

COMPUTERNETZE

Prof. Dr. Wolf-Fritz Riekert
 Fachhochschule Stuttgart – Hochschule der Medien (HdM)
 University of Applied Sciences Stuttgart – School of Media

<mailto:riekert@hdm-stuttgart.de>
<http://v.hdm-stuttgart.de/~riekert>

INHALTSÜBERSICHT

- Teil 0: Allgemeines
- Teil 1: Bitübertragungsschicht
- Teil 2: Sicherungsschicht
- Teil 3: Vermittlungsschicht
- Teil 4: Transportschicht
- Teil 5: Verarbeitungsschicht

LITERATUR

A.S. Tanenbaum: Computernetzwerke. 3. rev. Auflage.
 Prentice Hall, 2000. Abbildungen aus dem Buch im Web unter
<http://www.cs.vu.nl/~ast>. (Standardwerk)

M. Payer: Computervermittelte Kommunikation.
 URL: <http://www.payer.de/cmcs/cmcs0.htm>, Datum des Zugriffs:
 1.2.2002. (Ausführliches Skript, guter Gesamtüberblick über
 das Thema mit vielen Links auf weitere Ressourcen)

GLIEDERUNG

- Teil 0: Allgemeines
 - Einführung
 - Schichten, Dienste, Schnittstellen, Protokolle
 - Referenzmodelle (OSI, TCP/IP, hybrides Modell)
 - Schichten Übergreifendes
- Teile 1 bis 5: Gliederung des Vorlesungsstoffs nach den fünf Schichten des hybriden Referenzmodells von Tanenbaum
 - Aufgaben der jeweiligen Schicht
 - Beispiele (z.B. aus Internet, lokale Netze, Telefonie), gehen zum Teil über die jeweils betrachtete Schicht hinaus, werden aber zusammenhängend dargestellt.

NUTZEN VON COMPUTERNETZEN FÜR UNTERNEHMEN

- Gemeinsame Nutzung von Ressourcen: Daten, Programme, Geräte (z.B. Drucker)
- Zuverlässigkeit durch redundante Haltung von Ressourcen
- Kosteneinsparung durch viele preiswerte PCs und wenige große Server (Client-/Server-Modell)
- Skalierbarkeit: Einfache Aufrüstung der Rechenkapazität durch zusätzliche Computer im Netz
- Kommunikation auf Abteilungsebene, im gesamten Unternehmen, mit befreundeten Unternehmen (z.B. Lieferanten), weltweit (z.B. allen potentiellen Kunden)
- elektronischer Geschäftsverkehr (E-Business) mit anderen Firmen („Business to Business“ = B2B) und mit Endkunden („Business to Customer“ = B2C)
- Integration des Telefonverkehrs („Voice over IP“)

NUTZEN VON COMPUTERNETZEN FÜR BÜRGER

- Interpersonelle Anwendungen: Email, Chat, SMS, Fax
- Unterhaltung: Internet-Radio/TV, Video on Demand, Musik-Download, Spiele
- Informationen: News, digitale Zeitungen, WWW allgemein
- Auskunftsdienste: Fahrplan, Veranstaltungen, Telefonnr.
- Geschäftsverkehr („Business to Customer“ = B2C): Online-Shopping, Internet-Banking, Ticketservice ...
- Behördenverkehr („Administration to Customer“ = A2C): Ersparnis von Behördengängen, elektr. Steuererklärung ...
- Privates Büro: „SOHO“ (Small Office, Home Office), Telearbeit, Teilnahme am Arbeitsmarkt (Bewerbungen), Fortbildung (E-Learning), selbständige Tätigkeit

BEDEUTUNG DER SOFTWARE FÜR COMPUTERNETZE

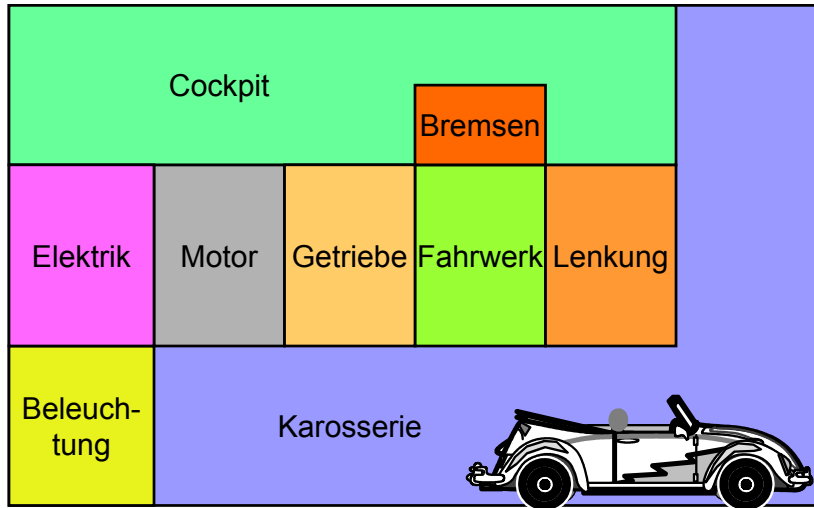


- Software ist inzwischen die entscheidende Komponente zur Bereitstellung von Netzwerkdiensten geworden
- Der überwiegende Teil dieser Vorlesung ist mit Netzwerksoftware befasst.
- Netzwerksoftware: ein komplexes Feld, das einer besonderen Strukturierungstechnik bedarf
 - ⇒ Strukturierung in Form von Schichten oder Ebenen

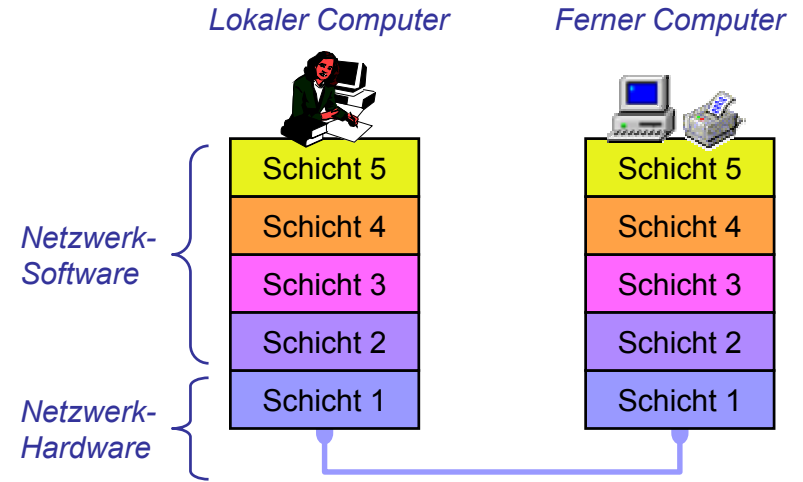
WARUM SCHICHTEN?

- **Modularisierung** der Netzwerksoftware. Jede Schicht ist ein eigener Modul. Zwischen den Modulen gibt es feste **Schnittstellen**. Für das Verständnis des Ganzen ist es nicht wichtig, wie ein Modul intern funktioniert, er kann als „Blackbox“ betrachtet werden. Dies dient der **Reduzierung der Komplexität** und vereinfacht die Arbeit für die Systementwickler.
- Schichten sind vertikal geordnet. Jede Schicht hat **nur Schnittstellen mit der unmittelbar darüber und der unmittelbar darunter liegenden Schicht**. Dies hat eine weitere Reduzierung der Komplexität zur Folge.
- Die festen Schnittstellen erlauben es, **Schichten auszuwechseln**, ohne die darüber oder darunter liegenden Schichten zu beeinflussen (Beispiel: Übergang von einem Token-Ring-Netz zu einem Ethernet).

BEISPIEL FÜR MODULARISIERUNG: AUTOMOBIL



SCHICHTEN GLIEDERN NETZWERKSOFT- UND HARDWARE

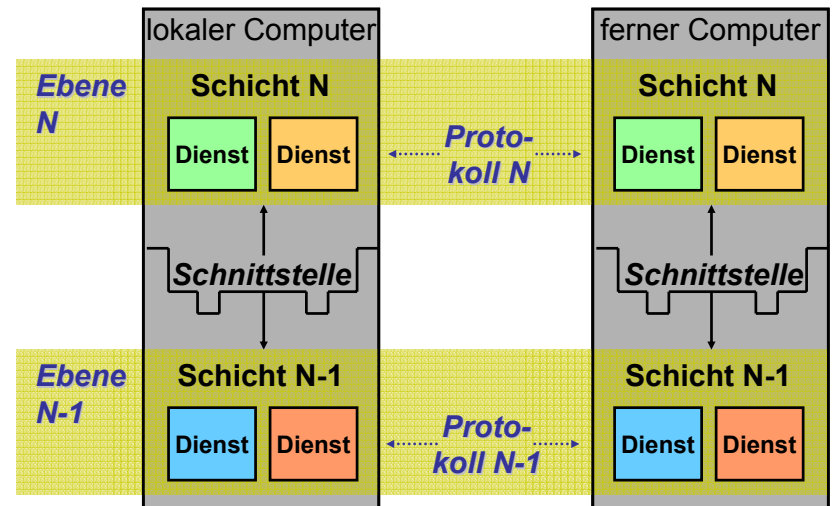


SCHICHTEN, DIENSTE, PROTOKOLLE, SCHNITTSTELLEN (1)

Netzwerksoftware wird in Form von **Schichten** (layers) aufgebaut.

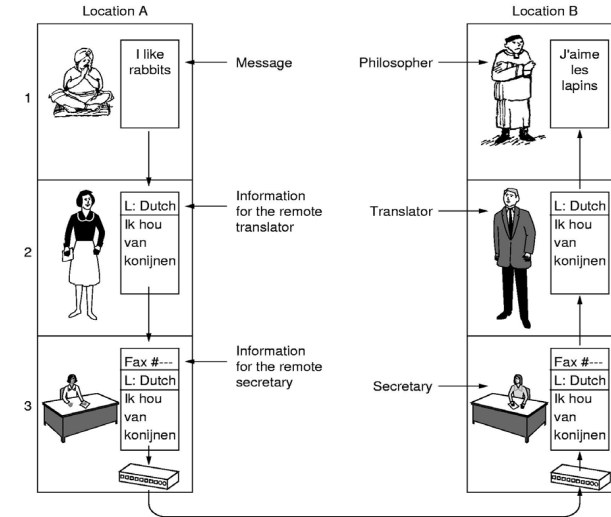
- Diese Schichten realisieren (**Netzwerk-)**Dienste (services).
- Schichten kommunizieren mit Schichten derselben Ebene (sogenannten Peers) auf fremden Computern. Diese Kommunikation befolgt **Protokolle** (= Regeln und Konventionen für die Kommunikation)
- Kommunikation erfolgt mittelbar (indirekt) über Dienste der nächsttieferen Schicht.
- Zwischen zwei angrenzenden Schichten existiert eine **Schnittstelle**. Diese legt fest, wie die Dienste der unteren Schicht von der oberen Schicht in Anspruch genommen werden können.

SCHICHTEN, DIENSTE, PROTOKOLLE, SCHNITTSTELLEN (2)



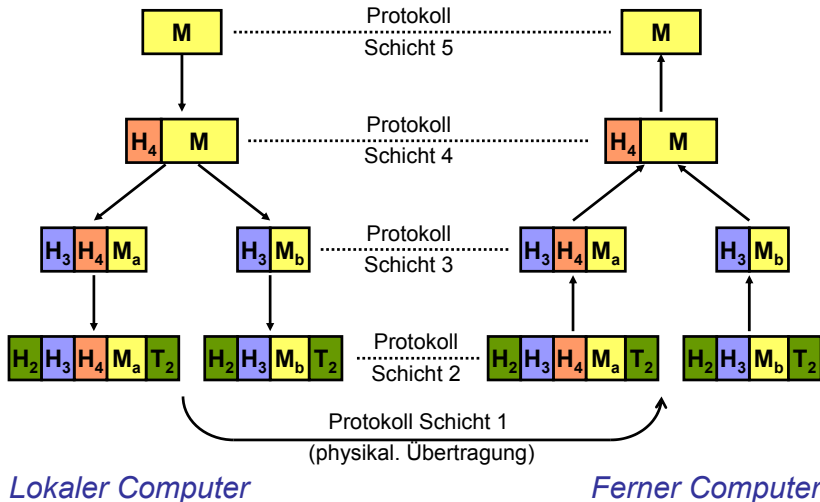
Netzwerkdienste werden in Form von Software realisiert. Hierzu stehen verschiedene Techniken zur Verfügung:

- **Anwendungsprogramme** auf der obersten Schicht. Beispiel: Mailtool, Web-Browser
- **„Unterprogramme“** in „Unterprogrammibliotheken“ (klassisch) bzw. „Methoden“ oder „Member Functions“ von Klassen und Objekten in „Klassenbibliotheken“ (objektorientiert). Diese werden von der höheren Schicht aus durch Aufrufe aktiviert
- Als Hintergrundprozesse gestartete **„Serverprogramme“**, die periodisch oder ereignisgesteuert die nächsttiefere Schicht über Aufrufe abfragen und z.B. ankommende Nachrichten übernehmen und weiterverarbeiten.



Quelle:
Tanenbaum
2000

KOMMUNIKATION IN COMPUTERNETZEN (1)



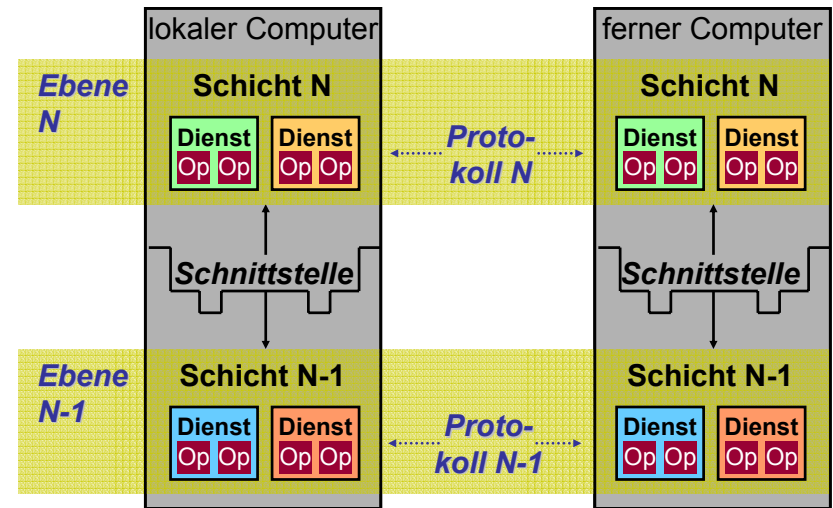
KOMMUNIKATION IN COMPUTERNETZEN (2)

- Abgehende Nachrichten M werden fortgesetzt an tiefere Schichten übergeben, bis die unterste Schicht erreicht ist, die die physikalische Übertragung übernimmt. Jede Schicht kann einen Nachrichtenkopf H („Header“) und ggf. auch einen Nachspann T („Trailer“) hinzufügen.
- Manche Schichten zerlegen größere Nachrichten in kleinere Teile. Umgekehrt können Schichten auch mehrere kleinere Nachrichten zu einer langen Nachricht zusammenfassen.
- Ankommende Nachrichten werden ausgehend von der untersten Schicht fortgesetzt an höhere Schichten übergeben. Dabei werden die der jeweiligen Schicht zugeordneten Header und Trailer entfernt.
- Beim Versand erfolgte Zerlegungen bzw. Zusammenfassungen von Nachrichten können beim Empfang von der jeweils zuständigen Schicht wieder rückgängig gemacht werden.

DIENTE UND DIENSTOPERATIONEN

- Dienste stellen eine Gruppe von **Dienstoperationen** bereit
- Beispiel Telefondienst: Dienstoperationen sind „Anrufen“, „Gespräch entgegennehmen“, „Gespräch beenden“, ...
- Ein Dienst wird dem Dienstinutzer (d.h. der nächsthöheren Softwareschicht bzw. auf der obersten Ebene einem Menschen) über eine Schnittstelle angeboten.
- Die Schnittstelle legt die Dienstoperationen und die Art der Schnittstellendaten (z.B. Adressangaben, Art und Format der zu übertragenden Daten, einstellbare Optionen) fest
- Die Implementation, d.h. der interne Aufbau der Dienstoperationen in Form eines Computerprogramms ist durch die Schnittstelle vor dem Dienstinutzer versteckt.
- Das Protokoll ist Gegenstand der Implementation, und deshalb (bei guten Implementationen) für den Dienstinutzer unsichtbar.

DIENSTOPERATIONEN IM SCHICHTENMODELL



DESIGNASPEKTE DER SCHICHTEN

- Adressierung
- verbindungsorientierte, verbindungslose Kommunikation
- logische Kommunikationskanäle
 - ⇒ Richtung: Simplex, Halbduplex, Vollduplex
 - ⇒ mehrere Kanäle (z.B. normale und dringende Daten)
- Multiplexen: Mehrere gleichzeitige Übertragungen über einen Kanal
- Fehlerüberwachung, -behebung
- Zerlegung von Nachrichten in Teile, Zusammenfassung
- Einhaltung der Reihenfolge der Nachrichten
- Geschwindigkeitsanpassung (z.B. langsamer Empfänger)
- Routing (Vermittlung von Datenpaketen durch das Netz)

VERBINDUNGSORIENTIERTE DIENSTE

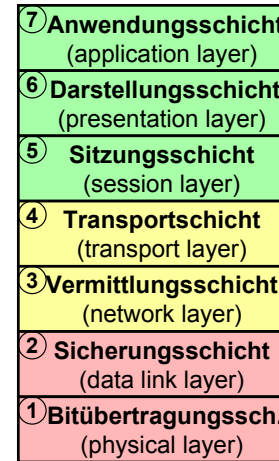
- Analogie: Telefonsystem
- 3 Phasen: Verbindungsaufbau, Nachrichtenübertragung, Verbindungsabbau
- Adressierung des Kommunikationspartners nur beim Verbindungsaufbau erforderlich, sonst genügt Bezug auf die aufgebaute Verbindung
- Empfang der Nachrichten in ursprünglicher Reihenfolge garantiert
- Varianten: Nachrichtenfolgen (Nachrichtengrenzen bleiben erhalten) bzw. Bytestrom (Nachrichtengrenzen bleiben nicht erhalten)
- Hohe Zuverlässigkeit (hinsichtlich Datenverlust) realisierbar durch Bestätigungsnachrichten. Automatische Korrektur von Fehlern durch Übertragungswiederholungen möglich.

VERBINDUNGSLOSE DIENSTE

- Analogie: Postsystem
- Kein Verbindungsaufbau, jede Nachricht trägt volle Adresse des Kommunikationspartners
- Nachrichten (Datengramme) werden nicht notwendig in ursprünglicher Reihenfolge empfangen
- Zuverlässigkeit (hinsichtlich Datenverlust) realisierbar durch Bestätigungsnachrichten, doch meist wird zugunsten höherer Übertragungsgeschwindigkeiten darauf verzichtet.
- Varianten
 - ⇒ unzuverlässiges Datengramm (analog: Postkarte)
 - ⇒ bestätigtes Datengramm (analog: Einschreiben mit Rückschein)
 - ⇒ Anfrage/Antwort (z.B.: Datenbankabfrage, WWW)

DAS OSI-REFERENZMODELL DER ISO

OSI (Open Systems Interconnection)



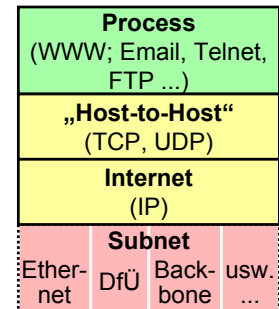
- Modell zur Verbindung offener Systeme (d.h. offen zur Kommunikation mit Systemen unterschiedlicher Hersteller)
- Festgelegt durch **ISO** (International Standards Organization) Ende 70er bis Anfang 80er-Jahre
- OSI sieht 7 Schichten vor und legt fest, was diese Schichten bewirken sollen
- OSI definiert keine Dienste und Operationen, ist daher keine Netzarchitektur
- In der Folge wurden aber auf der Basis von OSI Dienste und Operationen genormt und implementiert.

PRINZIPIEN BEI DER ENTWICKLUNG VON OSI

- Neue Schicht dort, wo neuer Abstraktionsgrad gefordert
- Jede Schicht sollte eine genau definierte Funktion erfüllen
- International genormte Protokolle berücksichtigen
- Informationsfluss zwischen Schichten sollte möglichst gering sein
- Anzahl der Schichten: Möglichst eine Schicht pro Funktion, aber die resultierende Netzwerkarchitektur sollte nicht unhandlich werden.

DAS INTERNET

Das **Internet** ist ein offenes Verbundnetz, das verschiedene existierende Netze als „Subnetze“ miteinander verbindet

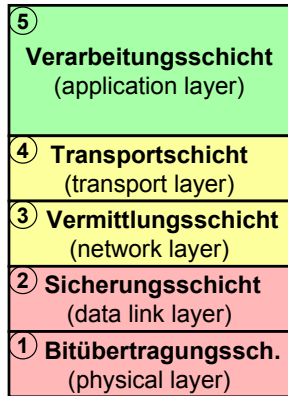


- Entstanden 1969 als **ARPANET** (gefördert durch US-amerikanische Militärforschungsinstitution „Advanced Research Project Agency“)
- Anfangs entwickelt durch verschiedene Universitäten und Forschungsinstitute
- Betrieb und Weiterentwicklung heute weitgehend auch durch kommerzielle Einrichtungen.

Pragmatische Entwicklungsphilosophie, folgt nicht dem OSI-Schichtenmodell. Dienste lassen sich grob in 3 Schichten innerhalb des Internet sowie 1 Subnetzschiicht strukturieren.

DAS HYBRIDE FÜNFSCHICHTEN-MODELL VON TANENBAUM

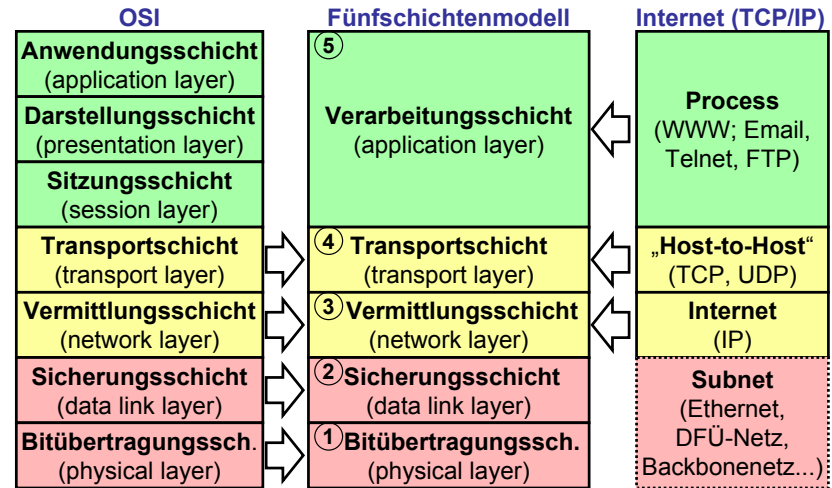
Im Lehrbuch „Computernetzwerke“ von Tanenbaum (2000) wird ein „hybrides“, d.h. aus OSI- und Internet-Modell abgeleitetes Fünfschichtenmodell vorgestellt.



- Vergrößerung des OSI-Modells: Die Schichten 5 bis 7 werden zu einer Schicht zusammengefasst.
- Übereinstimmung in Schicht 3 u. 4
- Verfeinerung des Internet-Modells: Die Subnetz-Schicht des Internet wird in die zwei entsprechenden OSI-Schichten 1 u. 2 aufgespaltet.

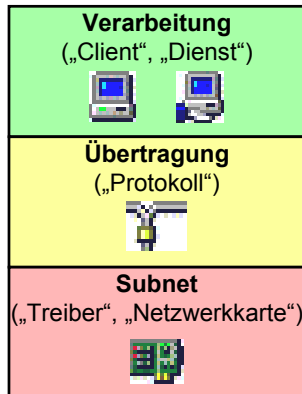
Nach diesem Modell wird im Folgenden vorgegangen.

FÜNFSCHICHTENMODELL, OSI UND INTERNET IM VERGLEICH

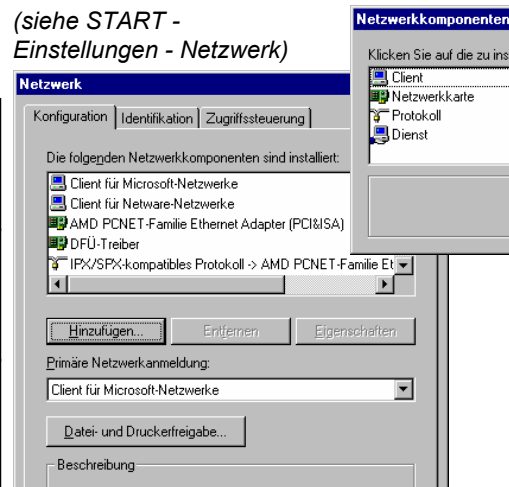


DAS SCHICHTENMODELL VON MICROSOFT WINDOWS

Microsoft Windows unterscheidet nur drei Schichten



(siehe START - Einstellungen - Netzwerk)

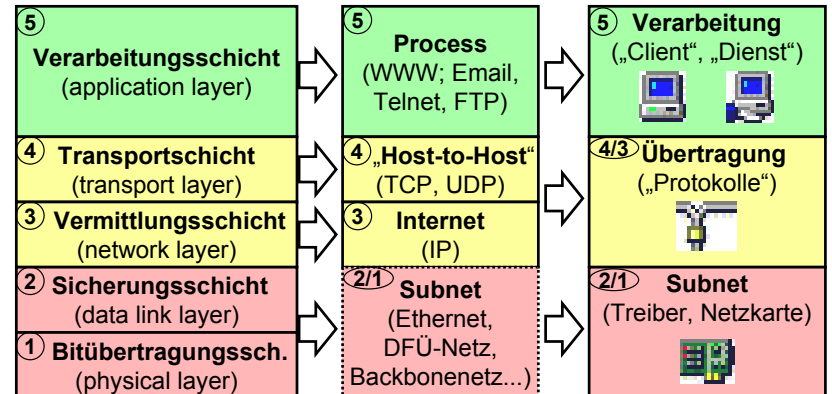


FÜNFSCHICHTENMODELL, INTERNET UND MICROSOFT IM VERGLEICH

Fünfschichtenmodell

Internet (TCP/IP) unterscheidet 3+1 Schicht

Microsoft Windows unterscheidet nur drei Schichten



GLIEDERUNG DES STOFFS NACH DEM HYBRIDEN MODELL

Verarbeitungssch. (application layer) ⑤

Anwendungsdienste: WWW, FTP, Telnet, E-Mail, File-/Printservices.
Netzsicherheit: Kryptographie, E-Commerce
Client-/Server-Programmierung mit Java, Perl ...

Transportschicht (transport layer) ④

Kommunikation zw. Prozessen: Adressierung, Flusssteuerung, Verbindung. Beisp.: TCP, UDP

Vermittlungssch. (network layer) ③

Kommunikation zwischen Computern: Vermittlung, Routing, Beisp.: IP

Sicherungsschicht (data link layer) ②

Übertragung von Rahmen im „Subnetz“, Fehlerüberwachung, Fluss-Steuerung, Beisp.: PPP

MAC-Teilschicht (media access control)

Kanalzuordnung: Lokale Netze (Ethernet/CDMA, Token Ring), Hubs, Switches, Bridges, GSM

Bitübertragungssch. (physical layer) ①

Bitrate, Übertragungsmedien, Telefonie, ISDN, Leitungsvermittlung, serielle Übertragung (RS232), Modem

1: BITÜBERTRAGUNGSSCHICHT (PHYSICAL LAYER)

Übertragung von rohen Bits über einen Übertragungskanal:

- Festlegung des physischen Übertragungsmediums
- mechanische, elektrische und prozedurale Festlegungen

Typische Festlegungen der Bitübertragungsschicht:

- Wieviel Volt entsprechen einer logischen 1 bzw. 0
- Wieviel Millisekunden dauert ein Bit
- Gleichzeitige Übertragung in beide Richtungen oder nicht?
- Wie kommt die erste Verbindung zustande und wie wird sie wieder gelöst
- Wie ist der Stecker für den Netzanschluss mechanisch aufgebaut?

2: SICHERUNGSSCHICHT (DATA LINK LAYER)

Fehlerfreie Übertragung von Daten mit Hilfe von „Datenrahmen“ und „Bestätigungsrahmen“

- Rahmen (*frames*) = voneinander abgrenzbare Bitfolgen
- Besondere Bitmuster als Rahmengrenzen, die innerhalb des Rahmens nicht auftreten dürfen
- Wiederholung der Übertragung im Fehlerfall, Erkennung und Eliminierung von Duplikaten
- Geschwindigkeitsanpassung
- **MAC-Teilschicht** (Media Access Control):
Regelung des Zugriffs auf das Übertragungsmedium in sog. Broadcast-Netzen, in denen alle Stationen denselben Kanal benutzen (Beispiel: Ethernet, Token Ring Net)

3: VERMITTLUNGSSCHICHT (NETWORK LAYER)

Steuerung des Betriebs des Subnetzes (der Subnetze):

- Routing, d.h. Auswahl der Paketrouten (statisch nach Tabellen oder dynamisch nach Netzauslastung etc.)
- Vermeidung von Staus bei hoher Netzbelastung
- Abrechnungsfunktion
- Verbindung heterogener Subnetze (z.B. mit unterschiedlichen Protokollen und Adressierungsarten)
- Die Vermittlungsschicht im Internet ist realisiert durch das Internet Protocol (IP)

4: TRANSPORTSCHICHT (TRANSPORT LAYER)

- Echte Ende-zu-Ende-Schicht: ermöglicht die Kommunikation zwischen zwei Prozessen auf unterschiedlichen Rechnern
- Verschiedene Arten von Transportdiensten möglich, z.B. verbindungsorientierter Transport (z.B. TCP), Datagramme (z.B. UDP) oder Broadcast an viele Empfänger
- Benennungsmechanismus für Transportverbindungen
- Ggf. Zerlegung der Nachrichten in kleinere Einheiten und Zusammensetzen in richtiger Reihenfolge beim Empfänger
- Multiplexen von Kanälen der Vermittlungsschicht, damit mehrere Prozesse gleichzeitig kommunizieren können
- Fluss-Steuerung zur Geschwindigkeitsanpassung

5: VERARBEITUNGSSCHICHT (APPLICATION LAYER)

- Realisiert durch Prozesse (ablaufende Programme), die miteinander über die Transportschicht kommunizieren
 - ⇒ In der Regel Unterscheidung von Clientprozess (Dienstanforderer) und Serverprozess (Dienstbringer)
 - ⇒ Beispiele: Telnet-, FTP-, Email-, WWW-Server u. -Clients
- Verarbeitungsschicht in unserem Hybridmodell entspricht der **Anwendungsschicht** im OSI-Modell, umfasst aber zusätzlich die Aufgaben der folgenden zwei OSI-Schichten
 - ⇒ **Sitzungsschicht** (session layer): Verwaltung von sog. Sitzungen, z.B. Login Sessions oder Filetransfers
 - ⇒ **Darstellungsschicht** (presentation layer): Kodierung von Daten auf standardisierte Weise, z.B. Buchstaben, Zahlen, Geldbeträge, Rechnungen usw.

TEIL 1: BITÜBERTRAGUNGSSCHICHT (PHYSICAL LAYER)

Übertragung von rohen Bits über einen Übertragungskanal:

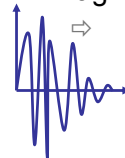
- Festlegung des physischen Übertragungsmediums
- mechanische, elektrische und prozedurale Festlegungen

Typische Festlegungen der Bitübertragungsschicht:

- Wieviel Volt entsprechen einer logischen 1 bzw. 0
- Wieviel Millisekunden dauert ein Bit
- Gleichzeitige Übertragung in beide Richtungen oder nicht?
- Wie kommt die erste Verbindung zustande und wie wird sie wieder gelöst
- Wie ist der Stecker für den Netzanschluss mechanisch aufgebaut?

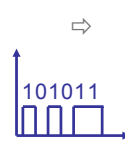
ANALOGUE UND DIGITALE SIGNALE

- **Analoge Signale:** Kontinuierliche Veränderungen physikalischer Größen (z.B. elektrische Spannung, magnetische Feldstärke) mit der Zeit



⇒ Mikrophone, Lautsprecher, Rundfunk, Fernsehen, klassische Telephonie, Compact-Kassetten oder Schallplattenspieler beruhen alle auf der Verarbeitung analoger Signale

- **Digitale Signale:** Abrupter Wechsel zwischen diskreten physikalischen Zuständen (z.B. stromführend / nicht stromführend) mit der Zeit



⇒ Moderne Computertechnik, Compact Disks sowie die modernen digitalen Varianten der Telephonie, digitale Video- und Audiotechnik beruhen alle auf der Verarbeitung digitaler Signale

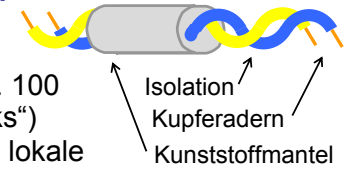
Verschiedene Medien sind zur Übertragung von Signalen geeignet:

- **Elektrische Übertragungsmedien** (Kabel)
 - ⇒ Gut geeignet für analoge Signale
 - ⇒ Mit Einschränkungen (geringe Reichweite) für digitale Signale
- **Elektromagnetische Wellen** (Funk)
 - ⇒ Für analoge Signale („Wellen“)
- **Optische Übertragungsmedien**
 - ⇒ Für digitale Signale („Ein-/Aus-switchen von Licht“)
 - ⇒ **Lichtwellenleiter** (Glasfaserkabel)
 - ⇒ Übertragung ohne Leiter (Infrarot, Laserstrecken)



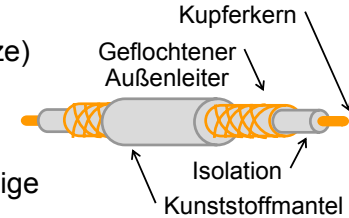
- **Verdrillte Kabel**, meist **Kabelpaare** (engl.: *twisted pair*)

- ⇒ preiswert
- ⇒ max. Übertragungsrate ca. 100 Mbit/s auf 100m (mit „Tricks“)
- ⇒ Beispiel: Telefonleitungen, lokale Computernetze



- **Koaxialkabel** (für sogenannte Breitbandnetze)

- ⇒ max. Übertragungsrate ca. 2 Gbit/s auf 2 km
- ⇒ Beispiel: Fernsehantenne, Kabelfernsehen, breitbandige Computernetze



- Bei der Übertragung von Signalen über elektrische Leitungen (verdrillte Kabelpaare, Koaxialkabel) treten bei zunehmender Leitungslänge Verluste auf.
 - ⇒ Dies verringert den Rauschabstand, d.h. das Verhältnis zwischen Signalstärke und Störungen.
 - ⇒ Abhängig vom Kabeltyp und von der maximal verwendeten Datenrate gibt es eine maximale nutzbare Leitungslänge.
- Abhilfe: Durch Verwendung von **Repeatern** (elektrischen Verstärkern) in regelmäßigen Abständen können die Signale aufgefrischt werden.
- Repeater arbeiten auf der Ebene der Bitübertragungsschicht, d.h. rein elektrisch. Die zu übertragenen Informationen werden nicht interpretiert.

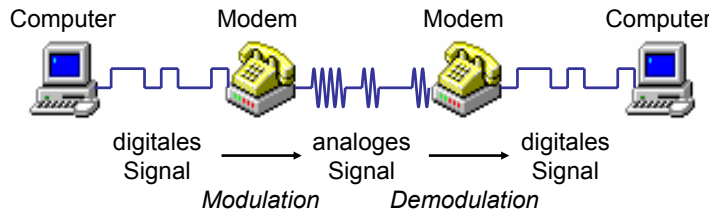
- Verschiedene Bereiche des elektromagnetischen Spektrums geeignet (s.a. Abb. 2.11 in Tanenbaum 2000)
- **Radiowellen** (10kHz-1GHz): Lang-, Mittel-, Kurzwelle, Amateurfunk, UKW (FM), Fernsehen, Mobilfunk
 - ⇒ rundstrahlend (omnidirektional)
 - ⇒ geradlinige Ausbreitung bei höheren Frequenzen (UKW, Fernsehen, Mobilfunk)
- **Mikrowellen** (1GHz-100GHz):
 - ⇒ Strahlung lässt sich bündeln mit Parabolantennen (quasioptische Ausbreitung)
 - ⇒ Richtfunkstrecken
 - ⇒ Satellitenkommunikation

- **Lichtwellenleiter** (Glasfaserkabel):
 - ⇒ Übertragungsrate ähnlich Koaxialkabel (im Gigabit-Bereich, potenziell noch besser)
 - ⇒ verwendet für Hochgeschwindigkeitsnetze und Fernnetze
 - ⇒ Totalreflektion von Lichtwellen verringert Verluste
 - ⇒ erfordert LED (*Light Emitting Diode* = Lichtdiode) oder Laser als Sender, Fotodiode als Empfänger
 - Lichtwellenübertragung ohne Leiter:
 - ⇒ **Infrarot** (z.B. zur Verbindung von PCs, Notebooks, Palmtops und Druckern in einem einzelnen Raum)
 - ⇒ **Laserstrecken** zur Informationsübertragung
- Sichtbares Licht, Infrarot und Ultraviolett zählen eigentlich auch zu den elektromagnetischen Wellen.

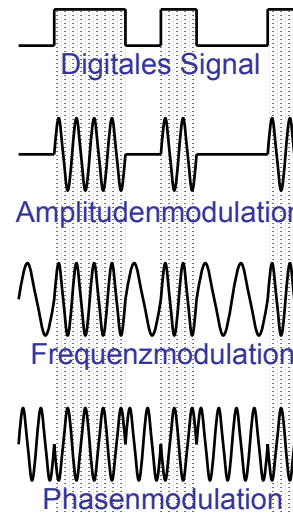
- Jeder analoge Übertragungskanal besitzt eine Grenzfrequenz, d.h. Schwingungen mit höheren Frequenzen werden nicht mehr übertragen. Diese Frequenz heißt auch die **Bandbreite**.
- Frequenzen werden gemessen in Hz (Hertz): **1 Hz = 1/sec**
- Der Begriff Bandbreite stammt aus der Rundfunktechnik: Die Bandbreite entspricht dem „Breite“ eines Senders auf der Rundfunkskala.
 - ⇒ Beispiel: Wenn ein Sender auf der Frequenz f Signale mit einer Bandbreite b überträgt, kann der nächste Sender (theoretisch) erst wieder auf der Frequenz $f+b$ bzw. $f-b$ senden, ohne dass es zu Störungen kommt.
 - ⇒ Die Bandbreite eines Senders ist maßgeblich für die höchste durch den Sender übertragene Frequenz und damit für die effektive Klangqualität.

- Die Leistungsfähigkeit eines digitalen Übertragungskanals wird in **Bit/s** (Anzahl übertragener binärer Zustände pro Sekunde) gemessen und als **Datenrate** bezeichnet.
- Oft müssen digitale Signale zur Übertragung oder Aufzeichnung in analoge Signale gewandelt werden
 - ⇒ Es besteht ein linearer Zusammenhang zwischen der Bandbreite eines analogen Kanals und der maximal erzielbaren Datenrate. Darüber hinaus wird die Datenrate durch den Rauschabstand (= Signalstärke / Stärke des Rauschens) beeinflusst.
 - ⇒ **Shannons Theorem** (1948):
Max. Datenrate = Bandbreite $\log_2(1 + \text{Rauschabstand})$

- Die direkte Übertragung digitaler Signale über elektrische Kabel oder Funk stößt auf Schwierigkeiten, da rechteckig geformte Signale hohe Frequenzanteile enthalten, d.h. zur korrekten Übertragung ist eine hohe Bandbreite erforderlich.
- Deshalb wurden verschiedene Modulationsverfahren entwickelt, um digitale Signale in analoge Schwingungen mit möglichst geringer Bandbreite umzusetzen.
- Ziel ist es, bei gegebener Bandbreite die maximale Datenrate (entsprechend Shannons Theorem) zu erreichen
- Wandlung zwischen Digital- und Analogsignalen durch **Modems** (Modulator/Demodulator)

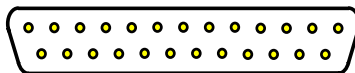


- Modems setzen digitale Signale in analoge um (**Modulation**) und umgekehrt (**Demodulation**).
- Modems ermöglichen dadurch die Übertragung von digitalen Signalen über das Telefonnetz
- Es sind Datenübertragungsraten von bis zu 56 kbit/sec möglich, die über das Telefonnetz bei einer nutzbaren Bandbreite von max. 4000 Hz analog übertragen werden.

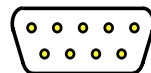


- Amplitudenmodulation:** Entsprechend dem digitalen Signal wird die Amplitude (Stärke) einer analogen Schwingung verändert.
 - Frequenzmodulation:** Hierbei wird die Frequenz einer analogen Schwingung verändert.
 - Phasenmodulation:** Der zeitliche Ablauf einer analogen Schwingung wird um einen bestimmten Anteil ihrer Schwingungsperiode verschoben.
- Für Modems werden in der Regel **Kombinationen** aus Amplituden- und Phasenmodulation benutzt.

- Serielle Schnittstellen werden zum Anschluss von verschiedenen Geräten an Computer verwendet:
 - ⇒ Modems, Computermaus, Bildschirmterminals, manchmal auch Drucker.
- Seriell bedeutet: Die Datenbits werden **nacheinander** übertragen. Es gibt in jede Richtung nur eine Datenleitung.
 - ⇒ Im Gegensatz zu parallelen Schnittstellen, z.B. zur häufig genutzten Centronics-Druckerschnittstelle. Hier werden 8 Bit = 1 Byte parallel übertragen.
- Anschluss über Kabel mit 25- oder 9-poligen Steckern.

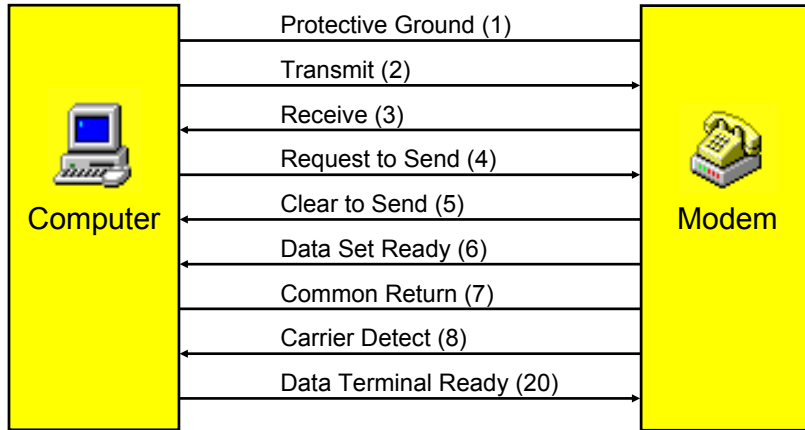


25-poliger Stecker

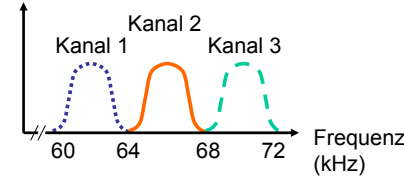


9-poliger Stecker

- Serielle Schnittstellen sind durch die Norm RS-232-C der Electronic Industries Association (EIA) und die weitgehend identische Empfehlung V.24 der CCITT genormt. (Eine noch neuere Norm ist durch RS-449 gegeben.)
- Mechanische Abmessungen der Stecker genau genormt.
- Elektrische Festlegung:
 - ⇒ Spannungen über 4 Volt = **high**, unter 3 Volt = **low**.
 - ⇒ Maximale Kabellänge = 15m bei 20 Kbit/sec
- Funktionale Spezifikation:
 - ⇒ Festlegung der Bedeutung der Anschlüsse (Stifte) der Stecker (siehe nächste Folie).
 - ⇒ Datensignale sind „active low“, d.h. ein gesetztes Bit entspricht einer Spannung unter 3 Volt.
 - ⇒ Steuersignale sind „active high“, d.h. ein gesetztes Bit entspricht einer Spannung über 4 Volt.

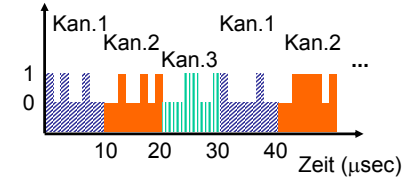


Multiplexverfahren dienen dazu, um über einen (meist: physischen) Kommunikationskanal mehrere logische Kommunikationskanäle zu realisieren:



Frequenzmultiplexverfahren

(Abk.: FDM= frequency division multiplexing): Das verfügbare Frequenzspektrum wird auf verschiedene logische Kanäle aufgeteilt, ähnlich wie auf einer Rundfunkskala. Geeignet für **analoge** Kanäle.



Zeitmultiplexverfahren

(Abk.: TDM=time division multiplexing): Die logischen Kanäle erhalten abwechselnd nacheinander Zugriff auf den physikalischen Kommunikationskanal. Geeignet für **digitale** Kanäle.

- Das Zeitmultiplexverfahren wird gerne verwendet, um in der klassischen Telefonie viele Gespräche gleichzeitig auf einer einzigen Glasfaserleitung zu übertragen.
- Das Zeitmultiplexverfahren funktioniert aber nur mit digitalen Signalen zufriedenstellend.
- Deshalb müssen Telefongespräche vor der Übertragung über lange Strecken in der Regel in digitale Form gewandelt werden und nach der Übertragung wieder zurückgewandelt werden.
- Die hierfür verwendeten Wandler heißen Codec (**Coder/Decoder**)

Die Funktionsweise von Codecs beruht auf der **Pulscode-modulation (PCM)** zur Digitalisierung von Analo signalen.

- **Coderfunktion**: Am Startpunkt der Übertragung wird die Stärke des Analo signals periodisch gemessen und in 8 Bit (256 diskrete Werte) kodiert (in USA oft: 7 Bit = 128 Werte).
- Diese Messung erfolgt mit einer Abtastrate von **8000/sec** (d.h. alle 125µsec)
 - ⇒ Nach einem Theorem von Nyquist (1924) kann damit eine Grenzfrequenz von 4000 Hz rekonstruierbar übertragen werden. (Allgemein: Die Abtastrate muss doppelt so hoch sein wie die Grenzfrequenz.)
- Die Übertragung erfolgt über Kanäle (sog. **PCM-Kanäle**) mit der Datenrate $8 \times 8000 = 64000 \text{ Bit/sec}$ (in USA meist $7 \times 8000 = 56000 \text{ Bit/sec}$).
- **Decoderfunktion**: Am Zielpunkt werden die übertragenen digitalen Werte in elektrische Spannungstufen gewandelt.

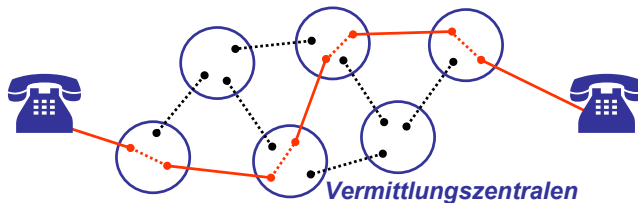
MODEMS UND CODECS: WAS IST DER UNTERSCHIED?

- **Modems** ermöglichen die Übertragung digitaler Signale über analoge Übertragungsstrecken.
 - ⇒ Senderseite: Digital-Analog-Wandlung durch Modem
 - ⇒ Analoge Übertragung (über Zweidraht-Telefonleitung)
 - ⇒ Empfängerseite: Analog-Digital-Wandlung durch Modem
- **Codecs** ermöglichen die Übertragung analoger Signale über digitale Übertragungsstrecken.
 - ⇒ Senderseite: Analog-Digital-Wandlung durch Codec (Ergebnis: Reihe von digitalen Messwerten)
 - ⇒ Digitale Übertragung (über Glasfaser)
 - ⇒ Empfängerseite: Rekonstruktion des ursprünglichen analogen Signals durch Codec

IN DER PRAXIS VERWENDETE ZEITMULTIPLEXINGVERFAHREN

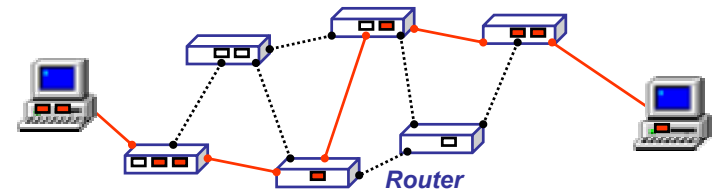
- Der **E1-Träger** (CCITT-Norm) ist ein Kanal, der die Daten von 32 PCM-Kanälen im Zeitmultiplexing überträgt:
 - ⇒ 2 PCM-Kanäle werden für Steuerdaten (Zeichengabe) verwendet, daher 30 PCM-Kanäle nutzbar
 - ⇒ Datenrate: $32 \times 64000 \approx 2 \text{ Mbit/sec}$
 - ⇒ Weiteres Multiplexing liefert Kanäle mit 128, 512, 2048 und 8192 PCM-Kanälen
- Der **T1-Träger** (in Nordamerika und Japan verwendet) überträgt 24 PCM-Kanäle mit 64000 Bit/sec (davon meist nur 56000 Bit/sec nutzbar) sowie einen 1-Bit-Synchronisationskanal mit 8000 Bit/sec:
 - ⇒ 1 PCM-Kanal dient der Zeichengabe, daher 23 nutzbar
 - ⇒ Datenrate: $24 \times 64000 + 8000 \approx 1,5 \text{ Mbit/sec}$
 - ⇒ Weiteres Multiplexing liefert die Träger $T2 = 4 \times T1$, $T3 = 6 \times T2$, $T4 = 7 \times T3$


LEITUNGSVERMITTLUNG (CIRCUIT SWITCHING)



- angewendet in der klassischen Telefonie
- Leitungen verbinden Telefone mit Vermittlungszentralen und Vermittlungszentralen untereinander.
- Vor der Kommunikation muss ein Ende-zu-Ende-Pfad aus miteinander verbundenen Leitungen eingerichtet werden.
- Nach der Kommunikation wird der Pfad wieder abgebaut.
- In der Praxis ist alles etwas komplizierter, da Leitungen auch gemultiplext werden können.

PAKETVERMITTLUNG (PACKET SWITCHING)



- Es wird für die Dauer der Kommunikation keine Verbindung hergestellt.
- Nachrichten werden in einzelne Datenpakete  zerlegt (erfordert Digitalisierung)
- Statt Vermittlungszentralen werden sogenannte Router genutzt. Datenpakete werden in den Routern zwischengespeichert und weitergeleitet, sobald eine Leitung in Richtung des Ziels frei ist.

Merkmal	Leistungs- vermittlung	Paket- vermittlung
Durchgehender „Kupferpfad“	Ja	Nein
Verfügbare Bandbreite	Fest	Dynamisch
Potenzielle Verschwendung von Bandbreite bzw. Datenrate	Ja	Nein
Übertragung mit Zwischenspeicherung	Nein	Ja
Durchgängig selbe Route benutzt	Ja	Nein
Verbindungsaufbau notwendig	Ja	Nein
Punkt möglicher Überlastungen	Beim Verbindungsaufbau	Bei jedem Paket
Gebührenberechnung	Pro Minute	Pro Paket

(Quelle: Tanenbaum 2000, Abb. 2.36)

Integrated Services Digital Network (ISDN):

- voll digitales leitungsvermittelltes Telefonsystem
- konzipiert durch CCITT (Comité Consultatif International Télégraphique et Téléphonique, dem internationalen „Telekom-Dachverband“) im Jahr 1984
- Ziel: Integration von Sprach- und Datendiensten
- Neue Leistungsmerkmale gegenüber der klassischen Telefonie (dem „*plain old telephone service*“ - POTS):
 - ⇒ Mehrfachrufnummer, Rufnummernübermittlung, Anrufumleitung, geschlossene Benutzergruppen, Dreierkonferenz, Anklopfen, Fangschaltung, Makeln, Parken, Subadressierung, Tarifinfo am Display, detaillierte Rechnung

ISDN-ARCHITEKTUR

- Basiskonzept: digitale Pipeline zwischen Kunden und Telephongesellschaft
- Die „letzte Meile“ zum Kunden erfolgt über vorhandene Zweidrahtleitungen
- Zweierlei Arten von Anschlüssen:
 - ⇒ für Privatkunden und kleine Firmen: sog. NT1-Dose zum Anschluss eines Buskabels für bis zu acht Geräte (Telefon, Fax, Computer ...)
 - ⇒ für größere Unternehmen: Anschluss einer Nebenstellenanlage an eine oder mehrere NT1-Dosen

DIE ISDN-SCHNITTSTELLE

Die ISDN-Pipeline unterstützt mehrere Kanäle, die durch das Zeitmultiplexverfahren aufgeteilt werden, darunter als wichtigste:

- B: digitaler Kanal für Sprache oder Nutzdaten (PCM-Kanal mit 64000 Bit/sec bzw. in USA 56000 Bit/sec)
- D: digitaler Kanal (16000 Bit/sec) für Steuerdaten

Wichtigste Anschlusstypen sind:

- Basisanschluss: 2B + 1D
- Primärmultiplexanschluss: 30B + 1D (entspricht dem E1-Träger), in USA: 23B + 1D (entspricht T1-Träger)

ISDN ist stark auf die (heute) „schmalbandige“ Übertragung mit 64000 Bit/sec hin orientiert und wird daher auch N-ISDN genannt = Narrow ISDN (Schmalband-ISDN) im Gegensatz zu Breitband-ISDN (siehe folgende Folie).

BREITBAND-ISDN (B-ISDN)

Breitband-ISDN:

- digitale virtuelle Leitung zum Übertragen von Paketen mit fester Größe mit 155 Mbit/sec.
- Revolutionäre Technologie, die vorhandenen Kabelleitungen auf der letzten Meile zum Kunden müssen ersetzt werden
- Übertragungsleistung reicht aus für Video on Demand
- Beruht auf dem Verfahren ATM (Asynchronous Transfer Mode)
 - ⇒ ATM ist paketvermittelt, nicht leitungsvermittelt
 - ⇒ Durch Bandbreitenreservierung auf den Routern können aber „virtuelle Leitungen“ reserviert werden

ASYMMETRIC DIGITAL SUBSCRIBER LINE (ADSL)

- Neuer Dienst für Telefon-Teilnehmer („**Subscriber**“) als Übergangslösung in Richtung Breitband-ISDN/ATM
- **Asymmetrische** (d.h. unterschiedliche) Datenraten für die Übertragung von der Telefongesellschaft zum ADSL-Teilnehmer (Downlink) bzw. umgekehrt (Uplink)
- Durch fortschrittliche Modulationstechniken kann die bestehende Zweidrahtverkabelung verwendet werden.
- ADSL kombiniert:
 - ⇒ einen herkömmlichen Analogkanal (alternativ einen N-ISDN-Basisanschluss mit 2 PCM-Kanäle) für die klassische Telefonie
 - ⇒ einen digitalen Downlink-Kanal mit bis zu 6 Mbit/sec
 - ⇒ einen digitalen Uplink-Kanal mit bis zu 640 Kbit/sec
- **Anwendung:** Video on Demand, Surfen im Web (beides erfordert hohe Datenraten für Downlink, geringe für Uplink)

MEHR ZU ADSL

- ADSL hat Ähnlichkeiten mit einer Modem-Übertragung.
- Im Gegensatz zu Modem-Übertragung muss ADSL nur die „letzte Meile“ bis zu einem Glasfaseranschluss (i.d.R. beim lokalen Vermittlungsamt) überbrücken.
- Dadurch fällt die Beschränkung auf die max. Bandbreite 4000Hz (entsprechend 64000 oder 56000 Bit/sec bei einer 8- bzw 7-Bit-Abtastung mit der Rate 8000/sec) weg.
- Dennoch erfordert ADSL sehr fortschrittliche Modulationsverfahren, damit die bestehende Kupferverkabelung genutzt werden kann.
- Von der deutschen Telekom wird das ADSL-Produkt TDSL angeboten. TDSL bietet in der Standardausführung
 - ⇒ Downlinkraten von max. 768 kbit/sec
 - ⇒ Uplinkraten von max. 128 kbit/sec

TEIL 2: SICHERUNGSSCHICHT (DATA LINK LAYER)

Übertragung von Daten zwischen zwei benachbarten (d.h. direkt durch eine Leitung verbundenen) Computern

- Fehlerfreie Übertragung von Daten mit Hilfe von **Rahmen** (= voneinander abgrenzbare Bitfolgen)
 - ⇒ besondere Muster als Rahmengrenzen, die innerhalb des Rahmens nicht auftreten dürfen
 - ⇒ „Datenrahmen“ und „Bestätigungsrahmen“
 - ⇒ Wiederholung der Übertragung im Fehlerfall, Erkennung und Eliminierung von Duplikaten
 - ⇒ Geschwindigkeitsanpassung
- **MAC-Teilschicht** (Media Access Control): Regelung des Zugriffs auf das Übertragungsmedium in einem sog. **Broadcast-Netz**, in dem alle Stationen denselben Kanal benutzen

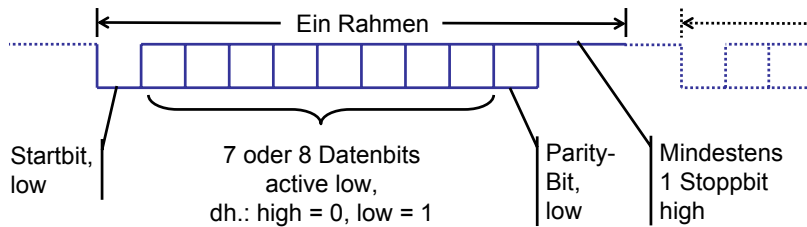
Die Sicherungsschicht beschreibt, wie zwei **benachbarte** Computer miteinander kommunizieren. Die Kommunikation kann dabei über zwei Arten von Kanälen laufen:

- **Punkt-zu-Punkt-Kanäle** verbinden genau **zwei Stationen** im Netz miteinander. Beispiele:
 - ⇒ Langstreckenverbindung zwischen zwei benachbarten Routern in einem Internet-Backbone-Netz
 - ⇒ Die Telefonleitung zwischen dem Modem eines Internetbenutzers und dem Modem des Providers
 - ⇒ die serielle Leitung zwischen einem Computer und einem Modem.
- **Broadcast-Kanäle** (engl. *broadcast* = Rundfunk) verbinden **eine Gruppe von Stationen** im Netz miteinander. Dies wird im Rahmen der MAC-Teilschicht behandelt. Beispiel:
 - ⇒ Lokales Netz auf Basis Ethernet

Auf der Sicherungsschicht werden die Daten in Form von Rahmen, d.h. abgrenzbaren Bitfolgen übertragen. Hierfür gibt es verschiedene Möglichkeiten:

- Parallele Übertragung von Datenrahmen. Beispiel Parallelport zur Druckeransteuerung. Ein Rahmen bestehend aus 8 Bit wird über acht parallele Leitungen + weitere Steuerungsleitungen gleichzeitig übertragen
- Serielle Übertragung von Datenrahmen. Eine Bitfolge wird über eine einzelne Leitung nacheinander übertragen. Dies erfordert Anfangs- und Endekennungen für Datenrahmen
 - ⇒ Asynchrone serielle Übertragung: Zwischen den Datenrahmen sind Pausen
 - ⇒ Synchrone serielle Übertragung: es gibt keine Pausen, bzw. in Pausen werden „Leerzeichen“ gesendet.

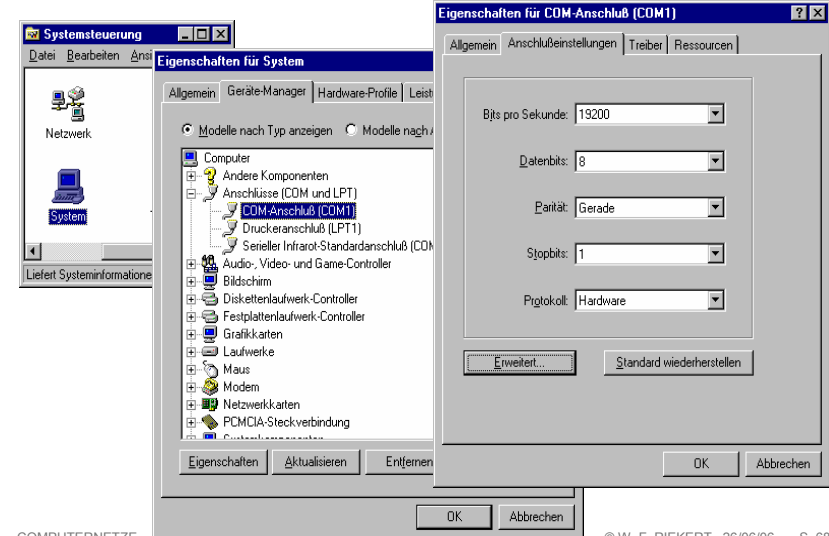
SERIELLE DATENÜBERTRAGUNG MIT RAHMEN FESTER LÄNGE



Rahmen für die serielle Datenübertragung nach V.24 / RS232C:

- Rahmen fester Länge (z.B. 11 Bit im Beispiel)
- zeichenorientiert: Übertragung von 7- oder 8-Bit-Zeichen
- asynchron, d.h. zwischen Zeichen sind Pausen möglich
- Fehlerkontrolle durch optionales Parity-Bit (Prüfsummenbit)
- Start des Rahmens angezeigt durch Startbit (low)
- Ende angezeigt durch mindestens 1 Stopbit (high)

EINSTELLUNG DER SERIELLEN SCHNITTSTELLE



- **Anfangskennung**
- ggf. Adresse (falls mehrere Stationen eine Leitung benutzen)
- ggf. Nummerierung des Rahmens (für evtl. Wiederholung)
- **Daten**
- ggf. Prüfsumme
- **Endekennung**

Achtung: Anfangs und Endekennung dürfen innerhalb des Rahmens nicht vorkommen, damit das Protokoll eindeutig interpretierbar bleibt.

Abhilfe: Stopfen (Stuffing), s. Tanenbaum (2000) Abb. 3.4

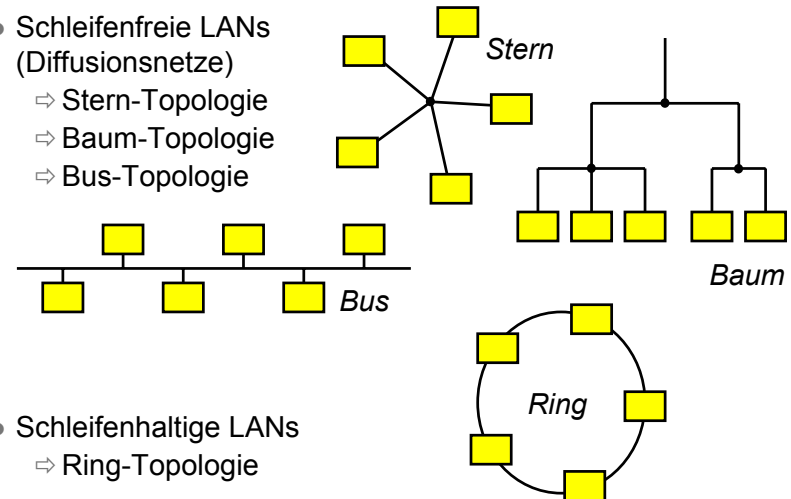
Protokoll für die Punkt-zu-Punkt-Verbindung zwischen dem Computer eines Internet-Benutzers und dem Server eines Providers über das Telephonnetz.

- Übertragung der Nachrichten in Form von „Rahmen“ mit Anfangs- und Endekennung sowie Fehlererkennung.
- Unterstützt verschiedene Protokolle auf der Vermittlungsschicht, z.B. IP (Internet), IPX (Novell), AppleTalk usw.
- Automatische Vergabe von Internet-Adressen (Adressen der Internet-Vermittlungsschicht, bestehend aus 4 Zahlengruppen, z.B. 193.196.176.114)
 - ⇒ Dadurch voller Internet-Zugang ohne besondere lokale Netzwerkkonfigurationsaufgaben möglich

(siehe auch Tanenbaum 2000, Abb. 3.26)

- dienen zur Vernetzung von PCs und anderen Computern innerhalb einer Institution
- ermöglichen die Nutzung von gemeinsamen Ressourcen (Betriebsmitteln):
 - ⇒ Dateien
 - ⇒ Geräte, insbesondere Drucker
 - ⇒ Rechenleistung
 - ⇒ allgemeine Netzwerkdienste (Email, WWW, ...)
- bestehen (wie alle Netze) aus Hardware (Leitungen, Elektronik, Computern) und Software
- sind typischerweise Broadcast-Netze, d.h. es wird nur ein Übertragungskanal für alle Computer genutzt
 - ⇒ ein Fall für die MAC(Media Access Control)-Teilschicht (Teilschicht der Sicherungsschicht)

- Schleifenfreie LANs (Diffusionsnetze)
 - ⇒ Stern-Topologie
 - ⇒ Baum-Topologie
 - ⇒ Bus-Topologie



- Schleifenhaltige LANs
 - ⇒ Ring-Topologie

- **Kabel**
 - ⇒ **Koaxialkabel**, auch als BNC (broadband network cable) bezeichnet, teuer, heute seltener
 - ⇒ **verdrillte Kabelpaare**, auch als TP („twisted pairs“) bezeichnet, preisgünstig, heute viel verwendet
- **Lichtwellenleiter** (LWL, Glasfaserkabel) in LANs seltener verwendet, da Technologie vergleichsweise teuer



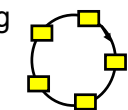
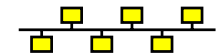
- Jede Art der Vernetzung ist durch maximale Kabellängen beschränkt
- **Ausweg:**
 - ⇒ Ausweitung des Netzes durch Aufteilung in einzelne **Netzsegmente**
 - ⇒ Verbindung der Segmente durch sogenannte Repeater
- Ein **Repeater** (oder **Hub**) arbeitet rein elektrisch und gehört deshalb im Schichtenmodell zur Bitübertragungsschicht 1: Er empfängt, verstärkt und überträgt Signale in beide Richtungen. Die verbundenen Segmente bilden jedoch zusammen **EIN** Netz.

Bridges (Brücken) dienen zur Verbindung unterschiedlicher LANS (mit unterschiedlichen Netztypen, Datenformaten und Übertragungsraten). Bridges arbeiten auf der Ebene 2 des Schichtenmodells (Sicherheitsschicht)

Nutzen von Bridges:

- Verbindung historisch gewachsener separater LANs
- Je ein LAN für einzelne, voneinander entfernte Gebäude vereinfacht Verkabelung
- Bessere Lastverteilung durch Aufteilung von LANs
- Große Gebäude erfordern Aufteilung von LANs, um maximale Kabellängen einzuhalten
- Zuverlässigkeit: Teilausfälle legen nicht das Gesamtnetz lahm
- Sicherheitsaspekte

- **Ethernet**
 - ⇒ Steuerungsverfahren: Kollisionserkennung und nochmaliges Senden nach zufallsgesteuerter Zeit
 - ⇒ Topologie: Bus (klassisch)
- **Token-Ring-Netze**
 - ⇒ Steuerungsverfahren: zyklisches Weiterreichen einer Sendeberechtigung (Token) zwischen den Stationen
 - ⇒ Topologie: Ring (klassisch)
- **Wireless LAN (WLAN)**
 - ⇒ Steuerungsverfahren: ähnlich wie Ethernet
 - ⇒ Übertragungsmedium: Funk



VERTEILER: ÜBERGANG ZUR STERNTPOLOGIE

Die klassischen Netzwerktopologien haben heute ihre Bedeutung verloren. In der Praxis werden heute **Verteiler** verwendet, an die die Computer sternförmig über TP- oder Glasfaserkabel angeschlossen sind. Der Verteiler „enthält“ dann die jeweilige Topologie.

Vorteile:

- Statt der teureren Koaxialkabel können die preisgünstigeren TP-Kabel verwendet werden, da der Verteiler die Funktion eines Repeaters übernimmt, der die Signale auffrischt.
- Computer mit Fehlfunktionen können durch den Verteiler vom Netz abgehängt werden und können die Übertragungen der anderen Teilnehmer nicht stören.

ARTEN VON VERTEILERN: HUBS UND SWITCHES

Es gibt zwei Arten von Verteilern:

- **Hubs** („Naben“) sind im einfachsten Fall elektrische Verstärker (Repeater) für die Signale und unterstützen nur eine Datenübertragung zu einem Zeitpunkt. Die Geschwindigkeit des Netzes teilt sich auf die Teilnehmer auf. Hubs arbeiten auf der Ebene 1 (Bitübertragungsschicht).
- **Switches** (Analogie: Switchboards der ersten Telefongeneration) unterstützen mehrere gleichzeitige Datenübertragungen durch das „Durchschalten“ von Verbindungen, so dass mehrere Teilnehmerpaare mit voller Geschwindigkeit des Netzes kommunizieren können. Switches arbeiten auf der Ebene 2 (Sicherheitsschicht) bzw. auf der Ebene 3 (Vermittlungsschicht).

PRAKTISCHE BEDEUTUNG DER VERSCHIEDENEN LAN-SYSTEME

- **Ethernet** ist Marktführer
 - ⇒ relativ preisgünstig in den einfachen Versionen
 - ⇒ von der Geschwindigkeit her ausbaufähig von 10 Mbit/s Gesamtleistung bis 1000 Mbit/s pro angeschlossenes Gerät (mit Switch)
- **Token Ring**
 - ⇒ vor allem benutzt in IBM-Großrechnerumgebungen (z.B. Banken)
 - ⇒ derzeit nicht schneller als 100 Mbit/s
 - ⇒ Marktanteil rückläufig („Toter Ring“)
- **Wireless LAN**
 - ⇒ zunehmende Bedeutung parallel zum Ethernet

ETHERNET

- Zugrundeliegendes Steuerungsverfahren: **CSMA/CD** (Carrier Sense Multiple Access Collision Detect)
 - ⇒ **Multiple Access**: Mehrere Stationen haben die Möglichkeit, auf das Medium (die Datenleitung) zuzugreifen (aber nicht gleichzeitig)
 - ⇒ **Carrier Sense**: Abhören des Kanals vor und bei dem Senden
 - ⇒ **Collision Detect**: Gleichzeitiger Zugriff auf das Medium wird erkannt.
- Verfahren genormt durch IEEE 802.3 / ISO 8802.3 ähnlich wie die anderen LAN-Systeme Token Ring (IEEE 802.5 / ISO 8802.5) und Wireless LAN (IEEE 802.11 / ISO 8802.11).

Übertragungsgeschwindigkeit 10MBit/s (veraltet)

10Base5: Basis-Ethernet, dickes Koax. („Yellow Cable“), max. 500m Länge

10Base2: „Cheapernet“, dünnes Koax., max. 200m

10BaseT: **Standard-Ethernet**, Hub oder Switch, verdrehtes Paar, max. 100m

Übertragungsgeschwindigkeit 100MBit/s (heute Standard)

100BaseT: **Fast-Ethernet**, Hub oder Switch, verdrehtes Paar, max. 100m

100BaseF: dto. aber mit Glasfaserkabel, max. 2000m

Übertragungsgeschwindigkeit 1000MBit/s (derzeit High End)

1000BaseT und 1000BaseF: **Gigabit-Ethernet**

Nächste Generation: 10 Gigabit-Ethernet (noch sehr teuer)

- Derzeit verbreitete Ethernet-Version mit 100Mbit/sec
- Viele handelsübliche Netzwerkkarten und Switches verstehen auch noch den älteren Standard mit 10Mbit/sec
 - ⇒ Solche Netzwerkkarten können für eine Übergangszeit auch im langsameren Standard-Ethernet laufen
 - ⇒ Komfortablere Switches ermöglichen es, dass vereinzelt alte 10Mbit/sec-Adapterkarten in einem Fast-Ethernet mitlaufen können
- funktionieren beide mit preiswertem TP-Kabel (= „besseres Telefonkabel“).
- Der Einsatz von Switches erlaubt die maximale Übertragungsgeschwindigkeit pro Teilnehmerstation

CLENTS UND SERVER IM LAN

- **Server** (d.h. Computer mit zugehöriger Serversoftware) stellen die Betriebsmittel bereit:
 - ⇒ Dateien: Fileserver
 - ⇒ Geräte, insbesondere Drucker: Printserver
 - ⇒ Rechenleistung: Applikationsserver
 - ⇒ Netzwerkdienste: Webserver, Mailserver ...
- Von **Clients** aus (d.h. von den Arbeitsplatzcomputern der Mitarbeiter) können diese Betriebsmittel genutzt werden.
- Vorsicht: Unter den Begriffen Client und Server wird oft auch die (Client- bzw. Server-)Software verstanden (z.B. Webserver, Email-Client) und nicht ein Computer
- Ein Computer kann je nach installierter Software auch zugleich als Client und Server dienen

NETZWERKSOFTWARE

- Die Serversysteme werden ausgestattet mit einem Netzwerk-Server-Betriebssystem:
 - ⇒ Novell IntranetWare (alter Name: „Netware“)
 - ⇒ Windows NT Server
 - ⇒ oder UNIX
- Die Clientsysteme benötigen sog. Requestersoftware (zzgl. zum eigentlichen Betriebssystem)
 - ⇒ wird entweder vom Netzwerksoftware-Hersteller geliefert (z.B. Novell Intranetware Client)
 - ⇒ oder in allen Microsoft-Windows-Betriebssystemen für die wichtigsten Netzwerktypen (NT, Novell) integriert

- Klassisch: Datei- und Druckdienste basierend auf **NetBIOS** (Network Basic Input Output System)
 - ⇒ Netzwerklaufwerke
 - ⇒ Netzwerkdrucker
- Heute: zusätzlich alle wichtigen Internetdienste durch Unterstützung der Internet-Protokolle. Hierdurch wird das LAN zum **Intranet**.
 - ⇒ E-Mail
 - ⇒ WWW
 - ⇒ News
 - ⇒ Telnet
 - ⇒ FTP
 - ⇒ ...

Steuerung des Betriebs des Subnetzes (der Subnetze):

- Routing, d.h. Auswahl der Paketrouten (statisch nach Tabellen oder dynamisch nach Netzauslastung etc.)
- Vermeidung von Staus bei hoher Netzbelastung
- Abrechnungsfunktion
- Verbindung heterogener Subnetze (z.B. mit unterschiedlichen Protokollen und Adressierungsarten)
- Beispiele:
 - ⇒ IP: Internet Protocol
 - ⇒ ATM: Vermittlungsschicht der ATM-Netzarchitektur (ATM= Asynchronous Transfer Mode), Basis für Breitband-ISDN der Telekom-Unternehmen

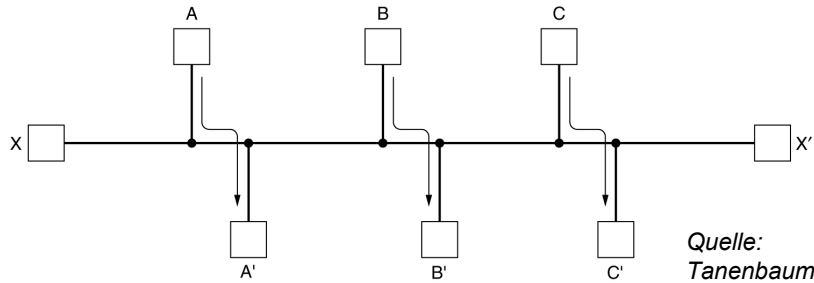
- Die Dienste sollen unabhängig von der Topologie des Subnetzes sein
- Die nächsthöhere Schicht, die Transportschicht, muß von der Anzahl, der Art, und der Topologie der vorhandenen Subnetze abgeschirmt werden
- Die für die Transportschicht vorgesehenen Netzadressen müssen ein einheitliches Nummerungsschema darstellen, auch zwischen LANs und WANs

Konsequenz:

Die Schnittstellen der Vermittlungsschicht nach oben sind noch netzweit einheitlich und verstecken die Unterschiede der Subnetze. Auf den nächsttieferen Schichten (Sicherheit, Bit-Übertragung) sind diese Unterschiede jedoch vorhanden.

- **Routing**: Weitervermitteln von Nachrichten in einem Netz auf der möglichst günstigsten Route. Hierzu gibt es sog. Routingalgorithmen.
 - ⇒ Günstig kann bedeuten: Schnell, preiswert
 - ⇒ Andere Ziele des Routingalgorithmus sind: Einfachheit, Robustheit, Stabilität, Fairness
- Das Routing wird im Wesentlichen von sogenannten **Routern** übernommen, speziellen Vermittlungscomputern, auf denen die Routingalgorithmen implementiert sind und die über eine Datenbasis verschiedener Übertragungsrouten verfügen.
- Ein normaler Computer, der eine Nachricht über eine ihm unbekannt Route schicken muss, nimmt hierzu die Dienste des nächstgelegenen Routers in Anspruch

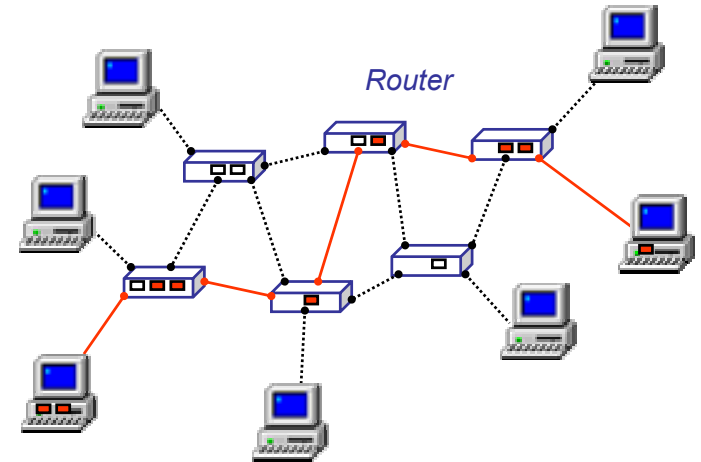
ROUTING: KONFLIKT ZWISCHEN FAIRNESS UND OPTIMALITÄT



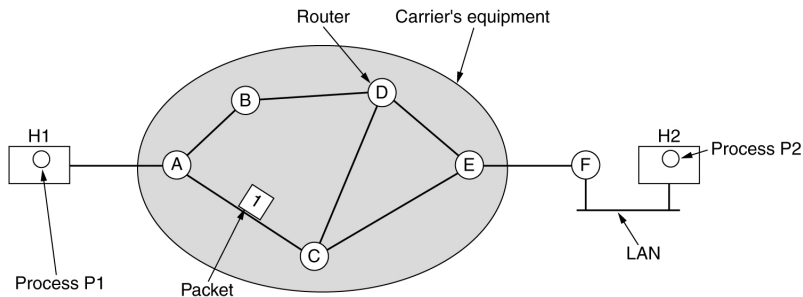
Optimal wäre dieses Netz ausgenutzt, d.h. der Durchsatz wäre am größten, wenn die Paare A-A', B-B', und C-C' ununterbrochen kommunizieren dürften.

Fair wäre es, wenn das Paar X-X' ebenso oft wie die anderen zum Zuge käme wie die anderen kommunizierenden Paare.

WEITERVERMITTLN VON NACHRICHTEN DURCH ROUTER



PAKETVERMITTLUNG NACH DEM PRINZIP "STORE & FORWARD"



Quelle: Tanenbaum

VERBINDUNGSORIENTIERUNG AUF EBENE 3: JA ODER NEIN?

Zur Frage der Verbindungsorientierung der Vermittlungsschicht gibt es zwei gegensätzliche Lager mit folgenden Auffassungen:

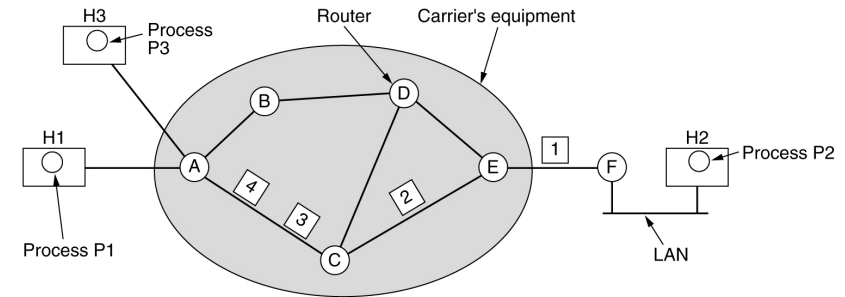
- Auffassung der **Netzbetreiber** (z.B. Telekoms): Vor jeder Übertragung muss eine **Verbindung** hergestellt werden (siehe Telefonsystem). Diese Verbindung erhält eine spezielle Kennung und wird so lange benutzt, bis die Verbindung abgebaut wird. Qualitätsmerkmale wie Datenrate, Fehlerüberwachung und Flusststeuerung sind beim Verbindungsaufbau verhandelbar.
- Auffassung des **Internet-Lagers**: Das Subnetz ist prinzipiell unzuverlässig. Fehlerüberwachung und Flusststeuerung sind Aufgabe der Transportschicht. Daher: **verbindungslose Dienste** auf der Vermittlungsschicht. Versenden einzelner Datenpakete, die alle mit der vollständigen Empfängeradresse „beschriftet“ sind.

DIE VIRTUELLE VERBINDUNG

Verbindungsorientierte Vermittlungsdienste lassen sich mit dem Konzept der virtuellen Verbindung realisieren:

- Beim Aufbau einer virtuellen Verbindung wird eine Route, d.h. ein Übertragungsweg über eine Reihe von Routern (Vermittlungscomputern) festgelegt.
- Über diese Route werden alle Daten übertragen, bis die Verbindung abgebaut ist.
- Jeder Router führt Buch über die existierenden virtuellen Verbindungen, die über ihn führen.
- Vorteil: Datenpakete müssen keine kompletten Zieladressen enthalten, nur Verbindungsnummern.
- Vorteil: Vermeidung von Überlastung, da beim Aufbau der Verbindung Bandbreite reserviert werden kann.
- Nachteil: Routerausfälle auf der vorab gewählten Route bedeuten Abbruch der Verbindung.
- Beispiel: ATM (Asynchronous Transfer Mode)

DIE VIRTUELLE VERBINDUNG: IMPLEMENTATION MIT TABELLEN



A's table		C's table		E's table	
H1	1	C	1	A	1
H3	2	C	2	C	2
In	Out	A	2	E	2
		A	1	E	1
		E	1	F	1
		E	2	F	2

Quelle: Tanenbaum

ATM: EIN VERBINDUNGSORIENTIERTER VERMITTLUNGSDIENST

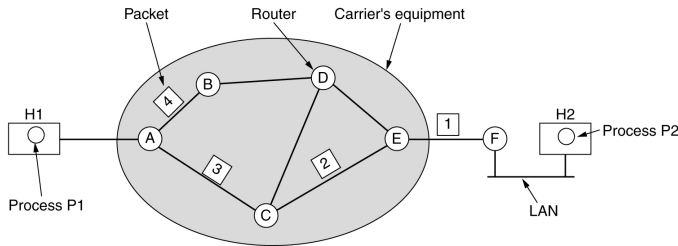
- ATM (Asynchronous Transfer Mode), ein vom Internationalen Verband der Telekom-Unternehmen (ITU) genormter Dienst, ist die Grundlage für den Telekommunikationsdienst B-ISDN (Breitband-ISDN)
- ATM stellt einen verbindungsorientierten Vermittlungsdienst bereit
- ATM sieht oberhalb der Vermittlungsschicht auch nur verbindungsorientierte Dienste vor, denn im Normalfall gelten verbindungslose Dienste oberhalb verbindungsorientierter Dienste nicht als gutes Design
- Allerdings ist es möglich und auch sinnvoll, Internet-Protokolle oberhalb von ATM zu führen. Dann sind auch verbindungslose Dienste oberhalb ATM möglich.

VERBINDUNGSLOSE VERMITTLUNGSDIENSTE

Verbindungslose Vermittlungsdienste lassen sich mit Datengrammen („Daten-Telegrammen“) realisieren:

- Verbindungslose Vermittlungsdienste übertragen die Daten in Form voneinander unabhängiger Datengramme.
- Routen werden nicht im Voraus festgelegt, nachfolgende Pakete können auch andere Routen nehmen
- Jedes Datengramm muss die volle Zieladresse enthalten (z.B. ein Dutzend Byte oder mehr)
- Nachteil: Volle Zieladressierung bei kleinen Datenpaketen großer Overhead (Mehr „Aufschrift“ als Inhalt)
- Vorteil: Verfahren sehr robust. Falls ein Router ausfällt, wird andere Route gewählt.
- Vorteil: Schnell, vor allem bei kleinen Datenmengen, da auf Verbindungsauf- und abbau verzichtet wird.
- Beispiel: IP (Internet Protocol)

VERBINDUNGSLOSE VERMITTLUNG: IMPLEMENTATION



A's table		C's table	E's table
initially	later		
A : -	A : -	A : A	A : C
B : B	B : B	B : A	B : D
C : C	C : C	C : -	C : C
D : B	D : B	D : D	D : D
E : C	E : B	E : E	E : -
F : C	F : B	F : E	F : F

Dest. Line

Quelle: Tanenbaum

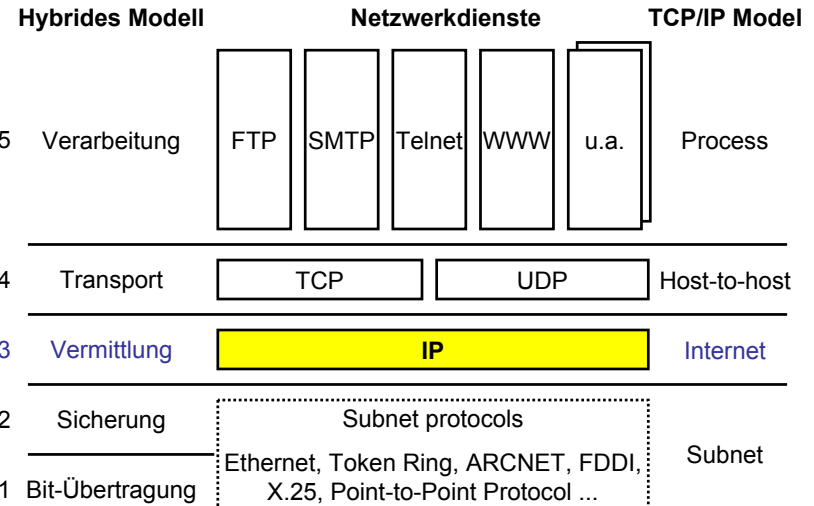
IP: EIN VERBINDUNGSLOSER VERMITTLUNGSDIENST

- IP (Internet-Protocol), die Vermittlungsdienst des Internet, ist ein verbindungsloser Datengrammdienst.
- Es wird über IP-Adresse (Subnet ID / Host ID) ein Rechner in einem Netzwerk („Subnet“) adressiert.
- Zuverlässigkeit nicht garantiert („Best Effort“). Zuverlässigkeit ist die Aufgabe von Diensten höherer Schichten (TCP)
- Unterhalb von IP sind beliebige (auch relativ unzuverlässige Subnetze möglich)
- Oberhalb von IP existieren verbindungslose und verbindungsorientierte Transportdienste
 - ⇒ TCP: verbindungsorientierter Transportdienst
 - ⇒ UDP: verbindungsloser Transportdienst

DAS INTERNET ALS VERBUNDNETZ

- Das Internet ist ein Verbundnetz, das unterschiedliche „Subnetze“ verbindet, z.B.:
 - ⇒ DFÜ-Netz (d.h. Datenfernübertragungsnetz): dient u.a. zur Verbindung von Internet-Benutzern und Internet-Providern über Telefonleitungen.
 - ⇒ Backbone-Netze: Netz aus schnellen Übertragungsstrecken zwischen sogenannten Routern
 - ⇒ Lokale Netze: z.B. vom Typ Ethernet oder WLAN
- Jede Art von Subnetz hat eigene Vorgaben für die Gestaltung von Bitübertragungsschicht und Sicherungsschicht. Das Internet schränkt diese nicht ein.
- Die Vermittlungsschicht im Internet (IP) ist jedoch einheitlich festgelegt, unabhängig vom zugrundeliegenden Subnetz.

IP (INTERNET PROTOCOL): DER INTERNET-VERMITTLUNGSDIENST

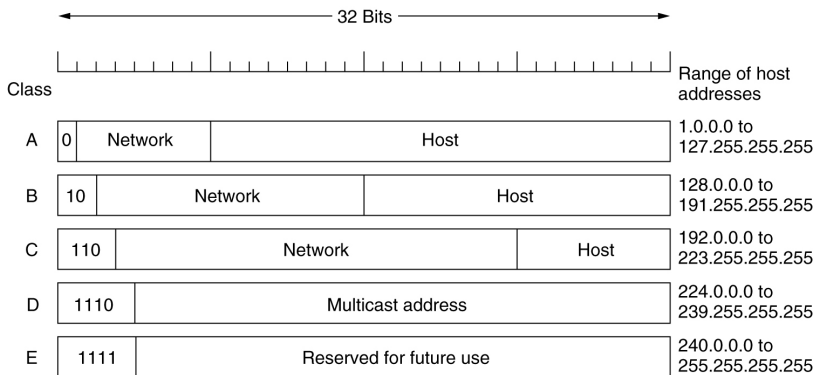


Die Adressierung im Internet erfolgt über Internet-Adressen (auch „IP-Adressen“ genannt)

- IP-Adressen bestehen aus vier durch Punkte getrennte Zahlengruppen, z.B. **193.196.176.30**
- In der derzeit gebräuchlichen Internet-Version IPv4 ist jede Zahlengruppe durch 8 Bit dargestellt und kann die Werte 0 bis 255 annehmen. Dadurch sind 2^{32} = rund 4 Milliarden Internetadressen möglich.
- In der künftigen Internet-Version IPv6 werden 16 Bit (statt 8) für 8 (statt 4) Zahlengruppen verwendet, die hexadezimal notiert werden. Beispiel für eine IPv6-Adresse: **2001:0db8:85a3:08d3:1319:8a2e:0370:7344**. Dadurch sind künftig 2^{128} = ca. $3,4 \cdot 10^{38}$ unterschiedliche Internetadressen möglich.

- IP-Adressen: 32 Bit lange Zahlen, typischerweise als vier Zahlengruppen dargestellt, z.B.: 193.196.176.30
- IP-Adressen sind aufgeteilt in Network ID und Host ID
- Ursprünglich gab es nur drei Arten (Klassen) von Aufteilungen:
 - ⇒ Class A: 7 Bit Network ID, 24 Bit Host ID
 - ⇒ Class B: 14 Bit Network ID, 16 Bit Host ID
 - ⇒ Class C: 21 Bit Network ID, 8 bit Host ID
- Um den Adressraum besser auszunutzen, verwendet man heute klassenlose Adressformate,
 - ⇒ Hier erfolgt die Aufteilung in Subnet ID und Host ID mit Hilfe einer so genannten Subnet-Maske

KLASSEN VON IP-ADRESSFORMATEN



Quelle: Tanenbaum

KLASSENLOSE ADRESSFORMATE AUFTEILUNG MIT SUBNETZMASKE

Eigenschaften für TCP/IP

Subnet-Id = 193.196.176.0 = 11000001.11000100.10110000.00000000

Host-Id = 379 = 1.01111011

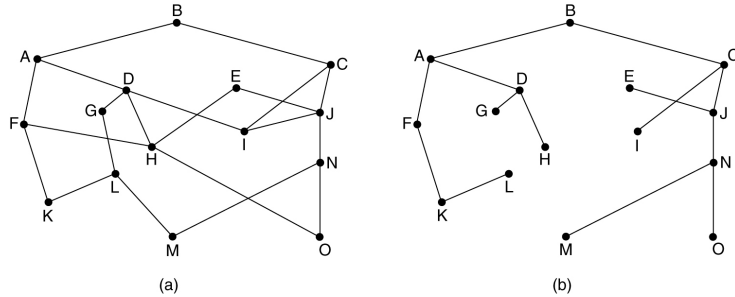
IP-Adresse: 193.196.177.123

Subnet-Maske: 255.255.254.0

Binary breakdown of IP: 11000001.11000100.10110001.01111011

Binary breakdown of Mask: 11111111.11111111.11111110.00000000

VERMITTLUNGsalgorithmen: Das Optimalitätsprinzip

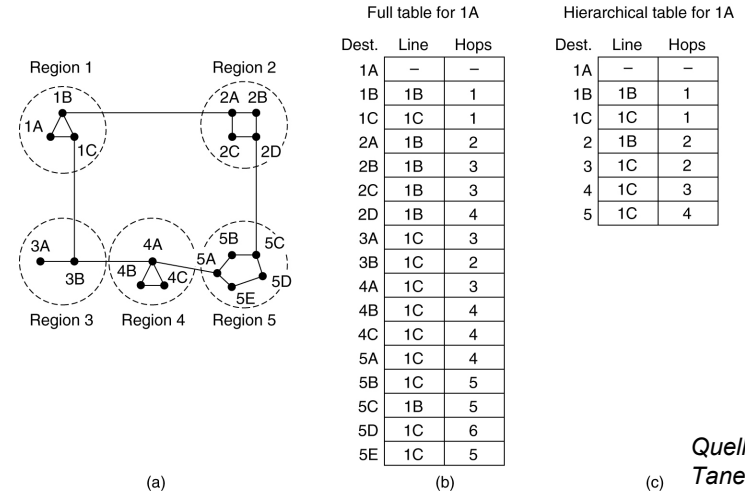


Quelle: Tanenbaum

Optimalitätsprinzip: Wenn der optimale Pfad von B nach M über N führt, dann ist auch der Teilpfad bis N der optimale Pfad von B nach N.

Folge: Alle von B ausgehenden optimalen Pfade bilden einen Baum, den so genannten „Sink Tree“ („Abflußbaum“)

HIERARCHISCHES ROUTING



HIERARCHISCHES ROUTING IM INTERNET ÜBER DIE SUBNET-ID

- IP-Adressen bestehen aus zwei Teilen, der Subnet-Id (identifiziert das Subnetz) und der Host-Id (identifiziert den Computer im Subnetz).
- Mit Hilfe der Subnet-Mask, die für jedes Subnetz festgelegt ist, lässt sich die Host-Id von der Subnet-Id trennen.
- Jeder Router hat Tabellen, die die Menge aller IP-Adressen in verschiedene, u.U. auch sehr große Subnetze zerlegen (dargestellt durch Subnet-Id und Subnet-Mask).
- Diese Tabellen beschreiben, welche Subnetze der Router über eine Netzwerkkarte direkt erreicht und welche nur über einen benachbarten Router erreicht werden.
- Auf diese Weise kann ein Router stets entscheiden,
 - ⇒ ob er ein IP-Paket selbst direkt zustellen kann
 - ⇒ oder ob er es an den nächsten zuständigen Router weiterleiten muss und welcher Router das ist.

DER IP-HEADER: DER KOPFTEIL VON IP-DATENGRAMMEN

IP-Datengramme bestehen aus Kopfteil (Header) und Textteil. Wichtige Datenelemente des Headers sind:

- Version:** z.Zt.=4, Im künftigen IPV6 = 6
- Total Length:** Länge von Header+Text
- Source Address:** IP-Adresse des Senders
- Destination Address:** IP-Adresse des Empfängers
- Time to Live:** Ein Zähler, der bei jeder Teilstrecke, d.h. bei jedem Router heruntergezählt wird, dient zur Begrenzung der „Lebensdauer“ eines Pakets
- Protocol:** Bezeichnung des Transportprozesses, i.d.R. TCP oder UDP

- **MAC-Adresse** oder **Physikalische Adresse** (meist eine Ethernet-Adresse), z.B.: 00-A0-24-DF-F6-98, verwendet in MAC-Teilschicht der Sicherungsschicht (Nr. 2). Liegt bereits hardwaremäßig in der Netzwerkkarte fest. Nicht routingfähig, erreicht nur Computer im lokalen Netz
- **Internet-Adresse** (IP-Adresse), z.B.: 193.196.176.114 verwendet in Vermittlungsschicht (Ebene Nr. 3) des Internet, muss nach Absprache mit dem Netzwerkadministrator oder Internetprovider eingestellt werden
- **Domain-Name**, z.B.: mars.iuk.hdm-stuttgart.de verwendet in Transport- und Verarbeitungsschicht (Ebenen Nr. 4 und 5) des Internet, kann nach Absprache mit dem Netzwerkadministrator oder Internetprovider vergeben werden. Domain-Namen werden durch sog. Domain-Name-Server in IP-Adressen umgewandelt.

Um einen Computer in einem lokalen Netz manuell für die Nutzung des Internets einzurichten, müssen verschiedene Einstellungen vorgenommen werden (s.a. LAN-Praktikum):

- Festlegung der **eigenen IP-Adresse** und der **Subnet-Mask** des lokalen Netzes (erhältlich vom Netzwerkadministrator bzw. Internetprovider),
- Festlegung der IP-Adresse eines **Gateways**, d.h. des Routers, der den Zugang zum Rest des Internets herstellt und alle IP-Pakete erhält, die nicht im LAN bleiben sollen.
- Einrichtung des Domain Name Systems (DNS):
 - ⇒ Festlegung des **eigenen Domain-Namens** (in Absprache mit Netzwerkadministrator/Internetprovider)
 - ⇒ Festlegung der IP-Adresse des **Domain Name Servers**

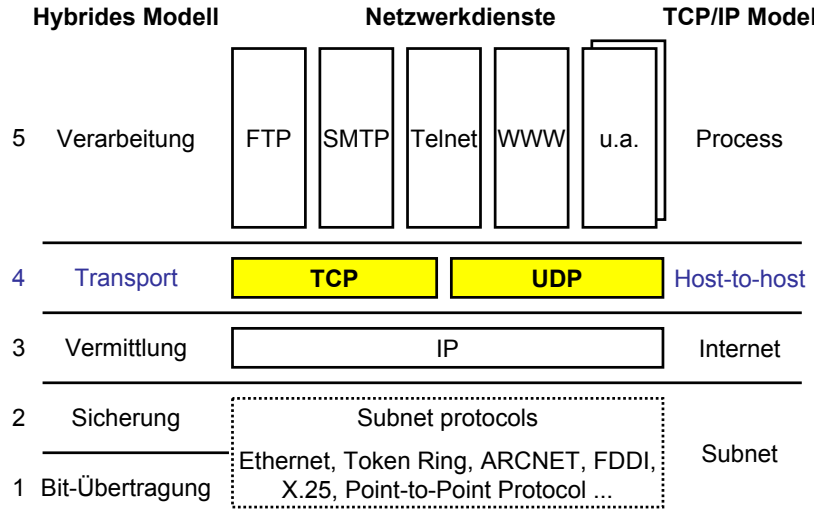
Für die Kommunikation mit anderen Hosts oder dem Gateway in einem LAN muss die IP-Schicht IP-Adressen in Adressen der Sicherungsschicht (MAC-Adressen) konvertieren, das sind meist Ethernet-Adressen (48-Bit lang, weltweit eindeutig):

Mögliche Lösungen:

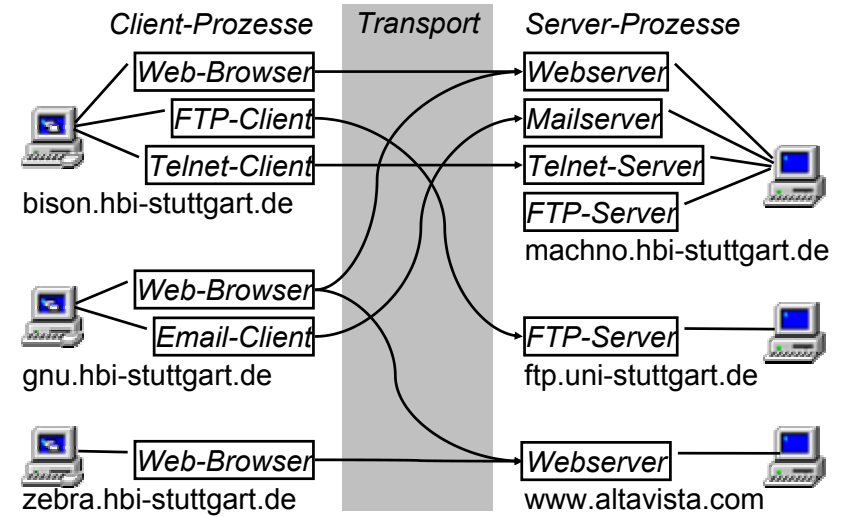
- Tabellen mit Zuordnung IP-Adresse - MAC-Adresse auf jeder Maschine
 - ⇒ pflegeaufwendig, fehleranfällig
- Vor dem Senden einer Nachricht zuerst ein Broadcast (Rundruf): „Wem gehört diese Internet-Adresse“ und lokales Abspeichern der Antwort (mit Verfallsdatum)
 - ⇒ Dies wird so realisiert im **ARP (Address Resolution Protocol)**

- Echte Ende-zu-Ende-Schicht: ermöglicht die Kommunikation zwischen zwei Prozessen auf unterschiedlichen Rechnern
- Verschiedene Arten von Transportdiensten möglich, z.B. verbindungsorientierter Transport (z.B. TCP) oder verbindungsloser Transport über Datagramme (z.B. UDP) oder als Broadcast an viele Empfänger
- Benennungsmechanismus für die Endpunkte einer Kommunikationsbeziehung zwischen zwei Prozessen
- Ggf. Zerlegung der Nachrichten in kleinere Einheiten und Zusammensetzen in richtiger Reihenfolge beim Empfänger
- Multiplexen von Kanälen der Vermittlungsschicht, damit mehrere Prozesse über dieselbe Übertragungsrouten quasi gleichzeitig kommunizieren können
- Flusststeuerung zur Geschwindigkeitsanpassung

INTERNET-TRANSPORTDIENSTE: TCP UND UDP



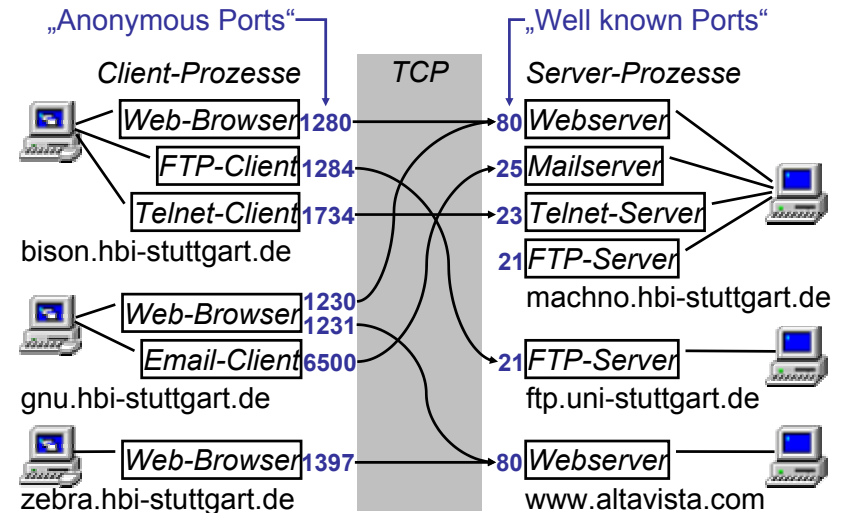
TRANSPORTSCHICHT: KOMMUNIKATION ZWISCHEN PROZESSEN



TRANSMISSION CONTROL PROTOCOL (TCP)

- Internet-Dienst der Transportschicht
- Verbindungsorientiert (Phasen Verbindungsaufbau, Datenübertragung, Verbindungsabbau)
- Verlässlicher Verbindungsaufbau, geregelter Verbindungsabbau
- Zuverlässigkeit: verlustfreie, fehlerfreie Datenübertragung; richtige Reihenfolge der Nachrichten
- Zerlegung der Nachrichten in kleinere Einheiten und Zusammensetzen in richtiger Reihenfolge beim Empfänger
- Vollduplex: Beide Seiten können jederzeit senden und empfangen
- Datenstromartige Schnittstelle, Nachrichtengrenzen bleiben nicht erhalten

PORTS ALS SERVICE ACCESS POINTS FÜR DEN TCP-DIENST



PORTS ALS ENDPUNKTE VON TCP-VERBINDUNGEN

- Ports bilden die **Endpunkte** (Service Access Points) von TCP-Verbindungen. Intern sind die Ports Tabelleneinträge, in denen die TCP-Software über die vorhandenen Verbindungen Buch führt.
- Ports werden mit **Nummern** bezeichnet. Diese Nummern sind innerhalb eines Computers eindeutig.
- An bestimmten, per Konvention bekannten Ports (*well-known ports*, Portnummer in der Regel kleiner als 1024) warten **Serverprozesse**, bis ein Clientprozess mit ihnen Verbindung aufnimmt.
- **Clientprozesse** benutzen untereinander unterschiedliche, ansonsten weitgehend beliebige Ports (*anonymous ports*, Portnummer i.d.R. größer als 1023), um eine Verbindung zu den Ports von Serverprozessen aufzunehmen.
- **Verbindungen** sind eindeutig definiert durch Angabe von IP-Adresse (oder Computernamen) und Portnummer auf Client- und auf Serverseite.

WELL-KNOWN PORTS

Kleine Portnummern bis ca. 1023 sind entsprechend einer Übereinkunft aller Internet-Serverbetreiber für bestimmte Serverprozesse (sog. Demons) vorgesehen. Beispiele:

Port	Transportdienst	Serverprozess	Zweck
21	TCP	FTP Demon	File Transfer
23	TCP	Telnet Demon	Virtuelles Terminal
25	TCP	SMTP Demon	Versenden von Email
37	UDP	Time Demon	Uhrzeit-Server
79	TCP	Finger Demon	Info über Benutzer
80	TCP	HTTP Demon	Webserver
139	TCP	NETBIOS	File-/Printservices

Eine vollständige Liste aller well-known Ports befindet sich auf jedem Unix-Rechner in der Datei `/etc/services`

BEISPIEL EINES PORTS

Portnummern sind oft sichtbar in WWW-Adressen (URLs).

Beispiel:

<http://www.theautochannel.com:8080/>

(Datum des letzten Zugriffs 27. April 2000)

Der Webserver auf dem Computer mit dem Domainname `www.theautochannel.com` akzeptiert Verbindungen auf dem Port 8080.

Normalerweise verwenden Webserver den Port mit der Nummer 80. Deshalb dient die 80 als Voreinstellung („Default“), wenn in der URL keine Portnummer angegeben ist.

USER DATAGRAM PROTOCOL (UDP)

- ein Internet-Dienst der Transportschicht (Host-to-host), ebenso wie TCP
- Verbindungsloser Dienst
- Schnittstellen zu UDP sind ähnlich gestaltet wie die zu TCP, zur Adressierung werden ebenfalls Ports verwendet
- UDP-Ports unterscheiden sich von TCP-Ports; ein UDP-Port kann dieselbe Nummer haben wie ein TCP-Port, ohne dass die beiden Ports etwas miteinander zu tun haben
- Es werden Datagramme übertragen
- Nachrichtengrenzen bleiben erhalten
- Erhaltung der Reihenfolge der Datagramme nicht garantiert
- Zuverlässigkeit nicht garantiert („Best Effort“)
- Schneller als TCP

Der TCP-Dienst wird durch sogenannte Sockets-Software realisiert. Im folgenden orientieren wir uns an der Software, die für die Programmiersprache Java angeboten wird.

- Socket ist das englische Wort für Steckdose
- Sockets sind Software-Objekte (bestehend aus Daten und Funktionen), die mit einem Port auf dem lokalen Rechner assoziiert sind.
- Zwei Prozesse, die miteinander kommunizieren möchten, müssen hierzu je einen Socket erzeugen und zwischen den Sockets eine Verbindung herstellen.
- Miteinander verbundene Sockets stellen ihren Prozessen Lese- und Schreibfunktionen bereit. Alles was auf einen Socket geschrieben wird, kann an dem anderen verbundenen Socket gelesen werden.

Sockets besitzen wie alle Objekte *Merkmale* (dargestellt durch Daten) und *Operationen* (dargestellt durch Funktionen)

- **Merkmale** (Daten) eines Sockets
 - ⇒ Ferne IP-Adresse, Ferner Port
 - ⇒ Lokale IP-Adresse, Lokaler Port
- **Operationen** (Funktionen) eines Sockets
 - ⇒ Konstruktor: Einen neuen Socket mit o.a. Merkmalen erzeugen und Verbindung herstellen
 - ⇒ Senden von Nachrichten
 - ⇒ Empfangen von Nachrichten
 - ⇒ Schließen der Verbindung

- Mit einem normalen Socket (auch Client Socket genannt) kann ein Clientprozess eine Verbindung zu einem Port auf einem fremden Computer herstellen.
- Erforderlich ist es aber, dass dort ein Serverprozess die Verbindungsanforderung akzeptiert.
- Dies geschieht mit Hilfe eines **Server Sockets**.
- Server Sockets werden durch Serverprozesse erzeugt.
- Server Sockets können auf Verbindungsanforderungen warten und diese akzeptieren.
- Das **Akzeptieren** einer Verbindungsanforderung bedeutet, dass serverseitig ein normaler Socket erzeugt wird, der mit dem anfordernden normalen clientseitigen Socket verbunden ist. Die eigentliche Kommunikation erfolgt dann über die Verbindung dieser zwei Sockets.

- **Merkmale** (Daten) von Server Sockets:
 - ⇒ **Lokale IP-Adresse, lokaler Port**
 - ⇒ **Backlog** = Länge der Warteschlange für Verbindungsanforderungen
- **Operationen** (Funktionen) von Server Sockets:
 - ⇒ **Konstruktor**: Erzeugen eines neuen Server Sockets mit den o.a. Merkmalen
 - ⇒ **Accept**: Warten bis Verbindungsanforderung eintrifft, dann diese akzeptieren und schließlich einen normalen Socket erzeugen, der die Verbindung bedient.
 - ⇒ **Schließen** des Server Sockets

- Ein Server Socket ist imstande, ganze Warteschlangen von Verbindungsanforderungen abzuarbeiten.
- Dabei kann ein Server Socket bereits neue Verbindungsanforderungen annehmen, wenn die alten Verbindungen noch nicht geschlossen sind.
- Es existieren dann mehrere Sockets, die demselben lokalen Port zugeordnet sind, aber mit unterschiedlichen fremden IP-Adresse/Port-Kombinationen verbunden sind.
- Meist wird für jeden dieser Sockets ein eigener (Sub-) Prozess gestartet, der dann die jeweilige Verbindung bedient. In Java können alternativ die effizienteren „Lightweight Processes“ oder **Threads** („Fäden“) verwendet werden. Mehrere dieser Threads können als „Handlungsfäden“ quasi gleichzeitig innerhalb eines einzelnen Betriebssystemprozesses ablaufen.

- Öffnen Sie verschiedene TCP-Verbindungen, indem Sie z.B. Telnet- und FTP-Sessions mit den Rechnern machno oder v öffnen oder indem Sie das WWW nutzen.
- Starten Sie in einer DOS-Box das Programm **netstat** bzw. **netstat -n**. Sie sehen dann die offenen TCP-Verbindungen angezeigt.

```

C:\WINDOWS>netstat

Active Connections

Proto Local Address           Foreign Address         State
TCP   rip:1088                v.hbi-stuttgart.de:ftp CLOSE_WAIT
TCP   rip:1368                machno.hbi-stuttgart.de:telnet ESTABLIS
TCP   rip:1380                www.altavista.com:80   TIME_WAIT
TCP   rip:1382                www.altavista.com:80   SYN_SENT
TCP   rip:1383                www.altavista.com:80   ESTABLISHED
TCP   rip:1384                www.altavista.com:80   ESTABLISHED
    
```

```

netstat
tcp 0 0 machno.hbi-stuttgart:www host-47.subnet-74.:4831 ESTABLISHED
tcp 0 0 machno.hbi-stuttgart:www surprise.mchh.sie:62050 TIME_WAIT
tcp 0 3998 machno.hbi-stuttgart:www ABD1481A.ipt.aol.c:1294 ESTABLISHED
tcp 0 0 machno.hbi-stuttgart:www kuen103.kuen.fh-he:1832 TIME_WAIT
tcp 0 1313 machno.hbi-stuttgart:www ABD1481A.ipt.aol.c:1293 ESTABLISHED
tcp 0 0 machno.hbi-stuttga:4898 machno.hbi-stutt:domain TIME_WAIT
tcp 0 0 machno.hbi-stuttga:4897 auth02.ns.de.uu.:domain TIME_WAIT
tcp 0 0 machno.hbi-stuttgart:www bw11olt.bluewin.c:62980 TIME_WAIT
tcp 0 0 machno.hbi-stuttgart:www stulir7-101-170.ra:1131 TIME_WAIT
tcp 0 0 machno.hbi-stutt:telnet ripc.hbi-stuttgart:1268 ESTABLISHED
tcp 0 0 machno.hbi-stuttga:www stulir7-101-170.ra:1126 FIN_WAIT2
tcp 0 1 machno.hbi-stuttga:4875 ierrs1.ier.uni-stu:smtp SYN_SENT
tcp 0 0 machno.hbi-stutt:telnet A8b2b.pppool.de:1113 ESTABLISHED
tcp 0 0 machno.hbi-stutt:telnet a02-18.dialin.msh.:1091 ESTABLISHED
tcp 0 0 machno.hbi-stutt:telnet astasp2.hbi-stuttg:2154 ESTABLISHED
tcp 0 0 machno.hbi-stuttga:ssh steinbock.hbi-stut:1299 ESTABLISHED
tcp 0 0 machno.hbi-stuttgart:20 r-44.stuttgart.ipd:1051 ESTABLISHED
tcp 0 0 machno.hbi-stuttgar:ftp r-44.stuttgart.ipd:1044 ESTABLISHED
    
```

```

netstat -n
tcp 0 0 193.196.176.135:80 62.158.165.55:2068 TIME_WAIT
tcp 0 0 193.196.176.135:80 151.189.0.129:45520 TIME_WAIT
tcp 0 0 193.196.176.135:23 62.155.181.56:1117 ESTABLISHED
tcp 0 0 193.196.176.135:80 195.186.27.6:4695 TIME_WAIT
tcp 0 0 193.196.176.135:80 195.186.27.6:4694 TIME_WAIT
tcp 0 0 193.196.176.135:80 195.186.27.6:4693 TIME_WAIT
tcp 0 0 193.196.176.135:80 195.186.27.6:4692 TIME_WAIT
tcp 0 0 193.196.176.135:80 195.186.27.6:4650 TIME_WAIT
tcp 0 0 193.196.176.135:80 195.186.27.6:4643 TIME_WAIT
tcp 0 0 193.196.176.135:80 213.200.1.41:1421 TIME_WAIT
tcp 0 0 193.196.176.135:80 142.231.43.247:1615 FIN_WAIT2
tcp 0 0 193.196.176.135:80 142.231.43.247:1612 TIME_WAIT
tcp 0 0 193.196.176.135:80 142.231.43.247:1611 TIME_WAIT
tcp 0 2 193.196.176.135:23 193.196.176.114:1268 ESTABLISHED
tcp 0 1 193.196.176.135:4875 129.69.80.12:25 SYN_SENT
tcp 0 0 193.196.176.135:23 213.6.139.43:1113 ESTABLISHED
tcp 0 0 193.196.176.135:23 212.4.224.82:1091 ESTABLISHED
tcp 0 0 193.196.176.135:23 193.196.176.132:2154 ESTABLISHED
tcp 0 0 193.196.176.135:22 193.196.176.91:1299 ESTABLISHED
tcp 0 0 193.196.176.135:20 62.180.41.44:1051 ESTABLISHED
tcp 0 0 193.196.176.135:21 62.180.41.44:1044 ESTABLISHED
    
```

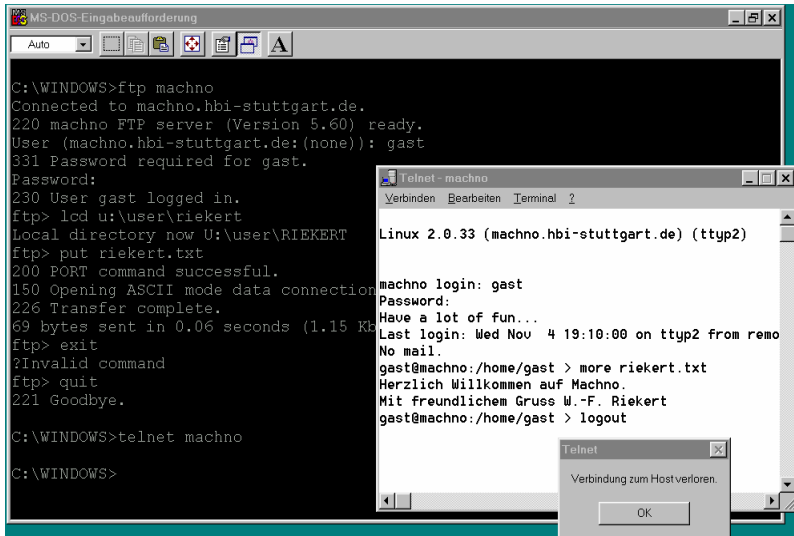
TEIL 5: VERARBEITUNGSSCHICHT (APPLICATION LAYER)

- Realisiert durch Prozesse (ablaufende Programme), die miteinander über die Transportschicht kommunizieren
 - ⇒ In der Regel Unterscheidung von **Clientprozess** (Dienstanforderer) und **Serverprozess** (Diensterbringer)
 - ⇒ Beispiele: Telnet-, FTP-, Email-, WWW-Server u. -Clients
- Verarbeitungsschicht in unserem Hybridmodell entspricht der **Anwendungsschicht** im OSI-Modell, umfasst aber zusätzlich die Aufgaben der folgenden zwei OSI-Schichten
 - ⇒ **Sitzungsschicht** (session layer): Verwaltung von sog. Sitzungen, z.B. Login Sessions oder Filetransfers
 - ⇒ **Darstellungsschicht** (presentation layer): Kodierung von Daten auf standardisierte Weise, z.B. Buchstaben, Zahlen, Geldbeträge, Rechnungen usw.

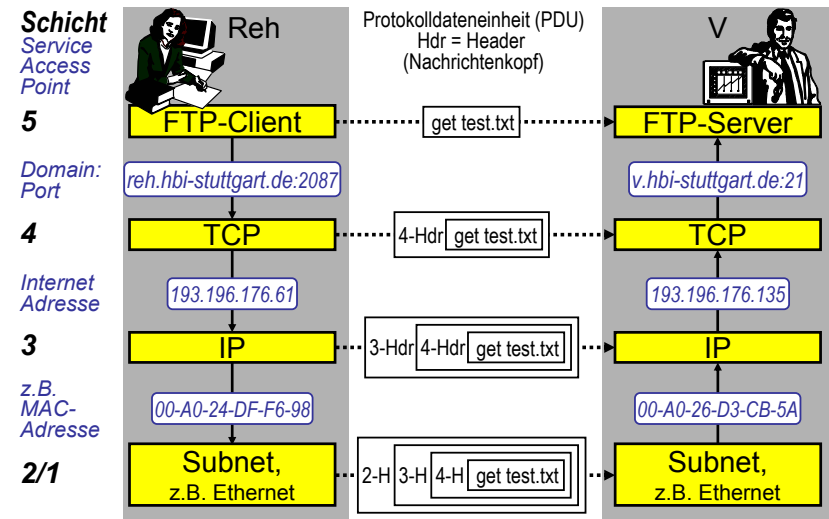
NETZWERKDIENTSTE DER VERARBEITUNGSSCHICHT

- Terminalemulation (TELNET, RLOGIN)
- Dateitransfer (FTP)
- Elektronische Post (SMTP, POP3, IMAP, MIME)
- WWW (HTTP, HTML, CGI, Java) - umfasst auch die vorgenannten Dienste
- Verzeichnisdienste (X500, NDS, LDAP, ADS, DNS)
- Net News (NNTP)
- netzbasiertes Fenstersystem (X Window System)
- Nutzung von fernen Programmen (RSH, REXEC, RPC, RMI, CORBA)
- Nutzung von fernen Dateien (NFS)
- Nutzung von fernen Datenbanken (z.B. JDBC)
- Synchrone Kommunikation (TALK, WRITE, Chat)
- Katalogdienste (Z39.50)
- Netzwerkmanagement (SNMP)

FTP-DIENST UND TELNET-DIENST: BILDSCHIRMABZÜGE



ROLLE DER SCHICHTEN AM BEISPIEL DES FTP-DIENSTES



DER TELNET-DIENST

- virtuelles Terminal
 - ⇒ Das klassische Bildschirmterminal eines „Mainframe“-Computers (Großrechners) wird emuliert
- Bildschirm funktioniert zeilenorientiert, nicht seitenorientiert
- Telnet-Client zur Darstellung des Bildschirmterminals
- Telnet-Server verbunden mit zeilenorientiertem Kommandointerpreter (z.B. UNIX Shell)
- Telnet ist verbindungsorientiert
- Verbindungseigenschaften aushandelbar, z.B.: Welche Art von Terminal wird emuliert?
- Konfigurierung des Telnet-Clients durch Eingabe von Escape-Sequenzen oder über Menüs des Telnet-Fensters
- Auf vielen Servern ist der Telnet-Dienst deaktiviert und durch SSH ersetzt (Vorteil: verschlüsselte Übertragung).

TELNET SESSION AUF DEM UNIX-RECHNER V

- *Zuerst Telnet aus DOS-Box starten:*
Start ⇒ Programme ⇒ MS-DOS-Eingabeaufforderung
C:\WINDOWS> **telnet v** (Hostname oder IP-Adresse)
- *Ein neues Fenster erscheint, der UNIX-Rechner fordert auf zum Login. Sie geben ein: user=it01, password=******
- *Sie können nun UNIX-Kommandos eingeben*
 - > **ls** Verzeichnis listen
 - > **more riekert.txt** Datei ausgeben
 - > **cp riekert.txt kopie.txt** Datei kopieren
 - > **cat kopie.txt** Kopie anzeigen
 - > **rm kopie.txt** Kopie löschen
 - > **logoff** Ausloggen
- *Telnet beenden über Menü oder „Kreuz“ am Fensterrand*

DER „TELNET-CLIENT“ ALS CLIENT-SIMULATOR

Das Protokoll, mit dem Telnet-Client und Telnet-Server kommunizieren, ist praktisch identisch mit der TCP-Kommunikationsschnittstelle

- Alle Zeichen, die man in das Telnet-Fenster hineintippt, sendet der Telnet-Client unverändert mit TCP an den Server weiter
- Alle Zeichen, die der Server über TCP an den Client schickt, zeigt dieser unverändert im Telnet-Fenster an.

Da man den Telnet-Client mit einer beliebigen Portnummer als Aufrufparameter starten kann (statt der standardmäßigen Portnummer 23), kann Telnet beliebige Server ansprechen, die Verbindungen über TCP akzeptieren und deren Protokoll mit druckbaren Zeichen auskommt. Allerdings sollte man in Telnet das lokale Echo einschalten, sonst sieht man nicht, was man tippt. Auch darf man sich nicht vertippen!

INTERNET MAIL

- Mail-Client und Mail-Server kommunizieren über die Protokolle **SMTP** zum Senden sowie **POP3** oder **IMAP** zum Lesen von Email
- Email-Nachrichten sind gegliedert in Header und den eigentlichen Nachrichtentext. Aufbau des Headers im Internet genormt durch **RFC822**
- Erweiterung des Headers durch **MIME** (Multipurpose Internet Mail Extensions), genormt durch **RFC1521**
 - ⇒ Typisierte Nachrichten (MIME Types)
 - ⇒ Mehrere Teile (Multipart Messages)

Siehe auch Abb.: 7.42 - 7.45 (Tanenbaum 2000)

EIN MIT TELNET SIMULIRTER EMAIL-CLIENT „SPRICHT“ SMTP (1)

```
C: telnet smtp.hdm-stuttgart.de 25
S: 220 smtp.hdm-stuttgart.de SMTP service ready
C: HELO adler.iuk.hdm-stuttgart.de
S: 250 smtp.hdm-stuttgart.de says hello to adler.iuk.hdm-stuttgart.de
C: MAIL FROM: <westbomke@hdm-stuttgart.de>
S: 250 sender ok
C: RCPT TO: <riekert@hdm-stuttgart.de>
S: 250 recipient ok
C: DATA
S: 354 Send mail; end with "." on a line by itself
C: From: westbomke@hdm-stuttgart.de
C: To: riekert@hdm-stuttgart.de
C: Subject: Hi
C: Grüße aus der HdM!
C: J.W.
C: .
S: 250 message accepted
C: QUIT
S: 221 smtp.hdm-stuttgart.de closing connection
```

Legende:
C = Client
S = Server

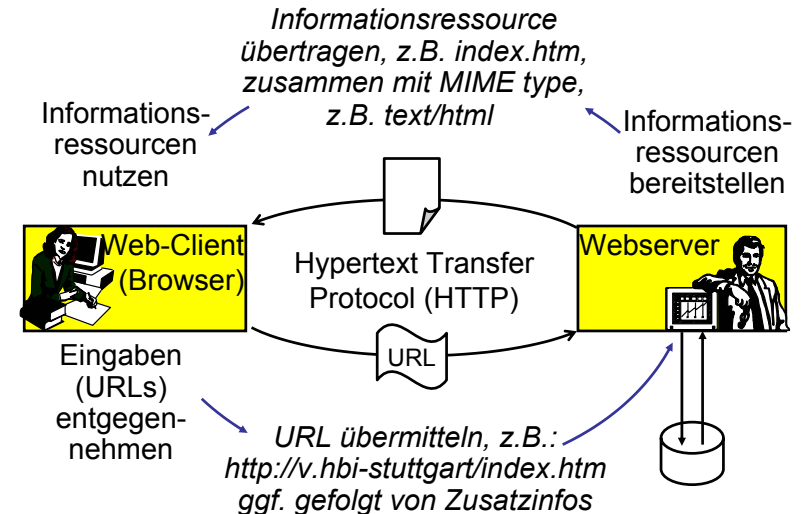
EIN MIT TELNET SIMULIRTER EMAIL-CLIENT „SPRICHT“ SMTP (2)

DAS WORLD WIDE WEB (WWW)

- Client:** Internet-Browser (z.B. Netscape Navigator, Microsoft Internet Explorer)
- Server:** Webserver (z.B. Microsoft Internet Information Server, Apache)
- Dienst:** Bereitstellen von Hypertextseiten und anderen Informationsressourcen (typisiert mit MIME Types) nach Angabe einer Adresse, der URL (Uniform Resource Locator)
- Art des Dienstes:** Verbindungsloser Anfrage/Antwort-Dienst
- Protokoll:** Hypertext Transfer Protokoll (HTTP).

Siehe auch Abb. 7.59-7.67 in Tanenbaum (2000)

WEB-CLIENT (BROWSER) UND WEBSEVER



- erhält eine Informationsressourcenanforderung, welche im Wesentlichen aus einer URL besteht
- stellt die Informationsressource bereit
 - ⇒ statisch: Informationsressource wird aus dem Dateisystem geholt
 - ⇒ oder dynamisch: Informationsressource wird durch ein Programm generiert
- legt den MIME-Type der Informationsressource fest: z.B. text/html oder anderer Typ (z.B. GIF-Grafik, Winword-Dokument, PDF-Dokument, ...)
- und schickt die Informationsressource zusammen mit dem MIME-Type an den Client (Internet-Browser) zurück

- verarbeitet die vom Web-Server erhaltenen Informationsressourcen abhängig von deren Typ
 - ⇒ direkte Anzeige: HTML-Seiten, GIF- bzw. JPEG-Grafiken
 - ⇒ direkte Ausführung: JavaScript, ActiveX Controls (letzteres nur Microsoft Internet Explorer)
 - ⇒ Anzeige/Ausführung über Plug-In (nachladbare Browser-Erweiterung): z.B. Acrobat Reader, Java Plugin
 - ⇒ Anzeige/Ausführung durch sog. Helper Application: z.B. Winword für Doc-Files usw.
- nimmt Eingaben von URLs an und leitet diese weiter an Web-Server
 - ⇒ Direkteingabe über Tastatur
 - ⇒ Anklicken von Hyperlinks (mit URL hinterlegte Bereiche)

UNIFORM RESOURCE LOCATOR (URL)

URLs adressieren weltweit eindeutig Informationsressourcen (d.h. Daten, Dienstprogramme und multimediale Dokumente):

Aufbau: *Protokoll://Domain:Port/Pfad*

Beispiel: <http://www.faw.uni-ulm.de:9876/RESEDA/RESEDA.html>

(Die Zeichen //, :, / sind syntaktische Kennzeichnungen für die verschiedenen Elemente der URL)

Protokoll: = Übertragungsprotokoll
(http: = Hypertext Transfer Protocol)

//Domain = Bezeichnung des Servercomputers im Internet

:Port = Kommunikationsport des Webserver-Programms, i.d.R. nicht erforderlich, da Standardwert = 80

/Pfad = Ortsangabe im Dateisystem des Servers, bestehend aus Verzeichnis(pfad) und Dateiname

URLs: VARIANTEN

Relative URLs: Hypertextseiten enthalten oft relative Links. Das Protokoll, die Domain und der Schrägstrich vor dem Verzeichnispfad werden dann weggelassen. Beispiele:

- english.html (d.h. die Seite liegt im gleichen Verzeichnis wie aktuelle Hypertextseite)
- ../cgi-bin/test.cgi (liegt im Nachbarverzeichnis cgi-bin)

Andere Protokolle: Außer http: sind noch andere Protokolle möglich: **https:** (verschlüsselte Datenübertragung im Web, z.B. für Internet Banking etc.), **ftp:** (Verwendung des klassischen File Transfer Protocols).

Wie ein Protokoll behandelt werden **mailto:** und **telnet:** (Aufruf des Mailsystems bzw. des Telnet-Clients für eine bestimmte Adresse, **file:** (lokaler Dateizugriff ohne Server).

EIN SIMULIERTER WEB-CLIENT (1)

- Starten Sie Telnet in einer DOS-Box mit folgendem Aufruf:
`telnet v.hdm-stuttgart.de 80`
- Stellen Sie das „Lokale Echo“ ein. Dazu zunächst mit Strg-+ in den Befehlsmodus gehen. Nach dem Befehl zweimal die Eingabetaste drücken, um in den Eingabemodus zu gehen:
`Strg + set local_echo`
- Geben Sie ein GET-Kommando ins Telnet-Fenster ein, gefolgt durch zweimaligen Druck der Eingabetaste:
`GET /~riekert/forms/test.htm HTTP/1.0`
- Der Webserver auf v.hdm-stuttgart.de schickt Ihnen dann den HTML-Code der Webseite zurück
- Probieren Sie dasselbe mit anderen URLs, eventuell auch mit URLs von CGI-Skripten.

EIN SIMULIERTER WEB-CLIENT (2)

```
C:\>telnet v.hdm-stuttgart.de 80
Microsoft (R) Windows 2000 (TM) Version 5.00 (Build 2195)
Willkommen bei Microsoft Telnetclient
Telnet Client ,Build 5.00.99206.1

Das Escapezeichen ist '^+'

Microsoft Telnet> set local_echo
Microsoft Telnet>
Microsoft Telnet> GET /~riekert/forms/test.htm HTTP/1.0
HTTP/1.1 200 OK
Date: Mon, 30 May 2005 19:00:18 GMT
Server: Apache/1.3.26 (Unix) Debian GNU/Linux PHP/4.1.2
Last-Modified: Mon, 30 May 2005 18:39:38 GMT
ETag: "37924-79-429b5dea"
Accept-Ranges: bytes
Content-Length: 121
Connection: close
Content-Type: text/html; charset=iso-8859-1

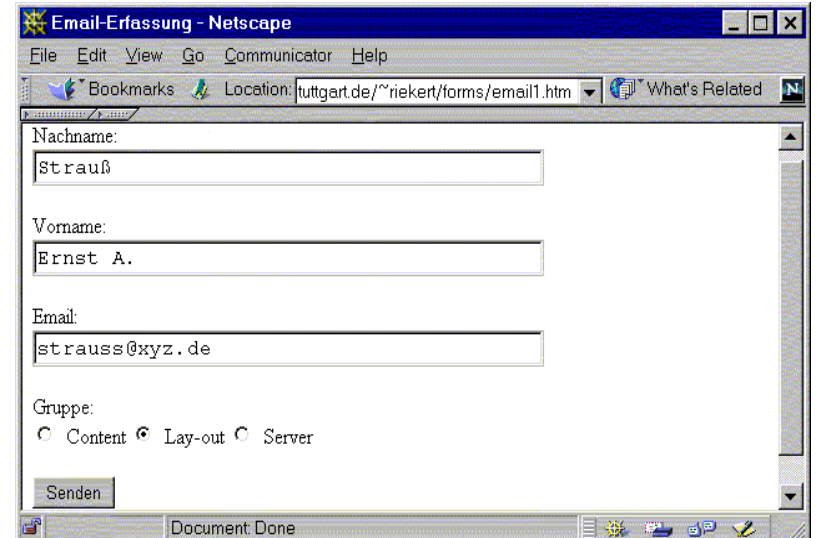
<HTML>
<HEAD> <TITLE> Hallo </TITLE> </HEAD>
<BODY>
  HTML-Testseite von Wolf-Fritz Riekert
</BODY>
</HTML>
```

WEBSERVER-PROGRAMMIERUNG

- Vom Web-Server bereitgestellte Informationsressourcen können dynamisch durch Programme erzeugt werden
- Der Pfadname in der URL bezeichnet das Programm, ggf. gefolgt durch mit Fragezeichen getrennte Parameter, z.B.:
`http://v.hbi-stuttgart.de/~sailer/cgi-bin/plan.cgi?Raum=w011`
Programmverzeichnis **Programm** **Parameter**
- Aufruf des Programms durch den Webserver über spezielle Schnittstellen, am weitesten verbreitet CGI = „Common Gateway Interface“, da von allen Webservern unterstützt
- Es gibt auch alternative herstellerabhängige Schnittstellen, z.B. ISAPI (Internet Information Server Application Programmer Interface) für den Internet Information Server (Webserver der Fa. Microsoft)

FORMULARE IM WWW

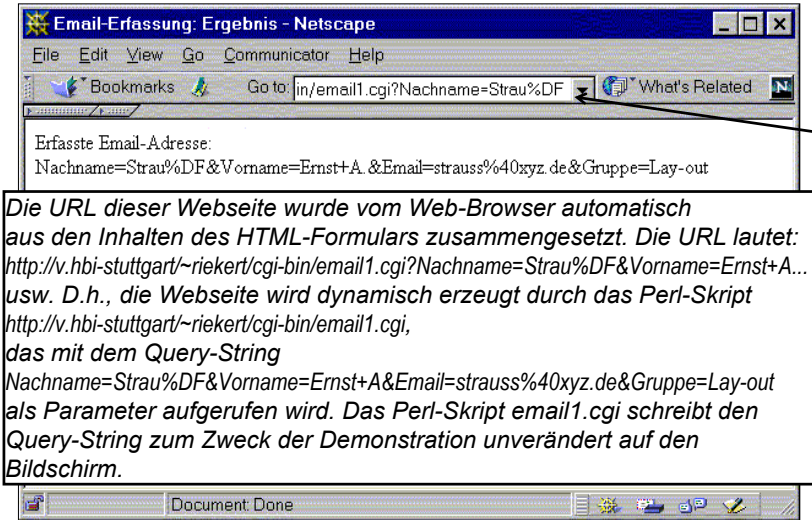
Z.B.: ERFASSUNG VON ADRESSEN



HTML-CODE DES FORMULARS ZUR ERFASSUNG VON ADRESSEN

```
<HTML>
<HEAD> <TITLE>Email-Erfassung</TITLE> </HEAD>
<BODY>
<FORM name="Formular"
  action=" ../cgi-bin/email1.cgi"
  method=get>
<P>Nachname:<BR><INPUT type=text name="Nachname" size=40>
<P>Vorname:<BR><INPUT type=text name="Vorname" size=40>
<P>Email:<BR><INPUT type=text name="Email" size=40>
<P>Gruppe:<BR>
  <INPUT type=radio name="Gruppe" value="Content"> Content
  <INPUT type=radio name="Gruppe" value="Lay-out"> Lay-out
  <INPUT type=radio name="Gruppe" value="Server"> Server
<P><INPUT type=submit value="Senden">
</FORM>
</BODY>
</HTML>
```

ERGEBNIS DES CGI-SKRIPTS email1.cgi



PERL-CODE DES CGI-SKRIPTS email1.cgi

```
#!/usr/bin/perl

$query_string = $ENV{"QUERY_STRING"};

print "Content-type: text/html\n\n";

print "<html>\n";
print "<head><title>Email-Erfassung: Ergebnis";
print "</title></head>\n\n";

print "<body>\n";
print "Erfasste Email-Adresse:<br>\n";
print "$query_string\n";
print "</body></html>\n";
```

email2.cgi: ANFÜGEN DER ADRESSE AN DAS ENDE EINER DATEI

```
#!/usr/bin/perl

$query_string = $ENV{"QUERY_STRING"};

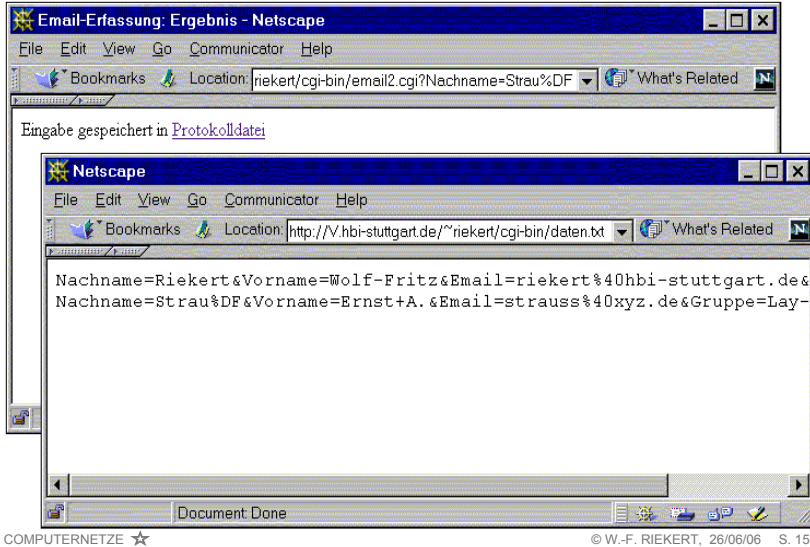
open (DATEN, ">>../data/daten.txt");
print DATEN "$query_string\n";

print "Content-type: text/html\n\n";

print "<html>\n";
print "<head><title>Email-Erfassung: Ergebnis";
print "</title></head>\n\n";

print "<body>\n";
print "Eingabe gespeichert in
  <A HREF=\" ../data/daten.txt\">Protokolldatei</A>";
print "</body></html>\n";
```

ERGEBNIS DES CGI-SKRIPTS email2.cgi



email3.cgi: WEITERE VERBESSERUNGEN

```
#!/usr/bin/perl
$query_string = $ENV{"QUERY_STRING"};

%Parameter = ();
@Formularfelder = split(/&/, $query_string);
foreach $Formularfeld (@Formularfelder) {
    ($name, $wert) = split(/=/, $Formularfeld);
    $wert =~ tr/+//;
    $wert =~ s/%([a-fA-F0-9]{2})/pack("C", hex($1))/eg;
    $wert =~ s/<!--(.*?)-->//g;
    $Parameter{$name}=$wert;
};

open (DATEN, ">>../data/email.txt");
print DATEN "$Parameter{Nachname};$Parameter{Vorname};"
print DATEN "$Parameter{Email};$Parameter{Gruppe}\n";

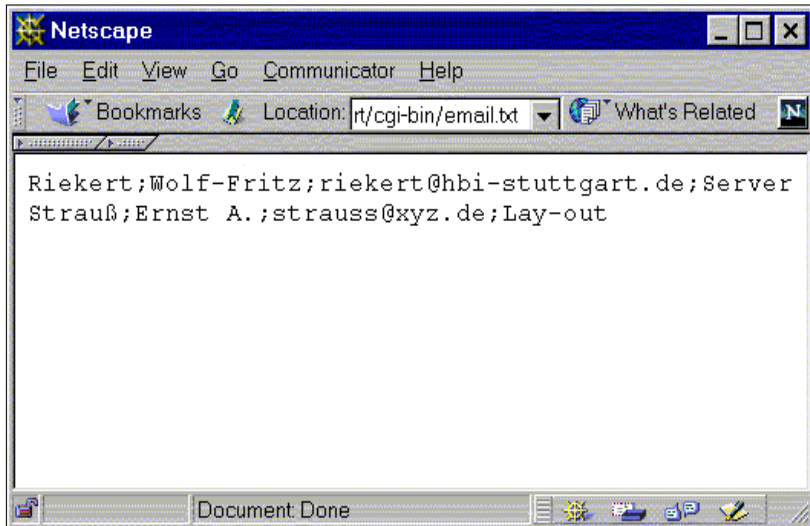
print "Content-type: text/html\n\n";
... usw...
```

*Parameter-
übernahme,
Wiederher-
stellung der
Sonder-
zeichen*

*Abgabe der
Parameter in
eine Datei*

Erzeugung der HTML-Ausgabe

ERGEBNIS DES CGI-SKRIPTS email3.cgi:



ÜBUNG ZUM WWW

Über <http://v.hbi-stuttgart.de/~riekert/forms/beispiele.htm> können die Perl-Skripts aus der Vorlesung aufgerufen werden.

- Probieren Sie die Skripts aus
- Kopieren Sie sich das Formular email1.htm auf die lokale Platte, fügen Sie noch weitere Eingabefelder hinzu und tragen Sie in der „action“ den absoluten Pfad auf das CGI-Skript email1.cgi ein (<http://v.hbi-stuttgart.de/~riekert/cgi-bin/email1.cgi>)
- Probieren Sie das neue Formular aus.