Rechnernetze

Computer Networks

Erik Jacobson

Fachbereich MND

Fachhochschule Frankfurt am Main

WS 2000/2001

Offene Verteilte Verarbeitung

(ODP = Open Distributed Processing):
Normen: ISO 10 764, X.900

Modelle und Methoden zur verteilten Datenverarbeitung.

Ein System wird unter verschiedenen Blickwinkeln, von unterschiedlichen Standpunkten aus betrachtet (Viewpoints on a system)

- Enterprise: Struktur des Unternehmens
- Engineering: Architektur
- Information: Datenbasen
- Computation: Software
- Technology: Hardware

ODP umfaßt:
- Hardware
- Betriebssysteme
- Netzwerke
- Programmiersprachen
- Datenbanken
- Verwaltung

.

.ODP beschreibt Funktionen wie:
- Speicherung
- Verarbeitung
- Zugang
- Kommunikation
- Management
- Sicherheit

Kapitel 1

Netze (networks)

Grundbegiffe (DIN/ISO 2382-18, ISO/IEC 2382-18.01.01)

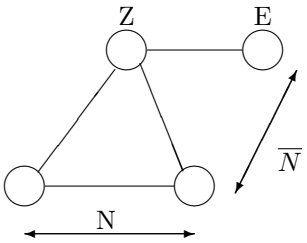


Bild n1p01

Netz:
Eine Anordnung von Knoten und den sie verbindenden Zweigen (Strängen).

Knoten (node) z.B. DV-Anlagen.
Zweige (branch) z.B. Leitungen.

Endknoten (end node): Ein Knoten, der an genau einem Zweig sitzt.
Zwischenknoten (intermediate node) Ein Knoten, der an mehreren Zweigen sitzt
Nachbarknoten (adjacent nodes): Zwei Knoten, die durch einen Zweig direkt miteinander verbunden sind.

Pfad (path): In einem Netz jeder beliebige Weg zwischen zwei Knoten. Ein Pfad kann aus einem oder mehreren Zweigen bestehen. Zwischen zwei Knoten kann mehr als ein Pfad existieren.

Allgemeine Netzwerke

a) Telefonnetz:
- Endknoten = Telefone
- Zwischenknoten = Vermittlungsstellen
- Zweige = Telefonleitung
- Problemstellung: Vermittlung, Wegewahl

b) Energieversorgungsnetze:
- Endknoten = Erzeuger und Verbraucher
- Zwischenknoten = Verteiler
- Zweige = Versorgungsleitungen
- Problemstellung: Energieverteilung

c) Computernetze:
- Endknoten = DV-Anlagen
- Zwischenknoten = DV-Anlagen
- Zweige = DFÜ-Leitungen
- Problemstellung: Datenübermittlung, Verteilte Verarbeitung

1.1 Computer-Netzwerk-Anwendungen

Klassen verteilter Datenverarbeitung

1.1.1 Datenverbund

Zweck: Datenübermittlung ohne Festlegung oder Koordinierung der (vorangehenden oder nachfolgenden) Datenverarbeitung.

Voraussetzung: Einheitliche Datenformate.

Aufgaben: Datenübertragung

Varianten
- On-line [synchron]: Zwei Partner (inter-)aktiv.
- Off-line [asynchron]: Jeweils ein Partner aktiv.

Beispiele: Datenträgeraustausch zwischen Banken, FTP, FTAM

Analogien: Telegramm, Telex, FAX

Anmerkungen:
– Die transportierten Datenmengen sind von sehr unterschiedlichem Umfang
– Der Datenträgeraustausch ist die Urform des Datenverbundes.

1.1.2 Informationsverbund

Zweck: Austausch von Nachrichten zwischen Wissensträgern (Menschen)

Voraussetzung:
Kommunikationsregeln (Protokolle), einschließlich einheitlicher Datenformate
Datenverbund als Grundlage

Aufgaben: Transport und Verteilung von Nachrichten

Varianten:
- On-line [synchron]: Zwei Partner (inter-)aktiv.
- Off-line [asynchron]: Jeweils ein Partner aktiv.

Beispiele: Elektronische Post (Electronic Mail, E-mail)

Analogien: Briefpost, Telefondienst

Anmerkungen:
– Die transportierten Datenmengen sind in der Regel vergleichsweise klein (1 KB bis 1 MB pro Tag)
– Information durch Informationsanbieter wird zur Handelsware (Dataware).

1.1.3 Funktionsverbund

Zweck: Gemeinsame Benutzung von peripheren Betriebsmitteln (resource sharing),
- Ein-/ Ausgabeperipherie, z.B. Plotter, Drucker, Terminals
- Massenspeicher, z.B. Platten, Optische Datenträger, Archive

Voraussetzung: Datenverbund als Grundlage
Kommunikationsregeln (Protokolle)
gemeinsame Verarbeitungsstruktur

Aufgaben:
automatische Verteilung von Datenströmen von und zu bestimmten Geräten.

Varianten: Server-Netze, Client-Server-Systeme

Beispiel: Novell-Netze

Anmerkung: in der Regel ein heterogenes Netz (verschiedenartige Rechner)

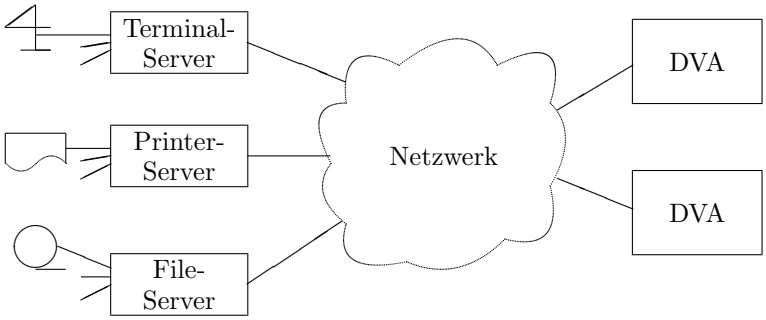


Bild n1p02

1.1.4 Leistungsverbund

Zweck: Verteilung von Aufgaben auf verschiedene Rechner entsprechend deren Ausstattung und Leistung (statisch).
DDP hier: Dedicated Data Processing

Voraussetzung: Datenverbund als Grundlage
Kommunikationsregeln (Protokolle)
gemeinsame Verarbeitungsstruktur

Aufgaben: Vermittlung von Daten nach Bedarf (Benutzeranfrage)

Beispiele:
– PCs für Büro-Anwendungen
– Workstations für Graphik-Anwendungen (CAD)
– Host als File-Server
– Backend-Rechner zur Bearbeitung aufwendiger Numerik (z.B. FEM).

Analogien: arbeitsteilige Produktion

Anmerkung: In der Regel ein heterogenes Netz (verschiedenartige Rechner)
Die Grenze zwischen Funktions- und Leistungsverbund ist unscharf. Im Zentrum steht oft ein "Host" (Sterntopologie).

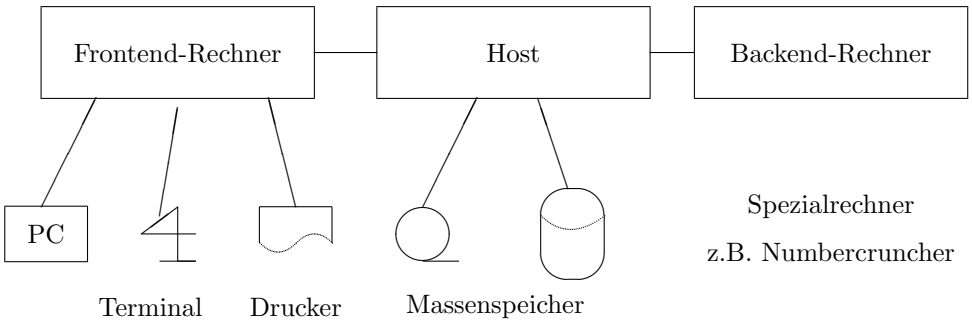


Bild n1p03

1.1.5 Lastverbund

Zweck: Verteilung von Aufgaben auf verschiedene Rechner entsprechend ihrer aktuellen Belastung zur optimalen Nutzung von Rechenleistung.

Voraussetzung: Homogener Rechnerverbund (gleichartige Rechner), Strategien und Algorithmen zur Verteilung.

Aufgaben: automatische (transparente) Verteilung von DV-Leistung

Beispiele: VAX- Cluster

Anmerkung: DDP hier: Distributed Data Processing

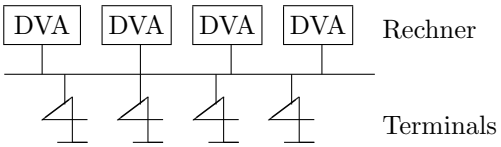


Bild n1p04

1.1.6 Verfügbarkeitsverbund

Zweck: Ausfallsicherung bei kritischen Systemen, i.a. Realzeitverarbeitung

Voraussetzung:
– homogener Verbund: Spiegelung
– inhomogener Verbund: Vermeidung gleichartiger Fehler.

Aufgaben:
(automatische) Umschaltung auf einen Reserverechner bei Ausfall eines Rechners.

Varianten:
- Mehrere Rechner an derselben Aufgabe (Hot-Stand-By), Mehrheitsentscheid
- Ein Rechner ist in Reserve (Cold-Stand-By)
- Ein Reserverechner ist bereit (Warm-Stand-By)

Hot-Standby:	n Programme	n Datensätze
Cold-Standby:	1 Programm	1 Datensatz
Warm-Standby:	n Programme	1 Datensatz
	1 Programm	n Datensätze

Beispiele: Flugzeug, Kernkraftwerk

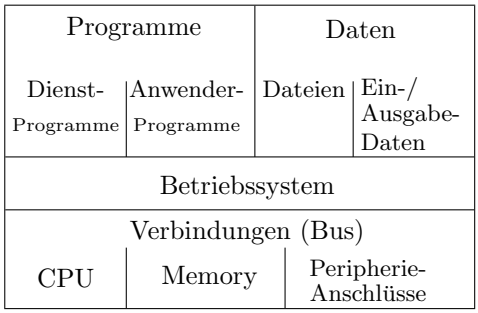
Analogien: Reserverad im PKW.

Anmerkung: extremer Lastverbund (Sicherheit durch Redundanz)

1.2 Rechnerkopplung

Minimalnetz aus zwei Rechnern (Netzausschnitt).

Rechner



. .Die Basismaschine (Hardware plus Betriebssystem) ist statisch, die Anwendung ist dynamisch. Die Programme ändern sich selten, die Daten ändern sich rasch.

Bild n1p05

Begriffe

Systembus: Bus innerhalb der Zentraleinheit eines Rechners.

Bandbreite: Übertragungskapazität (in Bits pro Sekunde, bps) eines Übertragungskanals, z.B. des Systembus (zu unterscheiden von der Bandbreite in der Elektrotechnik).

Bandbreiten-Längen-Produkt (BLP): Das Produkt aus Bandbreite B und physikalischer Länge L eines Übertragungskanals.

Kanal Ein- / Ausgabekanal zur Nahperipherie
Schnittstelle zu Massenspeichern (z.B. SCSI)

1.2.1 Multiprozessor-Systeme

Massiv-parallele Systeme

Merkmale: hohe Anzahl von Rechenwerken ($n \leq 2^{16} = 65536$)
Kopplung der Prozessoren z.T. durch reguläre Netze.

Voraussetzungen: gemeinsames (verteiltes) Betriebssystem.

Anwendung: Schaffung extremer Verarbeitungsleistung
Parallelverarbeitung eines Programms

Beispiele:
– Transputer unter OCCAM
– HYPERCUBE-Rechner
– SUPRENUM (GMD u.a.)

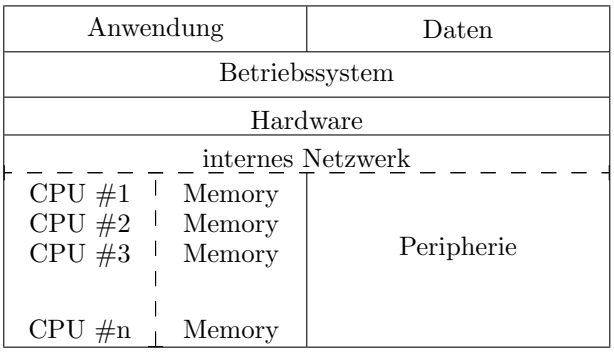


Bild n1p06

1.2.2 Mehrprozessor-Systeme

Parallelverarbeitung mehrerer Programme (Prozesse) oder Aufgaben (Tasks) durch mehrere CPUs (Multiprocessing) unter einem Betriebssystem.

Merkmale: Kopplung der Anwendungen über gemeinsame Speicherbereiche: Semaphore, Pipes (Intertask-Kommunikation).

Voraussetzungen: geeignetes Betriebssystem

Anwendung: – Lastverbund

Beispiel: DMA-Controller

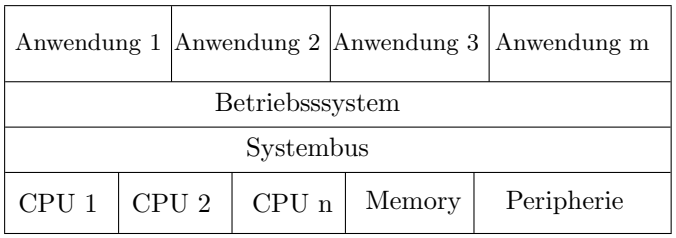


Bild n1p07

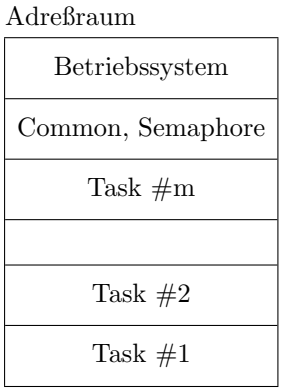


Bild n1p08

1.2.3 Buskopplung (homogen)

Kopplung von 2 gleichartigen Rechnern durch einen gemeinsamen Systembus.

Merkmale Merkmale für die Leistungsfähigkeit:
Datenübertragungsrate $d \approx 10Mbps$ (abhängig. von Takt f und Busbreite b)
Bandbreitenlängenprodukt ($BLP = s * d$) $BLP \approx 10 \cdot 10^6 bm/s$

Voraussetzungen: identische Rechner, modifiziertes Betriebssystem.

Anwendung – Vorzugsweise Lastverbund
– auch Verfügbarkeitsverbund

Beispiel: Tandem-Rechner

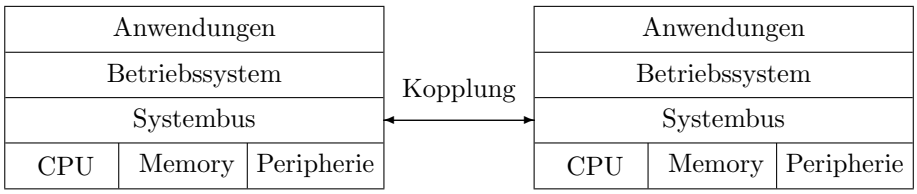


Bild n1p09

1.2.4 Kanalkopplung (inhomogen)

Prinzip: Ein Rechner 'sieht' den anderen als Massenspeicher.

Merkmale Kopplung: Heterogen (Verschiedene Rechner möglich)
Übertragung: Blöcke, Byte seriell (Bit parallel)
Datenrate: 5 Megabit pro Sekunde [Mbps] (Bandbreite)
Entfernung: ca. 10 Meter
Bandbreiten/Längenprodukt (BLP) $= 50 * 10^6 bm/s$

Voraussetzungen Gemeinsame (parallele) Schnittstelle (z.B. SCSI) einschließlich geeigneter Treiber

Anwendung – Funktionsverbund
– Leistungsverbund
– Lastverbund – Verfügbarkeitsverbund

Beispiele: Lokale Netze (LANs), NOVELL-Cluster

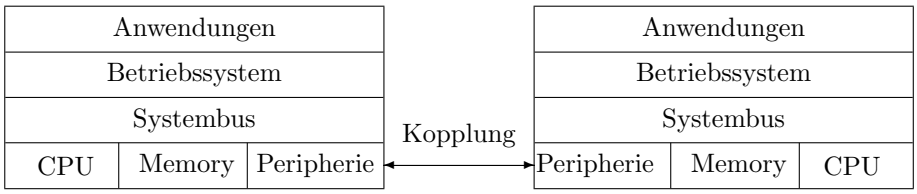


Bild n1p10

1.2.5 Indirekte Kopplung

Prinzip: Gemeinsame Platte, gemeinsamer Speicher

Merkmale Datenrate: $\frac{1}{2}$ * 5 Megabit pro Sekunde [Mbps] (Bandbreite)
Entfernung: ca. 2 * 10 Meter (Zwei Kabellängen)
Bandbreiten/Längenprodukt (BLP) $=50 \cdot 10^6 bm/s$
Vorteil: Rechner laufen asynchron, Daten werden auf Platte gepuffert.
Nachteil: Dual ported Drive (Gleichzeitig lesen und schreiben auf Platte)

Voraussetzungen: Hardware, d.h. geeignete Peripherie

Anwendung: Verfügbarkeitsverbund

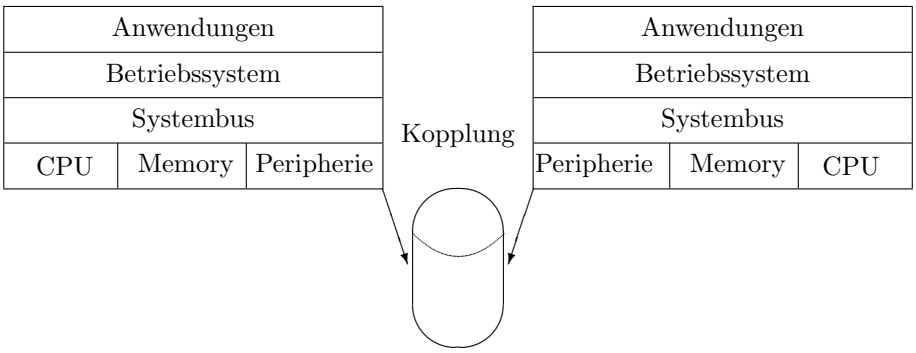


Bild n1p11

1.2.6 Lose Kopplung

Prinzip: Ein Rechner 'sieht' den anderen als Peripherie (z.B.: Terminal).

Merkmale Datenrate: 10^4 $bps \simeq 0,01$ $Mbps$
Entfernung: ca. $10^3 Meter = 1 Kilometer$
Bandbreiten/Längenprodukt $(BLP) = 10 * 10^6 bm/s$

Voraussetzungen Kompatible Terminalschnittstellen (in der Regel asynchron seriell nach RS-232C), geeignet Software auf Anwenderebene.

Anwendung: Datenverbund bis Leistungsverbund

Beispiel: Wide Area Networks (WANs)

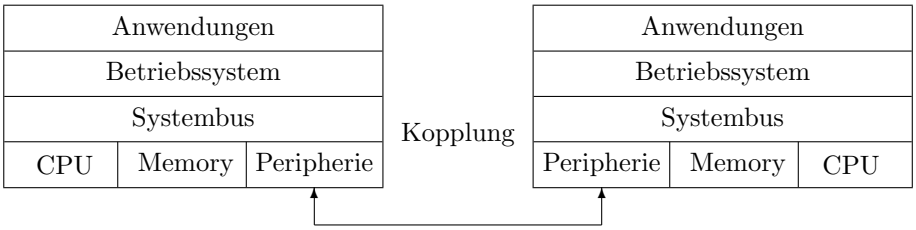


Bild n1p12

1.2.7 Datenträgeraustausch

Prinzip: Transfer von Daten auf Datenträgern

Merkmale Datenmenge: 1 Gigabyte = $8 \cdot 10^9 bit$
Entfernung: 5 km - 500 km
Transportgeschwindigkeit: ca. 20m/s (72 km/h)
Bandbreiten/Längenprodukt (BLP) = $160 \cdot 10^9 bm/s$
Vorteil: Hohe BLP, asynchron, off-line möglich
Nachteil: Unsicher, nur bedingt einsetzbar (ab 1 MB)

Voraussetzungen: Hardware, d.h. kompatible Peripheriegeräte

Anwendung: Datenverbund

Beispiel: Verteilung von Software per Post.

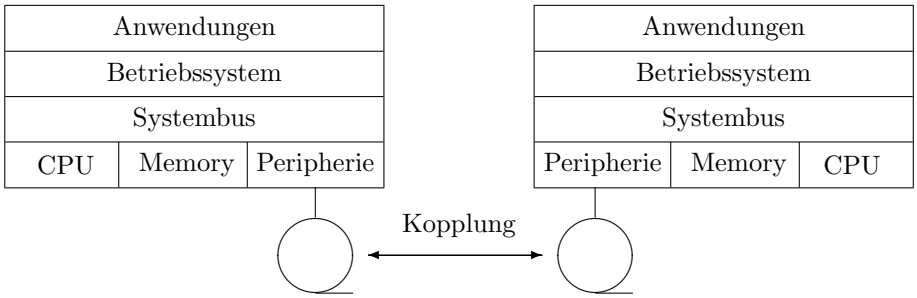


Bild n1p13

1.3 Kommunikation

Kommunikationsteilnehmer Kommunikationspartner:
– gleichgestellte Partner (peers)
– hierarchische Beziehung: Diensterbringer, Dienstbenutzer
Kommunikationsrollen – Sender (Datenquelle)
– Empfänger (Datensenke)
– Vermittler:
  – Verteiler (Schalter, switches)
  – Umsetzer (Anpassung, adapter)
  – Speicher (Puffer, stores)

1.3.1 Transferdiagramm (Ereignis/Zeit-Diagramme)

Das Transferdiagramm veranschaulicht die Datenübertragung zwischen zwei Kommunikations-Partnern, es zeigt die von ihnen verursachten und die bei ihnen eintreffenden Ereignisse im Zeitablauf.

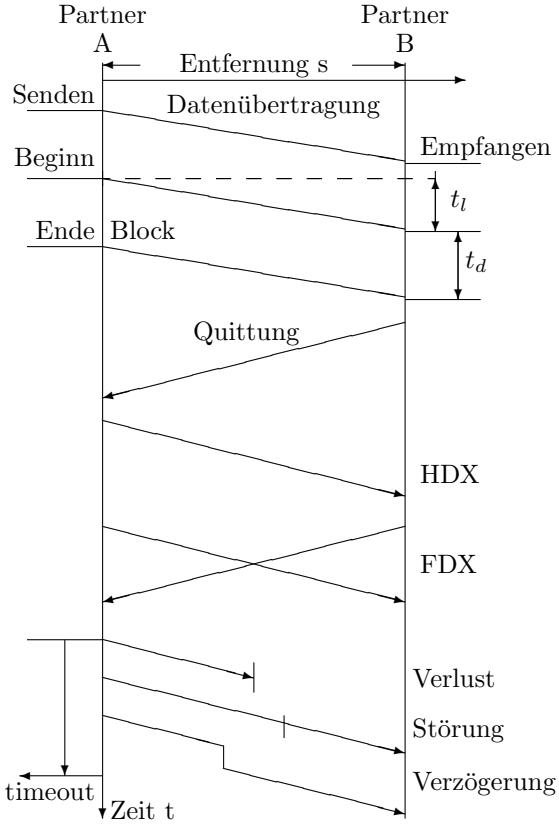


Bild n1p14

$t_l = s/c$ = Laufzeit
$t_d = D/r$ = Übertragungszeit
= Datenmenge / Übertragungsrate

Maßeinheiten:
$[D] = 1$ bit
$[r] = 1$ bit/s

HDX: half duplex
= Wechselsprechbetrieb
FDX: full duplex
= Gegensprechbetrieb
SMX: simplex = Richtungsbetrieb

1.3.2 Strategien und Protokolle

Kommunikationsstrukturen

a) Beziehungen (statisch)

– broadcasting	(Rundspruch):	einer an alle	1 : m
– multicasting	(Gruppenrundspruch):	einer an Gruppe	1 : n $\leq$ m
– mono cast (unicast)	(Punkt zu Punkt): point-to-point	einer an einen	1 : 1
– multipoint	Konferenzschaltung:	einige an einige	n : m

b) Zugriff (zum Medium, Netz) (dynamisch)

Chaotisch: Jeder und immer (RAM).
Time-sharing: Jeder zu seiner Zeit (Zeitintervalle), wenn er dran ist.
Resource-sharing: Jeder wenn er kann (und darf), wenn Betriebsmittel frei sind.

c) Kommunikationsprotokolle
(als Impulsdiagramme und Transferdiagramme)

– **Quittungsverfahren (acknowledging):**
nach der Übertragung erfolgt eine Bestätigung (Q)

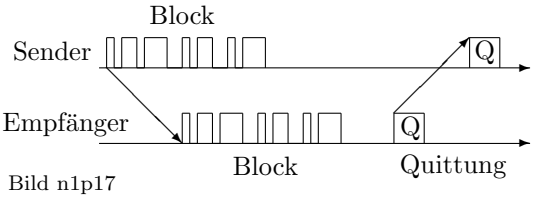


Bild n1p17

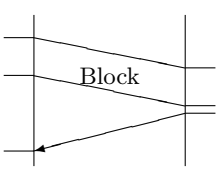


Bild n1p16

– **Handschlagverfahren (handshaking):**
die Übertragung wird von einer gegenseitigen Bestätigung der Bereitschaft begleitet.

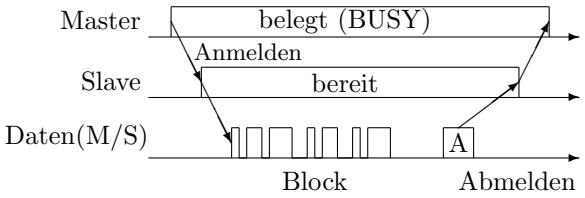


Bild n1p19

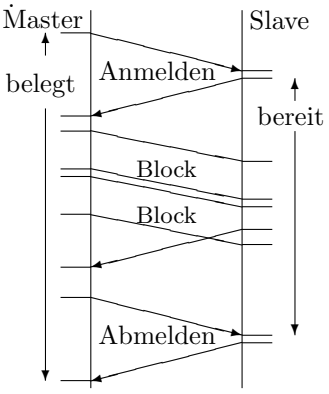


Bild n1p18

– **Flußsteuerung (flow control):**
Unterbrechung und Fortsetzung der Datenübertragung auf Anforderung (NAK und ACK) des Empfängers bei drohendem Pufferüberlauf. Dabei können Fragmente entstehen, für deren Behandlung weitere Vereinbarungen (Protokolle) getroffen werden müssen.

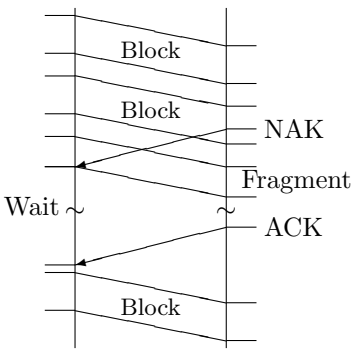


Bild n1p20

– **Synchronisation:**
nach jeder Übertragung wird auf eine Bestätigung gewartet; fällt diese negativ aus (NAK), wird die Übertragung wiederholt.

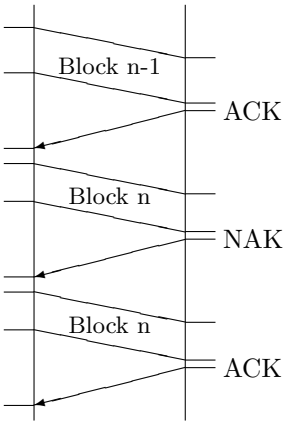


Bild n1p22

– **Fenstertechnik (windowing):**
es wird zunächst eine festgelegte Anzahl (hier w = 3) von Blöcken übertragen und dann auf eine Bestätigung für den ersten davon (n) abgewartet, bis der nächste Block (n+w) übertragen wird.

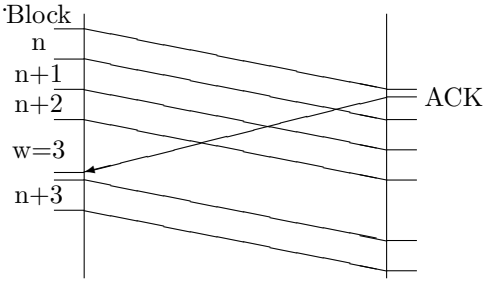


Bild n1p23

– **Zeitschrankenüberwachung (time-out):**. . Mit jeder Übertragung wird ein Timer (eine Stopuhr) gestartet; er wird bei rechtzeitiger Rückmeldung (ACK) zurückgesetzt(toc = time-out cancellation); andernfalls liefert er ein time-out Signal (Ereignismeldung) an den Sender.

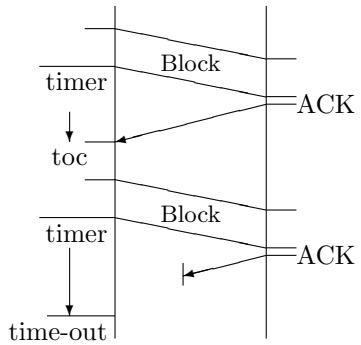


Bild n1p24

1.3.3 Der Übertragungskanal

Das BRM der Informationsverarbeitung

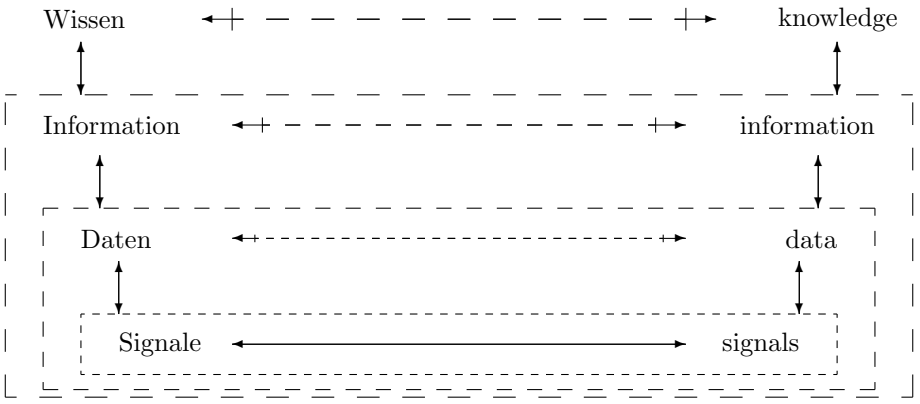


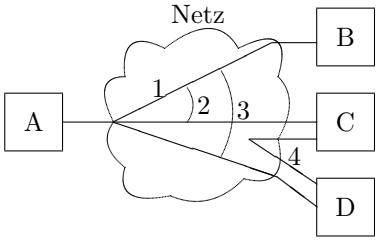
Bild b1p01

← – → : Kommunikation, Übermittlung
⟷ : Übertragung, Verarbeitung, Speicherung
↓ : Darstellung, Repräsentation, Codierung
↑ : Erwerb, Interpretation, Decodierung

Kapitel 2

Netzstrukturen

2.1 Verbindungsstrukturen



. .

1. Punkt-zu-Punkt-Verbindung (A → B)

– Mehrpunktverbindungen

2. Gruppenrundspruch (A → B,C)
3. Rundspruch (A → B,C,D)
4. Konferenzschaltung (A,C,D)

Bild n2p01

2.2 Netzklassen

Nach räumlicher Ausdehnung

2.2.1 GAN - Global area network, Globales Netz

Netz aus Netzen
In der Regel Kombination von WANs und LANs
Merkmale:
- Ausdehnung (s) ≈ 10.000 km (weltumspannend)
- Übertragungsrate (r) = 1 Mb/s (Megabit pro Sekunde, Mbps)
- Laufzeit $T_l \leq 4\ s$
- Das Netz ist meist privat

Anwendung – Informationsverbund
Beispiele
- SNA (IBM)
- DECnet (DEC)
- Internet

Probleme:
- Datensicherheit (Datenverlust, -verfälschung), Schutz durch Redundanz
- Datenschutz (Abhörsicherheit), Schutz durch Kryptographie

2.2.2 WAN - Wide area network, Weitverkehrsnetz

Merkmale:
- Ausdehnung (s) $\leq$ 10.000 km (landes- oder kontinentweit)
- Übertragungsrate (r) $\leq$ 100 Kb/s (Kilobit pro Sekunde, Kbps)
- Laufzeit $T_l \leq 1\ s$
- Das Netz ist öffentlich (der Betreiber kann eine öffentliche oder private Einrichtung sein).

Anwendung – Informationsverbund
Beispiele
- Telefon, Fax, ISDN
- DATEX-L
- DATEX-P

Probleme
- Einheitliche Protokolle von CCITT
- Dienste garantieren keine Sicherheit, keine Verschlüsselung
- Fehlerrate $\approx 10^{-6}$

2.2.3 MAN - Metropolitan area network, Regionales Netz

Merkmale:
- Ausdehnung (s) = 10 - 100 km (regional)
- Übertragungsrate (r) = 10 Mb/s (Megabit pro Sekunde, Mbps)
- Laufzeit $T_l \leq 0.1\ s$ (Lichtwellenleiter)
- Fehlerrate $\leq 10^{-8}$ pro Bit
- Das Netz ist öffentlich,

Anwendung: Datenverbund, Leistungsverbund
Beispiele
– DATEX-M
Problem: Mithören (Datenschutz)

2.2.4 LAN - Local area network, Lokales Netz

Merkmale:
- Ausdehnung (s) = 1 - 10 km (auf dem Gelände eines privaten Betreibers)
- Übertragungsrate (r) = 10 Megabit /Sek. (Mbps)
- Laufzeit $T_l \leq 100\ ms$
- Netz ist privat (Private Protokolle, private Nutzungsvorschriften)

Anwendung: Datenverbund .. Leistungsverbund
Beispiele
- Ethernet
- Token Ring
- Nebenstellenanlage (Private automatic branch exchange, PABX)

Probleme
- Mithören (Datenschutz)
- Management

2.2.5 Cluster

Homogenes Verbundsystem,
Merkmale:
- Ausdehnung (s) = 10 m (in einem Raum)
- Übertragungsrate (r) = 1 Gb/s (Gigabit pro Sekunde, Gbps)
- Laufzeit $T_l \leq 1ms$
- Netz ist (sehr) privat

Anwendung: Datenverbund .. Lastverbund
Beispiele
– Bürosysteme
Probleme
- Durchsatz
- Management

2.2.6 VLAN - Very local area network Sehr lokales Netz

Merkmale:
- Ausdehnung (s) $\leq$ 10 m (in einem Gehäuse)
- Übertragungsrate (r) = 0,1 Gb/s (Gigabit pro Sekunde, Gbps)
- Laufzeit $T_l \leq 1\mu s$
- Netz ist (sehr) privat
- Kopplungsstärke steigt mit sinkender Entfernung.

Anwendung Lastverbund .. Verfügbarkeitsverbund
Beispiele
- Transputer
- Multiprozessorsysteme, z.B. Suprenum

Probleme
- Durchsatz
- Geschwindigkeit

2.3 Netztopologien

Netz := Knoten + Zweige

2.3.1 Vollständiges Netz

(Vollständig vermaschtes Netz)
Jeder Knoten wird mit jedem anderen verbunden.

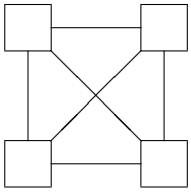


Bild n2p02

. .

Konfiguration:
n Knoten (gleichberechtigte)
z Zweige: $z = n \cdot (n-1)/2$
Pfadlänge p = 1
oder mehr, falls die Knoten auch als Vermittler (Zwischenknoten) agieren

Vorteile: schnell,
sicher (abhörsicher, ausfallsicher)
Nachteile: teuer,
starr, unflexiblel (nicht leicht erweiterbar)
Anwendung: Verfügbarkeitsverbund

2.3.2 Stern

Ein zentraler Vermittler (Zwischenknoten)

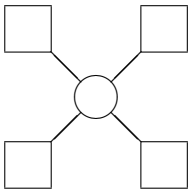


Bild n2p03

. .

Konfiguration:
1 zentraler Vermittler (Zwischenknoten)
n Endknoten
z Zweige: $z = n$
Pfadlänge p = 2

Vorteile: flexibel: leicht erweiterbar
billig: ein Anschluß je Endknoten (der Vermittler ist teuer)
Nachteile: langsam: Stau (congestion) im Vermittler, Flaschenhals (bottle neck)
unsicher (Ausfall des Vermittlers, Abhören im Vermittler),
Anwendung: Daten- bis Leistungsverbund, vorzugsweise im LAN
Beispiele: Hub (Nabe), Verteiler
Konzentrator
Multiplexer

Varianten:

Terminals

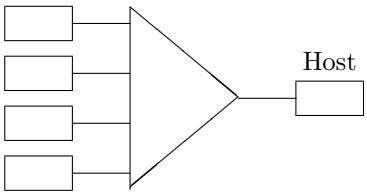


Bild n2p04

Konzentrator (vgl. Terminal-Server), unsymmetrischer Stern

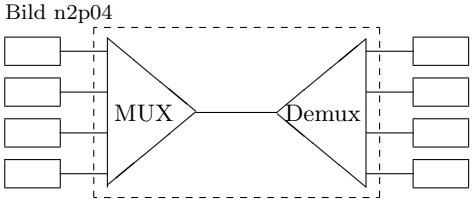


Bild n2p05

Multiplexer (1 virtueller Knoten, zweigeteilter Vermittler)

2.3.3 Baum

= gekoppelte Sterne

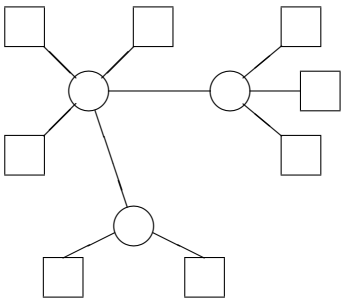


Bild n2p06

Konfiguration:
n Knoten
n_1 Endknoten
n_2 Zwischenknoten)
z Zweige: $z = n - 1$
Pfadlänge: $2 \leq p \leq n_2 + 1$
Es gibt genau 1 Pfad zwischen 2 (End)Knoten

Vorteile: flexibel, billig (vgl. Stern)
sicherer: bei Ausfall eines Vermittlers bleiben Subnetze erhalten.
Nachteile: unsicher (Ausfall eines benachbarten Vermittlers)
langsam: mehrere Flaschenhälse in einem Pfad
Anwendung: Daten- bis Leistungsverbund im LAN und im WAN

Variante: Hierarchischer Baum

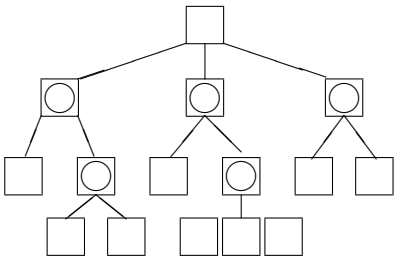


Bild n2p07

. .

Knoten sind sowohl Vermittler als auch Verarbeitungsrechner (Hybridknoten).
Einsatz: CIM

2.3.4 Maschennetz

= gekoppelte Bäume

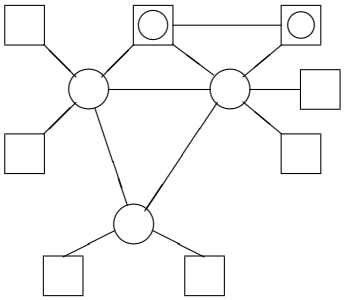


Bild n2p08

. .

Konfiguration:
n Knoten (Hybrid-, End- oder Zwischenknoten)
n_1 Endknoten
z Zweige: $z \geq n$
Pfadlänge: $1 \leq p < n - n_1 + 1$
Es gibt mindestens 1 Paar von (End)Knoten, zwischen denen es mehr als 1 Pfad gibt.

Vorteile: flexibel
relativ billig, Zwischenknoten sind teuer
relativ ausfallsicher:
bei Ausfall eines Vermittlers kann es Ausweichstrecken geben
Nachteile: Wegewahl im Zwischenknoten
Management für das Gesamtnetz
Jede Änderung eines Zwischenknotens betrifft ganzes Netzmanagement
Anwendung: Daten- bis Leistungsverbund, typische Topologie für WAN

2.3.5 Liniennetz

= verkümmerter Baum

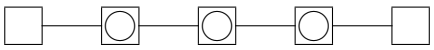


Bild n2p09

Konfiguration:
2 Endknoten
n-2 Zwischenknoten (Hybridknoten)
z Zweige: $z = n - 1$
Pfadlänge: $1 \leq p < n - 1$
Es gibt genau 1 Pfad zwischen 2 Knoten

Vorteile: einfache Struktur
billig
relativ flexibel (erweiterbar)
relativ ausfallsicher: Überbrückung eines Knotens bei Ausfall, andernfalls entstehen Subnetze
Nachteile: abhörbar
langsam: viele Zwischenknoten
Belastung der (hybriden) Zwischenknoten
Anwendung: selten

2.3.6 Ring

= geschlossenes Liniennetz

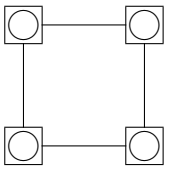


Bild n2p10

Konfiguration:
n hybride Knoten
z Zweige: $z = n - 1$
Pfadlänge: $1 \leq p < n - 1$
Es gibt genau 2 Pfad zwischen 2 Knoten (bidirektionaler Ring)
oder genau 1 Pfad zwischen 2 Knoten (unidirektionaler Ring)

Vorteile: einfach,
relativ billig
relativ flexibel (erweiterbar)
relativ ausfallsicher (Überbrückung oder Subnetze)
Nachteile: abhörbar
Engpässe in den hybriden Vermittlern
Management zur Koordinierung und Lastverteilung
Anwendung: Daten- bis Lastverbund, typisch für LAN

Varianten:
1. Trennung von Verarbeitungs- und Vermittlungsaufgaben in den Knoten

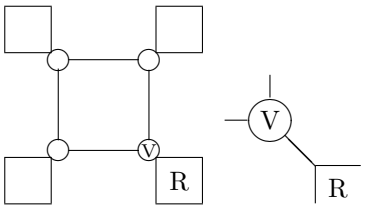


Bild n2p11

. .

In den Knoten werden sehr lokale Subnetze eingeführt: der Vermittler V (eine Netzwerkkarte mit eigenem Prozessor) wird über eine Buskopplung mit dem Verarbeitungsrechner R gekoppelt; er bildet einen rudimentären Stern.

2. Stern-Ring-Netz

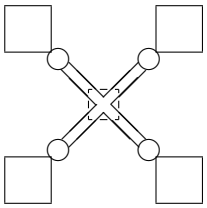


Bild n2p12

. .

- Zur strukturierten Verkabelung wird ein Sternkoppler eingefügt, zu dem von allen Knoten 2 Leitungen führen.
- zur dynamischen Konfigurierung können im Sternkoppler ausgefallene Knoten (automatisch) überbrückt werden.

2.3.7 Bus

= Stern ohne Vermittler

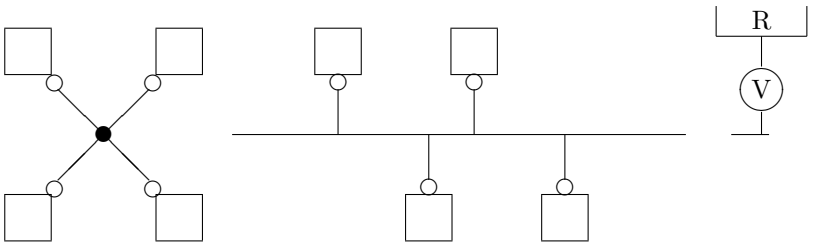


Bild n2p13

Die Vermittlungsaufgaben werden dezentralisiert; in jedem Endknoten gibt es einen integrierten Vermittler in Form einer Netzwerkkarte (vgl. 1. Variante des Ring). Die Anschlüsse am gemeinsamen Stamm (trunk) sind passiv.

Konfiguration:
n bzw. 2n Knoten: n Endknoten (und n Zwischenknoten)
Zweige: $z = 1$ *bzw.* $n + 1$
Pfadlänge: $p = 1$ *bzw.* 3
Es gibt genau 1 Pfad zwischen 2 Endknoten

Vorteile: einfachste Topologie
sehr flexibel (leicht erweiterbar)
billig: nur die Endknoten müssen aufgerüstet werden
ausfallsicher
Nachteile: abhörbar: alle hören alles (broadcasting)
Engpaß auf dem gemeinsamen Zweig
Gefahr durch Dauerbelegung (jabber = sabbern)
Zugriffsprotokolle notwendig
Anwendung: Daten- bis Verfügbarkeitsverbund, typisch für LAN

2.3.8 Reguläre Netze

Struktur des Netzes läßt sich als Koordinatensystem beschreiben. Fällt ein Rechner aus, so ist das Netz nicht mehr regulär.

Vorteile: reguläre Architektur, vereinfachte Protokolle
Nachteile: Lösungen für Spezialanwendungen, nicht universell einsetzbar
Anwendung: Last- und Verfügbarkeitsverbund; VLANs;
massiv parallele Systeme (Superrechner).

a) Gitter = d-dimensionales Liniennetz

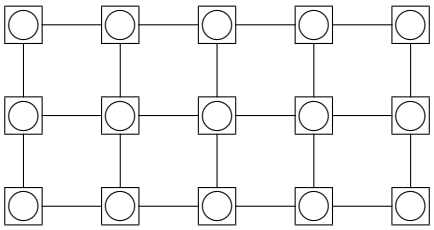


Bild n2p14

. .

Konfiguration:
Kantenlängen a, b, ...
n Knoten (Hybridknoten): $n = a \cdot b...$
z Zweige: $z = (a-1) \cdot (b-1)...$
Pfadlänge: $1 \leq p < (a-1) + (b-1) + ...)$
Kommunikation in der Regel nur zwischen Nachbarknoten.

Knoten-Klasse	Zweige pro Knoten	Anzahl Knoten
- Ecken	d	2^d
- Kanten	d+1	$2 \cdot (a-2+b-2+...)$
- Innen	2d	$(a-2)(b-2)...$

b) Hyperwürfel = d-dimensionaler Würfel der Kantenlänge 2
es gibt nur 2^d Eckknoten mit je d abgehenden Zweigen
d=3 . .d=4

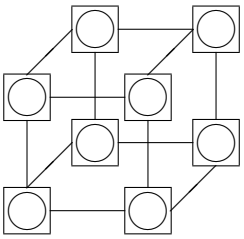


Bild n2p15

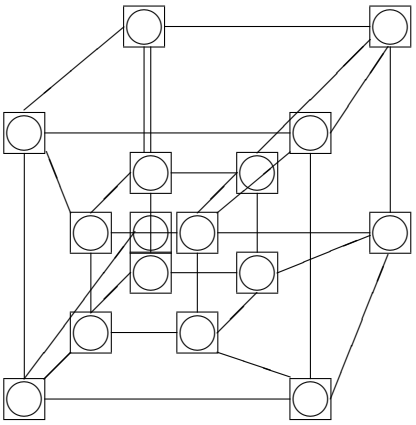


Bild n2p15a

Realisierung: Hypercube (Thinking Machines) mit $n = 2^{16} = 65536$ Rechnern (d=16)

c) Hyperring = d-dimensionaler Ring
d=2 ..d=3 (3-dim Erweiterung eines 2-dim Gitters)

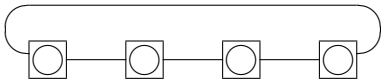


Bild n2p16

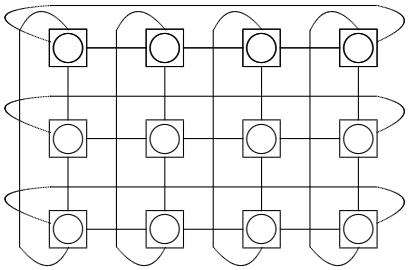


Bild n2p17

d) Spidernet (Spinnennetz)
= Hybrid aus Stern und Ringnetz

. .

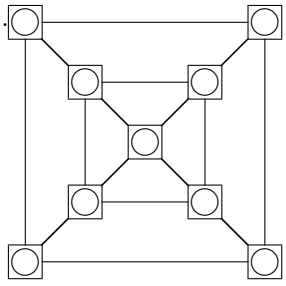
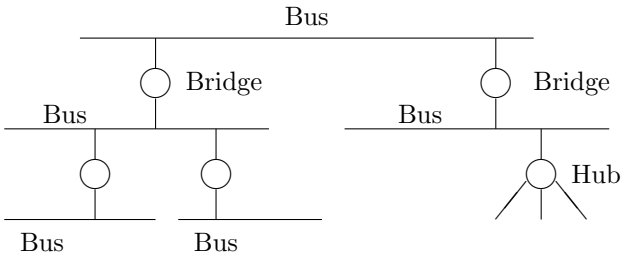


Bild n2p18

2.3.9 Mischtopologien

Verknüpfungen von Subnetzen mit jeweils einheitlicher Topologie.

a) Ethernet LAN = Baum von Bussen und Sternen



..

die angschlossenen Endknoten sind nicht hier eingezeichnet

Bild n2p19

b) FDDI-Backbone

Ring mit Ringen, Bussen und Sternen

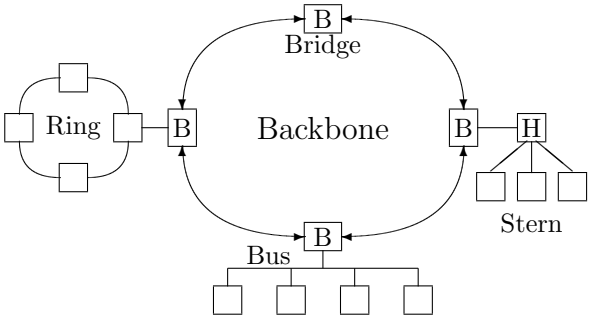


Bild n2p20

2.4 Netzarchitekturen

Datenübermittlung in Netzen, Vermittlungsstrukturen

Übersicht

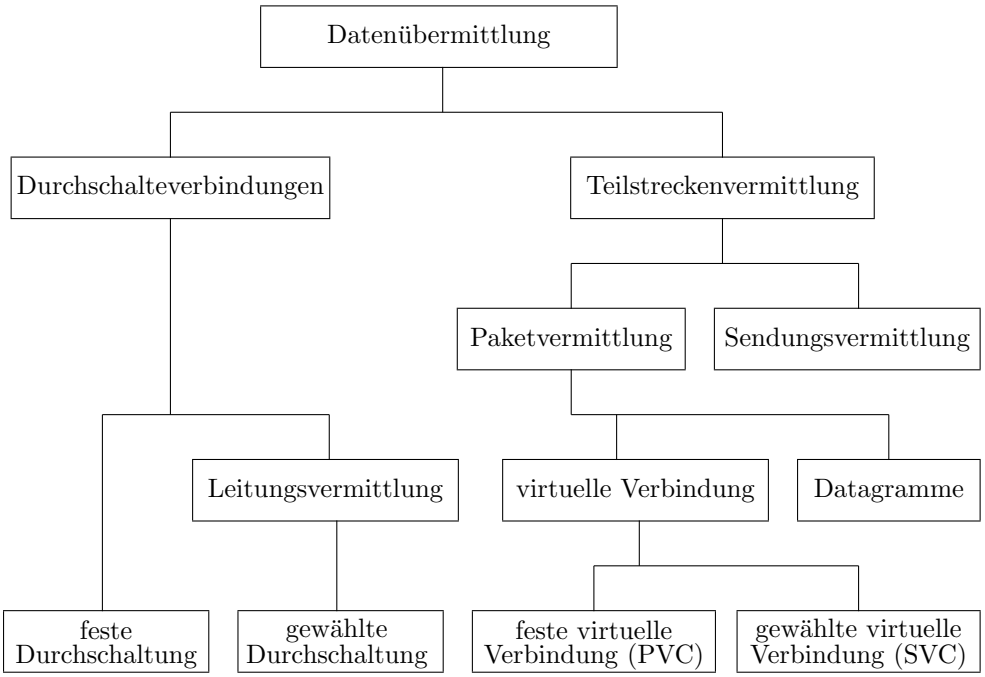


Bild n2p21

2.4.1 Feste Verbindungen

Zwei Prinzipien:
- Punkt-zu-Punkt-Verbindung: Durchschalteverbindung, Standleitung, abgeschlossenes Medium wie Kabel, Lichtwellenleiter (LWL)
- Mehrpunktverbindung (Rundspruch): gemeinsames offenes Medium, wie z.B. Bus, Äther

Merkmale:
- Passives Netz, keine aktiven Komponenten
- Physikalische Grundlage für alle Netze

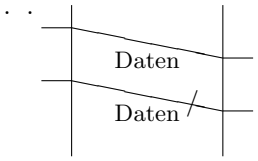


Bild n2p22

Vorteile:
- ständig verfügbar,
- keine Verzögerungen für Verbindungsaufbau,
- schnell: keine Signalverzögerung,
- maximale Übertragungsgeschwindigkeit: $c^* = c_0/n = c_0/\sqrt{\mu\varepsilon} \approx 200000 km/s$
- einfach: keine vorgegebenen Protokolle, nur vorgegebene Signalbandbreite
- ausfallsicher: keine Unterbrechung, höchstens Übertragungsstörungen,
- geringe Bitfehlerrate
- abhörsicher nur bei Punkt-zu-Punkt-Verbindungen

Nachteile:
- unflexibel, da physikalisch festgelegt.
- unwirtschaftlich, da in der Regel nicht ausgelastet
- Engpass bei Überlastung
- abhörbar bei Rundspruch
- in der Regel nur SMX (simplex) oder HDX (half duplex)

Topologien: Bus (Rundspruch), Linie (Sequenz von Punkt-zu-Punkt-Verbindungen)
Anwendung: Basis für alle anderen Architekturen
Beispiele:
- HfD (= Hauptanschluß für Direktruf der Telekom)
- SFV (= digitale Standardfestverbindung der Telekom)
- Äther im Funkverkehr (Ursprung des Ethernet)
- Luft für (Ultra-)Schallübertragung

2.4.2 Wählverbindungen

Leitungsvermittlung, gewählte Durchschaltung (circuit switching)

Merkmale:
Drei Phasen der Wählverbindung:

- Verbindungsaufbau
- Datenübertragung wie bei fester Verbindung, Standleitung
- Verbindungsabbau (-auslösung) durch Kommunikationspartner (A, B) oder durch das Netz (Vermittler V)

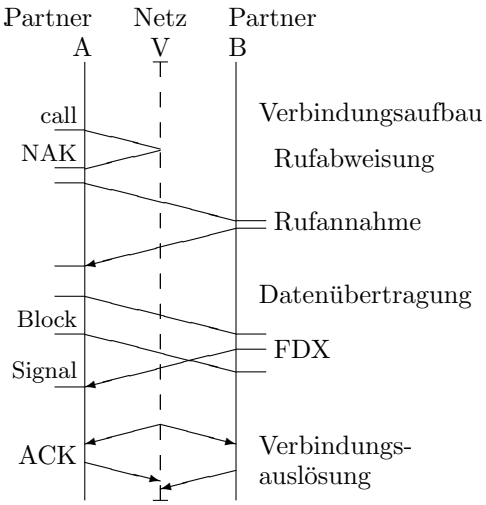


Bild n2p23

Vorteile:
- flexibel: Auswahl eines Partners möglich
- bessere Netzauslastung durch (zeitliche) Mehrfachnutzung

Nachteile:
- erfordert mindestens einen Zwischenknoten (Vermittler),
- Vermittler bildet Engpaß,
- Protokolle des Vermittlers müssen beachtet werden,
- unsicher (Ausfall des Vermitters, Abhören),
- Verbindungsaufbau /-abbau ist (zeit)aufwendig

Topologien: Stern u.ä. (Masche, Baum...)
Anwendungen: im LAN und WAN
Beispiele:
- Telecom: Datex-L, Telefax
- Privat: PABX (private automatic branch exchange) = Nebenstellenanlage

2.4.3 Teilstreckenvermittlung

Speichervermittlung (Store and forward)

Merkmale:
- segmentierter Datenstrom: die Daten werden in abgschlossenene Mengen (Nachrichten, Paketen) übertragen.
- die Daten werden in Zwischenknoten (Vermittler V) empfangen und gespeichert (store)
- der Pfad zum nächsten Knoten muß (statisch oder dynamisch) vorgegeben sein oder ermittelt werden (Wegewahl, routing, switching)
- die Daten werden nach Empfang weitergegeben (forward).

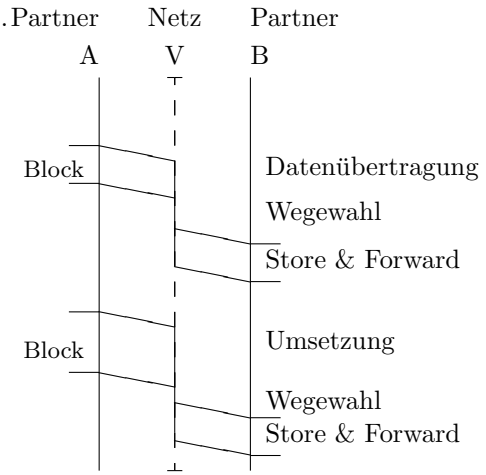


Bild n2p24

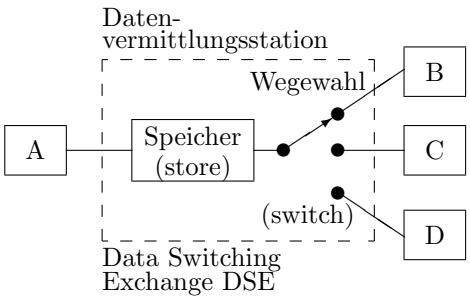


Bild n2p27

Vorteil: Asynchrone Datenübermittlung
- Sender und Empfänger müssen nicht gleichzeitig bereit sein
- Sender und Empfänger können mit unterschiedlichen Datenraten arbeiten
- Sender und Empfänger können unterschiedliche Protokolle verwenden

Nachteile:
- aufwendige Vermittler
- Verzögerung bei der Zwischenspeicherung

Topologien: Stern u.ä. (Masche, Baum...), Linie, Ring
Anwendungen: im WAN
Beispiele:
– Telecom: Datex-P

2.4.4 Sendungsvermittlung

Nachrichtenvermittlung (message switching)

Merkmale:
Nachrichten beliebiger Länge werden übertragen.
Vorteile:
- einfache Protokolle
- schneller, da weniger Protokollaufwand (Overhead)

Nachteile:
– Speicher in den Zwischenknoten können überlaufen
Anmerkung: künftig wegfallend

2.4.5 Paketvermittlung

(packet switching)

Merkmale:
- Pakete fester Minimal- und Maximalgröße
- Eine Nachricht wird in m Pakete zerlegt oder n Nachrichten werden in ein Paket gepackt.

Vorteile:
– geringere Speicherprobleme
Nachteile:
– höherer Verwaltungsaufwand bei den Anwendern (Teilnehmern A, B):
Zerlegen von Nachrichten zu Paketen und Zusammenfassen von Paketen zu Nachrichten

2.4.6 Datagramme

Daten-Telegramme

. .

Merkmale:
- Datenpakete mit Ziel- (und Quell)-adresse.
- Jedes Datagramm kann seinen eigenen Weg gehen

Vorteile:
- optimale Auslastung des Netzes
- Umgehung von Engpässen (Ausfällen)

Nachteile:
- Wegewahl für jedes Datagramm
- Datagramme können zwischen denselben Knoten unterschiedliche Laufzeiten haben
  z.B. auf den Wegen (a) ACDB oder (b) AEB
- Datagramme können unbemerkt verlorengehen
- Datagramme können fälschlicherweise dupliziert werden
- Empfänger muß eine Kontrolle durchführen

Topologien: Maschennetze
Anwendungen: im WAN und im GAN (Internet)
Beispiele:
– IP (Internet Protokoll)

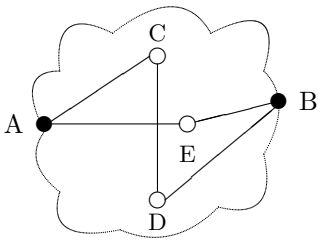


Bild n2p25

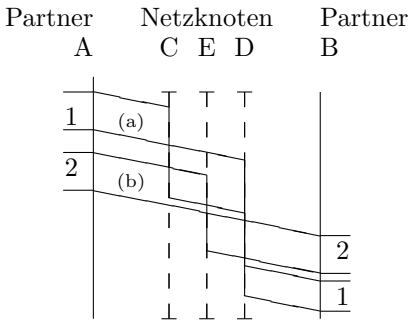


Bild n2p26

Virtuelle Verbindungen

Alle Pakete einer Nachricht durchlaufen denselbe Weg im Netz Bei fester virtueller Verbindung ist der Weg vorher festgelegt worden, bei gewählten virtuellen Verbindungen wird der Weg beim Verbindungsaufbau festgelegt und beim Abbau freigegeben.

2.4.7 Feste virtuelle Verbindung

PVC (permanent virtual circuit)

Merkmale:
- Fester Leitweg für jeden Pfad zwischen 2 Endknoten,
- alle Vermittler liegen fest.
- In jedem Vermittler gibt es dazu eine 1:1 - Zuordnung von ankommenden und abgehenden Leitungen (ports), die bei Einrichtung der Verbindung in Tabellen festgelegt wird.

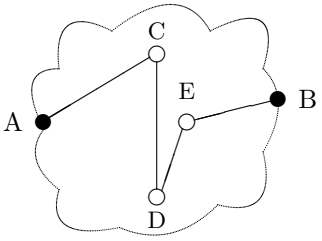


Bild n2p30

Vorteile:
- schnell, da kein Verbindungsaufbau notwendig
- effektiv, da die Datenpakete keine Adressen benötigen

Nachteile:
- starr, da keine Auswahl des Zielknotens möglich ist

Anwendungen: im WAN

Beispiele:
– DATEX-P10 der Telekom

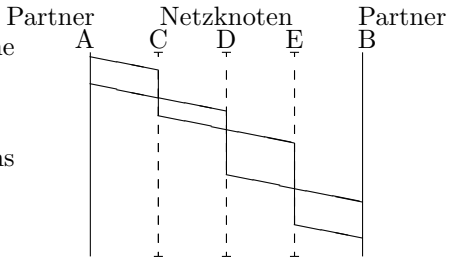


Bild n2p28

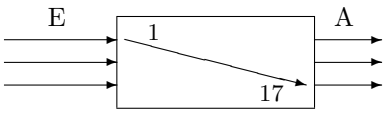


Bild n2p31

2.4.8 Gewählte virtuelle Verbindung

SVC (switched virtual circuit)

Merkmale: 3 Phasen . .

- Verbindungsaufbau durch ein Datagramm (Scout).
- Datenübertragung wie bei einer festen virtuellen Verbindung.
- Verbindungsabbau durch einen der Kommunikationspartner (A oder B) oder durch einen Netzknoten

Vorteile:
- flexibel: Partnerwahl möglich
- bessere Auslastung des Netzes
- bessere Ausnutzung der Betriebsmittel in den Vermittlern

Nachteile:
- langwierige Aufbauphase
- kritische Aufbauphase (nicht immer wird hier der optimale Weg gefunden)

Anwendungen: im WAN und im GAN

Beispiele:
– DATEX-P20 der Telekom

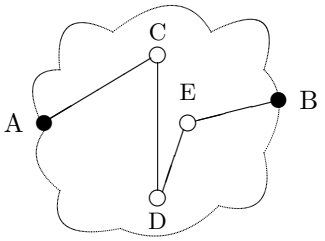


Bild n2p30

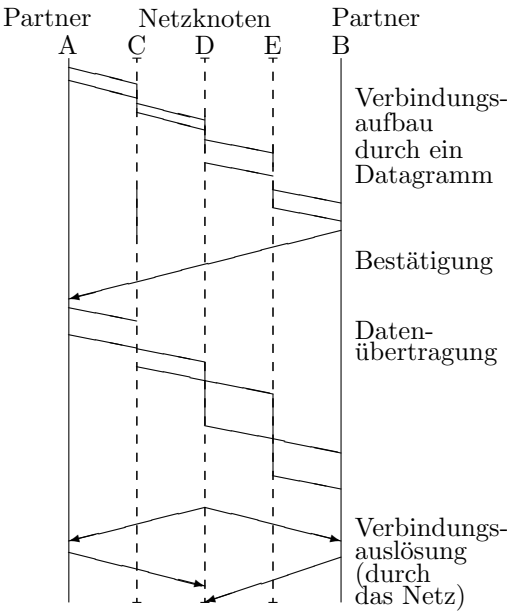


Bild n2p29

.

Kapitel 3

Das ISO-OSI-7-Schichtenbasisreferenzmodell

(OSI-BRM)

Open Systems Interconnection, OSI

Deutsche, Europäische und Internationale Norm: **EN ISO/IEC 7498: 1995**
entwickelt von der ISO (International Organization for Standardization) und der IEC (International Electrotechnical Commission) in Zusammenarbeit mit ITU-T (International Telecommunications Union, Telecommunication Standardization Sector) als CCITT/ITU-T Recommendation X.200.
Beginn der Normung 1977, erste Ausgabe 1984.

3.1 Kommunikation Offener Systeme

3.1.1 Das OSI-Environment

Anwendungen in Offenen Systemen (Kommunikationssystemen) arbeiten zusammen und bilden eine Assoziation (association), eine verteilte Anwendung.

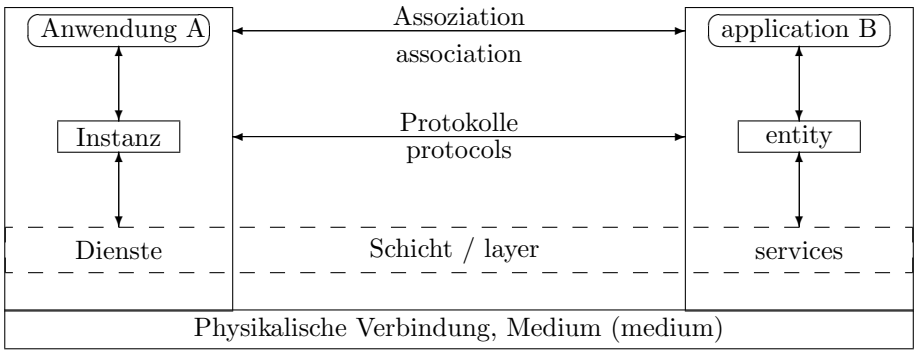


Bild n3p01

Begriffe aus ISO 7498:
- Reales System (real system):
  DV-Anwendung = Computer, Bediener und Anwender
- Reales offenes System (real open system):
  Reales System, das den Bedingungen von OSI entspricht
- Offenes System (open system): Abstraktes Modell eines realen offenen Systems
- Open Systems Interconnection (OSI):
  Verbindung 'Offener Systeme', entsprechend den Standards (von ISO oder IEC)
- OSI-Referenzmodell (RM):
  ein Modell, das die allgemeine Prinzipien von kommunizierenden 'Offenen Systemen' beschreibt

3.1.2 Das Schichtenmodell

Das Architekturkonzept:
- Unterteilung in Schichten
- Instanzen (Partner) bilden in jeder Schicht die aktiven Elemente
- Instanzen kommunizieren miteinander nach bestimmten Regeln (Protokolle)
- Instanzen kommunizieren mit Hilfe von Diensten der darunter liegenden Schicht (Basismaschine).

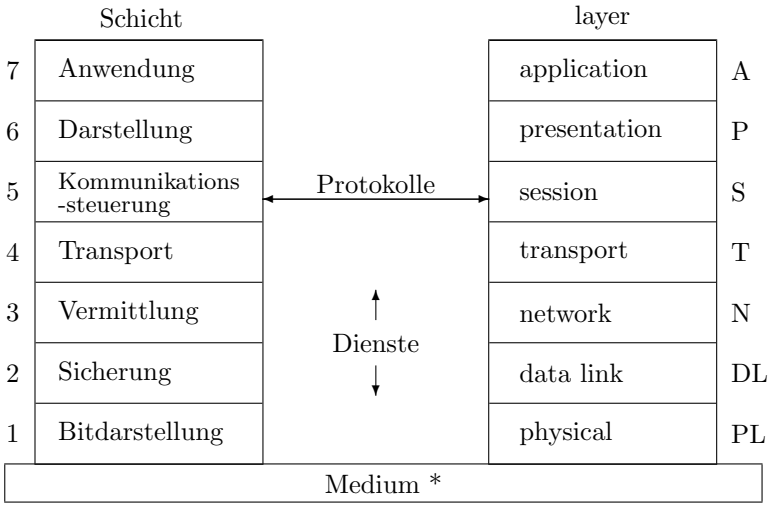


Bild n3p02

- keine Schicht darf umgangen werden, aber eine Schicht kann leer sein!
- obere Schichten sind Dienstbenutzer
- untere Schichten sind Diensterbringer
- die Schnittstelle heißt Dienstzugangspunkt (Service Access Point, SAP)

3.1.3 Modell einer Schicht

a) Hierarchie (Vertikale Struktur)
- Jede Schicht bildet eine abstrakte Maschine, die Funktionen ausführt.
- Die Funktionen werden von Instanzen ausgeführt.
- Je höher die Schicht, um so abstrakter und komplexer sind die Funktionen.

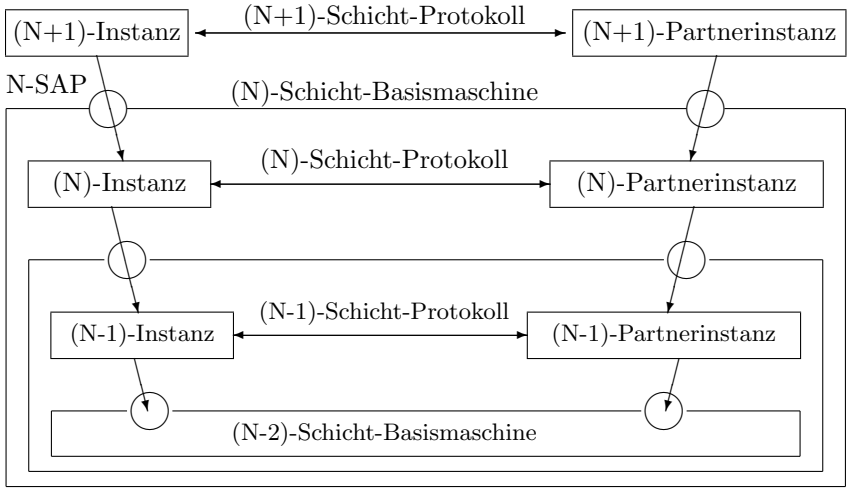


Bild n3p03

- (N+1)-Instanz ist ein Dienstbenutzer in der Schicht (N+1).
- (N)-Schicht-Basismaschine ist Diensterbringer der Schicht (N) für (N+1)-Instanzen
- (N-1)-Instanz ist ein Diensterbringer in der Schicht (N-1) für (N)-Instanzen

Dienstzugangspunkte (service access points, SAP):

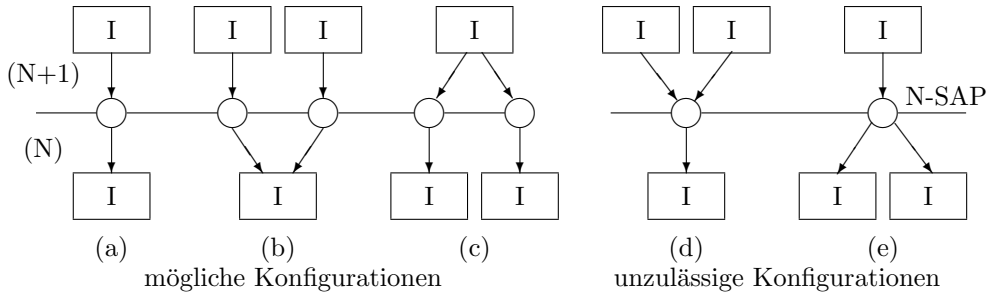


Bild n3p04

- Eine N-Instanz (I) bedient einen (a) oder mehrere (b) Dienstzugangspunkte (N-SAPs), d.h. erbringt einen oder mehrere Dienste bzw. führt eine oder mehrere Funktionen aus.
- Eine (N+1)-Instanz benutzt einen oder mehrere (N)-Dienstzugangspunkte (c).

- Ein (N)-Dienstzugangspunkt (N-SAP) wird immer nur von einer Instanz bedient und nicht von mehreren (d).
- Ein (N)-Dienstzugangspunkt wird immer nur von einer (N+1)-Instanz benutzt (b) und nicht von mehreren (e).
- Dienstzugangspunkte haben zwei Adressen:
  – eine aus Sicht der darüberliegenden Schicht (N+1)
  – eine andere aus Sicht der eigenen Schicht (N)

b) Kommunikation (Horizontale Struktur)

Verbindung (connection):
- Jedes System besitzt in jeder Schicht (N) mindestens ein Paar von Instanzen, die als Partner mit Hilfe eines Diensterbringers zusammenarbeiten.
- Instanzen kooperieren nach festgelegten Regeln (Protokollen).
- es gibt verbindungsorientierte Dienste (Connection Oriented Network Services, CONS) z.B. Standleitungen, die inder Regel (voll- oder halb-)duplex sind
- es gibt verbindungslose Dienste (Connectionless Network Services, CLNS) (z.B. Datagramme), die unidirektional (simplex) ablaufen.
- d.h. es gibt reale und virtuelle Verbindungen

Protokoll (protocol):
Satz von Regeln für die Kommunikation zwischen zwei Partnerinstanzen innerhalb derselben Schicht; es sind Regeln für die Benutzung von Diensten der darunter liegenden Schicht.

Kommunikation (communication):
Die Kommunikation zwischen zwei Partnerinstanzen erfolgt durch Dienstprimitive (service primitives), den Austausch von Nachrichten (Protokolldatenenheiten) mit Hilfe von Diensten.

3.1.4 Modell einer Instanz

Eine (N)-Instanz ist eine Funktionseinheit innerhalb einer Schicht (N), welche die Schichten (N+1) und (N-1) als Nachbarn hat (ausgenommen die Schichten 1 und 7). – Die Schicht 0 (Physikalisches Medium) ist nicht im OSI-BRM enthalten, wird aber im folgenden so behandelt wie die übrigen Schichten. –

Eine Instanz kann, je nach Schicht in der sie sich befindet, als Hard- oder Soft- oder Firmware realisiert sein.

a) Aufbau einer Instanz
Eine (N)-Instanz kann in 3 Funktionsschichten gegliedert werden, die nach dem EVA-Prinzip (Eingabe - Verarbeitung - Ausgabe) zusammenarbeiten.

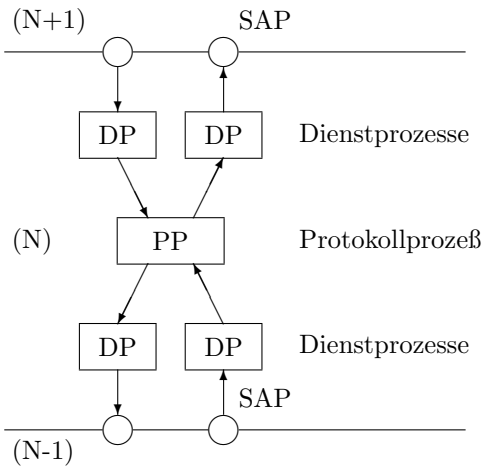


Bild n3p05

. .Dienstprozesse sorgen für die Annahme und Weitergabe von Dateneinheiten, ein Protokollprozeß führt die eigentliche Verarbeitung durch, d.h. er sorgt für die Einhaltung der Regeln des Protokolls, indem er die ankommenden Dienstdatenenheiten (Service Data Units, SDU) zusammen mit Protokollsteuerungsinformationen (Protocol Control Information, PCI) in eine neue Dateneinheit, die Protokolldateneinheit (Protocol Data Unit, PDU) pakt und an die darunterliegende Schicht weitergibt.

b) Datenübergabe zwischen den Schichten
Senden von Dateneinheiten:
Eine Protokolldateneinheit einer höheren Schicht, die (N+1)-PDU wird an einem (N)-Dienstzugangspunkt (Service Access Point, SAP) an die darunterliegende Schicht (N) abgegeben. Dort wird sie als (N)-Dienstdateneinheit (Service Data Unit, SDU) angenommen.

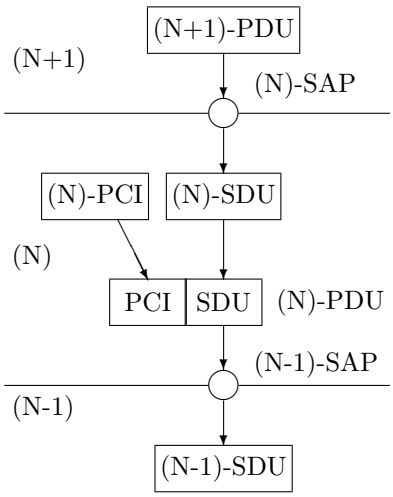


Bild n3p06

. .Der Protokollprozeß fügt die (N)-Protokollsteuerungsinformation (PCI) hinzu. Es ist Sache des Protokolls, an welcher Stelle die PCI zur SDU hinzugefügt werden, das kann am Anfang, am Ende oder zwischen den Daten der SDU sein (vgl. Start-Stop-Protokoll V.4). Ein Protokollprozeß kann auch PDUs ohne SDUs erzeugen, die dann nur Protokollsteuerungsinformation (N-PCI) enthalten und zur autonomen Koordination der entsprechenden Partnerinstanzen dienen. Die so entstandene (N)-Protokolldateneinheit (der Schicht (N)) wird an den Dienstzugangspunkt der darunterliegenden Schicht (N-1-SAP) weitergegeben. Dabei werden die Dateneinheiten immer umfangreicher.

Empfang von Dateneinheiten:
Beim Empfang beim Partner einer (N)-Instanz wird eine N-PDU an einem SAP angenommen, der Verpackungsvorgang rückgängig gemacht, von der N-Instanz die Protokollsteuerungsinformation N-PCI ausgewertet und dementsprechend eine N-SDU an die darüberliegende Schicht abgegeben bzw. ihr angeboten.
Beispiel: V.7 (s.w.u.)

c) Prozeßbeschreibung
Ein Endlicher Automat (Finite State Machine) wird beschrieben durch:
– das Eingabealphabet {E},
– das Ausgabealphabet {A},
– die Zustandsmenge {Z},
– den Anfangszustand Z_0,
– die Ausgabefunktion A = λ(E,Z),
– die Zustandsübergangsfunktion Z' = δ(E,Z)
Ein Endlicher Automat wird modelliert durch:
- Zustandsdiagramme
- Zustandstafeln, -übergangstabellen
- Beschreibungs- oder Spezifikationssprachen, z.B.:
  CC-Pascal (= Concurrent Pascal),
  CHILL (= CCITT High Level Language, Z.100)
  SDL (= Specification Description Language von CCITT, Z.200)

d) Komplexere Funktionen einer Instanz
1.) Segmentieren (Segmentation)

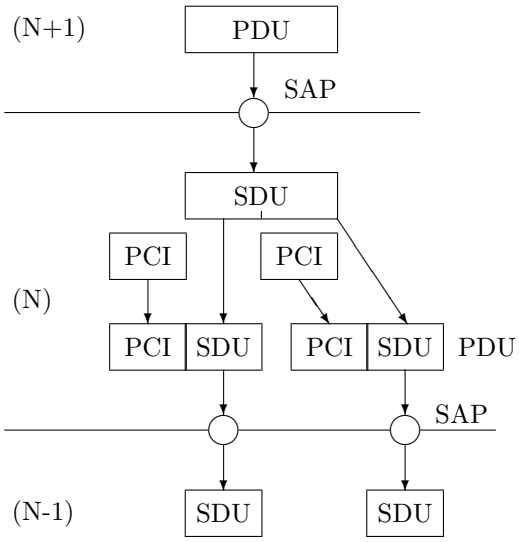


Bild n3p07

. .Dienstdateneinheiten (SDU), die zu groß sind, um durch die tieferliegende Schicht (N-1) transportiert zu werden, werden hier in Segmente geeigneter Größe zerlegt, mit eigener Protokollkontrollinformation (PCI), insbesondere einer Folgenummer versehen, und übertragen.
Auf der Gegenseite müsen die Segmente wieder in der richtigen Reichenfolge zusammengesetzt werden (Reassembling), und der darüberliegenden Schicht (N+1) übergeben werden.

2.) Blocken (Blocking)

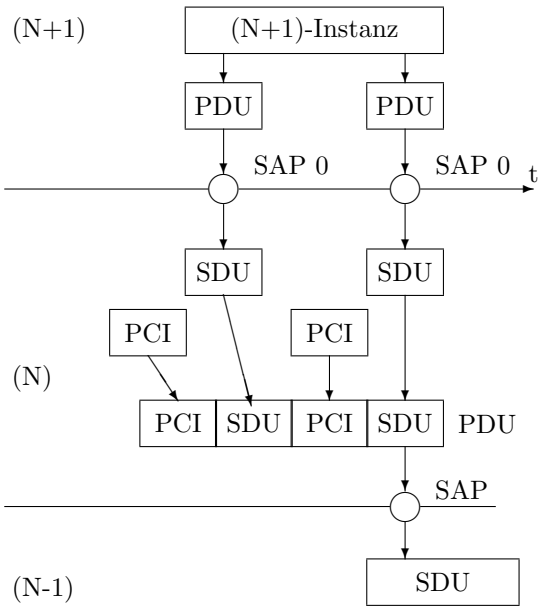


Bild n3p08

. .Wenn vom Dienstbenutzer (einer (N+1)-Instanz) zu kleine Pakete als SDUs angeliefert werden,
können sie (im Laufe der Zeit t) zu größeren Blöcken zusammengefaßt werden.
Zwischen den Paketen muß geeignete Protokollkontrontroll-information (PCI) eingefügt werden, um die Pakete auf der Gegenseite wieder trennen zu können (Deblocking).

3.) Multiplexen (Multiplexing)

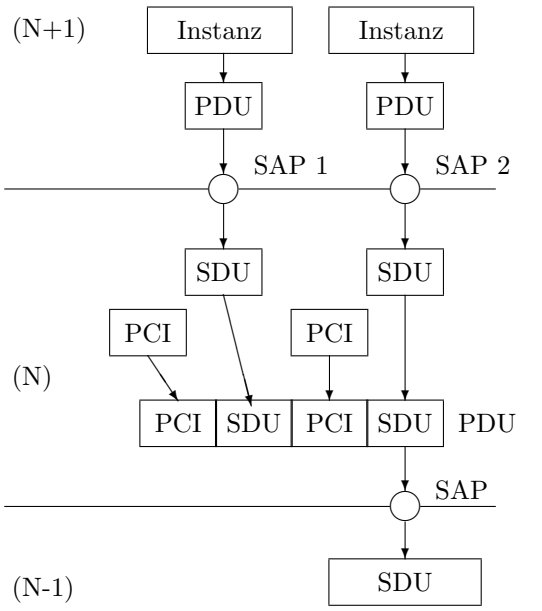


Bild n3p09

. .Wenn unterschiedliche (N+1)-Instanzen kleine Pakete anliefern, die vom Diensterbringer der Schicht N zu größeren Einheiten zusammengefaßt und an dieselbe Gegenstelle übertragen werden können, spricht man vom Multiplexen.
Das Gegenstück ist das Demultiplexen.

3.1.5 Dienste

a) Dienstgruppen

– Allgemeine Dienstgruppen, die in (fast) allen Schichten vorhanden sind:

- CONNECT	zum Verbindungsaufbau (in der Regel ein bestätigter Dienst)
- DISCONNECT	zum Verbindungsabbau (in der Regel ein bestätigter Dienst)
- DATA	zur Datenübertragung (in der Regel ein unbestätigter Dienst)
- ERROR	zur Fehlerbehandlung

– Schichtspezifische Dienstgruppen sind z.B.:

S-START_ACTIVITY	Beginn einer Aktivität der Schicht 5 (session)
P-SELECT_CONTEXT	Auswahl eines Contexts in der Schicht 6 (presentation)

b) Diensttypen

(1)	.request	Anforderung
(2)	.indicate	Anzeige
(3)	.response	Antwort
(4)	.confirm	Bestätigung

1–2 für unbestätigte Dienste
1–4 für bestätigte Dienste
1, 2, (4) für lokale Dienste

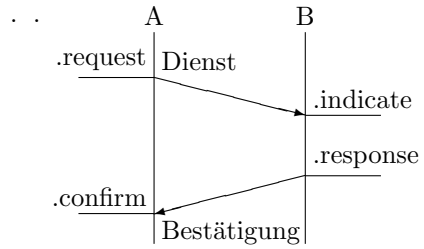


Bild n3p10

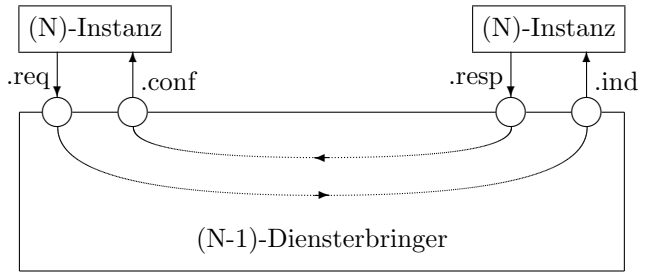


Bild n3p11

c) Dienstprimitive (Service Primitives)
Schicht-Gruppe.Typ [parameter]

Beispiele:

T-CONNECT.request	(∼ call, wählen, Wählton)
T-CONNECT.indicate	(∼ ring, klingeln, beim Partner)
T-CONNECT.response	(∼ off-hook, abheben)
T-CONNECT.confirm	(∼ connection indicator, Ende Rufton,Sprechbereitschaft)

Rückmeldung können aus verschiedenen Systemen kommen (z.B. Rufton als Bestätigung des Klingelns.

PL-DATA.request	(∼ senden, schreiben)
PL-DATA.indicate	(∼ empfangen, lesen)
DL-ERROR.indicate	(Lokaler Dienst, es gibt nur dieses Dienstprimitiv bei Error) ERROR-Anzeige kann sein: Interrupt oder Flag

3.1.6 Protokolle

- Vereinbarte Regeln zur Kommunikation zwischen Partnern.
- Regeln zur Nutzung der gemeinsamen Basismaschine.
- Festlegung des Protokollprozesses (der Protokollmaschine).
- Regeln für die Erzeugung und die Interpretation von PDUs.

Die Protokolle der einzelnen Schichten werden für jede Schicht gesondert in den folgenden Abschnitten dargestellt.

3.2 Die OSI-Schichten

- Jede Schicht hat eine darüberliegende und eine darunterliegende Nachbarschicht, ausgenommen die unterste und die oberste Schicht.
- Die einzelnen Schichten arbeiten weitgehend unabhängig voneinander.
- Die einzelnen Schichten werden unabhängig voneinander realisiert
- Die Instanzen der Schichten sind unterschiedlich realisiert,
- Die Instanzen können als Hard-, Soft- oder Firmware vorliegen
- Die Dienstzugangspunkte sind schichtspezifisch realisiert
- Jede Schicht bietet schichtunabhängige Dienste (z.B. CONNECT, DISCONNECT, DATA)
- Jede Schicht beinhaltet schichtspezifische Dienste
- Jede Schicht hat ihre eigenen Protokolle

Die Kommunikationspartner bilden
eine Assoziation (association) in den Anwendungsorientierte Schichten (7,6,5)
eine Verbindung (connection) in den Transportorientierte Schichten (4,3,2,1)

3.2.0 Das Medium

Zweck: Übertragung von Signalen
Dienste:
M - DATA.request (senden, Signale abgeben)
M - DATA.indicate (empfangen, Signale annehmen)
Instanzen: Das Medium (nur eine Instanz (!), daher nicht im BRM)
- Drähte (z.B. twisted pair)
- Coaxialkabel
- Lichtwellenleiter (LWL)
- Der "Äther" (bei Funk- oder Infrarotübertragung)
- Luft (bei akustischer Übertragung)

Dienstzugangspunkte, SAPs:
- Stecker, Kontakte (bei Drähten)
- Kontaktflächen (bei LWL)
- Antennen (beim Funk)
- Lautsprecher, Mikrofon (bei akustischer Übertragung)

Dienstdateneinheiten, SDUs: Signale
Protokolldateneinheiten, PDUs: keine
Funktionen: Transport von Energie.
Protokolle:
- Strom/Spannung (bei Drähten)
- Wellenlänge (bei LWL und Funk)
- Frequenz (bei akustischer Übertragung)

3.2.1 Die Bitübertragungsschicht (Physical layer)

Zweck: Bitübertragung zwischen Nachbarsystemen (Knoten)
Dienste: Datenübertragung zwischen Nachbarsystemen (Knoten)
PL-DATA.request
PL-DATA.indicate
Instanzen: Sender, Empfänger (beides Hardware)
Dienstzugangspunkte, SAPs: Stifte, Buchsen
Dienstdateneinheiten, SDUs: Bits
Protokolldateneinheiten, PDUs: Signale
Funktionen: Umsetzung von Bits in Signale und umgekehrt. (Signalcodierung)
Protokolle: Die Darstellung von Daten (Bits) erfolgt in der Regel durch isochrone Signale, in Schritten gleicher Länge (Taktschritten).
Die Datenübertragungsrate r wird in bps (Bits per second) gemessen, die Taktfrequenz (Baudrate) in Baud.
Kanalkapazität: Die maximale Datenübertragungsrate r wird durch die Art des Übertragungskanals bestimmt:

$\boxed{r_{max} = 2*B\ ld(n)}$ beim rauschfreien Kanal, mit B = Bandbreite (in Hz)
(die Baudrate ist 2*B) und n = Modulationstiefe (Anzahl unterschiedlicher Signalelemente), der ld = $\log_2$ (Logarithmus dualis) hat die Maßeinheit bit (Nyquist 1922).

$\boxed{r_{max} = B\ ld(1 + S/N)}$ beim rauschenden Kanal, S/N = Signal-Rausch-Verhältnis.
(Shannon 1948)

$\boxed{BER = \frac{1}{2}\ erf(X) \approx \frac{e^{-X}}{2\sqrt{\pi X}}}$ mit $\boxed{X = \frac{S}{N}\frac{\Delta f}{r}}$ ist die Bitfehlerrate (Bit Error Ratio)
(Sweeney 1992)

a) Basisbandverfahren(Gleichstromtelegraphie)
Die Darstellung von Daten erfolgt durch (Gleichstrom-)Signale, deren Pegel oder Pegelwechsel die Information enthalten. Die einzelnen Taktschritte bzw. die darin enthaltenen Signalelemente können von sehr kurzer Dauer sein und erscheinen meist als Züge von Rechteckimpulsen.
Ähnliche Verfahren findet man auch bei der Aufzeichnung von Daten auf Datenträgern (Magnetband, -platte, CD-ROM)

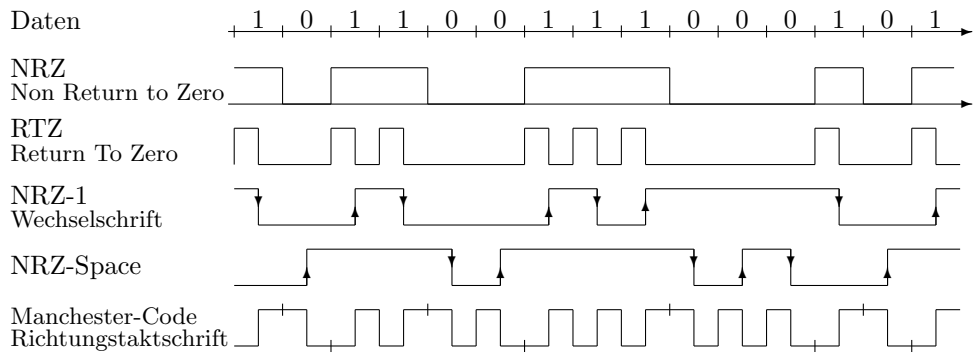


Bild n3p20

– **Pegelorientierte Basisbandverfahren:**
Zur Darstellung von Bits (Binärzeichen) werden Gleichspannungs- oder -stromwerte verwendet.

- Einfachstrom-Verfahren: Signale mit nur einer Polarität,
  z.B. TTL und 20-mA-Stromschleife (Current Loop): '1' = 20 mA, '0'= 0 mA.
- Doppelstromverfahren: Signale beider Polarität, z.B. V.28, V.11
- Richtungsschrift: Darstellung eines Bits durch einen konstanten Pegel des Signals über einen Taktschritt, z.B. NRZ (Non Return to Zero)
- Rückkehr nach Null, RTZ (Return To Zero): nur in der ersten Hälfte eines Taktschritts wird der dem Bit entsprechende Pegel dargestellt, in der zweiten Hälfte geht der Pegel auf Null zurück.

Pegel: z.B.:		
	V.28 (V.24)	'1' = –15 bis –3 V
		'0' = +15 bis +3 V
	V.11	'1' = –5 bis –0,3 V
		'0' = +5 bis +0,3 V

– **Flankenorientierte Basisbandverfahren:** (Taktschrift)
Zur Darstellung von Bits (Binärzeichen) wird der Wechsel zwischen zwei Pegeln verwendet, der in der Regel in der Mitte des Taktschritts erfolgt.

- Wechselschrift: Darstellung eines Bits durch Übergang (transition) zwischen zwei Zuständen (Pegelwechsel) in der Mitte eines Taktschritts, falls das Bit einen bestimmten Wert hat.
  Der Übergang wird auch als Flanke bezeichnet. z.B. NRZ-1 ≡ NRZ-mark
  '1' = Pegelwechsel
  '0' = kein Pegelwechsel
- Wechseltaktschrift (Two Frequency Encoding):
  Pegelwechsel bei jedem Taktschritt, zusätzlicher Wechsel in der Taktmitte, falls das Bit einen bestimmten Wert hat, z.B. '1'. Dadurch wird für einen Wert doppelt so viele Flanken erzeugt, was eine doppelt so hohen Pulsfrrequenz entspricht.
- Richtungstaktschrift (Phase Encoding, PE):
  Darstellung des Bitwerts durch die Phasenlage eines isochronen Impulszuges, z.B.
- Manchester-Codierung (Bi-Phase-inverted): Darstellung eines Bits durch seinen Komplementwert in der ersten Hälfte eines Taktschritts und seines Werts in der zweiten Hälfte, bzw, durch einen negativen (0) oder positiven (1) Pegelwechsel in der Mitte des Taktschritts.
- Manchester-Differenzcodierung: Darstellung eines Bits durch Abwesenheit (0) oder Vorhandenseins (1) eines Pegelwechsels am Beginn eines Taktschritts, während in der Mitte immer ein Pegelwechsel erfolgt. Dieses Verfahren liefert dieselben Impulszüge wie die Wechseltaktschrift, allerdings um 1/2 Taktlänge verschoben.
- Alle Taktschriften erlauben die Wiedergewinnung der Datenrate.

b) Breitbandverfahren (mit Modulation eines Trägers)
Träger ist eine Schwingung oder Welle mit der Darstellung $y(t) = A \cdot sin(\omega t + \varphi)$

Die Gesamtheit der Signalparameter (A, ω, φ) stellt den Signalzustand dar. Alle 3 Signalparameter (A, ω, φ) können zur Darstellung von Daten (Bits) verwendet werden.

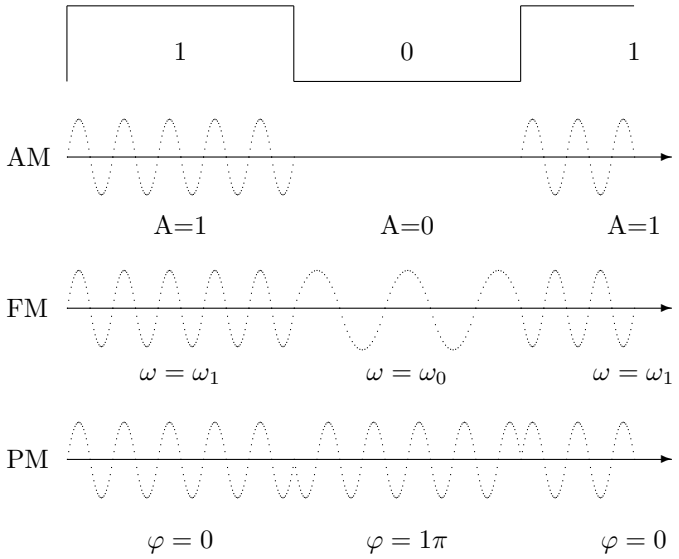


Bild n3p22

Eine Modulation kann kontinuierlich (analog) erfolgen oder
diskret (digital) als Umtastung, die mehrere (n) Zustände zuläßt, dann können pro Taktschritt mehr Daten übertragen werden:
Die Daten- oder Bitrate (bit rate) r gemessen in Bits pro Sekunde (bit/s oder bps) ist dann gegeben durch:

$$r = f_i \cdot log_2(n)$$

wobei f_i die Modulationsrate (Schrittgeschwindigkeit), d.h, die Zahl der Taktschritte pro Sekunde, gemessen in Baud, und n die Zahl der unterschiedlichen Signalzustände ist.

- Amplitudenmodulation (AM),
  Amplitudenumtastung (Amplitude Shift Keying, ASK):
  Darstellung von (Binär)zeichen durch unterschiedliche Werte der Amplitude A einer Schwingung oder Welle.
  NB: es empfiehlt sich, den Amplitudenwert $A = 0$ nicht zu verwenden, da sonst eine Unterbrechung der Übertragung als Signal gewertet wird.
  Diese Modulation ist störungsanfällig.
- Frequenzmodulation (FM),
  Frequenzumtastung (Frequency Shift Keying, FSK):

Darstellung von (Binär)zeichen durch unterschiedliche Werte der Frequenz ω einer Schwingung oder Welle. Falls Trägerfrequenz und Modulationsfrequenz nicht in einem ganzzahligen Verhältnist stehen, treten an den Schrittgrenzen diskontinuierliche Änderungen des Signals auf.

- Phasenmodulation (PM),
  Phasenumtastung (Phase Shift Keying, PSK):
  Darstellung von (Binär)zeichen durch unterschiedliche Werte der Pasenlage φ einer Schwingung oder Welle. Hier treten an den Schrittgrenzen fast immer diskontinuierliche Signaländerungen auf.
- PDM = Phasendifferenzmodulation:
  Darstellung des Binärzeiches durch Abwesenheit (0) oder Anwesenheit (1) eines Phasenwechsels in der Taktmitte (vgl. Wechselschrift).
- Pulscodemodulation, PCM: Vorgang, bei dem ein Signal abgetastet, jeder Abtastwert unabhängig von den anderen Abtastwerten quantisiert und die sich ergebende Folge von quantisierten Werten durch Codierung in ein digitales Signal umgesetzt wird. (IEV 702-06-59). Zum Beispiel:
- Pulsfrequenzmodulation, PFM: Pulszeitmodulation, bei der sich die Pulsfrequenz in Übereinstimmung mit einer gegebenen Funktion des Wertes des modulierenden Signals ändert. (IEV 702-06-58)

c) Eine Auswahl von Normen zur Breitbandübertragung:
CCITT (ITU-T) unterscheidet zwischen 2-Draht-Leitungen (über das öffentliche Telefonnetz - public switched telephone network PSTN bzw. general switched telephone network PSTN) und (gemieteten 4-Draht-Standleitungen - point-to-point 4-wire leased telephone-type circuits)

- V.21: Modem 300 Baud, FM (FSK), full duplex FDX, 2-Draht-Leitung
  A: '0' = 1180 Hz '1' = 980 Hz
  B: '0' = 1850 Hz '1' = 1650 Hz
- V.22: Modem 600 - 1200 Baud, PM (PSK), half duplex, 2-Draht-Leitung
  Trägerfrequenzen: Kanal 1: 1200 Hz, Kanal 2: 2400 Hz.

A:	600 Baud	synchron	2-wertige PM:	'0'	=	$+90°$
				'1'	=	$-90°$
B:	1200 Baud	asynchron	4-wertige PM:	'00'	=	$+90°$
				'01'	=	$0°$
				'10'	=	$-90°$
				'11'	=	$180°$

- V.23 Modem 600/1200 Baud auf 2-Draht-Wählleitungen: HDX-Übertragung mit 2-wertiger FSK ('0' = 450 Hz, '1' = 390 Hz) mit Signalisierungshilfskanal (75 Baud).
- V.27 Modem 4800 bps auf 4-Draht-Standleitungen: FDX-Übertragung von 3- Bitgruppen durch 8-wertige Phasendifferenzmodulation mit 1600 Baud auf 1800 Hz Träger.

- V.29 Modem 9600 bps auf 4-Draht-Standleitungen: FDX-Übertragung von 2-, 3- oder 4-Bitgruppen im kombinierten ASK und PSK mit 2400 Baud auf 1700 Hz Trägerfrequenz. Darstellung der Signalcodierung in Polarkoordinaten (A, φ)

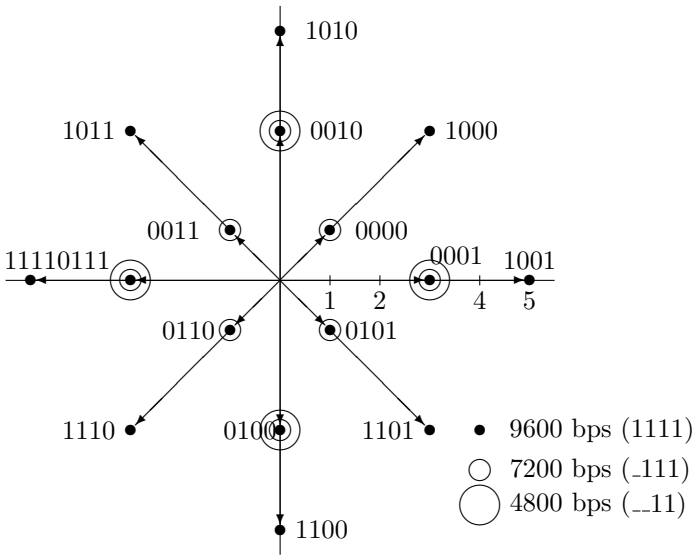


Bild n3p24

- V.32bis Modem mit Übertragungsraten bis zu 14400 bps auf 2-Draht-Leitungen im öffentlichen Telefonnetz. (A duplex modem operating at data signalling rates of up to 14 400 bit/s for use on the general switched telephone network and on leased point-to-point 2-wire telephone-type circuits)
  FDX-Übertragung auf der gleichen Trägerfrequenz (f=1800 Hz) mit Echounterdrückung, kombinierte ASK und DPSK mit 2400 Baud, zusätzlicher Codierung von 6 in 7 bit Gruppen (Trellis-Codierung) zur Fehlererkennung. (bei 4800 bps uncodiert, vgl. V.29).

Modem = Modulator + Demodulator
– **Amplitudenmodulation**
– **Amplitudenumtastung:** (stark vereinfacht)
Zur Modulation dient im einfachsten Fall ein Gatter, das das Trägersignal (mit der Trägerfrequenz f_0 durchläßt oder sperrt)
Zur Demodulation dient ein einfacher Einweggleichrichter (Diode D) und ein RC-Glied zur Glättung (Filter).

Modulator

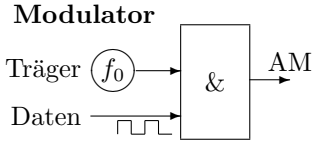


Bild n3p25a

.

.Demodulator

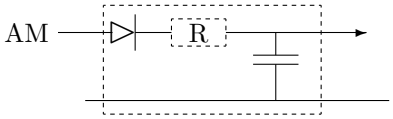


Bild n3p26a

– **Frequenzmodulation**
– **Frequenzumtastung:** (stark vereinfacht)
Zur Modulation dient im einfachsten Fall ein Umschalter zwischen 2 Trägerfrequenzen (f_0 und f_1), der hier aus 2 Gattern dargestellt wird, die mit dem Datensignal bzw. seinem Inversen angesteuert werden wie bei der Amplitudenumtastung und jeweils das eine oder das andere Trägersignal durchlassen. Anschließend werden beide Signale zusammengeführt (+) und übertragen.
Zur Demodulation sind 2 Filter (Bandpässe) notwendig, die auf die beiden Trägerfrequenzen abgestimmt sind, und entweder das eine oder andere Trägersignal durchlassen. Anschließende werden beide Signale (in Gegenrichtung) gleichgerichtet, wie bei der Amplitudenmodulation und zusammengeführt; diese Demodulation enthält eine gewisse Redundanz.

Modulator

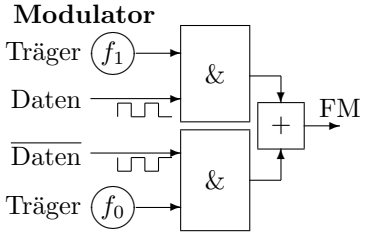


Bild n3p25b

Demodulator

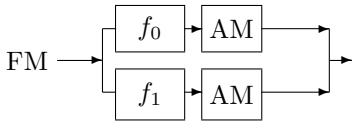


Bild n3p26b

– **Phasenmodulation**
– **Phasenumtastung:** (stark vereinfacht)
Zur Modulation dient im einfachsten Fall ein Umschalter zwischen dem Trägerfrequenzsignal (f_0) und seinem Inversen (180° Phasendrehung) wie bei der Frequenzumtastung.
Zur Demodulation wird das phasenmodulierte Signal mit einem Referenzsignal gleicher Frequenz (f_0) gemischt; in diesem Beispiel werden beide Signale einfach addiert, so daß das phasenrichtige Signal verdoppelt und das phasengedrehte Signal anulliert wird. Voraussetzung hierfür ist, daß die Amplituden und die Phasen von Träger und Referenzsignal übereinstimmen. Anschließend muß wieder eine Amplitudendemodulation (Gleichrichtung) erfolgen.

Modulator

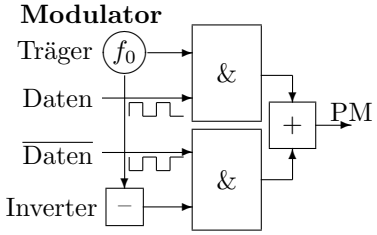


Bild n3p25c

Demodulator

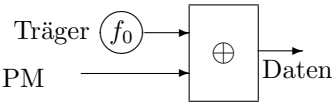


Bild n3p26c

3.2.2 Die Sicherungschicht (Data-link layer)

Zweck: Gesicherte Datenübertragung zwischen Nachbarsystemen.

Dienste:
DL-CONNECT.**
DL-DISCONNECT.**
DL-DATA.*
DL-ERROR. indicate (wichtig!)
{.** bedeutet bestätigte Dienste: request, indicate, response, confirm}
{.* bedeutet unbestätigte Dienste: request, indicate}

Instanzen: Hardware, und zwar:
– ICs, z.B.
UART (=Universal Asynchronous Receiver/Transmitter),
PUSART (=Programmable Synchronous/Asynchronous Receiver/Transmitter)
– (Netzwerk-)Karten.

Dienstzugangspunkte, SAPs:
Register von Integrierten Bausteinen (USART, PUSART)

Dienstdateneinheiten, SDUs: Bits, Bytes, Blöcke, Pakete

Protokolldateneinheiten, PDUs: Bits, Bitgruppen (Nibbles)

Funktionen:
- Aktivieren und Deaktivieren von Systemverbindungen für die Dienste CONNECT und DISCONNECT
- Datenübertragung (Ein-/Auspacken von Nutzdaten SDU/PDU) für den DATA-Dienst
- Datensicherung, für ERROR.indicate
  Fehlererkennung z.B. mit Prüfbits, ... , CRC (= Cyclic Redundancy Check)
  Fehlerbehebung z.B. mit ECC
- Flußkontrolle (Start-/Stopbetrieb)
- Splitten auf mehrere PL-Schicht-Diensterbringer, z.B. auf Parallelübertragung

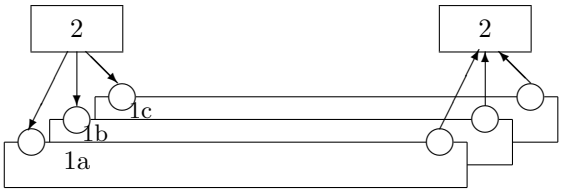


Bild n3p30

Protokolle:
Eine Auswahl von Normen und Standards:
- ISO 8886 (OSI- Data Link Service)
- ISO 3309 (HDLC Frame Structure)
- ISO 7776 (HDLC - X.25 LAPB - Link Access Procedure Balanced mode)

- ISO 8471 (HDLC address resolution and negotiation)
- ISO 7478 (multi link procedures)
- CCITT V.24 (serielle Schnittstelle)
- CCITT X.21 (seriell, synchron)
- CCITT X.25 (Datex-P)
- CCITT T.70/71 (Dienste und Protokolle im Telexnetz)
- CCITT I.440/441 (ISDN Interface)
- IEEE 802 = ISO 88802 = CCITT X.200 (LAN-Protokolle)

Beispiel: V.24 (RS-232 C, DIN 66020)
Die V.24-Schnittstelle ist viel älter als ISO-OSI. Sie läßt sich aber mit etwas Umsicht in OSI einordnen: Ein Teil von V.24 und den damit zusammenhängenden CCITT-Empfehlungen gehören zur Schicht 1 (PL), die übrigen zur Schicht 2 (DL).

– Zur Schicht 1 (Bitdarstellung) gehören:
a) Die Signale
- TxD Transmit Data = DATA.request
- RxD Receive Data = DATA.indicate
- CTS Clear to send ≈ CONNECT. indicate
- RTS Request to send ≈ CONNECT. request
- DTR Data terminal ready
- DSR Data set ready

Die letzten vier Signale dienen zur Modemsteuerung (handshaking)

b) Die Pegel nach V.28 (s.w.o.)

– Zur Schicht 2 (Sicherung) gehören:

1. Unterschiedliche Datenübertragungsprotokolle
a) ein asynchrones Protokoll: V.4

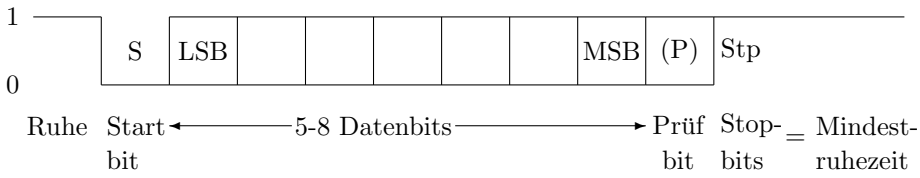


Bild n3p31

b) verschiedene synchrone Protokolle, z.B.
- BSC (= binary synchronous communication protocol)
- SDLC (Synchronous Data Link Control)
- HDLC (High-level Data Link Control)
- LAPB (Link Access Control Balanced mode)

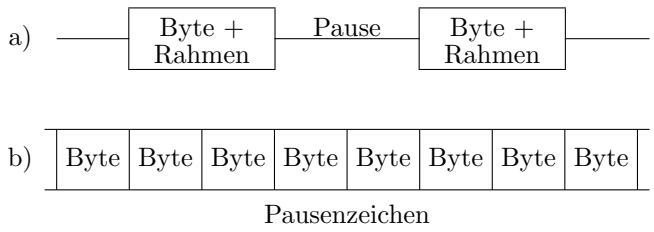


Bild n3p32

- a) Asynchrone Übertragungsprotokolle zeichnen sich dadurch aus, daß es Ruhezeiten beliebiger Länge (größer als die Mindestruhezeit der Stop-bits) gibt, nach denen 'asynchron' also zu einem beliebigen Zeitpunkt, ein Bitrahmen (s.o.) übertragen wird.
- b) Synchrone Übertragungsprotokolle ergeben einen andauernden Strom von Bits und Bytes, während der Pausen werden Pausenzeichen (z.B. das ASCII-Zeichen 'SYNC' = 16_{16}) übertragen.

NB: Ein anderes bekanntes Protokoll der Schicht 2 ist die parallele Schnittstelle nach Centronics. Hier werden die Daten auf mehrere (8) Schicht-1-Maschinen gesplittet und dort nach TTL-Protokoll übertragen.

2. Unterschiedliche Fehlererkennungsprotokolle

- a) beim asynchronen Protokoll V.4:
  optionale Paritätsprüfung für n Datenbits,
  Rahmenprüfung auf Eintreffen der Stoppbits nach 1+n+p Schritten,
  Flußprüfung auf Einhalten der Mindestruhezeit (s Stoppbits).
- b) bei den synchronen Protokollen: BSC, SDLC, HDLC, LAPB wird keine Prüfung auf der Bit/Zeichenebene durchgeführt, sondern meist in höheren Schichten, wo Prüfsummen über Zeichenfolgen erzeugt und auf Schicht 2 als 'normale' Daten übertragen werden.

3.2.2.1 Verfahren zur Fehlererkennung und -behebung

a) Erkennbarkeit und Korrigierbarkeit von Fehlern nach Hamming (1950)

- der Hamming Abstand d ist der (theoretische) Mindestabstand von zwei korrekten Bitmustern = Anzahl der unterschiedlichen Stellen in zwei richtigen Bitmustern.
- Um x Fehler zu erkennen muß $d \geq x + 1$ sein.
- Um y Fehler zu korrigieren muß $d \geq 2 \cdot y + 1$ sein.
- Das heißt: $x \approx 2 \cdot y$
  es können nur halb soviele Fehler korrigiert werden, wie erkannt werden können.

Beispiel: 1 Prüfbit/Wort (gerade Parität)

0	0	0	0	0	0	richtig
0	0	0	0	0	1	falsch
0	0	0	0	1	0	falsch
0	0	0	0	1	1	richtig
0	0	0	1	0	0	falsch
0	0	0	1	0	1	richtig

Daraus ergibt sich: d = 2 (richtige Bitmuster unterscheiden sich an zwei Stellen);
x = 1
Erkennung von 1-Bit Fehlern
oder einer ungeraden Anzahl von Fehlern $(2k + 1)$ in einem Wort mit n > k/2 Bits.
y = 0
keine Möglichkeit für eine Fehlerkorrektur.

b) Fehlererkennung

1. Paritätsprüfung (parity check)
Das Prinzip:
An jedes zu übertragenden Wort (Bitfolge), die als Dienstdateneinheit (SDU) übernommen wird, wird ein Prüfbit (parity bit) angehängt derart, daß die Quersumme (modulo 2) "0" oder "1", also die Summe aller "1"-en gerade (even) oder ungerade (odd) wird.
In V.4 ist diese Paritätsprüfung frei wählbar und muß bei Sender und Empfänger übereinstimmen.

- keine Paritätprüfung
- gerade Parität (even parity)
- ungerade Parität (odd parity)
- Füllbit "0" (space)
- Füllbit "1" (mark)

Die Fehlerrate sei z.B. $p_0 < 10^{-4}$, und es werde eine Poissonverteilung angenommen, d.h. das Auftreten der Fehler sei voneinander unabhängig (was in der Realität nicht immer gwährleistet ist)
Die Wahrscheinlichkeit für einen 1-Bit Fehler in einem n-Bit Rahmen ist dann:

$$p_1 = n \cdot p_0 < 8 \cdot 10^{-4}$$

Die Wahrscheinlichkeit für 2 Bit-Fehler in einem n-Bit Rahmen ist:

$$p_2 = n \cdot p_0 \cdot (n-1) \cdot p_0 < 7 \cdot 8 \cdot 10^{-8} \approx 0.510^{-6}$$

2. Zyklische Polynomcodes (Cyclic Redundancy Check, CRC)
Das Prinzip:
- Eine Bitfolge mit n Datenbit wird als n-stellige Dualzahl D betrachtet
- Diese Zahl wird durch eine andere, das Generatorpolynom G mit m bit Länge, dividiert ($0 < m < n$): D div G = Q Rest P
- Der Rest P=P(D) wird an die Bitfolge D angehängt, das ergibt eine neue Zahl $R = D \cdot 2^m + P$, die ohne Rest (mod 2) durch G dividierbar ist.
- Diese Prüfung wird beim Empfang durchgeführt.
- Ist bei der Übertragung ein Fehler aufgetreten, so hat sich der Wert der übertragenen Zahl R um einen Wert E geändert und gibt bei der Division durch G nicht mehr 0 sondern E/G. Falls jedoch E durch G ohne Rest teilbar ist, wird kein Fehler erkannt.

Anmerkung:
Die hiergenannten Zahlen (D, G, P) stellen als Bitfolgen Dualzahlen dar: $\sum_{i=0}^{n} x_i \cdot 2^i$.
Für eine verkürzte Darstellung werden nur die Glieder der Reihe angeben, deren Bits '1' sind, in der Form: $x^m + x^l + ... + 1$. Den Dualwert erhält man durch Einsetzen von $x = 2$.
Der Grad des Generator-Polynoms G muß $m \geq 1$ sein, d.h. G≥ 3.

Beispiel:
Bitfolge: 100010010001
Zahl: $891_h = 2193$
Polynom: $x^{11} + x^7 + x^4 + 1$

Die Art der erkennbaren Fehler hängt stark vom Generatorpolynom G ab:
- um 1-Bit-Fehler zu entdecken, muß G ungerade sein, also LSB = 1 (s.o. $x^0 = 1$)
- um 2-Bit-Fehler zu entdecken, darf G kein Teiler von $2^n + 1$ sein
- falls G eine Primzahl ist oder große Primzahlen als Teiler enthält, werden ebenfalls viele Fehlerkonstellationen erkannt, z.B. Bündelfehler (bursts) bis zur Länge m.

Genormte Generatorpolynome:
- CRC-12: $x^{12} + x^{11} + x^3 + x^2 + x^1 + 1 = 6159 = 3 \cdot 2053$(3*prim)
- CRC-16: $x^{16} + x^{15} + x^2 + 1 = 98309 = 37 \cdot 2657$(37*prim)
- CCITT V.42: $x^{16} + x^{12} + x^5 + 1 = 69665 = 5 \cdot 13933$ (5*Primzahl)

Diese Prozedur kann recht einfach auf Hardware-Ebene (billig und schnell) realisiert werden. Benötigt werden nur XOR-Gatter und ein Schieberegister, in das an den Prüfpositionen ($x^i = 1$) XOR-Gatter eingefügt sind.

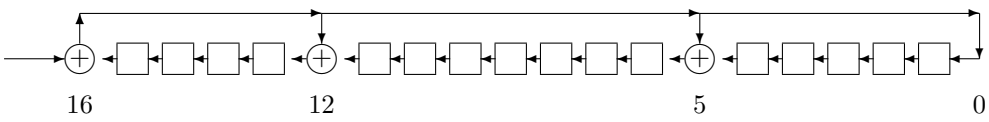


Bild n3p35

Hardware für den CRC nach CCITT V.42

c) Fehlerbehebung durch ECC (= Error Correcting Codes)

1. Blocksummenprüfung

	Block von Oktetts mit Querparität															
															Längsparität	
	←					Nutzdatenbytes							→		↓	
↑	1	1	1	0	1	1	1	0	0	0	1	1	0	0	1	0	↑
|	0	0	1	1	0	0	1	0	1	1	1	0	0	1	0	1	|
|	1	0	0	1	1	0	0	1	1	0	1	1	1	0	1	1	|
frames	1	1	1	0	1	1	1	0	0	0	1	0	1	0	1	0	Nutz-
|	0	1	0	0	1	0	1	1	1	1	1	0	1	0	1	0	bits
|	1	0	0	1	0	1	0	0	0	0	0	0	1	1	0	0	|
|	0	1	1	0	0	0	1	1	0	0	1	1	0	0	1	1	|
|	0	0	1	1	1	0	1	0	1	0	0	0	1	0	1	1	↓
↓ Quer	0	0	1	0	1	1	0	1	0	0	0	1	1	0	0	0	parität

1-Bit Fehler: ist erkennbar, lokalisierbar und somit korrigierbar
2-Bit Fehler: ist erkennbar, aber nicht eindeutig lokalisierbar und somit nicht korrigierbar
3-Bit Fehler: ist erkennbar, im 'worst case' wird er als 1-Bit Fehler behandelt und falsch korrigiert
4-Bit Fehler: im 'worst case' nicht erkennbar

2. Hamming Codes
Hier werden die Bits nicht in einer einfachen Blockmatrix angeordnet, sondern eher im Dreieck, in Blöcken wachsender Länge. Die Prüfbits werden ebenfalls für Gruppen von Bits erzeugt, die aber anders angeordnet sind.

Das Prinzip

- Die Nutzbits b_i $(i \in (1....m))$ werden in Gruppen wachsender Länge $(2^j - 1)$ mit gleichgroßen Lücken zusammengefaßt,
- je Gruppe wird ein Prüfbit p_j (gerader) Parität bestimmt $(j \in (1..r)$ mit $r \geq log_2(m + r + 1)$ b.z.w. $2^r - 1 \geq m + r)$
- die Prüfbits werden nach Vorgabe zwischen die Nutzbits eingesetzt
- Bei der Kontrolle wird festgestellt, welche Prüfbits falsch sind.

Anordnung von Nutz- und Prüfbits:

$p_0\ p_1\ d_1\ p_2\ d_2\ d_3\ d_4\ p_3\ d_5\ d_6\ d_7\ d_8\ d_9\ d_{10}\ d_{11}\ p_4...$

Die Bestimmung der Prüfbits:

p_0		d_1		d_2		d_4		d_5		d_7		d_9		d_{11}
	p_1	d_1			d_3	d_4			d_6	d_7			d_{10}	d_{11}
			p_2	d_2	d_3	d_4					d_8	d_9	d_{10}	d_{11}
							p_3	d_5	d_6	d_7	d_8	d_9	d_{10}	d_{11}

Fehlerkorrektur:
- Beim Empfang werden die Prüfbits neu bestimmt.
- Abweichende Prüfbits werden als "1" betrachtet, die anderen als "0".
- Aus diesen Bits wird ein Wort gebildet, dessen Wert ergibt unmittelbar die Position des verfälschten Bits.
- Dieses Bit negiert.
- Beim Auftreten von 2-Bit-Fehlern versagt dieses Verfahren und führt zu falschen Korrekturen !

d) Effizienz
Als Effizienz E eines Codes wird das Verhältnis der Anzahl n der Nutzbits zur Gesamtzahl (n+m) der übertragenen Bits bestimmt, wobei m die Zahl der Prüfbits ist.
$E = \frac{n}{n+m} < 1$
Als Redundanz R wird der Überschuß an Prüfbits betrachtet:
$R = 1 - E$

3.2.2.2 Rahmen (framing)

Aus- und Einpacken von SDUs in PDUs
Zweck: Abgrenzen (delimiting) und Synchronisation (synchronization) von Datenströmen, das Auffinden von Beginn und Ende einer SDU innerhalb einer PDU

a) Start-Stop-Prozedur V.4
bei der asynchronen seriellen Schnittstelle (V.24) Framing auf Bit-Ebene:
- Startbit = erstes Bit, das $\neq$ Ruhezustand ist
- n Datenbits mit n = 5, 6, 7, 8 (wählbar)
- p Prüfbit mit p = 0,1 (wählbar), Parität (odd, even) wählbar
- s Stopbits mit s = 1, 1.5, 2.0 (wählbar)

b) Synchrone serielle Prozeduren:
Framing auf Byte-Ebene (1 Byte = 1 Oktett = 8 bit)

BSC = Binary synchronous communication protocol, BiSync
(SDLC = Synchronous Data link control protocol)

Code-abhängig

Pausenzeichen	SYN	$16 ASCII
Start of Header	SOH	$01 ASCII
Start of Text	STX	$02 ASCII
End of Text	ETX	$03 ASCII
End of Block	ETB	$04 ASCII

Struktur:

SYN	SYN	SOH	header	STX	text	ETB/ETX	checksum	SYN

c) Bitorientierte Protokolle
Keine Strukturierung in Oktetts z.B. HDLC (= High level data link control protocol)
Rahmen werden durch Flags gebildet: HDLC - Flag = 0 1 1 1 1 1 1 0 = $ 7E

Falls dieses Zeichen ($7E) oder ähnliche im Datenstrom auftauchen, erfolgt Bitstuffing: Nach je fünf Einsen eine Null einschieben.
Beispiel:

Daten	DL-SDU:	0	1	1	1	1	1	1	1	Oktett
	PL-PDU:	0	1	1	1	1	1	0	1	1	Nonett

d) Rahmengrößen
Die Verfahren (a) - (c) beinhalten keine prinzipielle Einschränkung der Größe der Datenblöcke.
Mit der Rahmengröße wächst einerseits die Fehlerwahrscheinlichkeit, andererseits steigt die Effizienz.
a) bei V.4 sind 5 bis 8 Datenbits vorgesehen, eine Erweiterung ist denkbar.
b) bei X.25 (HDLC) sind es maximal 128 Bytes an Nutzdaten und 2 Bytes CRC Prüfsumme,
c) bei LAPB ist die Zahl der Nutzdaten nicht begrenzt.

3.2.2.3 Kompression (data compression)

Verringerung der zu übertragenden Datenmenge (entropy encoding)

a) Lauflängen-Codierung (run length encoding)
Falls mehr als 4 gleiche Zeichen (Bytes, Oktetts) aufeinander folgen, werden sie durch einen Rahmen ersetzt:
Falls $Zeichen = Flag$ muß in jedem Fall codiert werden.

	Flag	Zeichen	Länge
Oktetts	1	1	2

b) Statistische Codierung (statistical encoding)
Ersetzung von längeren, häufiger vorkommenden Zeichenketten durch Code-Symbole (vgl. Morse-Code: die häufigsten Zeichen sind die kürzesten "e" = ·)
Dazu muß der zu übertragende Text (Datei) zuerst auf die Häufigkeit der darin vorkommenden Zeichen (Oktetts) analysiert und einander zugeordenet werden. Aus den Zeichen und ihren Häufigkeiten werden neue, unterschiedlich lange Symbole gebildet und übertragen.

Huffman Codierung

Ziv-Lempel Algorithmus

c) Farbwerttabellen (Color Look Up Tables CLUT)
Bei Farbbildern sind die Pixel durch RGB–Werte (Rot Grün Blau) dargestellt. Meist je ein Oktett pro Farbe (RGB), d.h. 2^{24} mögliche Farbwerte, die selten ausgeschöpft werden. Nach Feststellung aller vorkommenden Farbwerte, werden diese in eine Tabelle gestellt (die kürzer ist als 2^{24}) und durch ihre Position in der Tabelle ersetzt (die Tabelle muß natürlich auch übertragen werden)

3.2.3 Die Vermittlungschicht (Network layer)

Zweck: Verbindung von Endsystemen (end-to-end-connection)
Dienste:
N-CONNECT.**
N-DISCONNECT.**
N-DATA.*
N-ERROR.indicate
N-EXPRESS_DATA.* (Datagramme)
N-EXPEDITED_DATA.* (Vorrangdaten)
N-QUALITY.request (Qualitätsanforderung)
N-RESET.* (Netzwerk reset)
{.** bedeutet bestätigte Dienste: request, indicate, response, confirm}
{.* bedeutet unbestätigte Dienste: request, indicate}

Instanzen: Hardware, Software, Firmware:
HW: (Intelligente) Controller
SW: Treiber, Netzwerksoftware z.B. Net BIOS
FW: Software in ROM eingebrannt (unveränderbar, um Vorschriften einzuhalten)

Dienstzugangspunkte, SAPs:
Adressen von Controller-Registern
Adressen von Routinen (z.B. im BIOS)
SW-Interrupt Vektoren ← Trap Instruction

Dienstdateneinheiten, SDUs: Nachrichten, Telegramme

Protokolldateneinheiten, PDUs: Bits, Bytes, Blöcke, Pakete
Funktionen:
a) Herstellen, Überwachen und Beenden von Verbindungen (Routing):

- Wahlleitungen
- PVC (Permanent Virtual Connection)
- SVC (Switched Virtual Connection)
- Datagramme

Einschließlich der Zuordnung von Betriebsmitteln im System (mit Hilfe des Betriebssystems)

b) Datenübertragung zwischen Endsystemen
c) Segmentieren und Reassembling von Paketen (Verkürzen von Nachrichten)
d) Flußregelung (z.B.: bei Pufferüberlauf)
e) Fehlerbehebung (durch Wiederholung) oder
Fehlermeldung (bei nichtbehebbaren Fehlern)
f) Multiplexen von Transportverbindungen

Protokolle:
- CONS = Connection Oriented Network Services, z.B. X.25 PVC und SVC
- CLNS = Connectionless Network Services, Datagramme
- LAN-Protokolle (IEEE 802 = ISO 8802)

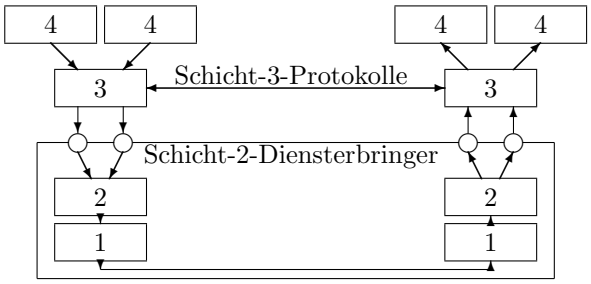


Bild n3p36

Normen und Standards:
CCITT X.21: DTE-DCE Interface for Circuit-Switched Public Data Networks
CCITT X.25: DTE-DCE Interface for Packet-Switched Public Data Networks
CCITT X.121: International Nubering Plan
CCITT X.213: OSI Network Services Definitions
CCITT X.223: OSI Network Protocol Definitions
CCITT E.163: Numbering Plan for the International Telefone Services
CCITT E.164=I.331 Numbering Plan for Integrated Services Data Network ISDN
CCITT I.330: ISDN Numbering and Addressing Principles
CCITT I.450 ISDN Network Services (layer 3)
CCITT T.90 Telematic Services
ISO 3166 Ländernamen
ISO 6523 Namen von Organisationen
ISO 8348 = X.213
ISO 8208, 8878 X.25 Packet Protocol (CONS)
ISO 8473 Datagram Protocol (CLNS)
ISO 8880 Network Services
ISO 8881 Anwendung von X.25 in LANs nach ISO 8802
ISO 8648 Interne Struktur der Schicht 3 (Unterschichten)
ISO 9542 Protokolle zwischen End- und Transitsystemen

Netzwerkklassen (Güte)

Typ A:	Fehlerfrei (LAN)
Typ B:	Keine Datenfehler aber Netzzusammenbrüche (z.B.: X. 25, WAN) Beim Zusammenbruch kommt der Dienst: N-RESET.indicate
Typ C:	Keine Datenfehler aber Verlust u. Verdopplung von Paketen und Systemzusammenbrüche (z.B.: ARPANET, Internet, GANs)

Verbindungen zwischen Endsystemen

a) Verbindungsarten:

- verbindungsorientiert (CONS) / verbindungslos (CLNS)
- Leitungsverbindungen / Virtuelle Verbindungen (Speichervermittlung, Paketdienste) kennzeichnend für virtuelle Verbindungen ist store and forward.

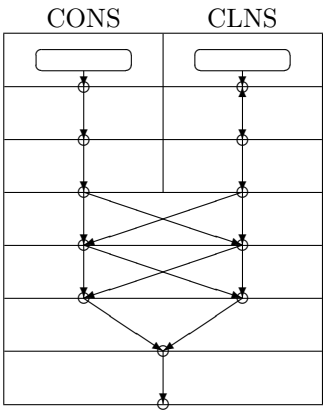


Bild n3p37

. .Im OSI-Stack werden verbindungslose Dienstanforderungen in der Regel auch durch verbindungslos arbeitende Diensterbringer (darunter liegende Schicht) erbracht. In den (transportorientierten) Schichten 3 und 4 kann hiervon abgewichen werden. So kann eine virtuelle Verbindung der Schicht 4 durch Datagramme der Schicht 3 hergestellt werden (bei TCP/IP). Um gekehrt kann es sein, daß für ein Datagramm der Schicht 4 auf der Schicht 3 eine Verbindung aufgebaut werden muß.

b) Routing (Wegewahl)

Grundlage sind feste Verbindungen (Leitungsverbindungen) der Schicht 1.
Virtuelle Verbindungen (PVC und SVC) werden auf Schicht 3 abgehandelt

PVC, feste virtuelle Verbindungen

In jedem Knotenrechner (A ... F) werden Tabellen eingerichtet, nach denen per Soft- oder Hardware Verbindungen zwischen Eingangs- und Ausgangsports hergestellt werden, den Zugangspunkten von Schicht-2-Basismaschinen. Diese Listen werden beim Einrichten eines PVC (z.B. Datex-P10) gefüllt und im Arbeitsspeicher gehalten. Für einen Restart müssen sie natürlich auch auf Massenspeichern (Festplatte) vorgehalten werden.

Vorteil dieser Methode ist eine einfachere Software,

Nachteil ist der notwendige manuelle Eingriff (wie beim Verdrahten einer Standleitung) und die dauernde Belegung von Betriebsmitteln; es ist also teuer.

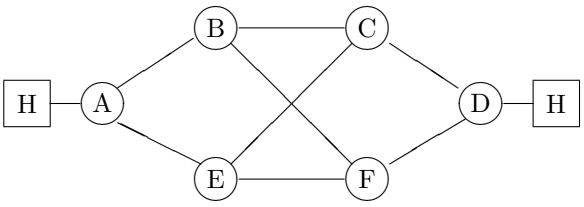


Bild n3p38

. .Beispiel:

Liste (A):

Eingang	Ausgang
port1 (H)	port6 (B)
.	.
.	.
.	.
port4 (E)	port5 (H)

SVC, gewählte virtuelle Verbindungen
Die Tabellen in den Knotenrechnern werden mit Hilfe von geeigneten Algorithmen (Routing-Algorithmen) dynamisch erstellt und verwaltet. (z.B.: 'shortest path').

N-CONNECT: Ein Datagramm durchläuft (als Scout) das Netz und hinterläßt eine Spur in Form von Listeneinträgen. Wenn der Scout am Zielrechner angekommen ist, wird eine Rückmeldung (Bestätigung) an den Quellrechner geschickt, und zwar genau über den Weg, den der Scout vorher ausgesucht hat. Diesen Weg nehmen dann auch die Daten.

N-DATA: Die Daten werden über die bestehende virtuelle Verbindung geleitet wie beim PVC.

N-DISCONNECT: Die Listeneinträge werden wieder gelöscht, die Betriebsmittel freigegeben.

Vorteil dieser Methode ist die bessere Ausnutzung von Betriebsmitteln,
Nachteil ist die Fehleranfälligkeit.

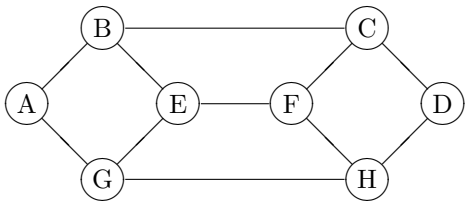


Bild n3p39

. .Beispiel: Liste (G)

Eingänge	Ausgänge
port 1 (A)	-
port 2 (E)	-
port 3 (H)	-

Die Eingänge sind fest zugeordnet, die Ausgänge werden dynamisch bestimmten Ports zugeordnet, die fest an andere Rechner angeschlossen sind.

c) Flußkontrolle, Datenflußsteuerung

Bei Store and Foreward (Speichervermittlung) kann der Speicher überlaufen, daher muß Datenfluß gesteuert werden. (vgl. <Ctrl/S> / <Ctrl/Q> am Terminal)

. .

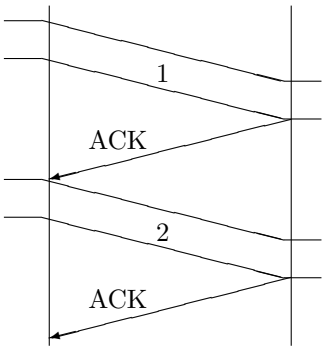


Bild n3p40a Stop & Go Betrieb: nach jedem Paket erfolgt eine Bestätigung (ACK)

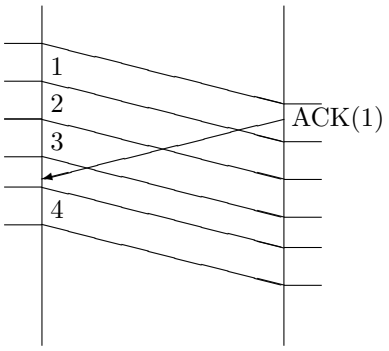


Bild n3p40 Sliding Window: der Sender erwartet eine Bestätigung (ACK) des Pakets i-n vor dem Senden des Pakets i (hier ist n=3 gesetzt).

Dienste nach X.213 ≡ ISO 8348
PDU-Paketgröße = 63 Byte

Funktionsaufrufe für verbindungsorientierte Dienste (CONS):
N-CONNECT.**(Ziel,Quelle, Bestätigt, Beschleunigt, Qualität(sstufe), Restdaten)
N-DISCONNECT.**(Ziel, Quelle, Daten)
N-DATA.*(Daten)
N-DATA_ACKNOWLEDGE.*()
N-RESET.*(Quelle, Grund)
{.** bedeutet bestätigte Dienste: request, indicate, response, confirm}
{.* bedeutet unbestätigte Dienste: request, indicate}

Funktionsaufrufe für verbindungsorientierte Dienste (CONS):
N-UNITDATA.*(Ziel, Quelle, Qualität(sstufe), Daten) ≤ 63 KByte
N-FACILITY.request(Qualität(sstufe))
N-FACILITY.indicate(Ziel, Qualität(sstufe), Grund)
N-REPORT.indicate(Ziel, Problem, Grund)

Dienste nach X.25 ≡ ISO 8208 (Datex-P)
Übermittlungsstruktur

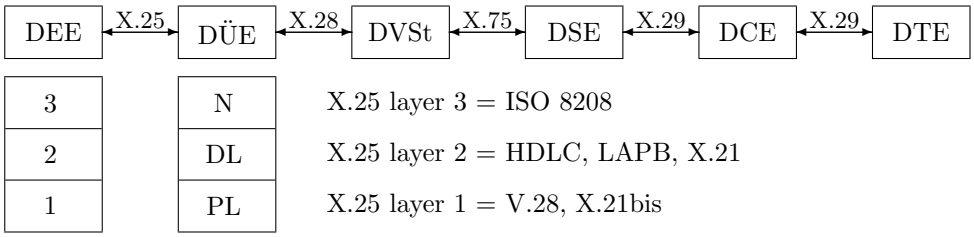


Bild n3p41

DEE = Datenendeinrichtung ≡ DTE = Data Terminal Equipment
DÜE = Datenübermittlungseinrichtung ≡ DCE = Data Circuit terminating Equipment
DVSt = Datenvermittlungssation ≡ DSE = Data Switching Exchange

Bei Endeinrichtungen (DTE), die nur im Start-Stop-Betrieb (V.24 asynch) arbeiten, muß zwischen DTE und DCE (Modem) ein spezielles Protokol (X.29) und zwischen DCE und DSE (hier PAD = Packet Assembler and Dissasembler genannt) eine spezielle Übertragungsprozedur (X.75) eingehalten werden.

CALL.request = N-CONNECT.request(indicate) – PDU (an Schicht 2):

← 8 Bit →	. .
0001 | group channel typ = 0000 1011	(1)
length of calling address | length of called address	
calling address	(2)
≈ .. ≈ called address	
00 | facilities length	
facilities	(3)
user data	max 16 Byte

1: header
Virtual Channel (VC)
= group + channel

2: Adreßformate gemäß X.121

3: z.B. R-Gespräch,

CALL.accepted = N-CONNECT.response(confirm) – PDU (an Schicht 2)
wie oben, mit:
– Typ = 0000 1111
– facilities werden bestätigt oder mit einem Gegenvorschlag beantwortet, z.B.:

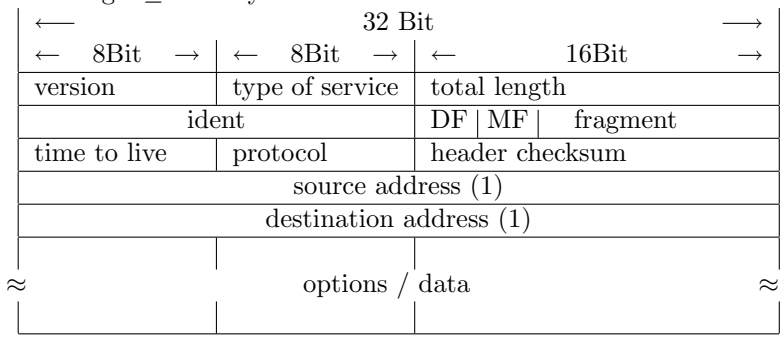
- 0000 0001 : use extended (7 bit) Numbers
- 0000 0010 : set nonstandard window size
- 0000 0011 : set nonstandard packet size
- 0000 0100 : set throughput class (75 bps bis 48 Kbps)
- 0000 1100 : Request reverse charging
- 0000 1101 : Accept reverse charging
- 0000 1111 : Select carrier (z.B. AT&T, Bell, TELENET, TYMNET,...)

Packet = N-DATA.request(indicate) – PDU (an Schicht 2):
wie oben, mit:
– Typ = PPPM SSS0 (LSB = 0 : Daten)

- PPP = Piggiback (Bestätigung des zuletzt erhaltenen Pakets, vgl. SSS)

- M = More: = 1 für Pakete einer Nachricht
- M = More: = 0 für das letzte Paket einer Nachricht
- SSS = Sequence number (aktuelle Paketnummer

Dienste im Internet (TCP/IP)
Das Internetprotokoll IP (= Internet Protocol) ist nicht genormt
TCP = Schicht 4
IP = Schicht 3
Struktur: Worte à 32 Bit = 4 Byte
Header: 5 Worte = 20 Byte
Gesamtlänge: $\leq$ 64K Byte



DF: Don't fragment
MF: More Fragments
time to live (TTL): Anzahl der Knoten, die durchlaufen werden können, bevor das Paket verworfen wird.
(1): im Format: aaa.bbb.ccc.ddd (jeweils von 0 bis 256)
Protocol Numbers

Decimal	Keyword	Protocol
0		Reserved
1	ICMP	Internet Control Message
2	IGMP	Internet Group Management
3	GGP	Gateway-to-Gateway
4		Unassigned
5	ST	Stream
6	TCP	Transmission Control
7	UCL	University College London
8	EGP	Exterior Gateway Protocol
9	IGP	any private interior gateway
10	BR-MON	BBN RCC Monitoring

(RFC-1060 Assigned Numbers March 1990)

3.2.4 Die Transportschicht (Transport layer)

Zweck: Verbindung von Teilnehmern in Endsystemen, Abschirmung der Anwendung vom konkreten Netzwerktyp, Entkopplung vom Netz.
"Die Transportschicht sorgt dafür, daß die Teilnehmer einander hören, das bedeutet aber nicht unbedingt, daß sie einander auch verstehen!"1
Dienste:
T-CONNECT.** (Bestätigter Verbindungsaufbau)
T-DISCONNECT.** (Bestätigter Verbindungsabbau)
T-DATA.* (unbestätigt, verbindungsorientiert oder verbindungslos)
T-ERROR.indicate
T-QUALITY.request (Qualitätsanforderung)
{.** bedeutet bestätigte Dienste: request, indicate, response, confirm}
{.* bedeutet unbestätigte Dienste: request, indicate}
Instanzen: Software: Netztreiber, z.B.:Net BIOS; Daemon-Prozesse (unter UNIX)
Dienstzugangspunkte, SAPs: Aufrufe von Routinen oder Prozessen, z.B.: SVC (= Super Visor Call), INT
Dienstdateneinheiten, SDUs: Dateien, Dokumente, Pakete
Protokolldateneinheiten, PDUs: Nachrichten
Funktionen:
a) Vermittlung zwischen Teilnehmern (CONNECT und DISCONNECT)
b) Umsetzung von Teilnehmeradressen (TSAPs) in Netzwerkadressen (NSAPs)
c) Fehlerfreie Datenübertragung
d) Multiplexen von Teilnehmerverbindungen
e) Splitten auf mehrere Systemverbindungen (Schicht-N-Maschinen)
f) Segmentieren zu großer Dateien (1 SDU → n PDUs)
g) Flußregelung
h) Wiederanlauf, Aufsetzen nach Störungen (N-RESET.ind)

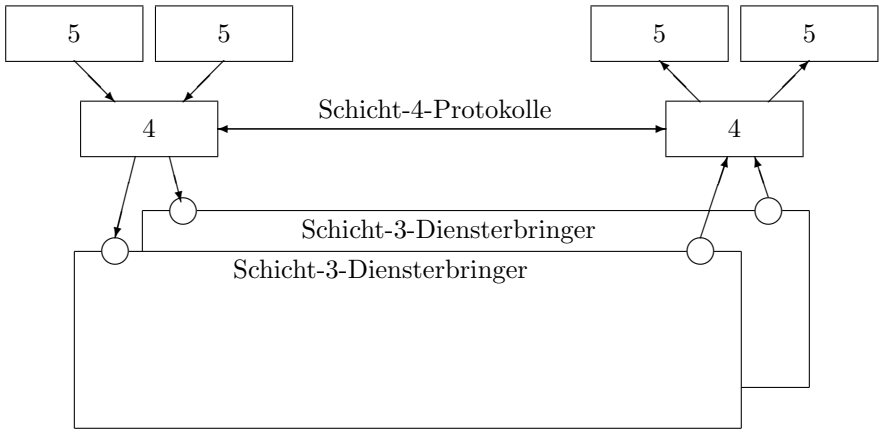


Bild n3p43

Klassen und Funktionen:

Class 0	simple class, Einfachklasse	Vermittlung, Datenübertragung,
TP0	für Typ-A-Netze	Segmentieren
Class 1	basic error recovery class,	= Class 0
TP1	Fehlerbehebungsklasse	+ Fehlerbehandlung (Wiederaufsetzen,
	für Typ-B-Netze	Flußregelung)
Class 2	multiplexing class,	= Class 0
TP2	Multiplexklasse	+ Multiplexen
	für Typ-A-Netze	+ Routing (Wegewahl)
Class 3	error recovery and multiple-	= Class 0
TP3	xing class, Fehlerbehebungs-	+ Multiplexen
	und Multiplexklasse	+ Routing (Wegewahl)
	für Typ-B-Netze	+ Fehlerbehandlung (Flußregelung)
		≈ Class 1 ∪ Class 2
Class 4	error detection and recove-	= Class 0
TP4	ry class, Fehlererkennungs-	+ Fehlererkennung + Fehlerbehebung
	und -behebungsklasse	+ Zeitschrankenüberwachung
	für Typ-C-Netze	+ Splitting

Protokolle:
- X.214 = IS 8072 OSI Transport-services
- X.224 = IS 8073 OSI Transport-protocols
- TCP Transmission Control Protocol (im Internet, verbindungsorientiert)
- UDP User Datagram Protocol (im Internet, verbindungslos)
- EHKP-4 Einheitliche höhere Kommunikations-Protokolle der Schicht 4 (vom Bundesministerium des Inneren)
- T.70 Telegraphendienste

Beispiele:
a) Internet Datagramme: UDP-PDU

⟵ 32 Bit ⟶	
IP-Senderadresse (0)	
IP-Empfängeradresse (0)	
leer | Protokoll (1)	UDP-Paketlänge (0)
Sender-Port (2)	Empfänger-Port (2)
Länge	Prüfsumme
Daten (Oktetts)	

. .
0: header
wird von der IP-Instanz abgetrennt und in den TCP-Header umgesetzt

1: Protokolltyp = 17. für UDP

2: Portadressen = SSAPs

b) IP-Sockets
IP-Socket = IP-Adresse + IP-Port

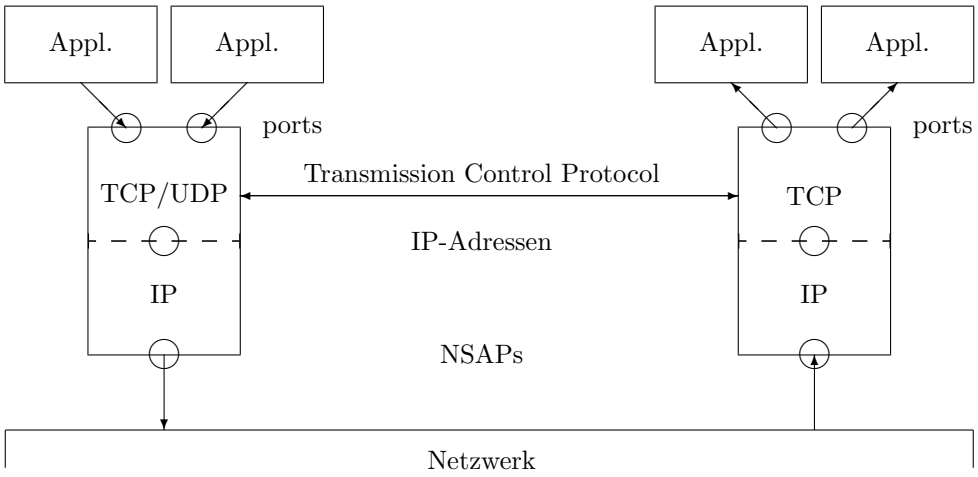


Bild n3p44

c) Internet-Pakete: TCP-PDU

⟵ 32 Bit ⟶	
Sender-Port	Empfänger-Port
Sequenznummer	
Quittungsnummer (piggyback)	
(1) | (2) | (3)	Fenstergröße
Prüfsumme	Urgent pointer
options	000
Daten (Oktetts)	

. .1: header length

2: leer

3: TCP-Flags

bit	Flag	Funktion
10	URG	use urgent data pointer
11	ACK	acknowledge → .confirm
12	EOM	End of Message
13	RST	RESET.ind
14	SYN	CONNECT.req
15	FIN	DISCONNECT.req

d) Die TCP-Protokollmaschine
(als Finite State Machine nach RFC 761)

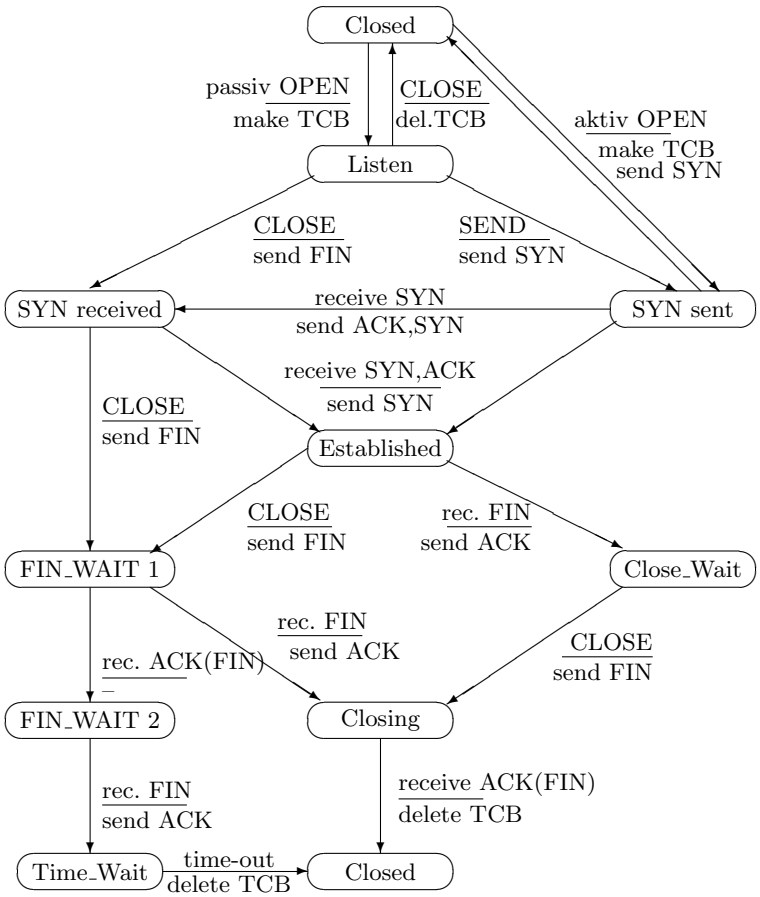


Bild n3p45

Function entries (SAPs to higher layer - Application)
OPEN (local port, foreign socket, active/passive, [, buffer size] [, timeout] [, precedence] [, security/compartment]) → local connection name
SEND (local connection name, buffer address, byte count, EOL flag, URGENT flag [, timeout])
RECEIVE (local connection name, buffer address, byte count)
CLOSE (local connection name)
ABORT (local connection name)
STATUS (local connection name)
This is an implementation dependent user command and could be excluded without adverse effect. Information returned would typically come from the TCB associated with the connection.

3.2.5 Die Kommunikationssteuerungsschicht (Sessions layer)

Zweck: Umgangsregeln für Netzteilnehmer
Teilnehmer:
Programme, Prozesse oder Tasks mit Instanzen der Schichten 5, 6 und 7 Internet-Anwendungen bilden jeweils ein unstrukturiertes Ganzes.

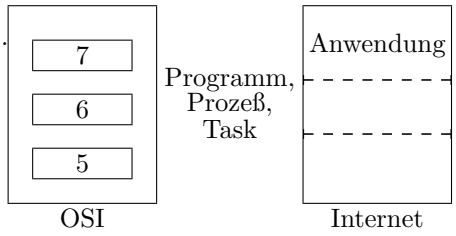


Bild n3p46

Dienste:
S-CONNECT.** (Bestätigter Verbindungsaufbau)
S-DISCONNECT.** (Bestätigter Verbindungsabbau)
S-DATA.* (Unbestätigte Datenübertragung, CONS oder CLNS)
S-SYNCH.** (Kommunikationssteuerung)
{.** bedeutet bestätigte Dienste: request, indicate, response, confirm}
{.* bedeutet unbestätigte Dienste: request, indicate}

Instanzen: Softwaremodule (Unterprogramme, Routinen) aus Library eingebunden

Dienstzugangspunkte, SAPs: Unterprogrammadressen

Dienstdateneinheiten, SDUs: Parameter (Zeiger auf Dateien)

Protokolldateneinheiten, PDUs: Dateien

Funktionen:
a) Auf- / Abbau von Verbindungen, insbesondere gesicherter Abbau
b) Datenübertragung incl. Fehlermeldung ohne Fehlerbehandlung
T-ERROR.ind → S-DATA.ind
c) Flußkontrolle
d) Synchronisation, Kommunikationssteuerung, Dialogverwaltung

Protokolle: Protokoll-Klassen:

BCS (basic combined subset)
- Vergabe von Senderechten (Token), immer nur ein Teilnehmer sendet,Halbduplex
- keine Synchronisation

BSS (basic synchronous subset)
- BCS plus Vorwärts- u. Rückwärtssynchronisation

BAS (basic activity subset)
- BSS plus Strukturierung von Sitzungen in Aktivitäten

Standards und Normen (Auswahl):
X.215 = IS 8326: Session-services
X.225 = IS 8327: Session-protocols (beide entsprechen der BAS-Klasse)
EHKP-5: Einheitliche Höhere Kommunikationsprotokolle des BMI (BSS-Klasse)
T.62: Teletex und FAX Prozeduren (BCS-Klasse)

Sitzungen (Sessions)
bilden Einheiten, die nicht unterbrochen werden können, z.B.
- Remote login (Telnet): eine Sitzung an einem entfernten Rechner
- File transfer (FTP, FTAM): Übertragung einer Datei oder Dateigruppe
- Remote Procedure Call (RPC): Ausführung einer Prozedur auf einem entfernten Rechner (Fernverarbeitung, JTM)

a) Eins-zu-Eins-Mapping: Eine Session entspricht einer Connection

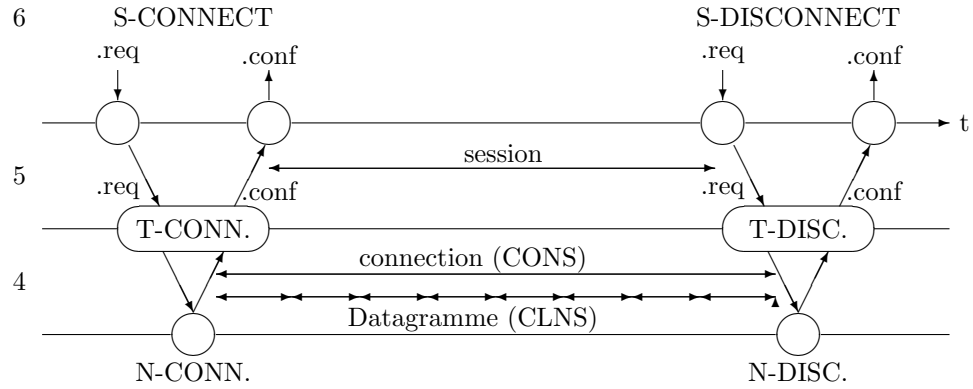


Bild n3p48

Während einer Sitzung werden beliebig viele Daten übertragen. Die zugehörigen Dienstprimitiven (S-DATA.*), die in entsprechenden Dienstprimitiven (T-DATA.*) der Transportschicht umgesetzt werden, sind hier nicht gezeigt.
Falls die Transportschicht verbindungslos arbeitet (CLNS), wird für jedes S-DATA (bzw. T-DATA) ein Datagramm versandt.

Der Verbindungsabbau (S-DISCONNECT) wird in der Regel beidseitig ausgehandelt (S-RELEASE), er kann vom Kommunikationspartner auch verweigert werden, dazu muß er in der Bestätigung eine geeignete Flagge setzen. Auch ein Abbruch ist möglich (S-U_ABORT).

b) Eins-zu-N-Mapping: Eine Session entspricht N Connections

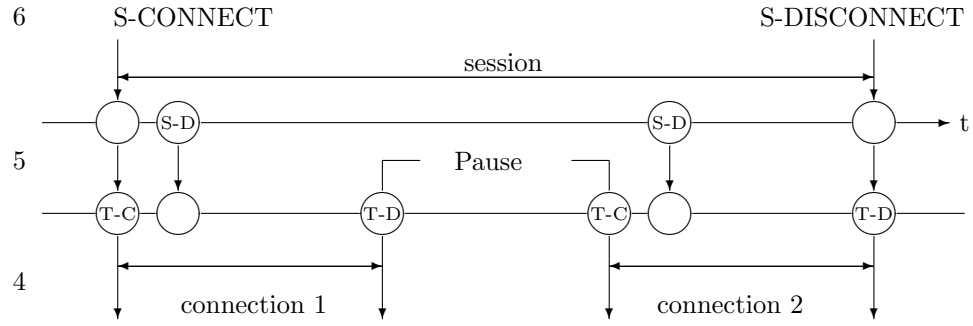


Bild n3p49

T-C = T-CONNECT T-D = T-DISCONNECT S-D = S-DATA

Die Bestätigungen bei Verbindungsaufbau (CONNECT) und -abbau (DISCONNECT) sind hier zur Vereinfachung weggelassen. Im Extremfall wird für jedes S-DATA eine neue Verbindung aufgebaut (T-C .. T-D) oder ein Datagramm geschickt bzw empfangen. Während der Pause wird die nicht benötigte Leitung durch die Schicht-5-Instanz freigeben z.B. um Kosten zu sparen, ohne daß der Schicht-6-Benutzer davon etwas bemerkt.

c) M-zu-Eins-Mapping: M Sitzungen über eine Connection (Multisession Connections)

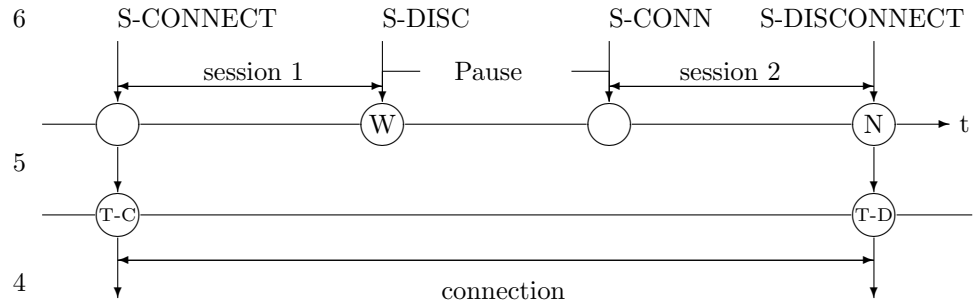


Bild n3p50

T-C = T-CONNECT T-D = T-DISCONNECT

Mehrere Sitzungen (z.B. Übertragungen von Dateien) werden nacheinander über ein und dieselbe Verbindung abgewickelt. Die Dienstprimitiven (S-DATA.* und T-DATA.*) sind hier nicht gezeigt. Zur Signalisierung an die T-Schicht-Instanz müssen die S-DISCONNECT.req. Primitiven noch zusätzliche Parameter enthalten (**W**ait, **N**o-wait).

d) Strukturierung in Aktivitäten

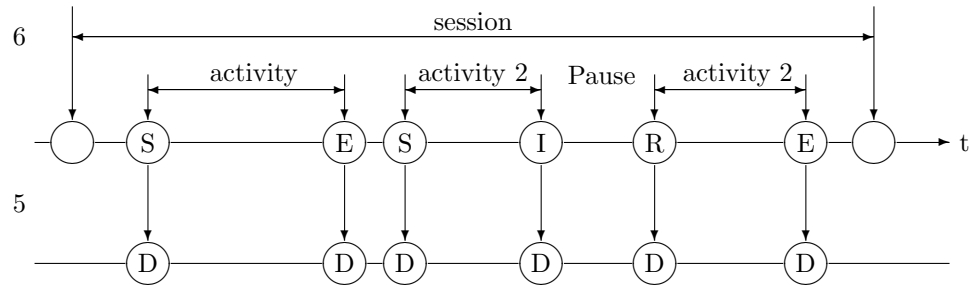


Bild n3p51

S = S-ACTIVITY_START E = S-ACTIVITY_END
I = S-ACTIVITY_INTERRUPT R = S-ACTIVITY_RESUME
D = T-DATA

Activity: ein in sich abgeschlossener Vorgang, z.B.
- Übertragung einer Datei als Ganzes
- eine Datenbank-Transaktion

Die Markierungen für Anfang und Ende einer Aktivität werden als Daten in eigenen Paketen übertragen.

Synchronisation: Vorsorge gegen Verlust von Daten
- Vorwärtssynchronisation (unbestätigt):
  Angabe von Wiederaufsetzpunkten für Rückwärtssynchronisation.
- Rückwärtssynchronistaion (bestätigt):
  Anforderung, an einen Wiederaufsetzpunkt zurückzugehen
- Haupt- und Nebensynchronisation: Wiederaufsetzpunkte verschiedener Ordnung:
  - Es gibt immer einen Hauptsynchronisationspunkt, zu dem zurückgesetzt werden kann, aber nicht weiter zurück (z.B. Anfang einer Datei)
  - Es kann mehrere Nebensynchronisationspunkte geben, zu denen zurückgesetzt werden kann (z.B. die Blöcke einer Datei)
  - Die Aktivitäten zwischen 2 Synchronisationspunkten bilden eine Dialogeinheit

a) Synchronisationspunkte (major, minor)
z.B.: Blocksicherung bei Datentransfer
major_synch ist bestätigt, nur der letzte kann benutzt werden
minor_synch ist unbestätigt, alle bis zum letzten major_synch können benutzt werden.

b) Resynchronisation (Wiederaufsetzen auf einen Synchronisationspunkt)
- Bei Datenverlust oder -fehlern
- Restart nach einem RESET.indicate,

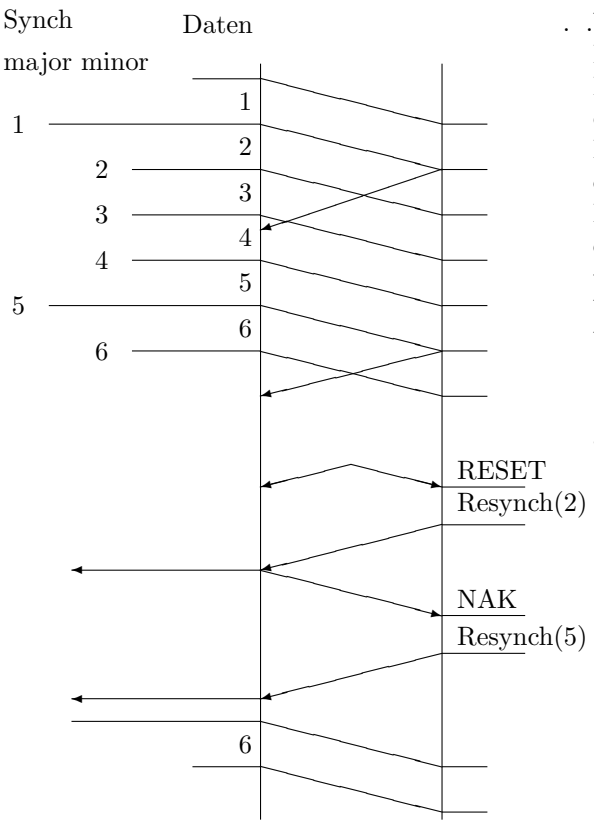


Bild n3p52

. .Wiederaufsetzen nach RESET.ind durch das Netz. Die Resynchronisation kann auf den letzten Hauptsynchronisationspunkt (5) oder den letzten Nebensynchronisationspunkt (6) erfolgen. Da (5) bestätigt wurde, ist dieses Wiederaufsetzen sicherer. Der Versuch auf einen früheren (Neben)Synchronisationspunkt (Resynch(2)) aufzusetzen, wird abgewiesen (NACK).

3.2.6 Die Darstellungsschicht (Presentation layer)

Zweck: Bereitstellung von Sprachmitteln
Dienste:
P-CONNECT.** → S-CONNECT.**
P-DISCONNECT.** → S-DISCONNECT.**
P-SYNCH_*.* → S-SYNCH_*.*
Diese Dienste werden nur zur S-Schicht durchgereicht
P-DATA.* (mit Umformung, Dolmetschen in gemeinsame Sprache)
Zur Bestätigung kann P-SYNCH benutzt werden
P-CONTEXT_SELECT.** (gemeinsame Sprache auswählen)
P-CONTEXT_DEFINE.** (gemeinsame Sprache definieren)
{.** bedeutet bestätigte Dienste: request, indicate, response, confirm}
{.* bedeutet unbestätigte Dienste: request, indicate}

Instanzen: Software, Tabellen (z.B.: CONTEXT_DEFINE baut Tabellen auf)

Dienstzugangspunkte, SAPs:
Funktionsaufrufe (von spezieller Software aus Bibliotheken)

Dienstdateneinheiten, SDUs: Dokumente

Protokolldateneinheiten, PDUs: Dokumente (aber in anderer Form)

Funktionen:
a) Umformen von Dokumenten (Dateien)
b) Auswahl aus Tabelle
c) Erstellung von Tabellen
Voreinstellung ist transparente byteweise Übertragung von Zeichen aus ASCII oder EBCDIC (Default context)

Protokolle:
Normen und Standards (Auswahl)
X.216 = ISO 88 22: P-Services
X.226 = ISO 88 23: P-Protocol
X.409: ASN 1 = Abstract Syntax Notation No. 1
T.50: International Alphabet No. 5 (IA5)
T.61: Basic Teletex Charecter Set (Zeichensatz für Teletex)
ISO 6937: 8-bit Zeichen
ISO 8859: 8-bit Graphikzeichen
ISO 7942: Graphic Kernel System
EHKP-6: Einheitliche Höhere Kommunikationsprotokolle des BMI

Datendarstellung

a) Texte: Zeichendarstellung in ASCII, EBCDIC, IBM-Code-1, 2, 3, IA5, etc.

b) Bytes: Anordnung der Bytes innerhalb eines Wort

	little endian				Wort/Byte	big endian			
	MSB			LSB	Adresse	MSB			LSB
Text	_	s	a	D	0/0	D	a	s	_
	_	t	s	i	1/3	i	s	t	_
	_	n	i	e	2/7	e	i	n	_
	t	x	e	T	3/11	T	e	x	t
Short	00	01		00 10		10 00		01	00
Integer			00 00 00 01					01 00 00 00	
Real	exp	man		tis se		se	tis	man	exp
Byte:	3	2	1	0		0	1	2	3
			Low Byte			Low Byte			
	DEC, Intel					IBM, Motorola			

Bei einer korrekten Übertragung von Texten werden andere Daten (Integer, Real, ...) verändert, bzw. umgekehrt.

c) Integer: Breite, Negativum (1er-, 2er-Komplement, Betrags-, Offset-Darstellung)

d) Real: Breite, Position und Darstellung von Exponent und Mantisse

e) Graphik: Vektorgraphik (GKS), Rastergraphik (Pixelanordnung), Farbdarstellung

f) Dokumente = { Text, Graphik, Numerik} & Anordnung (Layout)

g) Komprimierung

ASN.1 (Abstract Syntax Notation No. 1) X.208, X.209, X.409 (ISO 8824, 8825) Metasprache zur Darstellung von Dokumenten unter Verwendung von Oktetts (8-bit-Zeichen). Umwandlungen von der lokalen Datendarstellung in eine genormte Transferdarstellung.
Transfersyntax: TLV-Struktur (rekursive Record Struktur)
T : Typ, Tag (1 Oktett), z.T. genormt
L : Length (1 – n Oktetts)
V : Value (m Oktetts) = Wert oder TLV
Basic Encoding Rules (X.208)

Tag-Typen (X.209)

Typ	MSBs	Hexa
Universal	00	0 - 3F
Application	01	40 - 7F
Context-spec.	10	80 - BF
Private	11	C0 - FF

. .Universal-Tags (X.208)

Nummer (Bit 6 - 1)	Datentyp
1	Boolean type
2	Integer type
3	Bitstring type
4	Octetstring type
5	Null type
6	Object identifier type
7	Object descriptor type
8	External type
9	Real type
A	Enumerated type
B → F	Reserved for future use
10	Sequence and Sequence-of types
11	Set and Set-of types
12 → 16	Character string types
17, 18	Time types
19 → 1B	Character string types
1C → 3F	Reserved for future use

Beispiel (nach X.409): Personendaten
Name: John P Smith
Title: Director
Birthday: 11. July 1936

Tag	TLV Oktetts (in sedezimaler Darstellung) bzw Strings						
	60$	81 85$	V				
Personendaten:	T	L	61$	10$	V		
Name:			T	L	16$	04	John
IA5					T	L	V
					16$	01	P
					T	L	V
					16$	05	Smith
					T	L	V
			A0$	0A$	V		
Title:			T	L	16$	08	Director
IA5					T	L	V
...							

Verschlüsselung

a) Substitutionsverschlüsselung

Jedes Zeichen aus einem Entschlüsselungsalphabet wird in ein Zeichen aus einem Verschlüsselungsalphabet umgesetzt.

Ein Erraten (Knacken) der Verschlüsselung ist leicht möglich, da die (sprachspezifischen) Buchstabenhäufigkeiten erhalten bleiben.

Bei einer monoalphabethischen Substitutionsverschlüsselung ist
Verschlüsselungsalphabet = Entschlüsselungsalphabet

Die erste monoalphabethische Substitutionsverschlüsselung wurde von Julius Cäsar (Cesar Cipher) angewandt, bei der jedes Zeichen durch seinen um 3 versetzten Nachfolger ersetzt wurde:

a → d
b → e

Die (beliebige) Zahl (3), um die verschoben wird, bildet hier den Schlüssel.

b) Transpositionverschlüsselung

Die zu verschlüsselnden Zeichen werden in ein Raster angeordent, das nach vorgegebenen Regeln umgeordnet wird; das Verfahren ist zunächst monoalphabetisch.

Je nach Art der Umordnung ist eine solche Verschlüsselung mehr oder minder leicht zu durchbrechen.

Beispiele:

- Rückwärtsschreiben (den gesamten Text oder block- bzw. zeilenweise) Das ist ein Text → txeT nie tsi saD
- Vertauschen von Spalten einer 2-dimensionalen Anordnung

1234	1324
Das_	Dsa_
ist_	its_
ein_	eni_
Text	Txet

  Das ist ein Text → Dsa its eni Txet
- Vertauschen von Zeichen in 2- oder mehrdimensionalen Anordnungen z.B. im Rösselsprung.

c) Standardverfahren:

DES (Data Enscription Standard) verwendet ein symmetrisches Verfahren, bei dem mit dem gleichen Schlüssel ver- und entschlüsselt werden kann. Der Schlüssel stellt die Parameter für einen vorgegebenen Algorithmus bereit.

RSA (nach den Autoren Rivest, Shamir, Adelman) verwendet unterschiedliche Schlüssel zum Ver- bzw. Entschlüsseln (unsymmetrische Verschlüsselung). Dabei wird von jedem Teilnehmer der Schlüssel D zur Verschlüsselung öffentlich bekannt gegeben, während der Schlüssel E zur Entschlüsselung geheim bleibt. Notwendig ist hierbei, daß E nicht aus D abgeleitet werden kann; dies wird durch Verwendung großer Primzahlen erreicht.

3.2.7 Die Anwendungsschicht (application layer)

Zweck: Koordinierung von verteilten Anwendungen
(DDP = Distributed Data Processing)
Dienste: anwendungsspezifische Schnittstellen für die Anwendungsprozesse (Anwender, technische Prozesse)
z.B. beim Virtual Terminal:
A-CONNECT = login
A-DISCONNECT = logout

Instanzen: Software, Programme, Module

Dienstzugangspunkte, SAPs: anwendungsspezifisch
API (= Application Process Interface)

Dienstdateneinheiten, SDUs: Benutzerein- und -ausgaben

Protokolldateneinheiten, PDUs: Dokumente

Funktionen:
- Verarbeitung der Eingaben
- Erstellung der Ausgaben
- Verknüpfen und Zuordnen von lokalen Daten und übertragenen Informationen
- Informationsverarbeitung

Protokolle:
Standards (Auswahl):

X.400	Message Handling System (MHS) [E-MAIL]
F.300	Videotext
ISO 85 71	File Transfer Access + Management (FTAM) [Datenverwaltung]
ISO 86 13	Manufacturers Application Protocol (MAP + TOP) [CIM]
ISO 90 40/41	Virtual Terminal

Anwendungen (Auswahl)
- Internet-Anwendungen (Telnet, FTP, SMTP E-Mail, ...)
- ISO-Standards (VT, FTAM, MHS E-Mail,..)
- TOP (Technical and Office Protocol aus der Luftfahrt)
- MAP (Manufacturing Automation Protocol aus der Automobilindustrie)

ISO	Internet
VT (Virtual Terminal)	Telnet
FTAM (File Transfer, Access and Management)	FTP (File Transfer Protocol)
MHS (Message Handling System, X.400) ≡ MOTIS (Message Oriented Text Interchange System)	SMPT (Simple Mail Transfer Protocol)
The Direcory (X.500) (Informationsdienst)	DNS (Domain Name System) NIS (Network Information System) früher: Yellowpages
JTM (Job Transfer and Management)	—
—	NNTP (Network News Transfer Protocol

Beispiel:
Elektronische Post nach X.400

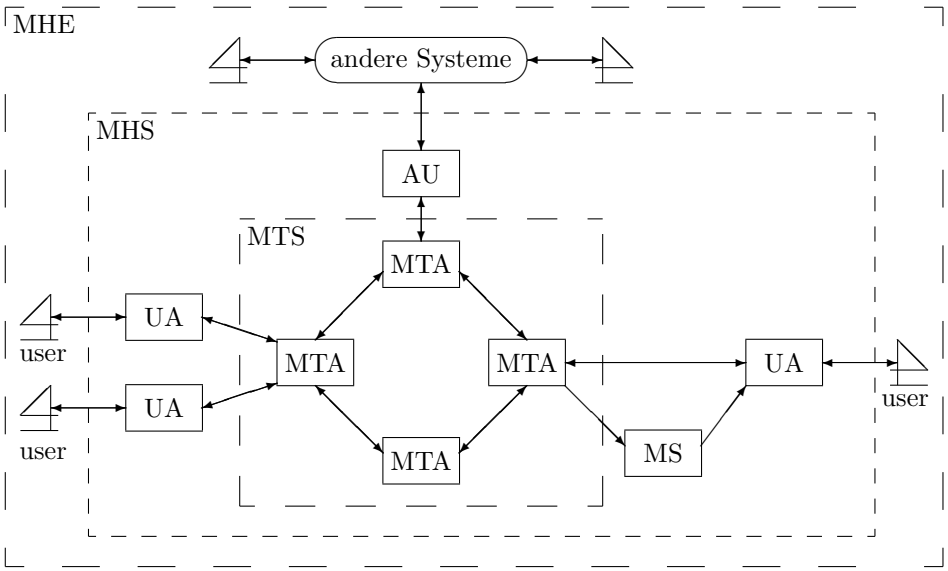


Bild n3p55

MHE = Message Handling Environment
MHS = Message Handling System
MTS = Message Transfer System
UA = User Agent
(Client, ein Programm als persönlicher Agent eines Anwenders)
MTA = Message Transfer Agent
(sind miteinander vernetzt und stellen einen verteilten Server dar)
AU = Access Units (Gateways, verbinden zu anderen Systemen)
MS = Message Storage (Mailbox des Anwenders)

Architektur (Dienste und Protokolle):

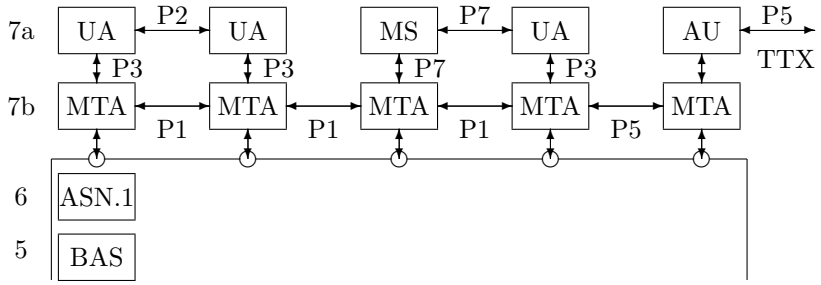


Bild n3p56

Protokolle:
P1 (X.411): MTS Transfer Protocol, zwischen MTAs
P2 (X.420): InterPersonal Messaging Protocol (IPM), zwischen UAs
P3 (X.411): MTS Access Protocol, Zugang zu den MTS-Diensten
P5 (X.430): Teletex Access Protocol, Zugang zu Teletex
P7 (X.419): Message Store Access Protocol, Zugang zur Mailbox

E-mail Adressen

key: domaine	Beispiel
C : country	ge = Germany
A : administrative management domain (admd)	dbp = Deutsche Bundespost
P : private management domain (prmd)	fh-frankfurt = Fachhochschule Ffm
O : organisation	fb06 = Fachbereich MND
OU: organisation unit	unit : BCN
S : surname (Nachname)	user : Jacobson
G : given name (Vorname)	vorn : Erik

Syntax
X.400 (MOTIS):
<[G=vorn;]S=user;[OU=unit;O=organisation;]P=prmd;A=admd;C=country>
konkret: <G=Erik;S=Jacobson;OU=BCN;O=fb06;P=fh-frankfurt;A=d400;C=de>

SMTP (Internet):
user@host.domaine
Die Umwandlung von X.400– in Internet-Adressen ist möglich, aber nicht eindeutig.
Erik.Jacobson@BCN.fb06d.fh-frankfurt.d400.de

Jede X.400–Adresse ist hierarchisch aufgebaut:

Country							
ADMDs							
PRMDs							
user							

Die Anwendungen im Internet:

<table>
<tr><td>7
6
5</td><td>Telnet Protocol (RFC 652)</td><td>FTP File Transfer Protocol (RFC 959)</td><td>SMTP Simple Mail Transfer Protocol</td><td>NNTP Network News Transfer Protocol</td><td>TFTP Trivial FTP (local) (RFC 783)</td><td>DNS Domain Name System</td></tr>
<tr><td>4</td><td colspan="3">TCP: Transmission Control Prototocol TCP (CONS)</td><td colspan="3">User Datagram Protocol UDP</td></tr>
<tr><td>3</td><td colspan="6">IP: Internet Protocol IP (CLNS)</td></tr>
<tr><td>2
1</td><td colspan="6">V.24, V.25, X.200 (ISDN), ISO 8802 (LAN)</td></tr>
</table>

Bild n3p58

3.3 Transitsysteme

Zwischenknoten (Intermediate Systems).
Aufgaben:
- Protokollumsetzung
- Wegewahl (Routing)
- Zwischenspeicherung (Store & Forward)

Im Gegensatz zu Endsystemen werden die obersten Instanzen durch Automaten (mit deterministischem Verhalten) repräsentiert.

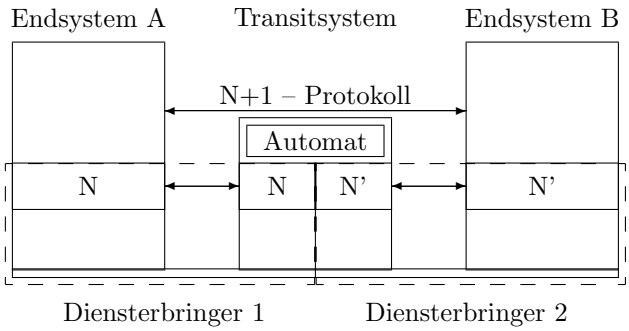


Bild n3p61

Klassen von Transitsystemen

Schicht 0: Medium
– Verstärker, Repeater (Signalregeneration)
– Schalter

Schicht 1: Bitdarstellungsschicht (Physical-Layer)
Umsetzung der Bitdarstellung
- Optokoppler (optisch/elektronisch
- Modem (Modulator/Demodulator):
  Basisband- ↔ Breitbandprotokoll
- Pegelumsetzung, z.B.
  V.28 ↔ TTL
- Codeumsetzer
  NRZ ↔ RZ
  RZ ↔ Manchester-Code

Schicht 2: Sicherungsschicht (Data-Link-Layer)
- Seriell-Parallel-Umsetzer
- Bridge (mit Prüfung der Datenkonsistenz, Prüfsummen)
- Modem mit Datenkompression (V.42, V.42 bis)

Schicht 3: Vermittlungsschicht (Network-Layer)
– Gateways: Protokollumsetzer zwischen Netzwerken mit unterschiedlichen Adreßräumen, z.B.

zwischen LAN und WAN
zwischen Token-Ring LAN und Ethernet LAN

– Router: Wegewahl in und zwischen Netzwerken (DSE, Data Switching Exchange)
– Brouter: Bridge mit Router- und Filter-Funktion (Adreßfilter)
– Hub (Nabe): Umsetzung auf andere Topologien (vgl. Ethernet)

Schicht 4: Transportschicht (Transport-Layer)
Keine Umsetzer vorgesehen, da nur zur Entkopplung von Vermittlungschicht und Teilnehmern vorgesehen.

Schicht 5: Kommunikationssteuerungsschicht (Session-Layer)
Keine Umsetzer vorgesehen, da Teilnehmer dieselben Regeln einhalten müssen!

Schicht 6: Darstellungsschicht (Presentation-Layer)
Noch keine automatischen Systeme zur Umwandlung von Datendarstellungen oder gar Sprachübersetzung vorhanden.

Nur gemeinsame Transfersyntax wird hier unterstützt.

Schicht 7: Anwendungsschicht (Application-Layer)
Umsetzer auf der Anwendungsschicht:
– Agenten.

z.B. Message Transfer Agent (MTA),
Informationsverarbeitung: Speichern, Sortieren, Weiterleiten, Vervielfältigen)

– Gateways (Protokoll-Umsetzer)

Access Units (AU) zwischen verschiedenen E-mail-Systemen, X.400 und Internet SMTP,
Gateways zwischenn NNTP und Bitnet

3.4 Schichten und Systeme

Die möglichen Wege von Daten durch die Schichten (Splitting und Multiplexing) und durch verschiedene Systeme:

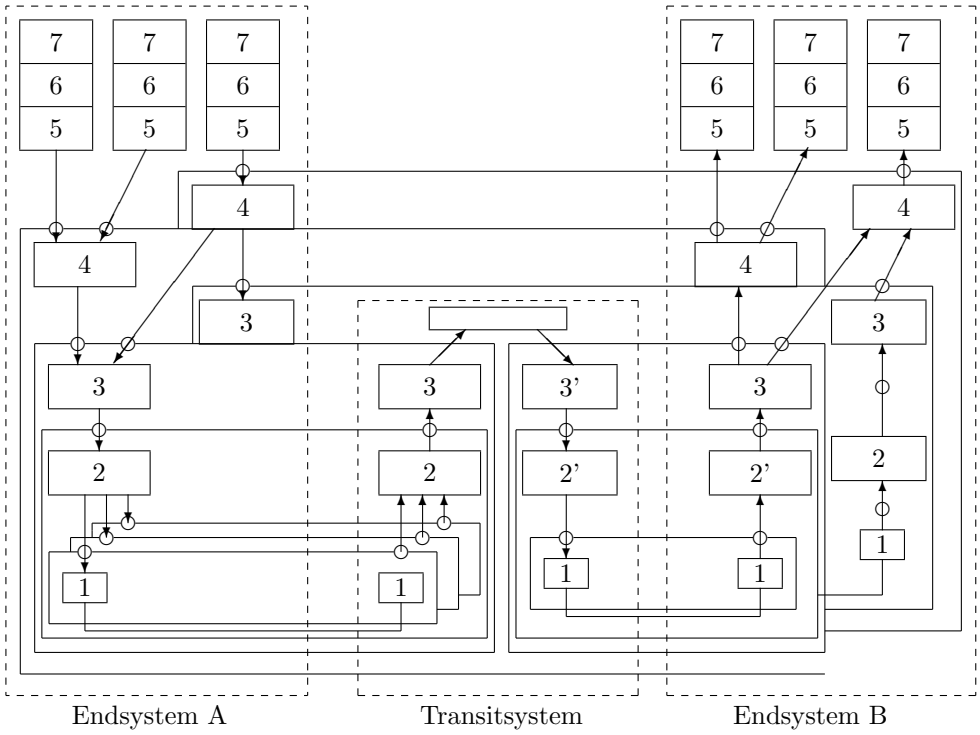


Bild n3p65

Funktionen:
Splitten in Schicht 2 und 4
Multiplexen in Schicht 3 und 4
Protokollumsetzung in Schicht 1, 2 und 3

.

Kapitel 4

Lokale Netze (LAN)

Lokale Netze sind privat (Jeder kann in seinem Netz machen was er will.)
Dieselben Verfahren werden auch in (öffentlichen) Regionalen Netzen (MAN, Metropolitan Area Networks) verwendet.

4.1 Normen und Standards

Die Schichten 0, 1 und 2 von IEEE und ISO (IEEE 802.X ≡ ISO 8802.X)

ISO-OSI ISO 8802-1
IEEE 802.1

2 Data Link Layer	2b LLC	ISO 8802-2 IEEE 802.2				
	2a MAC	Ethernet	Token Ring	Token Bus	Slotted Ring	FDDI
1 Physical Layer	1a Phys	CSMA/CD Bus				
	1b	ISO 8802-3 IEEE 802.3	ISO 8802-5 IEEE 802.5	ISO 8802-4 IEEE 802.4	ISO 8802-7 IEEE 802.7	ISO 8802-6 IEEE 802.6

LLC = Logical Link Control
MAC = Medium Access Control
FDDI = Fiber Distributed Data Interchange

Bild n4p01

4.2 Dienste und Protokolle der LLC-Schicht

LLC-Dienste (Schicht 2a) für die Schicht 3 (Vermittlungsschicht, z.B, IP)

1. Datagrammdienste (CLNS):

DL-UNIT_DATA.*(source-address, destination-address,
data,
priority)

2. Verbindungsorientierte Dienste (CONS) in Vorbereitung:
- DL-CONNECT.**
- DL-DISCONNECT.**
- DL-DATA.*
- RESET.*

{.** bedeutet bestätigte Dienste: request, indicate, response, confirm}
{.* bedeutet unbestätigte Dienste: request, indicate}

4.3 Dienste und Protokolle der MAC-Schicht

Einheitlich für alle Varianten als Dienste für die LLC-Instanzen.
Zur Zeit nur Datagrammdienste (CLNS)

MAC-UNIT_DATA.*(source-address, destination-address,
data,
status,
priority,
class)

4.4 CSMA/CD Ethernet-Bus (IEEE 802.3)

Carrier Sense: Medium dauernd abhören
Multiple Access: Jeder darf jederzeit auf das Medium zugreifen, falls es frei ist
Collision Detection: Kollisionserkennung und -behandlung, falls 2 Stationen gleichzeitig zu senden beginnen.

Varianten:
- 10 BASE 5: Yellow Cable (Thick wire), UHF-Kabel
- 10 BASE 2: Cheapernet (Thinwire), BNC-Kabel
- 10 BASE T: Twisted-Pair Ethernet (10 MHz)
- 10 BASE T: Twisted-Pair Ethernet (1 MHz)
- 10 BASE F: Lichtwellenleiter (vgl. FDDI, Fiber Distributed Data Interface)
- X BASE Y:
  X = Bitrate in MHZ (Mb/s), Y = Segmentgröße (in 100 m) oder Protokoll

Mischtopologie (baumförmig angeordnete Busse und Sterne):

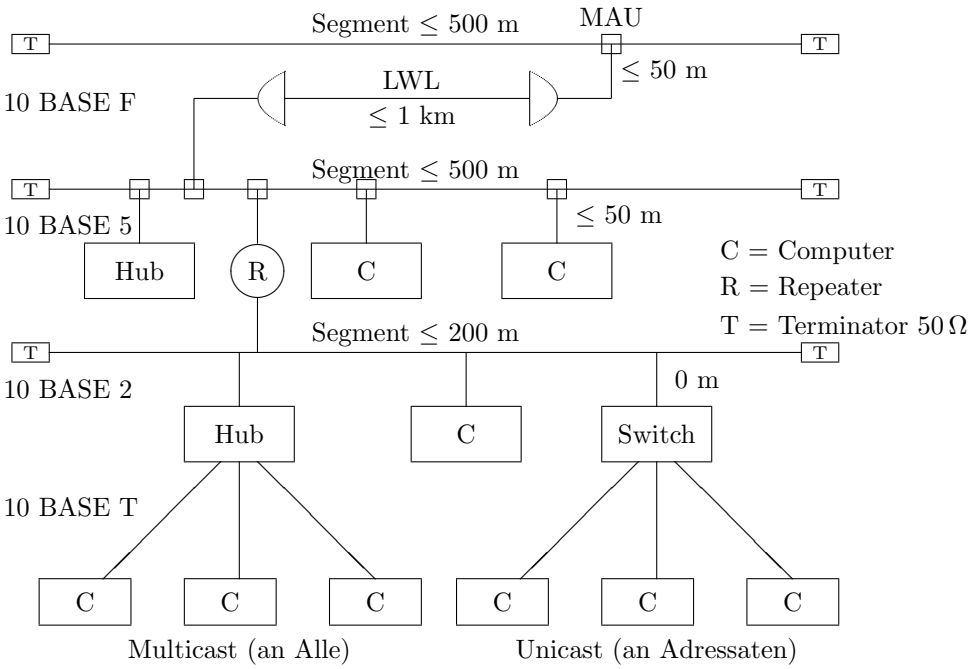


Bild n4p04

Regeln für die Ethernet-Segmente:

- Zwischen 2 Endknoten dürfen nur 2 Repeater (mit Signalverstärkung) liegen, sonst muß 1 Bridge (mit Datenaufbereitung) eingefügt werden.
- Ein Lichtwellenleiter (LWL) wird als 1 Repeater angesehen, maximale Länge ≤ 1 km.
- Transceiver-Cabel (AUI) bei 10 BASE 5 können eine maximale Länge von 50 m besitzen
- Die Mindestabstände von Stationen, bzw. Leitungszugriffspunkten (TAP = Trunc access point)
  - bei 10 BASE 5: 2,5 m
  - bei 10 BASE 2: 0.5 m
- maximale Anzahl der Knoten (Rechner) pro Segment:
  - bei 10 BASE 5: 100
  - bei 10 BASE 2: 30
  - bei 10 BASE T: 1

4.4.1 Zugriff zum Medium (Medium Access)

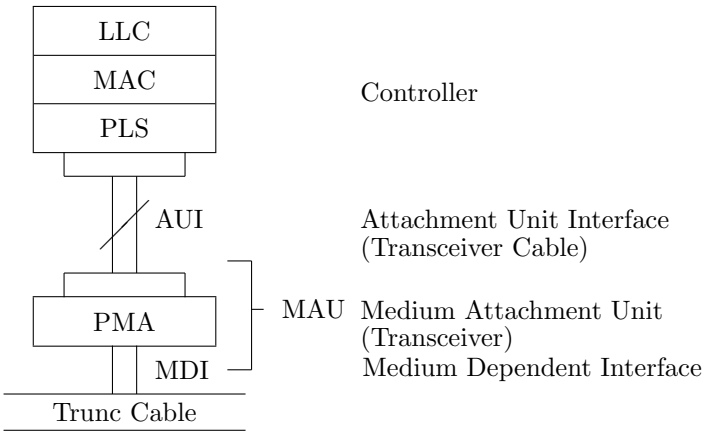


Bild n4p02

PMA: Physical Medium Attachment PLS: Physical Link System
MDI: Medium Dependent Interface AUI: Attachment Unit Interface
MAU: Medium Attachment Unit (Transceiver)

Thick-wire Ethernet (10 BASE 5)
Trunc Cable:
Yellow Cable, 50 Ω UHF- Coaxialkabel ($c^* = 0.77c_0$)
max. 500 Meter pro Segment
max. 100 Stationen pro Segment

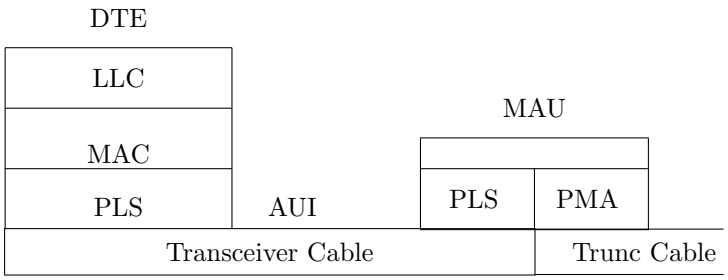


Bild n4p03

MAU: Transceiver (10 BASE 5 Medium Access Unit)

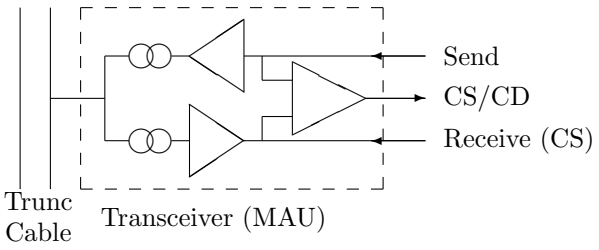


Bild n4p06

Thin-wire Ethernet (10 BASE 2)
Trunc Cable:
BNC-Coaxial Kabel (RG 58 U)($c^* = 0.65c_0$).
max. 185 200 Meter pro Segment
max. 30 Stationen pro Segment
MAU: BNC – T -Stück

Twisted-pair Ethernet (10 BASE T)
Trunc Cable:
Twisted-pair Cable, 50 Ω ($c^* = 0.60c_0$)
max. 20 Meter pro Segment
max. 1 Station pro Segment von Sternverteiler (Hub)
MAU: RJ-45 – Buchse

4.4.2 Bitdarstellung

Schicht 1:
Basisbandverfahren: Manchester Code(Bi-Phase-inverted)

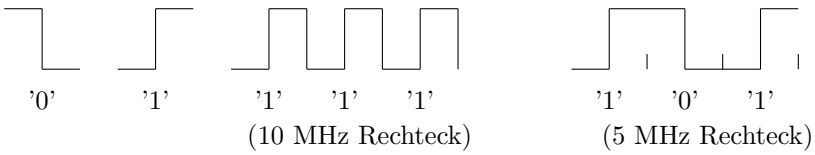


Bild n4p07

Pegel:
PLS: ± 0.9 V an 78 Ω erdfrei
PMA: unsymmetrisch ("H"= 0 V; "L"= -2.1 V (±10%) D.h. -84 mA an 25 Ω),
bei Collisionen können sich diese Pegel verdoppeln.

4.4.3 Ethernet Pakete

Paket = MAC-SDU (Frame)+ Start Frame (Präambel + Start Frame Delimiter)
serielle Oktetts

Oktetts	Feld	Inhalt
7	Preamble	7 * AA$
1	SFD	1 * AB$ = Start Frame Delimiter
2 / 6	Destination Address	
2 / 6	Source Address	
2	Length	Länge des Datenfelds
M	Data	(LLC-SDU)
	Pad	(Füll-Bytes)
4	FCS	Frame Check Sum (CRC)
	LSB MSB	transmitted left to right

Die maximale Paketgröße ist 1526 Oktetts: Präambel (7+1), Kopf (header) (14), Daten (M = 1500), CRC (4)
Die minimale Paketgröße ist 72 Oktetts; (M = 46) Oktetts mit Nutzdaten, andernfalls muß mit leeren Oktetts (Füll-Bytes, Pads) aufgefüllt werden.
Das Generatorpolynom des CRC lautet:
$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x^1 + x^0 = 4\,374\,732\,215 = 5 \cdot 7 \cdot 29 \cdot 4\,310\,081$

4.4.4 Kollisionsbehandlung

Begrenzte binär-exponentielle Zurückstellung (truncated binary exponential backoff)

Protokolle der Schicht 2a:

Mindestabstand der Pakete: 9.6μs

Standardzeitabschnitt (slot time): 51,2 $\mu s = t_R$

Festlegung der minimalen Paketlänge (72 Oktetts = 576 Bit $\equiv$ 57,6 μs)

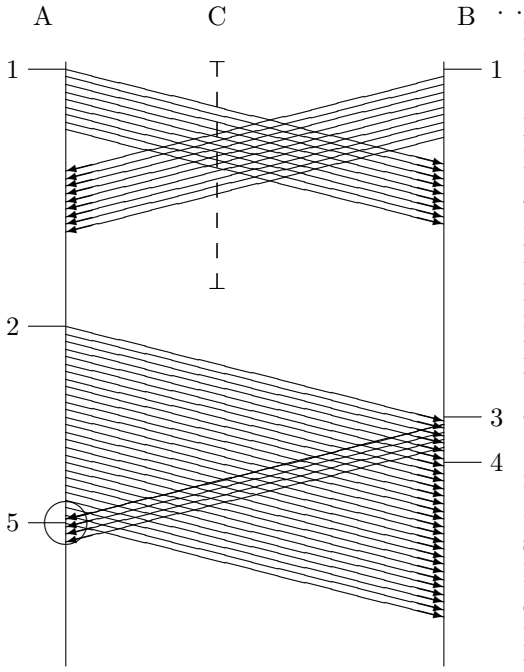


Bild n4p05

Betrachtet werden zwei Stationen A und B an den entferntesten Enden eines LAN (max. 2 Repeater dazwischen).
Im ersten Fall sind die Pakete zu kurz: Die Stationen A und B beginnen zur gleichen Zeit zu senden, da sie den Bus als frei erkennen. Ein Zwischenstation C kann eine Kollision erkennen (z.T. doppelte Pegel). Die beiden Sender merken nichts von einer Kollison.
Im schlimmsten Fall (unterer Bildteil) beginnt eine Station (B) unmittelbar vor dem Eintreffen des Signals einer anderen Station (A) zu senden und erkennt sofort eine Kollision; sie bricht nicht sofort ab sondern sendet ein Stausignal (jam signal) von minindestens 3.2 μs Länge. Die Gegenstation kann eine Kollision nur erkennen, wenn seine eigenes (minimales) Paket mindestens so lang ist, daß es bei Eintreffen des Stausignals noch gesendet wird.

Daraus ergibt sich, daß die Minimallänge eines Pakets länger sein muß als die Netzlaufzeit (round trip delay), also die Zeit t_R für den Hin- und Rücklauf eines Signals von einer Station am Ende eines Segments zur Station am anderen Ende des Segments. Diese ergibt sich aus der maximalen Größe eines Segments

$l_{max} = (2 \cdot 50m + 3 \cdot 500m + 2 \cdot 2 \cdot 50m + 2 \cdot 1000m)$

zu:

$t_{max} = 2 \cdot l_{max}/c^* = 2 \cdot 3800m/(0.77 \cdot 3 \cdot 10^8 m/s = 33\mu s < t_R$

Kollisionsbehandlung:
begrenzte binär-exponentielle Zurückstellung (truncated binary exponential backoff):
Nach der 1. Kollision warten die Partner eine Zeit $t_W = Z \cdot t_R$ wobei Z = Zufallszahl aus {0,1} ist.

Mit einer Wahrscheinlichkeit von 50% tritt eine weitere Kollision ein, je nachdem welche Werte bei A und B erzeugt wurden:

A	B	Folge
0	0	A und B senden sofort wieder → neue Kollision
1	0	A sendet zuerst
0	1	B sendet zuerst
1	1	A und B senden nach der Wartezeit gleichzeitig wieder → neue Kollision

Nach der n-ten Kollision warten die Partner die Zeit $t_W = Z \cdot t_R$ mit $Z \in \{0, 2^n - 1\}$
Dabei vermindert sich die Wahrscheinlichkeit für eine erneute Kollision.
Nach der 10. Kollision wird der Wert n = 10 für die Zurückstellung beibehalten.
Nach 16 Kollisionen wird mit einer Fehlermeldung an die LLC-Schicht abgebrochen.
Im schlimmsten Fall beträgt die Wartezeit einer Station:
$t_{max} \approx (2 + 6) \cdot 2^{10} \cdot t_R = 419430,4\mu s = 0.419s$

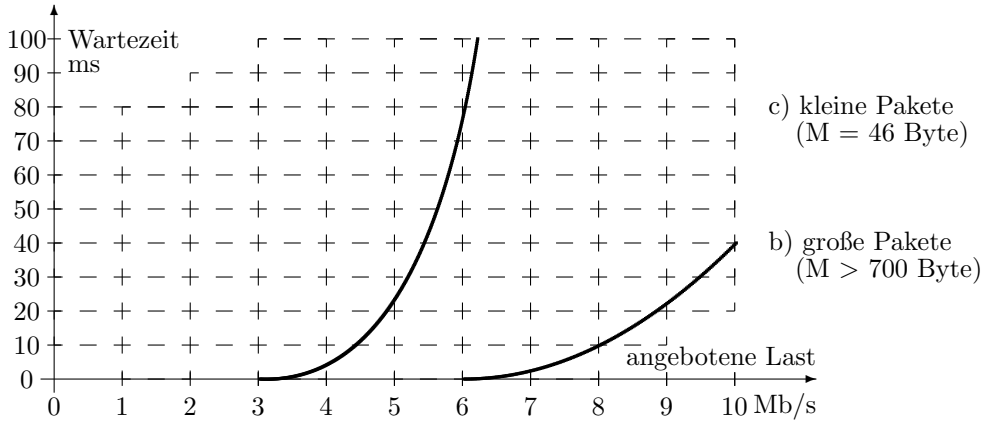


Bild n4p03b

Wartezeit in Abhängigkeit von angebotener Last und Paketgröße

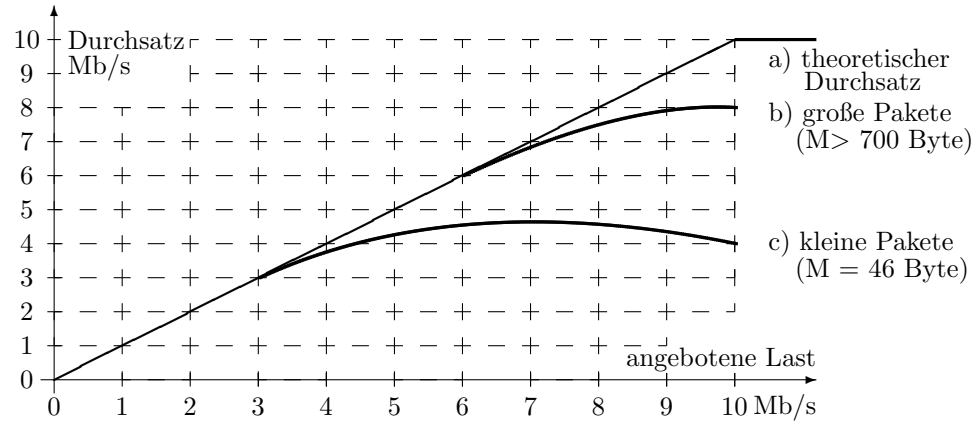


Bild n4p03a

Datendurchsatz in Abhängigkeit von angebotener Last und Paketgröße

4.5 Token Ring (IEEE 802.5)

Ring mit Sendeberechtigungsmarke (DIN ISO 8802-5)

Topologie: unidirektionaler Ring
Protokoll: Token-Verfahren (Token Passing), bitseriell

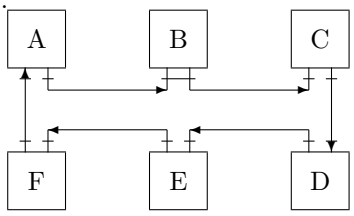


Bild n4p10

Schicht 0 (Medium):
Abgeschirmte verdrillte Leitungen (STP, shielded twisted pair)
Wellenwiderstand Z = 150 Ω
Automatische Überbrückung (bypass function) abgeschalteter Stationen in der TCU (Trunc Coupling Unit)
Signalregenerierung in der TCU
1 Bit Puffer (+ Verzögerung) in der TCU

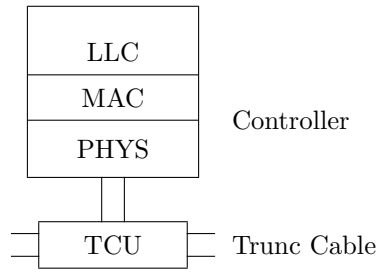


Bild n4p11

Schicht 1 (Bitdarstellung):
Pegel: ± 2 Volt, erdfrei, symmetrisch
Takt: 1MHz bzw 4 MHz
Daten und Signalelemente:
'0' = ⊓⊔ '1' = ⊔⊓ } Manchester-Code
'J' = ⎽⎽ oder ‾‾
(keine Pegeländerung über einen Takt)
'K' = ⊔⎽ oder ⊓‾
(Pegeländerung nur am Taktanfang)

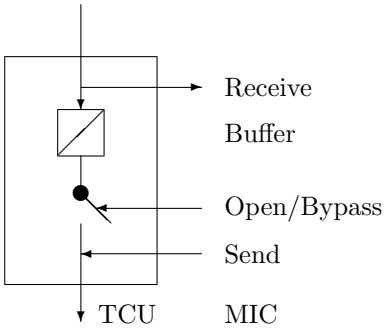


Bild n4p13

Schicht 2: Token Passing Protocol
- Ein 'Free-Token' kreist.
- Sendewillige Station belegt Token (T = 1) und hängt Daten an.
- Empfänger setzt Accepted Bit im FS (= 1. Bit im Frame Status).
- Sender löscht den Frame und sendet 'Free-Token'.

Schicht 2a (MAC):
MAC-PDU

Oktetts	Feld	Inhalt
1	SD	Starting Delimiter = `JK0JK000`
1	AC	Access Control (Token Byte) = `PPPTMRRR`
1	FC	Frame Control = `FFZZZZZZ`
2/6	Destination Address	je nach Netzkonfiguration werden 2
2/6	Source Address	oder 6 Byte für Adressen verwendet
M	Data	MAC-SDU = LLC-PDU für jedes LAN einstellbare Größe M z.B. M = 64 K für IP-Pakete
4	FCS	Frame Check Sum (CRC) vgl. Ethernet
1	ED	Ending Delimiter = `JK1JK1IE`
1	FS	Frame Status = `ACrrACrr`
	MSB LSB	transmitted left to right

AC: Access Control (Token Byte) = `PPPTMRRR`
PPP = Priorität, nur Stationen mit gleicher oder höherer Priorität dürfen dieses Token nutzen
T = Token bit ('0'= Free Token, '1'= Busy Token)
M = Monitorbit (wird von der Monitorstation von '0' auf '1' gesetzt)
RRR = Reservierungswünsche für die Priorität des nächsten Tokens

FC: Frame Control = `FFZZZZZZ`
FF = Format Type
00 = MAC-Frame, 01 = LLC-Frame, 1x = user defined
ZZZZZZ = control bits

ED: Ending Delimiter = `JK1JK1IE`
I = Intermediate indicator ('0'= single frame)
E = Error bit (kann von jeder Zwischenstation im Fehlerfall auf '1' gesetzt werden)

FS: Frame Status = `ACrrACrr`
A = Accepted; Zielstation setzt diese Bits von '0' auf '1', wenn die Adresse erkannt wurde
C = Copied; Zielstation setzt diese Bits von '0' auf '1', wenn die Daten übernommen (copiert) wurden

Andere Blockformate (Frames):

Token: | SD | AC | ED |

Abort: | SD | ED |

MAC-Protokoll: (ohne Prioritäten)

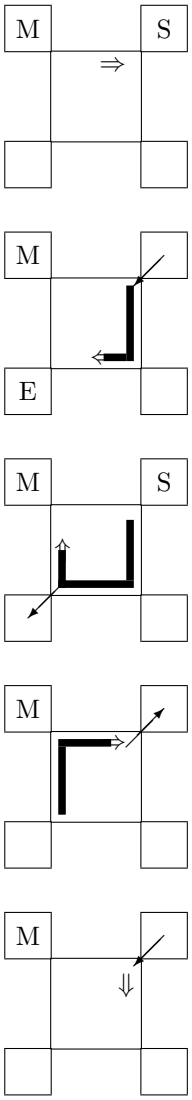


Bild n4p12

. .0. Eine Station im Ring ist Monitor
1. Monitor erzeugt Free Token
2. Stationen schleifen Token durch
3. Sendewillige Station S ändert das Token-Bit und hängt Adressen, Daten und CRC an; die gesamte Nachricht darf eine zeitliche Gesamtlänge (THT = Time Holding Token = 10 ms) nicht überschreiten
4. Empfänger E speichert die Nachricht und setzt AC-Bits
5. Monitor invertiert M-Bit und entfernt Nachrichten mit invertiertem M-Bit
6. Sender S entfernt die Nachricht vom Ring und erzeugt ein neues Free Token

Die Bitkapazität des Ringes von mindestens 24 Bit (TRL $\equiv$ $24\mu s$ bzw $6\mu s$) für die 3 Byte eines Free Token wird erzeugt durch je 1 Bit pro Station, die fehlenden müssen durch den Monitor bereit gestellt werden (Schieberegister).
Dies entspricht (bei 1 MHz Takt) einer physikalischen Ringumfang von $l = t \cdot c^* = 24 \cdot 10^6 s^{-1} \cdot \frac{2}{3}\ 3 \cdot 10^8 m/s = 4\,800m$
Faires Protokoll:
Bei insgesamt N Stationen im Ring kommt jede nach maximal (N - 1) Umläufen dran. Die maximale Wartezeit für eine Station ist $t_{max} = (N-1) \cdot THT$
Maximale Belegung: 100 %
Gefahr: Wenn Sender abstürzt und kein 'Free Token' generiert...
Jede Station muß senden und Monitor Station werden können.
Monitorfunktionen:
- Free Token erzeugen
- alle Token prüfen
- kreisenden Token entfernen, d.h. solche mit invertiertem M-Bit, und Free Token erzeugen, d.h. T-Bit auf '0' setzen und Daten entfernen.
- bei Tokenverlust ein neues Token erzeugen nach einem Timeout (TNT = Timer No Token $\approx TRL + N \cdot THT \leq 1s$).

Stationsfunktionen:
- CRC aller Pakete prüfen und im Fehlerfall das E-bit setzen.
- duplizierte Token mit eigener Adresse entfernen
- Monitorfunktion übernehmen, falls kein Free Token nach $t_{max} = TNT$ auftritt, dabei Konkurrenzen auflösen.

Managementfunktionen:
- Die Timer aller Stationen werden von einem zentralen Manager gesetzt.

4.6 Token Bus (IEEE 802.4)

Bus mit Sendeberechtigungsmarke (DIN ISO 8802-4)

Topologie: Bus
Protokoll: Token-Verfahren (Token Passing), logischer Ring

Schicht 0 (Medium):
Ethernet-Kabel

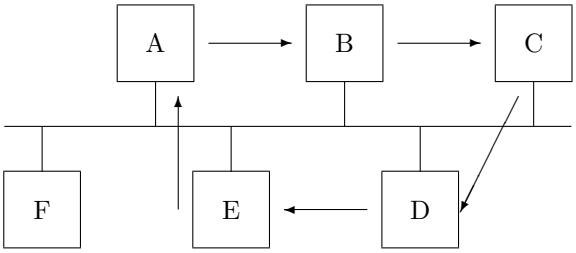


Bild n4p15

Token Passing Protocol
- Teilnehmer sendet Nachricht an X
- oder ein 'Free Token' an Nachbarn (z.B.: mit nächsthöherer Hardware-Adresse).
- Es darf nur der Teilnehmer senden der den Token hat
- Daten werden direkt geschickt und das Token an den Nächsten weitergereicht.
  Unterschiede zu CSMA/CD:
- höherer Aufwand im Protokoll
- höherer Overhead
- fairer Zugriff und berechenbare Wartezeit (vgl. Token Ring)

4.7 Fiber Distributed Data Interface, FDDI (IEEE 802.6)

Insbesondere als Backbone zur Verbindung von LANs über Gateways (GW) oder Bridges (B).

Topologie: bidirektionaler Ring mit Lichtwellenleiter (LWL)
Protokoll: Token-Verfahren (Token Passing), bitseriell

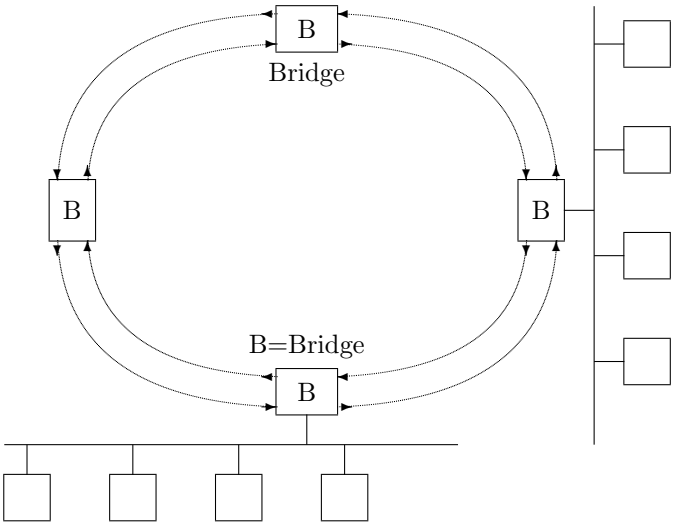


Bild n4p16

4.8 Asynchronous Transfer Mode, ATM

Verteilter Stern zur Übertragung von Daten, Sprache (Ton) und Bildern (Video)
Topologie: bidirektionaler Ring mit Lichtwellenleiter (LWL)
Protokoll: innerhalb des ATM: Frame Insertion. Das Prinzip beruht darauf, daß ähnlich wie bei der asynchronen seriellen Übertragung nach V.4, Nachrichten (Zellen) zu beliebigen Zeitpunkten in einen Datenstrom eingefügt werden. Dadurch ergeben sich nur minimale Verzögerungen, so daß Bilder und Sprache praktisch verzögerungsfrei übertragen werden können Der Aufbau der Zellen besteht aus einem Header mit 5 Byte Adreßinformation und 48 Byte Nutzdaten. Die Wegewahl im ATM-Netz erfolgt ähnlich wie in der OSI-Schicht 3, kann also feste (permanent) und gewählte (switched) Verbindungen unterstützen.

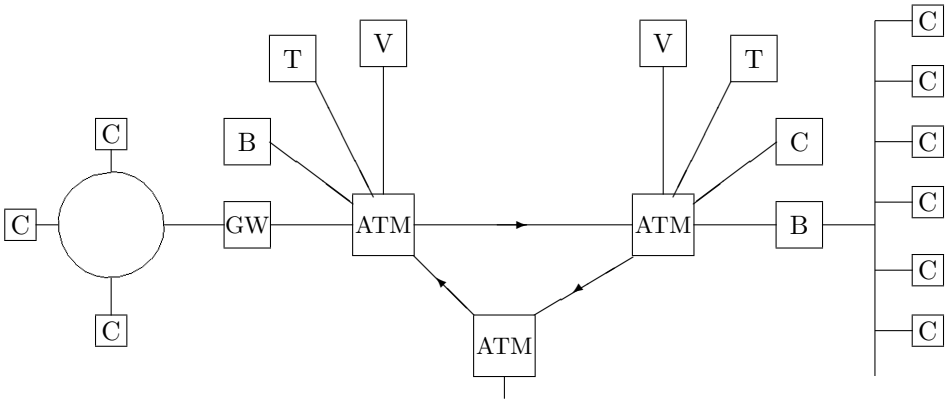


Bild n4p17

ATM als Backbone zur Verbindung von Telefonen (T), Video-Stationen (V) und Computern (C) in LANs über Gateways (GW) oder Bridges (B).

ATM wird auch als B-ISDN bezeichnet. B steht für Breitband, in dem Sinne, daß eine hohe Übertragungsbandbreite (Übertragungsrate) zur Verfügung steht; es ist nicht zu verwechslen mit der (Breitband-)Modulation der Schicht 1 (Physical Layer).

4.9 Wide Area Networks (WAN)

<table>
<tr><td></td><td colspan="2">Telematic Services</td><td colspan="6">Value Added Services</td></tr>
<tr><td>7</td><td>Telefax
F.160-
F.190</td><td>Teletex
F.200</td><td>MHS
X.400</td><td>Directory
X.500</td><td>FTAM
ISO
8571</td><td>JTM
ISO
8831</td><td>Virtual
Terminal
ISO 9040</td><td>BTX</td></tr>
<tr><td>6</td><td>T.6</td><td>T.61</td><td colspan="6">T.50, T.51
X.208, X.209, X.409 = ASN.1</td></tr>
<tr><td>5</td><td colspan="8">T.62
Control Procedures for Teletex and Group 4 Telefax</td></tr>
<tr><td>4</td><td colspan="8">T.70
Network Independent Basic Transport Services</td></tr>
<tr><td>3
2
1</td><td colspan="2">PSTN</td><td>PSPDN</td><td>CSPDN</td><td colspan="2">ISDN</td><td colspan="2">(LAN)</td></tr>
</table>

T.6	Facsimile coding scheme
T.61	Character repertoire for the international Teletex service
T.50	International alphabet No. 5 (IA5)
T.51	Coded character set for Telematic services
FTAM	File Transfer, Access, and Management
JTM	Job Transfer and Manipulation
ASN.1	Abstract Syntax Notation No. 1
PSTN	Public Switched Telephone Network
PSPDN	Packed Switched Public Data Network
CSPDN	Circuit Switched Public Data Network
ISDN	Intergrated Services Data Network
LAN	Local Area Network

4.9.1 Öffentliche Datennetze

1. Packet Switched Public Data Networks (PSPDN)
(paketvermittelnde öffentliche Datennetze)
Realisierung durch DATEX-P
DATEX-P10: PVC
DATEX-P20: SVC

Übermittlungsstruktur

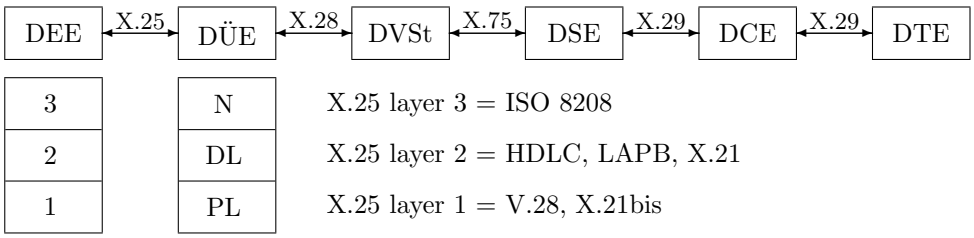


Bild n3p41

X.25: Public data networks: Interfaces - Interface between DTE and DCE for terminals operating in the packet mode and connected to public data networks by dedicated circuit

X.28: Public data networks: Interfaces - DTE/DCE interface for a start-stop mode data terminal equipment accessing the packet assembly/disassembly (PAD) facility in a public data network situated in the same country

X.29: Public data networks: Interfaces - Procedures for the exchange of control Information and user data between a packet assembly/disassembly (PAD) facility and a packet mode dte or another PAD

X.75: Public data networks: Transmission, signalling and switching - Packet-switched signalling system between public networks providing data transmission services

DEE = Datenendeinrichtung ≡ DTE = Data Terminal Equipment
DVSt = Datenvermittlungssation ≡ DSE = Data Switching Exchange
DÜE = Datenübermittlungseinrichtung ≡ DCE = Data Circuit terminating Equipment

Dienste nach X.25 ≡ ISO 8208 (Datex-P) s.a. 3.2.3
CALL.request = N-CONNECT.request(indicate) – PDU (an Schicht 2 : DL)

CALL.accepted = N-CONNECT.response(confirm) – PDU (an Schicht 2 : DL)

Packet = N-DATA.request(indicate) – PDU (an Schicht 2 : DL)

2. Circuit Switched Public Data Networks (CSPDN)
(leitungsvermittelnde öffentliche Datennetze)
Realisierung durch DATEX-L

4.9.2 Öffentliche Telefonnetze (PSTN)

Public Switched Telefone Networks im GSTN
(Global Switched Telefone Network)

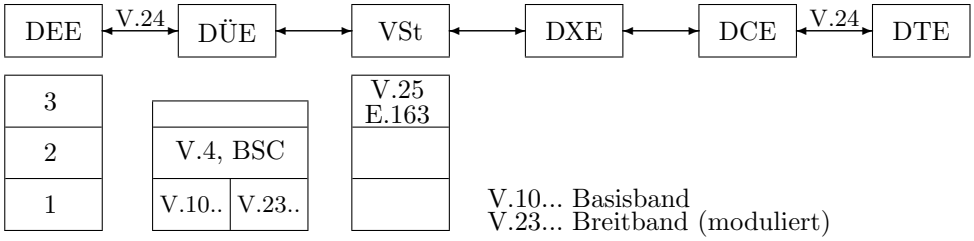


Bild n3p41a

V.25	Automatic calling/answering equipment in the GSTN
E.163	Numbering plan for the international telephone
V.24	Interchange circuits between dte and dce
V.4	Asynchronous start-stop procedure
BSC	Binary Synchronous Communication
V.10/11,	V.28, V.31 Electrical characteristics
V.21 ff	Modulation procedures on public switched or leased telephone lines

4.9.3 Dienste Integrierende Datennetze (ISDN)

Integrated Services Data Networks (I.200)

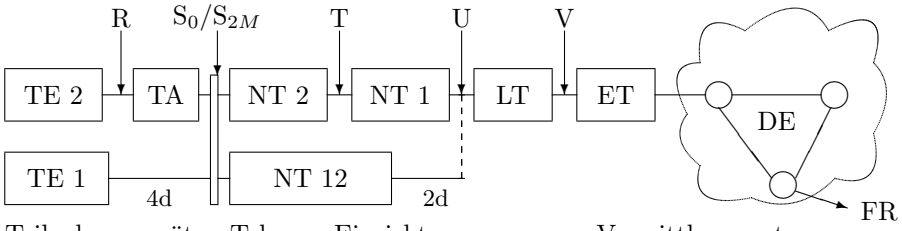


Bild n3p41b

- ET (Exchange Termination): Vermittlungsstelle
- LT (Line Termination): Leitungsabschluss
- NT (Network Termination): Netzabschluss
- TE (Terminal Equipment): Endgerät
- TA (Terminal Adaptor): Geräteanpassung (für Analoggeräte)

S_0	Basisanschluss	2 Nutzkanäle D_1 und D_2 mit je 64 kbit/s 1 Zeichengabekanal D mit 16 kbit/s
S_{2M}	Primärmultiplexanschluss	max 30 Nutzkanäle mit je 64 kbit/s 1 Zeichengabekanal D mit 64 kbit/s

4.10 Das Internet

Das Netz der Netze

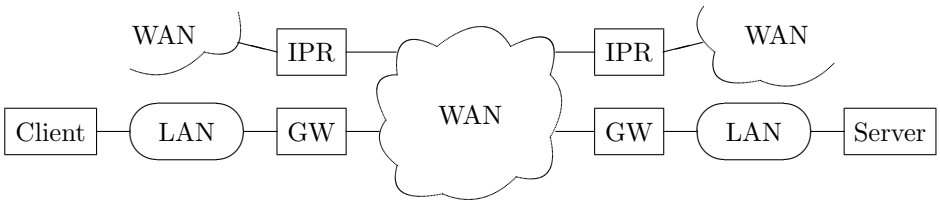


Bild n4p20 (IPR = Internet-Packet-Router)

Client-Server-Architektur

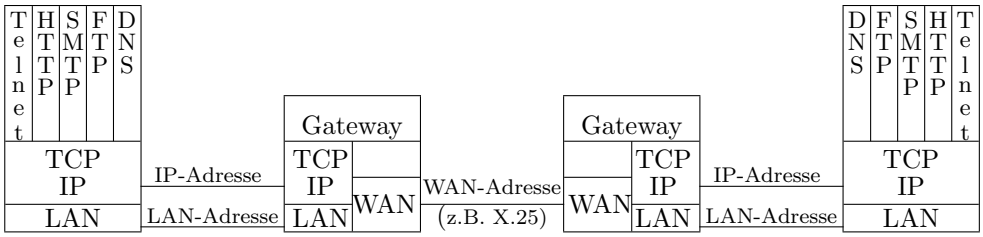


Bild n4p21

Dial-up Access

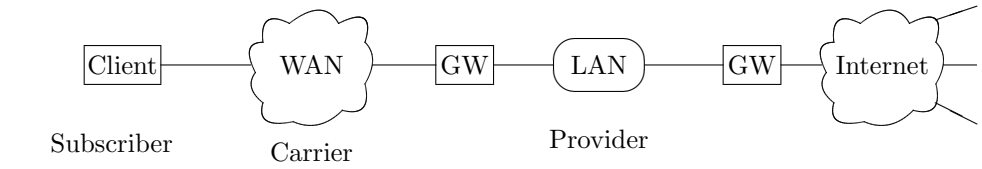


Bild n4p22

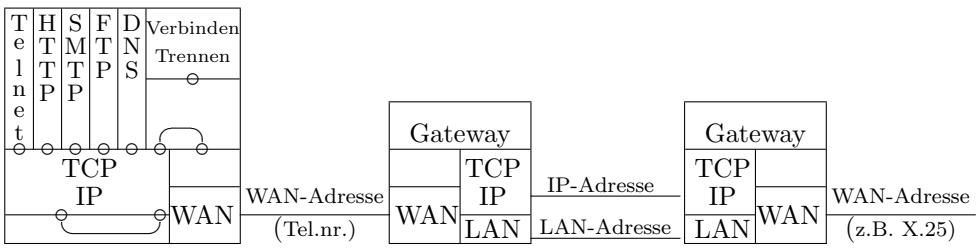


Bild n4p23

7 6 5	Telnet Protocol	HTTP Hyper text Transfer Protocol	SMTP Simple Mail Transfer Protocol	FTP File Transfer Protocol	NNTP Network News Transfer Protocol	TFTP Trivial FTP	DNS Domain Name Service
4	TCP: Transmission Control Protocol					UDP: User Datagram Protocol	
3	IP: Internet Protocol						ICMP IGMP
2 1	WAN & LAN Protocols						

TCP Ports
Well Known Ports (0 ... 1023)

Name	Nr.	Funktion
TCPMUX	1	TCP Port Service Multiplexer
RJE	5	Remote Job Entry
FTP	21	FTP (Control)
SSH	22	SSH Remote Login
Telnet	23	Telnet
SMTP	25	Simple Mail Transfer Protocol
NAME	42	Host Name Server
DOMAIN	53	Domain Name Server
WHOIS	63	Whois
TFTP	69	Trivial FTP
GOPHER	70	Gopher
FINGER	79	Finger
HTTP	80	HTTP / WWW
POP3	110	Post Office Preotocol 3
NNTP	119	Network News Transfer Protocol
NTP	123	Network Time Protocol

Registered Ports (1024 ... 49151)

Dynamic / Private Ports (49152 ... 65535)

Internet Standards

STD 1		Standardization Process	RFC 1600, 1610
STD 2		Assigned Numbers	RFC 1340
STD 3		Host Requirements	RFC 1122
STD 4		Gateway Requirements	RFC 1009
STD 5	IP	Internet Protocol	RFC 791, 950, 919, 922
	ICMP	Internet Control Message Protocol	RFC 792
	IGMP	Internet Group Multicast Protocol	RFC 1112
STD 6	UDP	User Datagram Protocol	RFC 768
STD 7	TCP	Transmission Control Protocol	RFC 761, 793
STD 8		Telnet Protocol	RFC 652, 854, 855
STD 9	FTP	File Transfer Protocol	RFC 959
STD 9	TFTP	Trivial File Transfer Protocol	RFC 783
STD 10	SMTP	Simple Mail Transfer Protocol	RFC 821, 822
STD 11	SMTP	Format of Electronic Mail Messages	RFC 822
STD 12	NTP-V2	Network Time Protocol (Version 2)	RFC 1119
STD 12	NNTP	Network News Transfer Protocol	RFC 977
STD 13	DNS	Domain Name Service	RFC 1034, 1035
STD 14	DNS-MX	Mail Routing and the Domain System	RFC 974
STD 15	SNMP	Simple Network Management Protocol	RFC 1157
STD 51	PPP	Point-to-Point Protocol	RFC 1661
STD 61	HTTP	Hypertext Transfer Protocol	RFC 1945

Example of the SMTP Procedure

This SMTP example shows mail sent by Smith at host Alpha.ARPA, to Jones, Green, and Brown at host Beta.ARPA. Here we assume that host Alpha contacts host Beta directly.

S: MAIL FROM:<Smith@Alpha.ARPA>
R: 250 OK
S: RCPT TO:<Jones@Beta.ARPA>
R: 250 OK
S: RCPT TO:<Green@Beta.ARPA>
R: 550 No such user here
S: RCPT TO:<Brown@Beta.ARPA>
R: 250 OK
S: DATA
R: 354 Start mail input; end with <CRLF>.<CRLF>
S: Blah blah blah...
S: ...etc. etc. etc.
S: <CRLF>.<CRLF>
R: 250 OK

The mail has now been accepted for Jones and Brown. Green did not have a mailbox at host Beta.

Kapitel 5

Netzwerkmanagement

Ziel des Netzwerkmanagements, der Netzwerkadministration, der Netzwerkverwaltung ist der ungestörte, sichere Betrieb eines Rechnernetzes.

5.1 Managementaufgaben

- Planung der Architektur und Installation
- Einrichtung der Domaine mit Namen und Adressen der Netzknoten
- Betrieb, Verwaltung, Administration
- Erweiterungen, Änderungen, Bestandsführung, Wartung
- Monitoring, Analyse, Tuning, Fehlersuche
- Werkzeuge
- Sicherheit

5.2 Managementkonzepte

- Hierarchie: generell wird eine (hierarchische) Baum-Struktur von Domainen eingesetzt, damit alle Beziehungen eindeutig bleiben.
- zentral verwaltet werden alle Objekte und Systeme innerhalb einer Domaine,
- dezentral erscheinen alle Systeme, die unterhalb einer Domaine angesiedelt sind.

5.3 Objekte im Netz

Objekte, die in einem verteilten System, einem Netzwerk auftreten, sind
- Daten
- Software, Programme
- Hardware, Geräte
- Personen

Diese Objekte können durch Daten beschrieben werden, die verwaltet werden müssen; dazu werden Datenbanksysteme eingesetzt.

Daten	Programme	Geräte	Personen
Technische Daten:			Personendaten
Speicherort	Speicherort	Aufstellungsort	Dienstort
Datum, Erzeuger	Version, Ersteller	Typ, Hersteller	Zuständigkeiten
Benutzer	Benutzer	Bediener	Abteilung
Bedeutung, Relevanz	Relevanz	Relevanz	Stellung
Beziehungen zu anderen:			
Daten und Programmen	Programmen und Daten	Geräten	Personen

Aber auch die Objekte selber müssen verwaltet werden, d.h. sie müssen
- geplant,
- beschafft,
- installiert,
- konfiguriert, d.h. an die Umgebung angepaßt
- optimiert,
- repariert,
- ersetzt,
- entfernt werden.

Beim Netzwerkmanagement werden vorwiegend die vorhandenen Geräte (Netzknoten) und die zwischen ihnen bestehenden Beziehungen (Zweige), die u.a. Kommunikationsprotokolle und deren Betriebsparameter enthalten, verwaltet.

Eine typische Datenstruktur (nach Jackson):

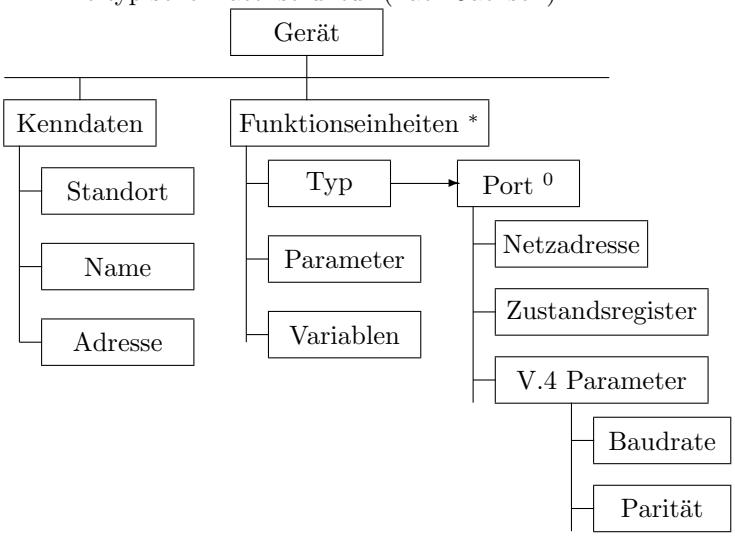


Bild n5p01

Die Beziehungen zwischen den Knoten können in Form von Matrizen dargestellt werden. Hier ein unvollständiges fiktives Beispiel:

Netz-knoten	Netzknoten Protokolle Parameter							
	1	2	3	4	5	6	7	8
1	——	V.4	ISDN	HfD				
	——	8,e,1	S0	HDLC				
2		====						
3			====					
4				====				
5					====			
6						====		
7							====	

Die Attribute der Beziehungen (Verbindungsparameter) können zu den unterschiedlichsten OSI-Schichten gehören, z.B.

- Schicht 1 (Bitdarstellung):
  - Pegel nach V.28, V.10, V.11
  - Modulationsverfahren V.21 (FSK) V.22 (PSK), ... V.32bis
- Schicht 2 (Sicherungsschicht):
  - V.4 (Start-Stop-Prozedur-Parameter: Anzahl der Daten, Prüf- und Stopbbits
  - HDLC
  - ISDN

- Schicht 3 (Vermittlungsschicht):
  - LAN-Protokolle nach ISO 8802 (Ethernet, Token Ring, Token Bus, ..): Ringumlaufzeiten, Time-Out-Limits
  - Datex-P (X.25)
  - Internet Protocol (IP): IP-Adressen
- Schicht 4 (Transportschicht):
  - Transmission Control Protocol (TCP): TCP-Adressen (Ports)
- Schicht 6 (Darstellungsschicht):
  - Verschlüsselungsalgorithmus
  - ASN.1 Anwendung

5.4 Managementwerkzeuge

- Verteilte Anwendungen (Client-Server-Archtekturen)
- Firmware in Vermittlern (Clients)
- Protokolle zur Kommunikation

5.5 Beispiele

5.5.1 OSI–Netzwerkmanagement (ISO 10 164)

Aufgabenbereiche:
- Konfigurationsmanagement: Zustand des Netzes erkennen und überwachen
- Leistungsmanagement: Kontrolle und Analyse des Durchsatzes und der Fehlerraten

- Fehlermanagement: Feststellen, Isolieren und Kontrollieren von Abweichungen im Netzwerkverhalten, insbesondere von Ausfällen
- Abrechnungsmanagement: Sammeln und Verarbeiten von Inanspruchnahme von Betriebsmitteln
- Sicherheitsmanagement: Zugangskontrolle zum Netzwerk

Funktionen:
- Mitlesen: Managenment-Informationen lesen
- Kontrolle: Geräteeinstellungen verändern
- Berichten: Zustandsberichte der Geräte

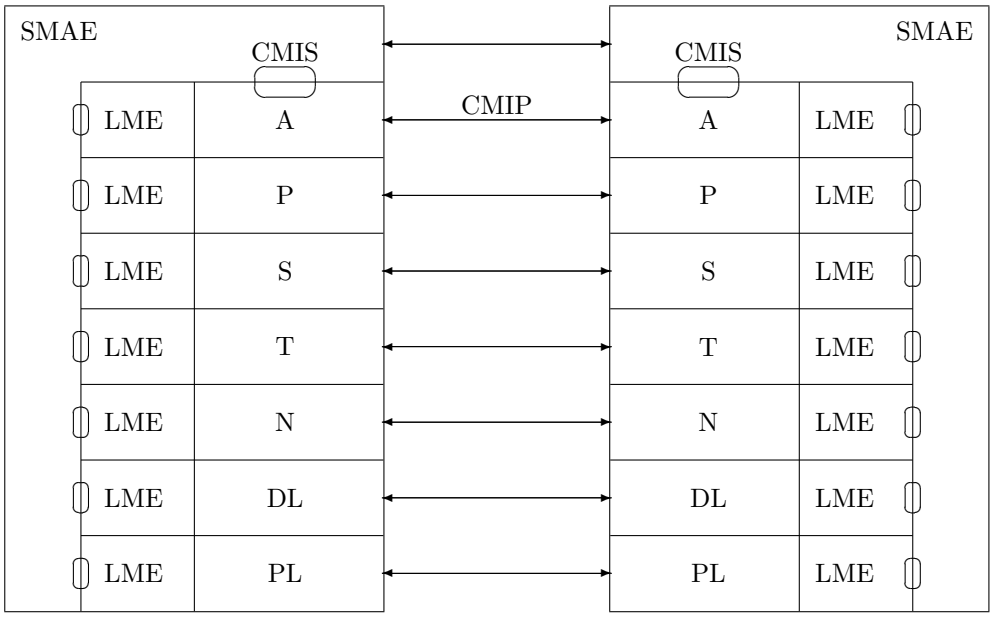


Bild n5p02

Architektur:

SMAE: System Management Application Entity
Netzwerkweite Verwaltung der OSI-Instanzen auf allen Schichten.

LME: Local Management Entity
Lokale Verwaltungsinstanz einer bestimmten OSI-Schicht mit Zugriff auf die (Protokoll-)Instanzen dieser Schicht

CMIP: Common Management Information Protocol
Anwendungsprotokoll zwischen den SMAEs.

CMIS: Common Management Information Service
Anwendungsdienste für SMAE in jedem System.

Dienste:

- A-CONNECT = Verbindungsaufbau zu einer SMAE
- A-DISCONNECT = Verbindungsabbau
- A-GET = Kontrolle: Geräteeinstellungen abfragen
- A-SET = Kontrolle: Geräteeinstellungen verändern
- A-CREATE = Management-Objekt erzeugen, z.B. Neues Gerät
- A-DELETE = Management-Objekt entfernen, z.B. Routing-Eintrag
- A-EVENT_REPORT = Ereignismeldung, z.B. Fehler
- A-ACTION = Ereignisbehandlung, z.B. Neustart, Reset

5.5.2 SNMP, Simple Network Management Protocol

Beispiel SNMP:

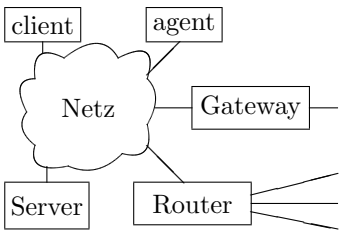


Bild n5p03

. .
- Server enthält MIB (Managed Objects Information Base) RFC 1158
  und RMON (Remote Monitor) RFC 1271
- alle übrigen Netzkomponenten (z.B. Gateway, Router) enthalten Agenten (agents)

5.5.3 NIC, Network Information Center

Aufgaben:
- Monitoring
- Strukturierung
- Information
- Verwaltung von Namen und Adressen
- Validierung von Produkten
- Festlegung von Schnittstellen und Protokollen
- Festlegung von Sicherheitsstandards

Kapitel 6

Sicherheit in Netzen

Der Begriff Sicherheit wird unterschiedlich definiert:
- In Deutschland wird hierunter ein bestimter Zustand eines Systems verstanden,
- im internationalen Bereich wird Sicherheit mehr als Vorgang betrachtet.
- in der ISO wird Sicherheit als die Minimierung der Verwundbarkeit von Werten und Ressourcen definiert [ISO-88]. Dort werden Sicherheitsdienste definiert, die spezifische Sicherheitsmechanismen zur Beseitigung von Schwachstellen (Risiken), also der Verwundbarkeit, bereitstellen.

6.1 Risiken

- Verlust der Vertraulichkeit (privacy)
  Vertraulichkeit bezieht sich stets auf Daten, die nur berechtigten Personen zugänglich sein sollen z.B. Personendaten (vgl. BDSG, Bundesdatenschutzgesetz).
- Verlust der Datenkonsistenz (integrity)
  Integrität bezieht sich auf Daten bzw. Informationen; diese dürfen nur von Befugten verändert werden. Eine Manipulation durch Unbefugte soll verhindert oder erkannt und nachgewiesen werden können. Manipulationen können auf den Verarbeitungssystemen erfolgen oder während der Übertragung. Auf den Verarbeitungssystemen erfolgen diese Angriffe z.B. durch Viren. Bei der Übertragung erfolgen diese Angriffe durch verschiedene Techniken (s.w.u.)
- Verlust der Verfügbarkeit (reliability)
  Verfügbarkeit eines EDV-Systems bezieht sich auf die gespeicherten Daten und insbesondere auf die korrekte Funktion des Systems, also den korrekten Ablauf der Verarbeitungsprogramme, dazu zählen sowohl die Soft- als auch die Hardwarefunktionen

Risiken entstehen durch einfache Bedienfehler oder durch bewußte Angriffe (z.B. Sabotage).

6.2 Risikoobjekte

– Daten: Verändern, Verfälschen, Löschen
Abhören, Entwenden, Mißbrauch
– Programme: Verändern der Funktionalität (Trojanische Pferde, Viren)
Behindern der Funktionalität (Blockieren)
– Hardware: Behindern der Funktionaliät (Bremsen)
Stören und Zerstören (Formatieren der Festplatte, mechanische Überlastung und Bruch durch übermäßige Plattenzugriffe, welche die vorgesehene Lebensdauer überschreiten lassen)
– Personen: Bloßstellung von persönlichen Geheimnissen, Bruch der Privatsphäre (privacy), z.B. zu Zwecken von Abwerbung oder Erpressung

6.3 Bedrohungen

Eine Bedrohung ist ein Umstand oder ein Ereignis, wodurch die Verfügbarkeit, die Integrität oder die Vertraulichkeit von Informationen oder ihrer Verarbeitung in einem EDV-System gefährdet werden kann.

6.3.1 Bedrohungen der Verarbeitungssysteme

- Verändern, Verfälschen, Löschen von Daten und Programmen.
- Abhören, Entwenden von Daten: Bruch der Vertraulichkeit (privacy)
- Vortäuschen einer anderen Identität: Verletzung der Authentizität

6.3.2 Bedrohungen bei der Kommunikation

Bedrohungen können durch eine nicht an der Kommunikation beteiligte Instanz (outsider) oder durch eine der kommunizierenden Instanzen (insider) erfolgen. Angriffe können aktiv oder passiv sein. Aktive Angriffe beinhalten Modifikationen, also schreibende Zugriffe, passive Angriffe erfolgen durch Abhören ohne Modifikation, also lesende Zugriffe.

6.3.2.1 Abhören von Nachrichten (Passiv)

Ein Angreifer (ein Outsider) kann Nachrichten zu jedem Zeitpunkt an jeder Stelle des Netzwerkes abhören (Verkehrsanalyse), das heißt er sieht das äußere Erscheinungsbild der Nachricht.
Beispiel: Sniffing im Internet

6.3.2.2 Abhören von Nachrichteninhalten (Mitteilungen) (Passiv)

Ein Angreifer kann Nachrichteninhalte, d.h. die Klartext-Mitteilung M sehen, wenn M unverschlüsselt übertragen wird, oder wenn er für eine verschlüsselte Mitteilung den Dechiffrierschlüssel kennt.
Beispiel: Monitoring, insbesondere Abhören von Paßworten

6.3.2.3 Modifikation von Nachrichten (Aktiv)

Ein Angreifer kann Nachrichten oder Bestandteile davon an jeder Stelle des Netzwerkes entfernen. Er kann ferner Daten in bestehende Nachrichten zusätzlich einfügen oder austauschen. Das kann Adreßinformationen oder Anwendungsinformationen betreffen.
Beispiel: aktives Sniffing im Internet, TCP Hijacking

6.3.2.4 Einspielen von Nachrichten (Aktiv)

Ein Angreifer kann zu jedem Zeitpunkt Nachrichten einspielen. Dies können
- vom Angreifer neu erstellte Nachrichten
- zuvor abgehörte Nachrichten sein.
Der Angreifer kann durch Entfernen und Einspielen von Nachrichten die vom Absender vorgegebene Reihenfolge verändern.

6.3.2.5 Maskerade (Aktiv)

Durch eine Kombination der vorhergehenden Angriffe kann ein Angreifer in einem Kommunikationsprotokoll die Identität eines anderen Teilnehmers vortäuschen.
Beispiele: Spoofing im Internet, DNS-Spoofing, IP-Spoofing, UDP-Spoofing

6.3.2.6 Leugnen von Nachrichten

Eine der an einem Kommunikationsprotokoll beteiligten Instanzen (Insider) kann anschließend leugnen, daß eine Nachricht von ihr abgesendet wurde.

6.3.2.7 Dienstverweigerung

Eine Instanz kann die Erfüllung der von ihr geforderten Aufgaben verweigern. Dieser Angriff kann vom Vermittler oder vom Empfänger, oder von einem Server in einer Client-Server-Beziehung, durchgeführt werden. Ein Angreifer kann ein Verteiltes System durch Einspielen großer Mengen von Daten oder Anfragen lahmlegen, so daß kein regulärer Dienst mehr erbracht werden kann (Blockieren).

6.3.3 Sonstige Bedrohungen

Social hacking
Beschaffung von sicherheitsrelavanten Informationen durch soziale Kontakte wie:
– Telefonanrufe (Vortäuschung einer autorisierten Person),
– Gesellschaften, Partys, Gelage (Preisgabe von Informationen im Rausch),
– Internet-Newsgroups, Internet-Chat, E-mail (Kommunikation mit Unbekannten).

6.4 Risikoanalyse

- Bedrohungsanalyse: Welche Motivatioen kann ein Angreifer haben ?
  - Nutzung von Ressourcen
  - Industriespionage
  - Sabotage
- Informationswertanalyse: Welche Daten stellen einen Sicherheitsrisiko dar (welche Folgen sind bei iherer Verfälschung zu befürchten) ?
  - Produktions- und Forschungsgeheimnisse (Wettbewerbsnachteile)
  - Personendaten (Abwerbung, Erpressung)
- Schwachstellenanalyse: wo könnte ein Angreifer ansetzen ?
  - Sicherheitslücken der Verarbeitungssysteme, Systemfehler
  - Administrationsfehler
  - Bedienfehler

6.5 Risikoabwehr

6.5.1 Systemsicherheit

6.5.1.1 Sicherheitskonzeption und -management (Security Policy)

Festlegung von Zuständigkeiten: in jedem Unternehmen muß für
– jedes Subnetz
– jeden Rechner
– jede Anwendung
genau ein Betreuer -Administrator- und seine Stellvertretung festgelegt werden.

6.5.1.2 Technische Schutzeinrichtungen

– Firewalls dienen zur Isolation Lokaler Netze (LAN) von Weitverkehrsnetzen (WAN) insbesondere vom Internet. . .

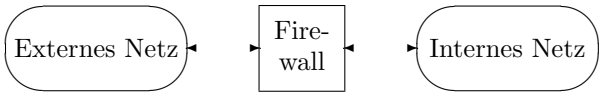


Bild n5p05

Ein Firewall besteht im einfachsten Fall aus einem Rechner, einem (dualhosted) Gateway; im ausgedehntesten Fall wird er von einem Netzwerk mit mehreren Komponenten gebildet (screening subnet)
– Verschlüsselung interner Daten (z.B. auf der Festplatte)
– OSI-Security ISO 7498-3

6.5.2 Übertragungssicherheit

6.5.2.1 Technische Methoden

- Verwendung sicherer Leitungswege (Abschirmungen, LWL)
- Verwendung verschlüsselter Leitungsprotokolle
  (geheime Bitdarstellung auf OSI-Schicht 1)

6.5.2.2 Kryptographische Methoden

Sichere Kommunikationsprotokolle geben Schutz auf den höheren Schichten, insbesondere auf Schicht 6 (Darstellungsschicht) des OSI-BRM. Dazu werden kryptographische Methoden zur Ver- und Entschlüsselung eingesetzt. Die verwendeten kryptographischen Methoden des Protokolls müssen den Kommunikationsteilnehmern bekannt sein, die Sicherheit beruht auf der Geheimhaltung von Schlüsseln.

Der prinzipielle Ablauf eines verschlüsselten Nachrichtenaustausches:
Ein Teilnehmer A transformiert eine Nachricht M mit einem Schlüssel E (Encryption) zu E(M) und sendet diese an B. B transformiert die empfangene Nachricht mit einem Schlüssel D (Decryption) zurück und erhält somit wieder M = D(E(M)). Ein Angreifer C kann in diesem Modell lesend und schreibend auf den Kommunikationskanal L_{ab} zugreifen.

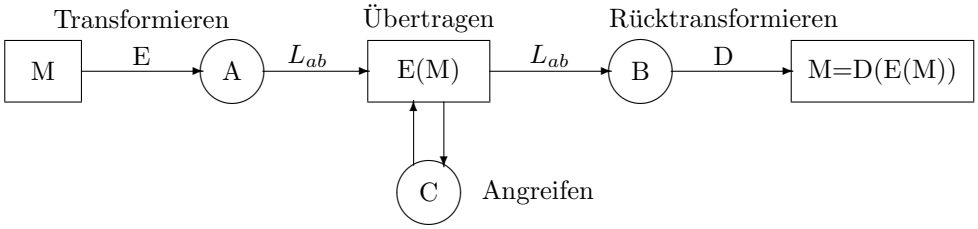


Bild n5p06

Beim **symmetrischen Kryptosystem** ist E = D = K.
Nur A und B sind im Besitz desselben geheimen Schlüssels K.
Beispiel: DES (Data Encryption System)

Beim **asymmetrischen Kryptosystem** ist $E \neq D$.
Jeder Teilnehmer besitzt ein Schlüsselpaar, bestehend aus einem Chiffrierschlüssel E_X, der öffentlich bekannt gemacht werden kann und einem Dechiffrierschlüssel D_X der nur dem Eigentümer dieses Schlüsselpaares bekannt ist.

Es ist rechentechnisch nicht mit vertretbarem Aufwand möglich, aus Kenntnis des Chiffrierschlüssels E den Dechiffrierschlüssel D abzuleiten.

Public Key Kryptosysteme sind solche, bei denen der Chiffrierschlüssel veröffentlicht wird. Der Chiffrierschlüssel wird als öffentlicher, der Dechiffrierschlüssel als privater Schlüssel bezeichnet.

Zur Übermittlung einer vertraulichen Nachricht von A an B verschlüsselt A seine Nachricht mit dem öffentlichen Schlüssel E_B von B, so daß dieser die Nachricht mit seinem eigenen privaten Dechiffrierschlüssel D_B wieder entschlüsseln kann.
Beispiel: RSA (Verfahren nach Rivest, Shamir und Adleman)

Elektronische Unterschrift
Will A eine Nachricht M an B senden, deren Urheberschaft (Authentizität) der Empfänger prüfen kann, so erzeugt sie mit ihrem eigenen privaten (De)Chiffrierschlüssel D_A eine chiffrierte Nachricht $D_A(M)$. B kann durch Anwendung des eindeutig A zugeordneten Chiffrierschlüssels E_A nachweisen, daß M von A stammen muß, wenn die Operation $E_A(D_A(M))$ die von A stammende Nachricht erzeugt. Der Ausdruck $D_A(M)$ wird dann auch als Elektronische Unterschrift oder Signatur von A bezeichnet, mit

der A bestätigt, daß die Mitteilung M von ihr stammt. Wenn der Chiffrierschlüssel E_X eines Teilnehmers X öffentlich bekannt ist, so kann eine von diesem erzeugte Elektronische Unterschrift von jedermann überprüft werden. Voraussetzung für dieses Verfahren ist, daß Chiffrier- und Dechriffrieroperationen vertauschbar sind, also D(E(M)) = E(D(M)) ist.

Schlüsselabsprache
Asymmetrische Kryptosysteme sind deutlich aufwendiger als symmetrische. Daher wird in der Regel zu Beginn einer Sitzung (S-CONNECT) ein asymmetrisches Kryptoverfahren verwendet um einen gemeinsamen symmetrischen Schlüssel zu vereinbaren, der nur für diese eine Sitzung gilt.
Beispiel: Diffie-Hellman Verfahren (1976)

Authentisierung
Unter Authentisierung von Teilnehmern oder Instanzen eines Rechensystems wird der Nachweis einer zuvor behaupteten Identität verstanden.
Beispiel: Benutzerauthentisierung beim Unix-login durch Paßworte.

Protokolle zur Teilnehmerauthentisierung und Schlüsselverteilung können direkt zwischen zwei Kommunikationspartnern abgewickelt werden (Zwei-Parteien-Protokolle) oder die Dienste einer vertrauenswürdigen Dritten Partei (Trusted Third Party, TTP) mit einbeziehen (Drei-Parteien-Protokolle). Wenn sehr viele Teilnehmer vorhanden sind, ergeben sich jedoch Probleme bei der Schlüsselverwaltung, insbesondere wenn Schlüssel nur eine begrenzte Lebensdauer haben. Dann werden Authentisierungsdienst benötigt.
Beispiel: Needham-Schroeder-Protokoll, das Kerberos-Protokoll (RFC 1510), das Diffie-Hellman Key Agreement und das Directory Authentication Protocol (X.500).

6.5.2.3 Anonyme Kommunikation

Die Anonymität bei der Kommunikation wird als wichtiger Aspekt des informationellen Selbstbestimmungsrechtes betrachtet und soll in diesem Zusammenhang Benutzer und Instanzen vor Rückschlüssen schützen, die aus der Beobachtbarkeit der Kommunikation resultieren. Anonyme Kommunikation hat daher aus Sicht des Datenschutzes einen Stellenwert, der mit dem der Vertraulichkeit von Daten vergleichbar ist. Anonymität kann den Sender (Originator), den Empfänger (Recipient) oder die Kommunikationsbeziehung betreffen.

Digitale Pseudonyme
Ein digitales Pseudonym dient zum Schutz von Individuen (Teilnehmern) vor der Erzeugung von Kommunikations-Profilen.

Ein Teilnehmer ist jederzeit in der Lage, mit Hilfe seines privaten Schlüssel die Eigentümerschaft des Pseudonyms nachzuweisen und gleichzeitig ist sichergestellt, daß das Pseudonym nicht von anderen verwendet werden kann.

Ein Teilnehmer kann mehrere solcher Pseudonyme besitzen und verwenden.

Ein digitales Pseudonym ist ein öffentlicher Schlüssel, mit dem Elektronische Unterschriften überprüft werden können, die von dem anonymen Eigentümer des zugehörigen privaten Schlüssels erzeugt wurden.

Inhaltsverzeichnis