwendungen (Application)	Anwendungs-orientiert	Anwendung	Ende zu Ende ( <a href="#">Multihop</a> )	<a href="#">HTTP</a> <a href="#">FTP</a> <a href="#">HTTPS</a> <a href="#">SMTP</a> <a href="#">XMPP</a> <a href="#">DNS</a> <a href="#">LDAP</a> <a href="#">NCP</a>	Daten	<a href="#">Gateway</a> , <a href="#">Content-Switch</a> , <a href="#">Proxy</a> , Layer-4-7-Switch
6	Darstellung (Presentation)						
5	Sitzung (Session)						
4	Transport (Transport)	Transport-orientiert	Transport	Punkt zu Punkt <a href="#">Punkt zu Punkt</a>	<a href="#">TCP</a> <a href="#">UDP</a> <a href="#">SCTP</a> <a href="#">SPX</a>	<a href="#">TCP = Segmente</a> <a href="#">UDP = Datagramme</a>	<a href="#">Router</a> , <a href="#">Layer-3-Switch</a>
3	Vermittlung-/Paket (Network)		Internet		<a href="#">ICMP</a> <a href="#">IGMP</a> <a href="#">IP</a> <a href="#">IPsec</a> <a href="#">IPX</a>	<a href="#">Pakete</a>	
2	Sicherung (Data Link)		Netzzugriff		<a href="#">Ethernet</a> <a href="#">Token Ring</a> <a href="#">FDDI</a> <a href="#">MAC</a> <a href="#">ARCNET</a>	Rahmen ( <a href="#">Frames</a> )	<a href="#">Bridge</a> , <a href="#">Layer-2-Switch</a>
1	Bitübertragung (Physical)					<a href="#">Bits</a> , <a href="#">Symbole</a> , Pakete	<a href="#">Netzwerkkabel</a> , <a href="#">Repeater</a> , <a href="#">Hub</a>

# Anno 1985



<https://youtu.be/uMPpQ5tOmus>

# TCP/IP-Referenzmodell

Kommunikationsregeln des Internets

# TCP/IP-Referenzmodell

- *"die Sprache, wenn zwei Computer über das Internet kommunizieren"*
- **Gruppe von Protokollen** (insgesamt > 500), die den **Datenaustausch** zwischen Rechnern über das **Internet** regeln
- Entstand Ende der 1960er unter dem Department of Defense (deshalb auch **DoD-Referenzmodell** genannt) und wurde 1983 offiziell zur **Standard-Protokollfamilie im ARPANET**
- ARPANET wuchs: Telefonstandleitung, Funk- sowie Satellitennetze
- Notwendigkeit für neue Netzwerk-Referenzarchitektur
  - Mehrere heterogene Netze nahtlos verbinden
  - Kommunikation (Quelle + Ziel) aufrechterhalten, auch nach Hardwareausfall
  - verschiedenste Anwendungen: Datendienste, Sprachübertragung, ...

# TCP/IP-Protokollfamilie

- Protokoll = Festlegung der Regeln einer Kommunikation
- Protokollfamilie = Zusammenfassung mehrerer Protokolle zu einem Gesamtpaket = Protokoll Stapel (protocol stack)
- Ursprünglich: Parallele Entwicklung von versch. Protokollfamilien:
  - TCP/IP
  - IPX/SPX (Novell)
  - OSI (Open System Interconnection)
  - DECnet
- TCP/IP hat sich in vielen Bereichen durchgesetzt



# Übersicht

OSI-Schicht	TCP/IP-Schicht		Beispiel
Anwendungen (7)	Anwendungen	Application	HTTP, UDS, FTP, SMTP, POP, Telnet, DHCP, OPC UA SOCKS
Darstellung (6)			
Sitzung (5)			
Transport (4)	Transport	Transport	TCP, UDP, SCTP
Vermittlung (3)	Internet	Internet	IP (IPv4, IPv6), ICMP (über IP)
Sicherung (2)	Netzzugang	Link	Ethernet, Token Bus, Token Ring, FDDI
Bitübertragung (1)			

# Netzzugangsschicht (link layer)

- **Paketvermittlung** als Antwort auf die Anforderungen (unterschiedliche Netze, Hardwareausfall)
- Netzzugangsschicht beschreibt, was die Verbindungen (serielle Leitungen, klassisches Ethernet, ...) dazu tun müssen (weil Internetschicht **verbindungslos** ist)
- Ist die Schnittstelle zwischen Hosts und Übertragungsleitungen

# Die Internetschicht (internet layer)

- Entspricht grob der OSI-Vermittlungsschicht
- Ziel: Hosts ermöglichen, in **jedes beliebige Netz Pakete einzuspeisen**, unabhängig vom Zielnetz
- Sogar **Reihenfolge egal**, Sortierung wird von höheren Schichten übernommen
- Analogie: Post
  - Adressen im Zielland (Zielnetz)
  - Sammelstellen (Gateways)
  - eigene Briefmarken, Umschläge ...
- definiert offizielles **Paketformat (IP-Paket)** und **Protokoll (IP)** + dazu **ICMP**
- Aufgabe:
  - IP-Pakete richtig zustellen (**Adressierung**)
  - **Paket-Routing**
  - **Überlastungen** vermeiden

# Die Transportschicht (transport layer)

- Ziel: Peer-Einheiten auf Quell- und Zielhosts verbinden
- Anfangs (historisch): 2 Ende-zu-Ende-Übertragungsprotokolle: TCP und UDP
- **TCP** (Transmission Control Protocol):
  - **Zuverlässiges, verbindungsorientiertes** Protokoll mit **Flusskontrolle**
  - Teilt eingehenden Bytestrom auf in einzelne Nachrichten und gibt diese an die Internetschicht weiter
  - Am Ziel werden sie wieder zusammengesetzt
- **UDP** (User Datagram Protocol):
  - **Unzuverlässiges, verbindungsloses** Protokoll
  - Für Anwendungen (Schichten darüber!) die auf korrekte Reihenfolge, Fehlererkennung oder Flusskontrolle der Transportschicht verzichten und das lieber selbst erledigen
  - Wird vor allem für einmalige Anfragen, sowie Anwendungen in Client-Server-Umgebungen verwendet die schnell sein müssen (Sprache, Video)

# Die Anwendungsschicht (application layer)

- TCP/IP-Modell hat **keine Sitzungs- oder Darstellungsschicht!**
- benötigte Funktionen werden **von den Anwendungen selbst** eingebunden
- Erfahrung mit OSI-Modell zeigte, dass das gut so war!
- Anwendungsschicht umfasst alle Protokolle höherer Schichten, z. B.:
  - TELNET (virt. Terminal)
  - FTP
  - SMTP
  - DNS
  - HTTP
  - RTP (Real-time Transport Protocol, für Echtzeitmedien wie Sprache oder Filme)

# Protokolle und Netzwerke im TCP/IP-Modell

