Datenkommunikation und Rechnernetze

Prof. Dr. rer. nat. Bernd E. Wolfinger

E-Mail: wolfinger@informatik.uni-hamburg.de

Web: www.informatik.uni-hamburg.de/TKRN

Wintersemester 2011/2012

18.143 Datenkommunikation und Rechnernetze (DKR)

Bernd Wolfinger

Lernziel.
Technische Kommunikationssysteme sind relevant für die rechnerinterne Datenübertragung und bilden überdies die Basis für Rechnernetze und Verteilte Systeme, woraus ihre wachsende Bedeutung resultiert. Innovative Datenübertragungstechniken, u.a. im Zusammenhang mit optischer Signalübertragung, Satelliten- und Mobilkommunikation, finden in jüngerer Vergangenheit verstärkten Einsatz.

Die VorlesungsteilnehmerInnen sollen erkennen, wie sich Nachrichtentechnik und Informatik im Bereich der technischen Kommunikation ergänzen und zu welchen Systemkonzepten diese "Symbiose" geführt hat. Die erworbenen Kenntnisse sollen vertieft werden durch detaillierte Diskussion einer Reihe existierender Architekturkonzepte, Protokolle und Dienste für Kommunikationssysteme und Rechnernetze, die gleichzeitig einen Eindruck von der Vielzahl möglicher Realisierungsvarianten vermitteln sollen.

Die Vorlesung beschäftigt sich somit zum einen mit

- Systemen, Algorithmen, Technologien und Diensten zur Kommunikation und präsentiert darüber hinaus Grundlagen für eine wissenschaftlich fundierte
- Planung, technische Realisierung, Konfigurierung, Analyse etc. von Kommunikations- und Rechnernetzen sowie ihrer Komponenten.

Stellung im Studienplan.
Hauptstudium

Voraussetzungen.
Grundstudium

Vorgehen.
Vorlesung; Integration von eLearning-Werkzeugen und Lernmodulen aus dem ELCH-Projekt TeleMuM in die Lehrveranstaltung

Periodizität.
jährlich zum Wintersemester

Eignung.
Bedingt geeignet für Lehramtsstudierende, Nebenfachstudierende.

Skript.
Dieses Skript wurde von Merlin Senger mit dem Textverarbeitungsprogramm TeXmacs unter Mandriva Linux aus den Folienkopien vergangener Semester erstellt. Die Abbildungen sind mit den Programmen Dia, OmniGraffle, Gimp und GnuPlot erstellt worden.

Skriptversion.
In der vorliegenden Version 2010.1 sind alle Korrekturwünsche berücksichtigt worden, die der Arbeitsgruppe TKRN seit der Veröffentlichung der ersten Skriptversion 2008.1 im Wintersemester 2008/2009 zugegangen sind. Allen Studierenden sei an dieser Stelle ein großer Dank für Ihre wertvollen Hinweise ausgesprochen. Sollten Sie Anmerkungen oder Verbesserungsvorschläge zu dieser Skriptversion haben, so senden Sie diese bitte an `kolesnikov@informatik.uni-hamburg.de`

Inhaltsverzeichnis

Kapitel 1
Einführung

1.1 Einige Grundbegriffe

Im Zentrum der Vorlesung "Datenkommunikation und Rechnernetze" steht die Kommunikation zwischen Informatiksystemen sowie zwischen Menschen mittels Informatiksystemen. Die hierbei wesentlichen zu diskutierenden Aspekte sind die Algorithmen, die Technologien und die Systeme, die zur Kommunikation verwendet werden.

Definition 1.1. *Unter* Kommunikation *verstehen wir koordiniertes symbolisches Handeln mehrerer Beteiligter unter Zuhilfenahme eines Mediums.*

Definition 1.2. *Unter einer* Nachricht *verstehen wir Zeichen oder kontinuierliche Funktionen, die zum Zwecke der Weitergabe Informationen auf Grund bekannter oder unterstellter Abmachungen darstellen.*[1.1]

Die Beteiligten der Kommunikation werden im folgenden *Kommunikationspartner (KP)* genannt, das zu Grunde liegende Medium *Übertragungsmedium (ÜM)*. Unter symbolischem Handeln verstehen wir konkret den Austausch von Nachrichten. Desweiteren unterscheiden wir unterschiedliche Aspekte einer Nachricht:

- Der *syntaktische Aspekt* bezieht sich auch den Aufbau der Nachricht,
- der *semantische Aspekt* auf die Bedeutung der Nachricht und
- der *pragmatische Aspekt* auf den Wert bzw. Nutzen der Nachricht für den Empfänger.

Definition 1.3. *Ein* allgemeines Kommunikationssystem *besteht aus einer Menge von mindestens zwei Kommunikationspartnern sowie einem Nachrichtentransportsystem (NTS) bei indirekter Kommunikation bzw. einem Übertragungsmedium bei direkter Kommunikation.*

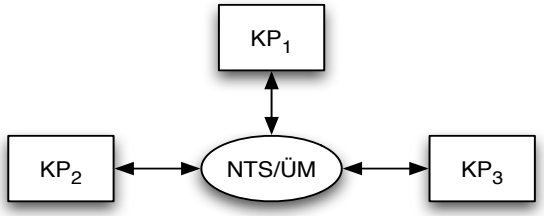


Abbildung 1.1. Schematische Darstellung eines Kommunikationssystems

1.1. vgl. DIN 44300

Die Rolle der Kommunikationspartner besteht also darin, unter Berücksichtigung gemeinsamer Regeln

- Nachrichten zu senden und zu empfangen,
- diese Nachrichten zu interpretieren,
- auf Nachrichten zu reagieren.

Die Rolle des Nachrichtentransportsystems besteht darin, einen Mechanismus für den Nachrichtenaustausch bereitzustellen.

Definition 1.4. *Besteht das Nachrichtentransportsystem seinerseits aus einer Menge von untereinander verbundenen Einzelkomponenten, über die die Nachrichten zwischen den Kommunikationspartnern indirekt weitergeleitet werden, so nennen wir das Kommunikationssystem auch* Kommunikationsnetz (KN).

Beispiele für Kommunikationsnetze sind öffentliche Paketvermittlungsnetze wie DATEX-P, ISDN-Netze und die Netze der Internet Service Provider (ISP). Abbildung 1.2 zeigt eine solche Konstellation. Die Kommunikationspartner des Kommunikationssystems B benutzen das Kommunikationssystem A als Nachrichtentransportsystem. Es sind daher Regeln für die Kommunikation zwischen folgenden Kommunikationspartnern nötig. Zum Beispiel:

- KP1 (B) und KP1 (A)
- KP1 (A) und KP2 (A)
- KP2 (A) und KP3 (B)
- KP1 (B) und KP3 (B)

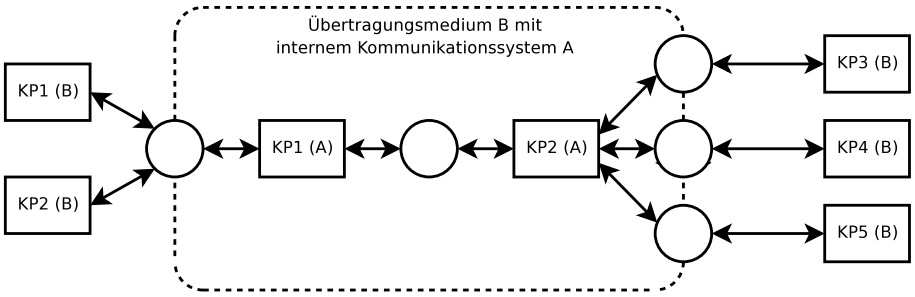


Abbildung 1.2. Ein Nachrichtentransportsystem, das selbst ein Kommunikationssystem darstellt

Merkmale der Kommunikation

Die Kommunikation zwischen Kommunikationspartnern weist in der Regel folgende Merkmale auf:

- Sie dient dem koordinierten Handeln der Kommunikationspartner.
- Sie unterliegt den Intentionen der Kommunikationspartner.
- Ein gemeinsame Verstehensgrundlage (Konventionen, Wissen) wird vorausgesetzt.
- Kommunikation kann sich auf den Kommunikationsprozess und seine Voraussetzungen beziehen (Metakommunikation).
- Sie ist mit den Erwartungen an die anderen Kommunikationspartner verbunden (Partnerbilder).

- Sie unterliegt dem Bestreben nach ökonomischem Verhalten.

Klassen von Kommunikationssystemen

Wir wollen im folgenden zwischen unterschiedlichen Klassen von Kommunikationssystemen unterscheiden. Eine wichtige Klassifikation betrifft die Art der Kommunikationspartner:

1. Zwischenmenschliche Kommunikation, wie etwa Kommunikation mittels einer gemeinsamen Sprache, Mimik oder Gestik. Als Nachrichtentransportsystem kann beispielsweise ein Fernsprechnetz für akkustische Signale dienen. Die formale Modellierung zwischenmenschlicher Kommunikation ist wegen der Komplexität der Kommunikationspartner schwierig.

2. Mensch-Maschine-Kommunikation, bei der ein Kommunikationspartner menschlich und ein anderer maschinell ist. Das Nachrichtentransportsystem könnte beispielsweise das Ein-/Ausgabesystem eines Rechners oder das Netz eines Internet Service Providers sein.

3. Kommunikation zwischen Maschinen[1.2], bei dem die Kommunikationspartner beispielsweise Rechner oder Prozesse sind. Als Nachrichtentransportsystem können Kupferleitungen, Funkverbindungen, ATM-Netze o.ä. dienen.

Abbildung 1.3 zeigt das typische Modell und die Begriffe eines Kommunikationssystems in der Nachrichtentechnik.[1.3]

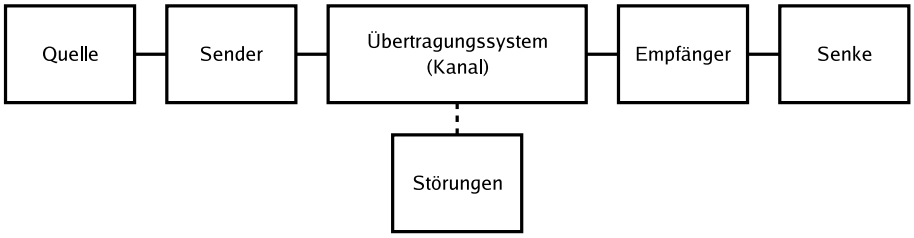


Abbildung 1.3. Nachrichtenübertragungssystem in der Nachrichtentechnik.

Die *Quelle* erzeugt die zu übertragenden Nachrichten, ein *Sender* wandelt die Nachrichten in eine für den Kanal geeignete Form. Der *Kanal* überträgt die Nachrichten als Signale (z.B. elektrische oder optische Impulse, Schallwellen), zum Empfänger. *Störungen* beeinträchtigen die Signale, was eine Verfälschung oder einen Verlust der Nachricht zur Folge haben kann. Der *Empfänger* wandelt die empfangenen Signale in eine für die *Senke* verständliche Form. Die Senke interpretiert, verarbeitet, speichert oder vernichtet die empfangenen Nachrichten.

Datenübertragungssysteme

Datenübertragungssysteme lassen sich in folgende vier Klassen einteilen:

- rechnerinterne Datenübertragung (Abb. 1.4) wie etwa zwischen Zentralprozessoren, E/A-Prozessoren und Hauptspeichermoduln

- Datenübertragung zwischen Rechner und Peripherie (Abb. 1.5) z.B. Sekundärspeicher

- Datenübertragung in einem lokalen Netz (Abb. 1.6), wie z.B. die Kommunikation zwischen Clientrechnern und Fileservern. Hierbei ist das Nachrichtentransportsystem nicht öffentlich.

- Datenfernübertragung (DFÜ) über große Entfernungen hinweg (Abb. 1.7). Hier werden Daten über öffentliche Weitverkehrsnetze übertragen.

1.2. Der Fokus der Vorlesung liegt auf dieser Klasse.

1.3. vgl. DIN 40146/1

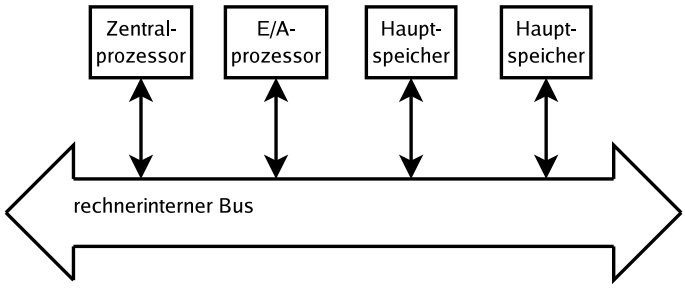


Abbildung 1.4. Rechnerinterne Datenübertragung (i.d.R. existiert eine Hierarchie von Bussystemen)

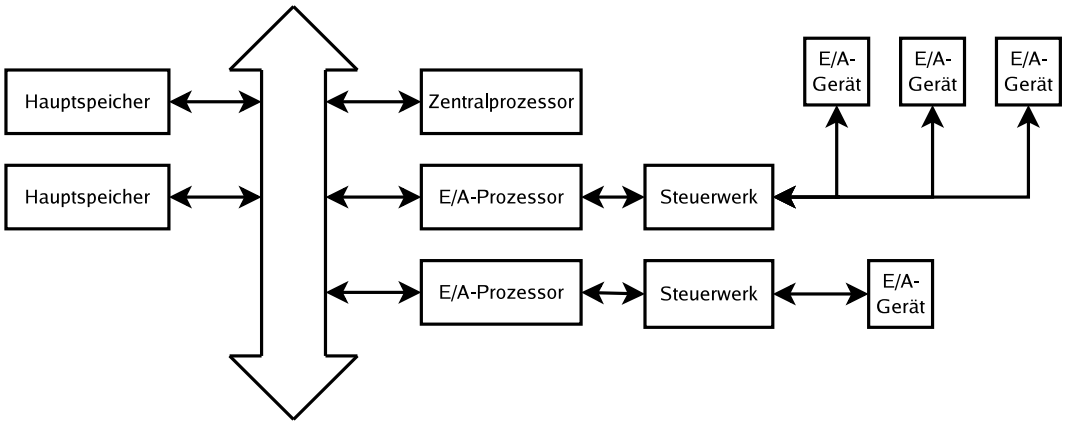


Abbildung 1.5. Datenübertragung zwischen Rechnern und lokaler Peripherie

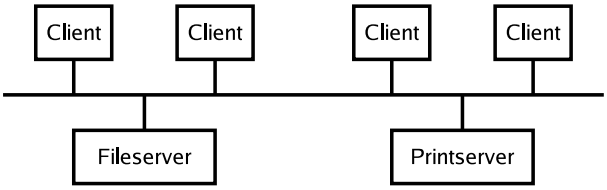


Abbildung 1.6. Datenübertragung im lokalen Rechnernetz

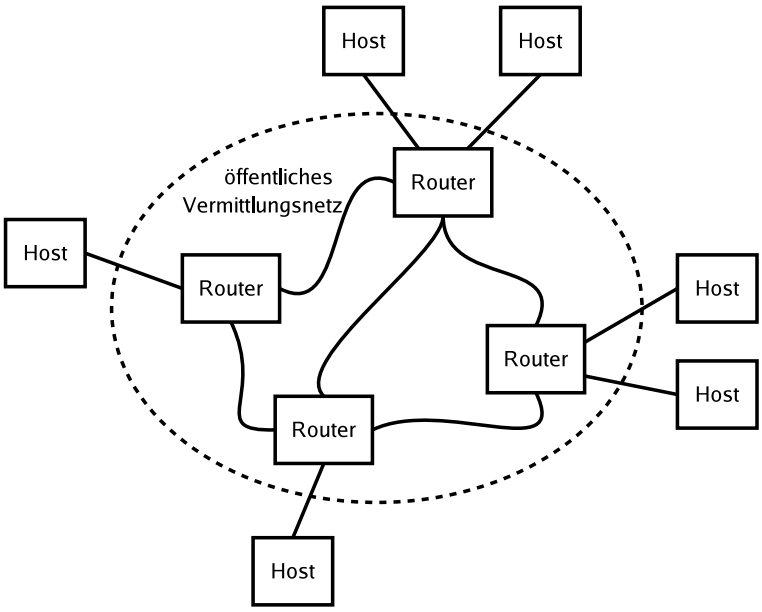


Abbildung 1.7. Datenübertragung im nicht-lokalen Rechnernetz

Definition 1.5. *Eine Menge von untereinander kommunikationsfähigen Kontenrechnern, die durch ein Kommunikationssystem zum Zwecke der Kommunikation und Kooperation verbunden sind, heißt* Rechnernetz.

Beispiele für die Knotenrechner eines Rechnernetzes sind Prozessor-Speicher-Paare mit lokalem Betriebssystem, Multiprozessorsysteme und "handheld computer" mit Anschluss an ein Mobilfunknetz. Beipiele für zugrunde liegende Kommunikationssysteme sind Mobilfunknetze, Satellitenkommunikationssysteme, das konventionelle Fernsprechnetz, das ISDN-Netz, auf Ethernet oder Token Ring basierende LAN[1.4]-Infrastrukturen u.v.m.

Definition 1.6. *Ein Rechnersystem oder Rechnernetz, bei dem der Ort der Ein-/Ausgabe vom Ort der Verarbeitung räumlich getrennt ist, sowie eine Datenfernübertragung mit Hilfe nachrichtentechnischer Systeme stattfindet, heißt* Datenfernverarbeitungssystem.

Ein Rechnernetz aus Terminals zur Datenerfassung, die die erfassten Daten über ein öffentliches Kommunikationsnetz zum zentralen Rechner zur dortigen Verarbeitung senden, und Teile des Resultats wiederum dem Benutzer präsentieren, stellt ein Beispiel für ein Datenfernverarbeitungssystem dar.

Definition 1.7. *Kommunikation über große Entfernungen unter Nutzung nachrichtentechnischer Übertragungsverfahren, sowohl für die Mensch-Mensch-, die Mensch-Maschine- als auch die Maschine-Maschine-Kommunikation nennen wir* Telekommunikation.

Der Begriff *Telematik* stellt ein Kunstwort aus den Wörtern Telekommunikation und Informatik dar, der durch die zunehmende Überlappung von Informatik und Nachrichtentechnik begründet ist.

Definition 1.8. *Ein* Verteiltes System (VS) *ist eine durch ein Kommunikationssystem lose gekoppelte Menge von Knoten, wobei*

- *die Knoten kooperieren, um Systemfunktionen auszuführen (verteilte systemweite Kontrolle)*
- *keine zwei Prozesse dieselbe Sicht des Systemzustands besitzen und insbesondere kein zentraler Prozess existiert, der andere Prozesse mit einer konsistenten, identischen Sicht des globalen Systemzustands versorgen kann.*

Mit loser Kopplung ist hier gemeint, dass die Kommunikation durch Nachrichtenaustausch und nicht über gemeinsamen Speicher stattfindet. Die Knoten stellen nicht unbedingt autonome Rechner dar. Es sei noch erwähnt, dass für den Begriff des Verteilten Systems in der Literatur zahlreiche, sich zum Teil widersprechende Definitionen existieren.

Verteilte Systeme und Rechnernetze sind also voneinander abzugrenzen. Bei Verteilten Systemen steht die Verteilung der Daten, Verarbeitungskapazitäten, Prozessen und vor allem der Kontrolle im Vordergrund. Beim Rechnernetz steht die Kommunikation von autonomen Rechnern im Vordergrund. Verteilte Systeme basieren nicht notwendigerweise auf einem Rechnernetz, sondern können auch innerhalb eines Gehäuses realisiert werden. Rechnernetze stellen nicht notwendigerweise Verteilte Systeme dar, da sie zentral organisiert sein können (z.B. zentrale Systemkontrolle, zentrale Datenhaltung).

1.2 Ziele des Einsatzes und Klassifikation von Rechnernetzen

Aus Endbenutzersicht werden u.a. die folgenden Erwartungen an Rechnernetze gestellt:

- Kommunikation zwischen Personen
    - elektronischer Briefverkehr (mail, news)

1.4. local area network, lokales Rechnernetz

  - elektronische Konferenzen
  - Zugriff auf Informationen
  - Dateitransfer
  - Informationssysteme (news, gopher, www)
  - Fachdatenbanken und Fachinformationszentren
  - 'video on demand'
- Benutzung entfernter Verarbeitungsrechner
  - Ferndialog (remote login)
  - Stapelverarbeitung (RJE, remote job execution/entry)
  - Telefonbanking, teleshopping
  - remote experiments
  - Kooperation von Programmläufen
  - client/server-computing
  - distributed computing
  - verteilte Anwendungen

Aus Sicht der Betreiber stellen Rechnernetze u.a. die folgenden Möglichkeiten dar: Eine Erhöhung der Zuverlässigkeit, Verfügbarkeit, des Grenzdurchsatzes, die Möglichkeit der Nutzung von spezialisierter Hardware und Software, sowie ein Ausgleich schwankender Anforderungen.

Wir betrachten nun zwei Klassifikationen für Rechnernetze. Die erste orientiert sich an den *Verbundarten* die zweite am *geographischen Verteilungsgrad*.

1.2.1 Klassifikation nach Verbundart

Im vorigen Abschnitt wurden unterschiedliche Einsatz- und Nutzungsmöglichkeiten dargestellt, die sich den Betreibern von Rechnernetzen bieten. Wir klassifizieren Rechnernetze somit im folgenden zunächst nach der Art der Nutzung aus Betreibersicht.

Datenverbund. Hier steht eine verteilte Datenhaltung im Vordergrund. Daten können durch *Replikation*, also durch Haltung von Kopien der Datenobjekte, und durch *Partitionierung*, also durch Aufteilung der Datenobjekte auf verschiedene Rechner verteilt sein. Auch Mischformen von Replikation und Partitionierung sind möglich.

Funktionsverbund. Es werden Rechner verbunden, die spezielle Aufgaben übernehmen. Hierfür werden die Rechner meist auch mit spezieller Hardware ausgestattet.

Lastverbund. Hier dient der Verbund der Rechner dem Lastausgleich (load sharing). Problematisch ist hierbei die Definition und Messung der Momentanauslastung sowie der zusätzliche Aufwand zum Verlagern von Aufträgen auf einen anderen Rechner.

Nachrichtenverbund. Beim Nachrichtenverbund wird das Rechnernetz zum Austausch von Nachrichten auf der Ebene kommunizierender menschlicher Endbenutzer verwendet. Ein Beispiel hierfür ist der Austausch von E-Mails.

Zuverlässigkeitsverbund. Bei hochkritischen Anwendungen kann durch mehrfache Berechnung durch verschiedene Rechnernetzknoten mit Hilfe unterschiedlicher Algorithmen und anschließendem Mehrheitsentscheid die Zuverlässigkeit der Berechnung erhöht werden (z.B. *2-aus-3-Systeme*).

Verfügbarkeitsverbund. Der Ausfall eines Rechners in einem Verfügbarkeitsverbund wird durch den weiterhin möglichen Zugriff auf einen Ersatzrechner transparent. Bei den Ersatzrechnern wird zwischen *hot standby* und *cold standby* unterschieden. Für Systeme, die beim Ausfall von Komponenten mit den verbleibenden Resourcen (mit geringeren Kapazitäten) weiterarbeiten können, ist der Ausdruck *gracefully degrading systems* entstanden.

1.2.2 Klassifikation nach geographischem Verteilungsgrad

Auch die Größenordnung der physikalischen Ausdehnung von Rechnernetzen wird oft als Kriterium für die Klassifikation verwendet. Im folgenden werden vier Klassen beschrieben. Besonders die englischen Abkürzungen werden sehr häufig verwendet.

Lokales Rechnernetz. Unter einem lokalen Rechnernetz (LAN, local area network) versteht man ein Rechnernetz, dessen physikalische Ausdehnung die Größenordnung von Räumen, Gebäuden oder des Geländes einer Institution besitzt. Die Entfernungen zwischen den einzelnen Rechnern liegen etwa zwischen 10 m und 1 km. Damit gilt beispielsweise das Netz des Fachbereichs Informatik als lokales Rechnernetz.

Regionales Rechnernetz. Erstreckt sich ein Rechnernetz über ein Gebiet von der Größe einer Stadt, eines kleinen Bundeslandes oder einer zusammenhängenden Industrieregion (z.B. das Rhein-Main-Gebiet), spricht man von einem regionalen Rechnernetz (MAN, metropolian area network). Die Entfernung der Knoten liegt in der Größenordnung von 10 km. Beispiele hierfür sind das Hamburger Hochgeschwindigkeits-Rechnernetz HHR und das Netz des Hamburger Telekommunikationsdienstleisters HanseNet.

Überregionales Rechnernetz, Weitverkehrsnetz. Bei einem überregionalen Rechnernetz oder Weitverkehrsnetz (WAN, wide area network) liegt die Knotenentfernung etwa zwischen 100 km und 1000 km. Das Gebiet hat also die Ausmaße eines Landes oder Kontinents. Das Deutsche Forschungsnetz (DFN) und das DATEX-P-Netz der Deutschen Telekom sind überregionale Rechnernetze.

Globales Rechnernetz. Umspannt das Rechnernetz den Planeten Erde, wie es beim globalen Internet (inclusive seiner Satellitenverbindungen) der Fall ist, spricht man von globalen Rechnernetzen.

Darüber hinaus sind Begriffe wie *interplanetary network*, *personal area network (PAN)* und *body area network (BAN)* entstanden, um Rechnernetze von heute noch wenig verbreiteten Größenordnungen zu beschreiben.

1.3 Topologieformen für Datenübertragungssysteme

Definition 1.9. *Unter der* physikalischen Topologie *verstehen wir die Art der Vermaschung der wesentlichen Hardwarekomponenten (z.B. der Vermittlungsrechner eines Kommunikationsnetzes). Entsprechend führen die (etablierten oder grundsätzlich möglichen) Kommunikationsbeziehungen zwischen kommunizierenden Prozessen zu einer* logischen Topologie.

Grundsätzlich ist der Topologiebegriff nahezu auf jeder Protokollschicht einer Rechnernetzarchitektur anwendbar. Beispielsweise besitzt ein Fileserver im Ethernet einen Bus als physikalische Topologie und einen Stern als logische Topologie (auf der Ebene des Dateitransfers). Abbildung 1.8 zeigt einige reine Topologieformen.

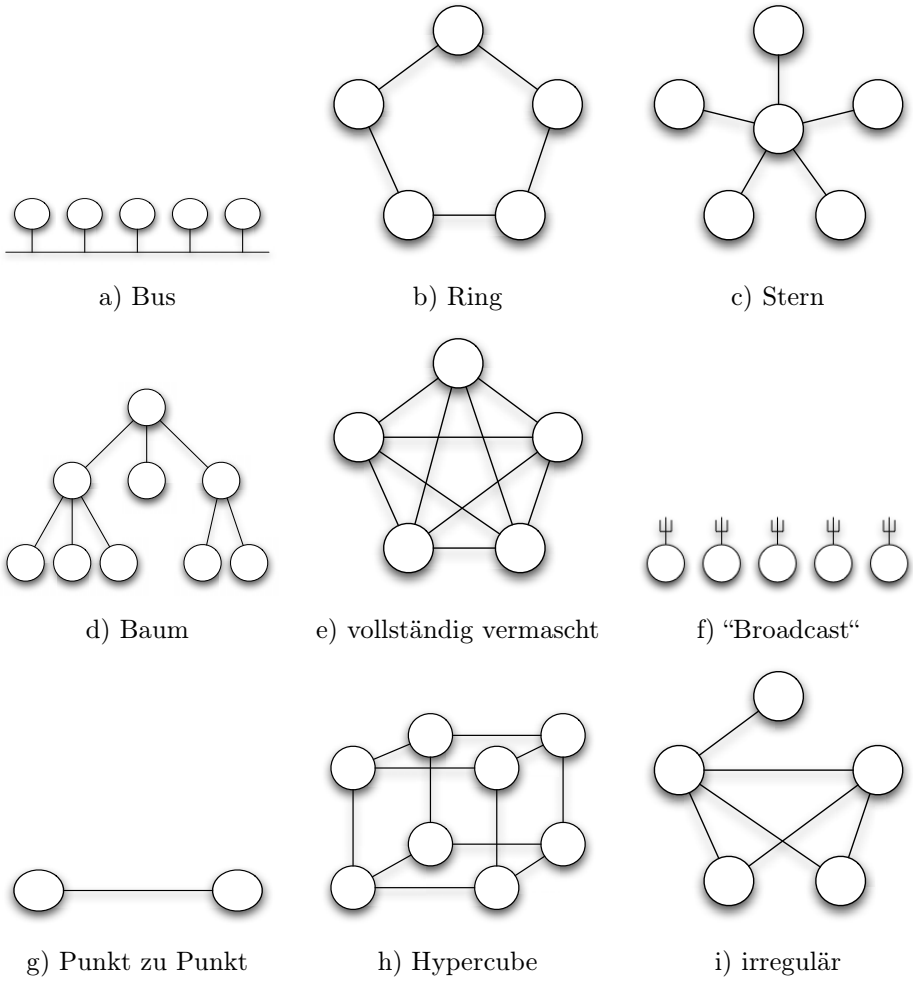


Abbildung 1.8. Reine Topologieformen

Das folgende Beispiel zeigt, dass bezüglich der Topologie unterschiedliche Interpretationen möglich sind. Gegeben ist die folgende Rechnerkonfiguration mit jeweils genau einer Leitung zwischen A_i und B.

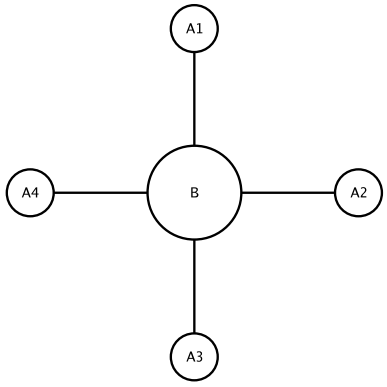


Abbildung 1.9. Gegebene Rechnerkonfiguration

Beispiel 1.10. Unterschiedliche Interpretationen sind für die gegebene Rechnerkonfiguration in Abbildung 1.9 möglich:

1. Wenn B ein Timesharing-Rechner und A_i seine Terminals sind, also keine Kommunikation zwischen den Rechnern A_i untereinander besteht, liegt die Interpretation als *Punkt-zu-*

Punkt-Verbindungen nahe.

2. Ist B ein zentraler Vermittlungsrechner (switch) und die Kommunikation erfolgt zwischen den Rechnern A_i , so ergibt sich eine *sternförmige* Topologie.

3. Simuliert der Rechner B den Rechnern A_i gegenüber einen Ring (vgl. wire center, Token-Ring LAN), so ist die Topologie aus Sicht der Rechner A_i ein *Ring*. Siehe Abbildung 1.10.

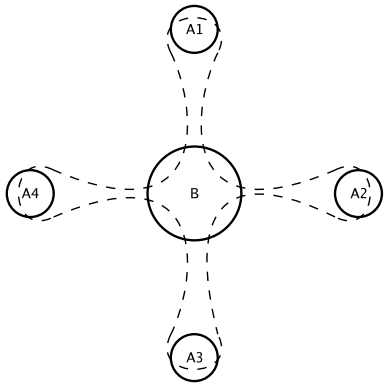


Abbildung 1.10. Interpretation als Ring

1.4 Dienste, Protokolle und andere Basiskonzepte

Definition 1.11. *Eine Menge von Konventionen, die die Kommunikation zwischen mehreren durch Botschaftenaustausch interagierenden Instanzen (z.B. kommunizierenden Prozessen) regeln, heißt* Kommunikationsprotokoll *oder kurz* Protokoll.

Der Begriff Botschaft wird hier als Synonym für Nachricht in ihrer allgemeinen, schichtenunabhängigen Bedeutung verwendet. Ein Kommunikationsprotokoll übernimmt folgende Aufgaben:

Botschaftensyntax. Festlegung der Struktur der ausgetauschten Botschaften (auch Protokolldateneinheiten)

Botschaftensemantik. Festlegung der Bedeutung der Botschaften und der möglichen Reaktionen

Timing. Festlegung der grundsätzlich möglichen, zeitlichen Abläufe von Ereignissen

Beispiele für die Aufgaben von Kommunikationsprotokollen sind die Regelung der Zugriffskontrolle für Kommunikationspartner auf ein gemeinsames Übertragungsmedium, die Organisation eines (nahezu) fehlerfreien bit-transparenten Transports von Daten über eine Leitung, die Regelung der Kommunikation zwischen Prozessen in verschiedenen Rechnern (Interprozesskommunikation) oder die Organisation eines Dateitransfers. Aus der GSS-Vorlesung sollten die Protokolle IP, TCP, HDLC, X.25, CSMA/CD u.a. bekannt sein.

1.4.1 Dienste und ihr Zusammenhang mit Protokollen

Definition 1.12. *Unter einem* Dienst (service) *verstehen wir eine implementationsunabhängige Beschreibung der Leistung, die seitens eines oder mehrerer Diensterbringer(s) einer Menge von Benutzern angeboten und auf Anforderung hin erbracht wird.*

Das Dienstkonzept spielt eine wichtige Rolle bei Rechnernetzen, Verteilten Systemen, Betriebssystemen und öffentlichen Netzbetreibern. Wir unterscheiden zwei Varianten der Diensterbringung in Rechnernetzen (die allerdings auch in Mischformen vorkommen).

1. *Abgesetzter Zugriff (remote access)* auf einen zentral erbrachten Dienst, wie z.B. in Client-Server-Beziehungen
2. *Verteilte Erbringung* eines Dienstes durch Kommunikation und Kooperation mehrerer Diensterbringer, wie z.B. in Peer-to-Peer-Beziehungen

1.4.2 Protokoll-/Diensthierarchien

Bei der verteilten Erbringung eines Dienstes S ist ein Protokoll P für die Kommunikation zwischen den beteiligten Diensterbringern erforderlich. Der Dienst S basiert dann auf dem Protokoll P. Es können unterschiedliche Protokolle als Basis für ein und denselben Dienst dienen.

Hinsichtlich der Komplexitätsreduktion, der Testbarkeit und der Standardisierungsmöglichkeiten ist eine Schichtung von Diensten sehr vorteilhaft. Abbildung 1.11 zeigt eine exemplarische, statische Protokollhierarchie.

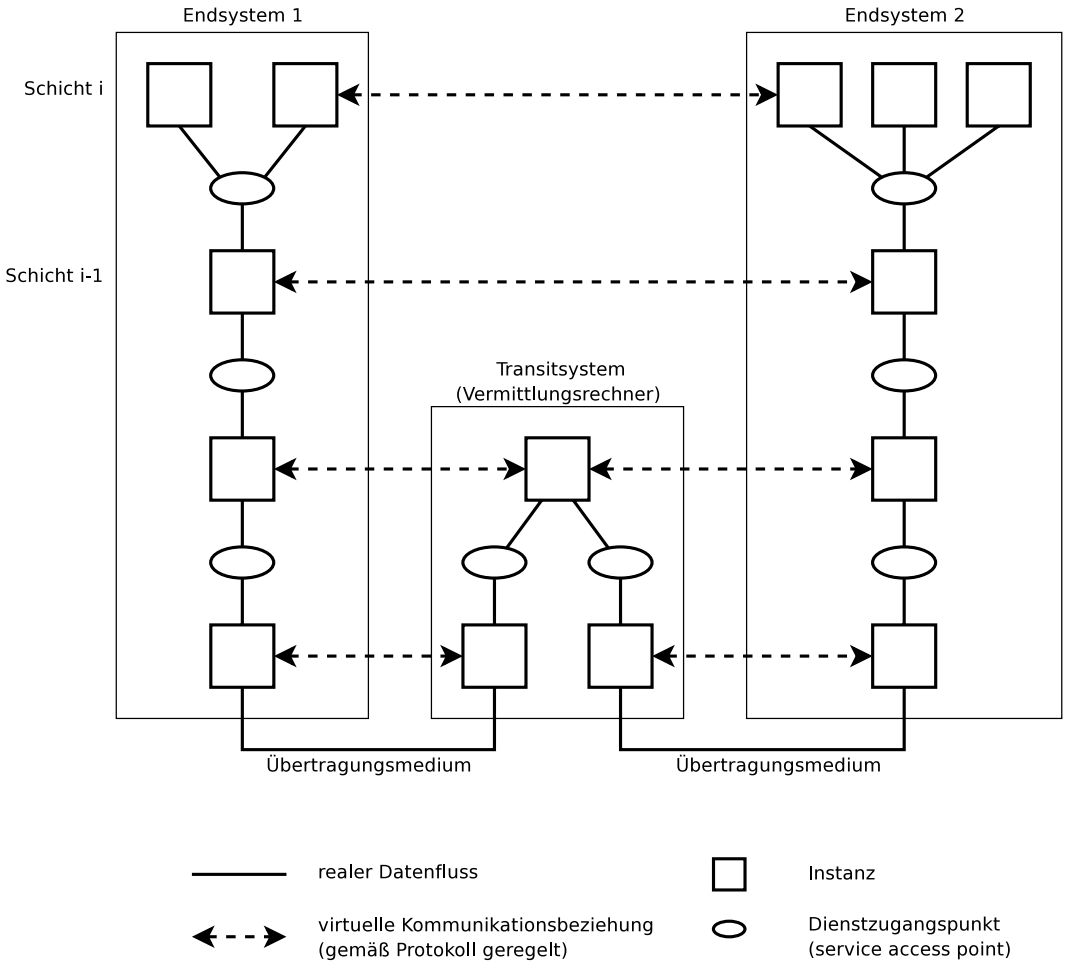


Abbildung 1.11. Exemplarische Protokollhierarchie

Neben der statischen Struktur ist auch das dynamische Verhalten darstellbar, um beispielsweise Protokollvorkommnisse, Verbindungsendpunkte (connection endpoints) oder etablierte Verbindungen zu verdeutlichen.[1.5]

1.5. vgl. T4-Vorlesung zu weiteren Details

Mit Hilfe des E-Learning-Werkzeugs *InternetExplorativ* können die dynamischen Abläufe im TCP/IP-Schichtenmodell durch benutzergesteuerte Animation dargestellt werden und lassen sich so schrittweise genauer untersuchen. Dies bezieht sich zunächst auf die Veränderung der einzelnen Datenpakete beim Durchlaufen der verschiedenen Protokollschichten vertikal.

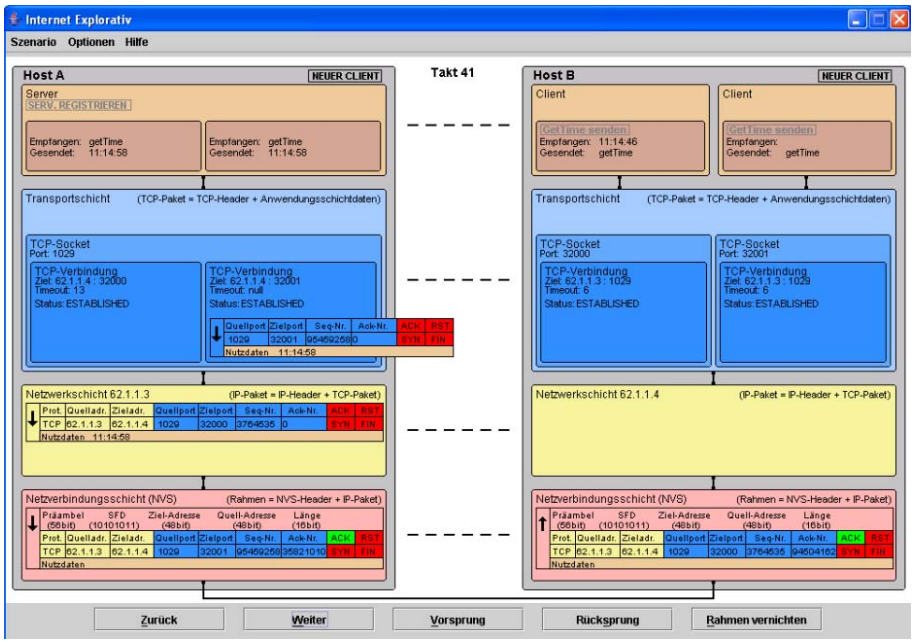


Abbildung 1.12. Screenshot des E-Learning-Werkzeugs InternetExplorativ

1.5 Chancen und Risiken globaler Vernetzung

Während die zunehmende globale Vernetzung viele Chancen bietet, entstehen auch gesellschaftliche Risiken und Probleme. Hier seien nur einige davon genannt:

- neue nützliche Anwendungen zur Unterstützung von Lehre, Arbeit, Medizin, Handel und Verkehr
- Mobilitätsunterstützung durch Mobilkommunikationsnetze
- sinkende Umweltbelastung durch globale Vernetzung
- leicht zugängliche Wissensbasen
- zeit- und ortsunabhängige Kommunikation.

Dem stehen unter anderem entgegen:

- Sicherheitsprobleme
- existenzbedrohende Abhängigkeiten von Netzinfrastrukturen
- neue rechtliche Probleme des Internet
- negative Auswirkungen auf die Persönlichkeit der Benutzer
- Verlust der Privatsphäre
- soziale Ungleichheiten (z.B. *Digital Divide*).

Kapitel 2
Grundlagen der Datenübertragung

2.1 Grundbegriffe

Definition 2.1. *Unter* Daten (data) *verstehen wir Zeichen oder kontinuierliche Funktionen, die zum Zwecke der Verarbeitung Information auf Grund bekannter oder unterstellter Abmachungen darstellen.* [2.1]

Definition 2.2. *Die physikalische Darstellung von Nachrichten oder Daten nennen wir* Signal (signal). *Diejenige Kenngröße des Signals, deren Wert oder Werteverlauf die Nachricht oder die Daten darstellen, nennen wir dabei* Signalparameter.

Digitale Daten sind durch Zeichen dargestellte Daten während analoge Daten durch kontinuierliche Funktionen dargestellt werden. Beispiele für Signalparameter sind Amplitude, Phase, Frequenz.

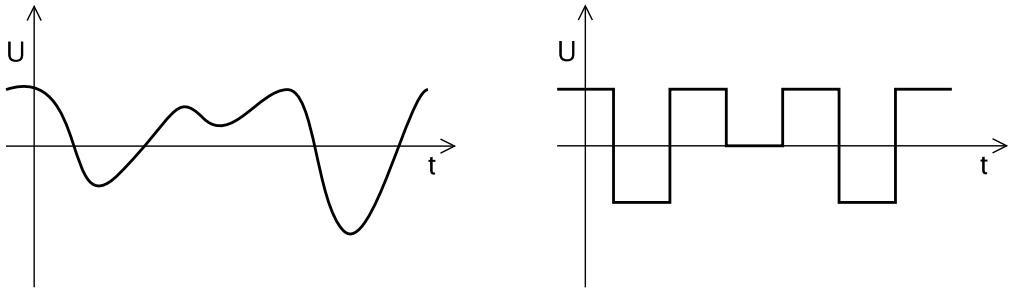


Abbildung 2.1. Signalfunktionen

Die Darstellung der Daten für den Kommunikationspartner unterscheidet sich im Allgemeinen von der Darstellung innerhalb des Übertragungsmediums. Es findet also in der Regel eine Anpassung des von der Nachrichtenquelle erzeugten *primären Signals* an die Eigenschaften des Übertragungsmediums statt. Beim Empfänger wird dann das primäre Signal zurück gewandelt (siehe Abbildung 2.2).

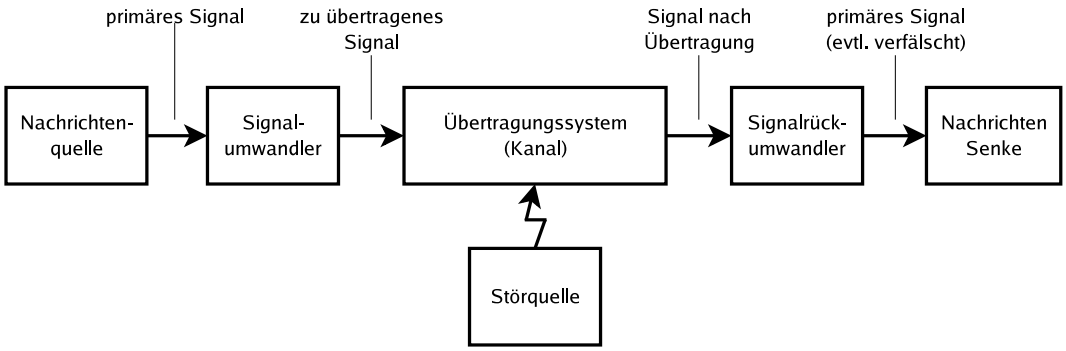


Abbildung 2.2. Verfeinerte Sicht eines Kommunikationssystems

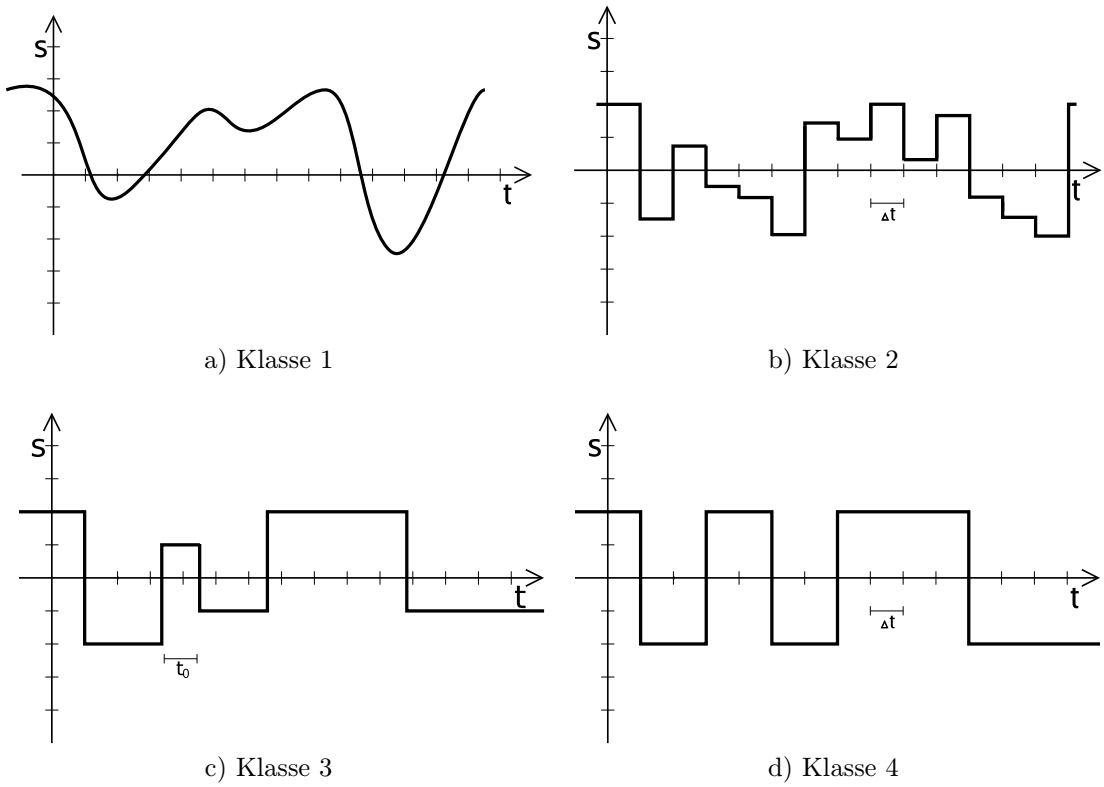
2.1. vgl. DIN 44300 und Definition 1.1

Sowohl die Zeitkoordinate als auch die Signalkoordinate[2.2] eines Signals kann diskret oder kontinuierlich sein:

- Mit einer diskrete Signalkoordinate ist gemeint, dass es eine vorgegebene abzählbare Menge M_s gibt, so dass für den Wertebereich M_σ des Signals s gilt: $M_\sigma \subset M_s$. Beim Spezialfall eines zweistufigen Signals ist $M_\sigma = M_s = \{s_1, s_2\}$.
- Eine diskrete Zeitkoordinate bedeutet, dass es eine abzählbare Menge M_t gibt, so dass für die Menge M_τ der Zeitpunkte mit einer Änderung der Signalkoordinate gilt: $M_\tau \subset M_t$. Im Spezialfall eines konstanten Zeittaktes ist dann $M_t = \{n \cdot \Delta t \,|\, n \in \mathbb{N} \wedge \Delta t > 0 \wedge \Delta t \text{ konstant}\}$.

Signale lassen sich deshalb in vier Klassen aufteilen:

1. Kontinuierliche Zeit- und Signalkoordinate (z.B. Sprache oder Musik). Diese Signale dieser Klasse nennen wir auch *analog*.
2. Diskrete Zeit- und kontinuierliche Signalkoordinate (mit dem wichtigen Spezialfall eines konstanten Zeittaktes Δt).
3. Kontinuierliche Zeit- und diskrete Signalkoordinate. In diesem Fall sind die Zeitpunkte für eine Änderung der Signalkoordinate beliebig. Für aufeinanderfolgende Änderungszeitpunkte t_α und t_β kann es aber Restriktionen des Typs $|t_\alpha - t_\beta| \geqslant t_0 > 0$ geben.
4. Diskrete Zeit- und Wertekoordinate (mit dem praktisch besonders relevanten Fall eines zweistufigen Signals mit konstantem Zeittakt). Diese Signale nennen wir auch *digital*.



a) Klasse 1 b) Klasse 2

c) Klasse 3 d) Klasse 4

Abbildung 2.3. Beispiele für die Signalklassen

Signalumwandlungen und Codierungen bei der Datenübertragung

Zum Zweck der Datenübertragung ist es oft notwendig, analoge Signale digital oder digitale Signale

2.2. Die Signalkoordinate entspricht i.a. dem Signalparameter.

analog zu übertragen. Im ersten Fall wird ein Signal der Klasse 1 schrittweise in ein Signal der Klasse 4 überführt. Durch *Quantisierung* wird der Wertebereich und durch *Abtastung* zu äquidistanten Zeitpunkten der Zeitbereich diskretisiert. Beide Reihenfolgen sind möglich. Das Signal wird also entweder zuerst in ein Klasse 3 oder in ein Klasse 2 Signal überführt.

Für die analoge Übertragung digitaler Signale gibt es unterschiedliche Ansätze. Im diesem Kapitel werden hierzu Amplituden-, Phasen- und Frequenzmodulation betrachtet. Zunächst erweitern wir hier indes unser Modell des Kommunikationssystems derart, dass je zwei Codierungen bzw. Decodierungen vorgenommen werden (Abb. 2.4).

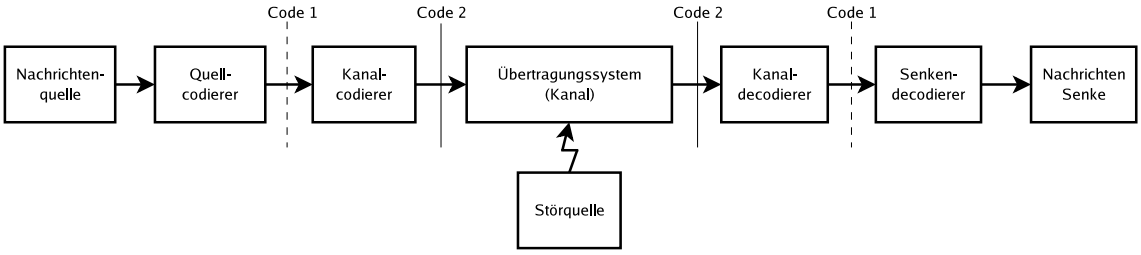


Abbildung 2.4. Doppelte Codierung bei der Datenübertragung

Der Quellcodierer bzw. Senkendecodierer kann dazu dienen,

- Geheimhaltung der übertragenen Nachrichten durch Ver- und Entschlüsselung zu gewährleisten (z.B. DES, siehe Kapitel 11),
- die Nachrichten als Folge von Binärsignalen zu übertragen (z.B. PCM, siehe Abschnitt 2.7),
- die in den Nutzdaten enthaltene Redundanz (z.B. bei Bewegtbildern, siehe Kapitel 7) zu entfernen.

Mögliche Ziele des Kanalcodierers bzw. -decodierers sind

- die Unterstützung einer empfangsseitigen Rückgewinnung des Taktes (z.B. Manchester-Codierung, vgl. Abschnitt 2.7)
- ein systematisches Hinzufügen von Redundanz zur Unterstützung einer empfangsseitigen Fehlerkorrektur und/oder Fehlererkennung (z.B. CRC, vgl. Abschnitt 2.12).

Die Abbildungen 2.5 und 2.6 stellen die Ersetzbarkeit analoger und digitaler Signale zum Zwecke der Datenübertragung dar. Abbildung 2.5 zeigt dabei, wie durch Quantisierung und Abtastung (in beliebiger Reihenfolge) ein Klasse 1-Signal in ein Klasse 4-Signal überführt werden kann. Umgekehrt zeigt Abbildung 2.6 wie ein digitales Signal bei Amplitudenmodulation analog übertragen wird.

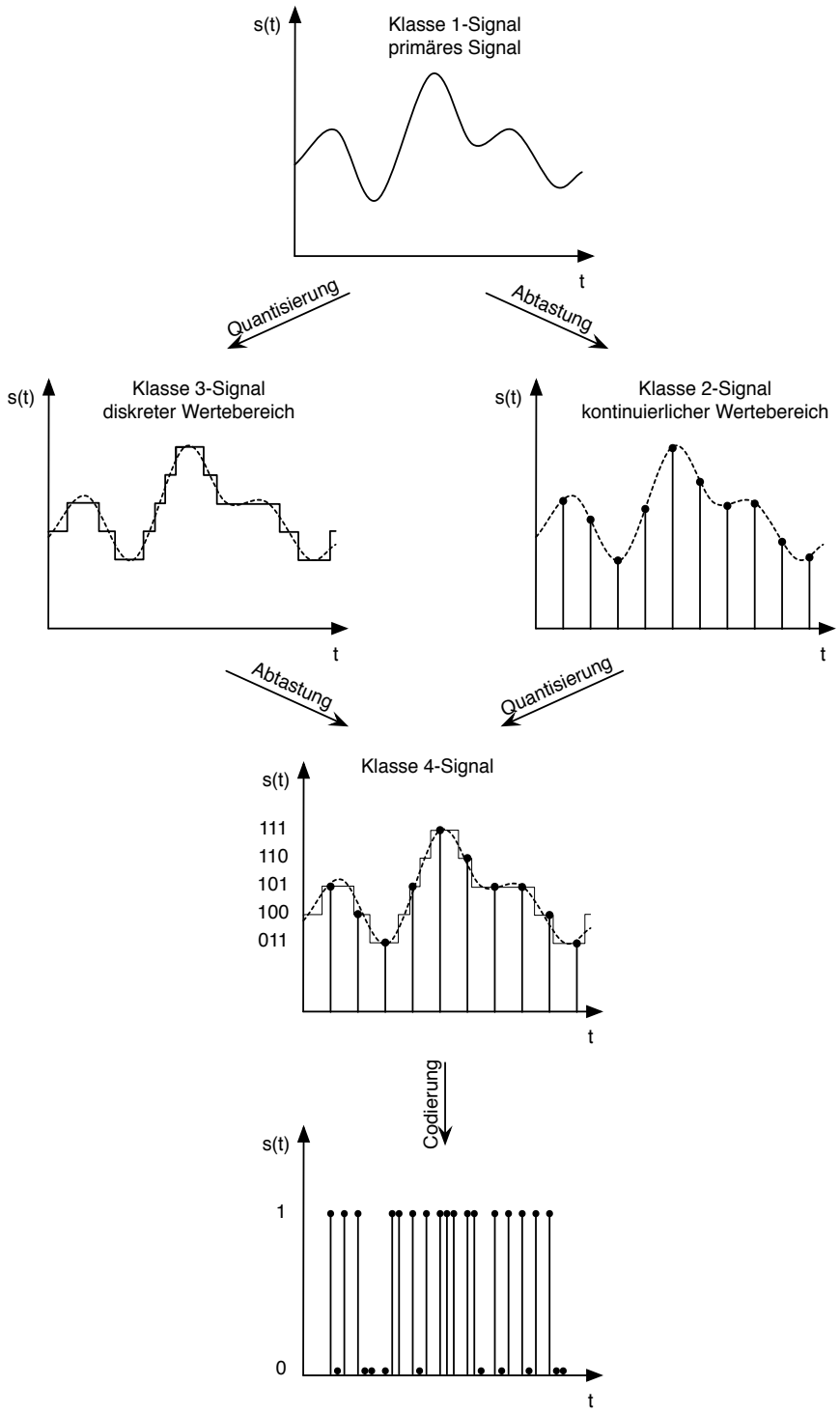


Abbildung 2.5. Überführung eines analogen in ein digitales Signal

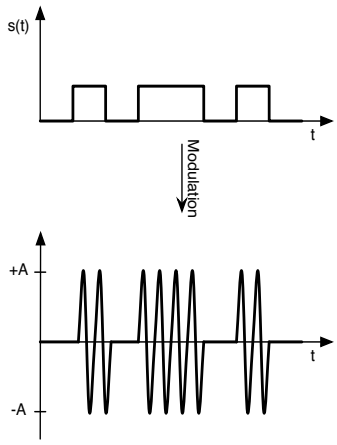
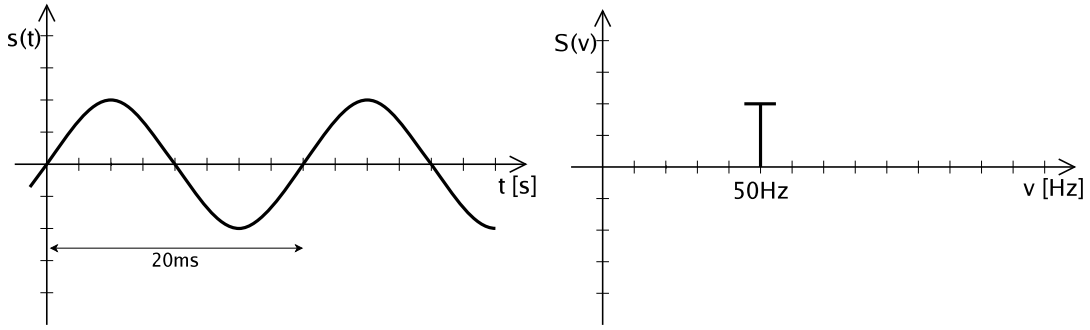


Abbildung 2.6. Analoge Übertragung digitaler Signale

Signaldarstellung im Zeit- und Frequenzbereich

Eine Signalfunktion $s(t)$ beschreibt den zeitlichen Verlauf des Signalparameters. Sie lässt sich mit Hilfe der Fouriertransformation in eine Spektralfunktion $S(\omega)$ transformieren. Die Spektralfunktion beschreibt die Verteilung der Signalenergie im Frequenzbereich (spektrale Energieverteilung). Abbildung 2.7 zeigt eine Sinusschwingung im Zeit- und Frequenzbereich. Aus typografischen Gründen wird in diesem Skript in einigen Abbildungen der Buchstabe v statt des griechischen ν für Frequenzen verwendet.



a) Signalfunktion b) Spektralfunktion

Abbildung 2.7. Sinusschwingung im Zeit- und im Frequenzbereich

Der mathematische Zusammenhang zwischen Signalfunktion $s(t)$ und Spektralfunktion $S(\omega)$ (mit der imaginären Einheit i, $i^2 = -1$) ist:

$$S(\omega) = \int_{-\infty}^{\infty} s(t) \cdot e^{-i\omega t}\,\mathrm{d}t$$

$$s(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} S(\omega) \cdot e^{i\omega t}\,\mathrm{d}\omega$$

Einen Sonderfall stellen periodische Signale dar. Ihre Transformierten lassen sich (einfacher) über die Fourierreihenentwicklung bestimmen. Ihre Spektralfunktion hat immer ein Linienspektrum (siehe Abbildung 2.7 b). Die Signalfunktion lässt sich also als additive Überlagerung von Sinus- und

Cosinusschwinungen (unterschiedlicher Amplitude) darstellen, wobei die Frequenzen ganzzahlige Vielfache der *Kreisfrequenz* $\omega = \frac{2\pi}{T} = 2\pi\nu$ sind.

$$s(t) = \frac{1}{2} \cdot c + \sum_{n=1}^{\infty} a_n \cdot \sin(n\omega t) + \sum_{n=1}^{\infty} b_n \cdot \cos(n\omega t)$$

Die Koeffizienten c, a_n und b_n – also die oben angesprochen Amplituden der Sinus- und Cosinusschwingungen – ergeben sich wie folgt. Sei $k \in \mathbb{N}$ eine beliebige, feste natürliche Zahl. Wir betrachten das Produkt $\sin(k\omega t) \cdot s(t)$. Durch Einsetzen obiger Gleichung und Ausmultiplizieren ergibt sich:

$$\sin(k\omega t) \cdot s(t) = \frac{1}{2} \cdot c \cdot \sin(k\omega t) + \sum_{n=1}^{\infty} a_n \cdot \sin(n\omega t) \cdot \sin(k\omega t) + \sum_{n=1}^{\infty} b_n \cdot \cos(n\omega t) \cdot \sin(k\omega t)$$

Weiter betrachten wir das Integral des Produktes in den Grenzen 0 und T.

$$\begin{aligned} \int_0^T \sin(k\omega t) \cdot s(t)\,\mathrm{d}t &= -\frac{c}{2} \cdot \frac{\cos(k\omega t)}{k\omega}\Bigg|_0^T \\ &+ \sum_{n=1}^{\infty} \int_0^T a_n \cdot \sin(n\omega t) \cdot \sin(k\omega t) \\ &+ \sum_{n=1}^{\infty} \int_0^T b_n \cdot \cos(n\omega t) \cdot \sin(k\omega t) \end{aligned}$$

Dieser Ausdruck lässt sich stark vereinfachen. Der erste und der letzte Summand ist immer 0. Für den mittleren Summanden gilt:

$$\int_0^T a_n \cdot \sin(n\omega t) \cdot \sin(k\omega t) = a_n \cdot \int_0^T \sin(n\omega t) \cdot \sin(k\omega t) = \begin{cases} a_n \cdot 0 & k \neq n \\ a_n \cdot \frac{T}{2} & k = n \end{cases}$$

Aus diesem Grund kann auch das Summenzeichen entfallen, da nur ein Term der Summe von Null verschieden ist. Es ist also

$$\int_0^T \sin(k\omega t) \cdot s(t)\,\mathrm{d}t = a_k \cdot \frac{T}{2}$$

Stellt man diese Formel nach a_k bzw. a_n um, so ergibt sich die zweite der folgenden Formeln für die Berechnung der Koeffizienten. Die Koeffizienten b_n lassen sich analog herleiten. Der Koeffizient c stellt den Spezialfall b_n mit $n = 0$ dar.

$$\begin{aligned} c &= \frac{2}{T} \cdot \int_0^T s(t)\,\mathrm{d}t \\ a_n &= \frac{2}{T} \cdot \int_0^T s(t) \cdot \sin(n\omega t)\,\mathrm{d}t \\ b_n &= \frac{2}{T} \cdot \int_0^T s(t) \cdot \cos(n\omega t)\,\mathrm{d}t \end{aligned}$$

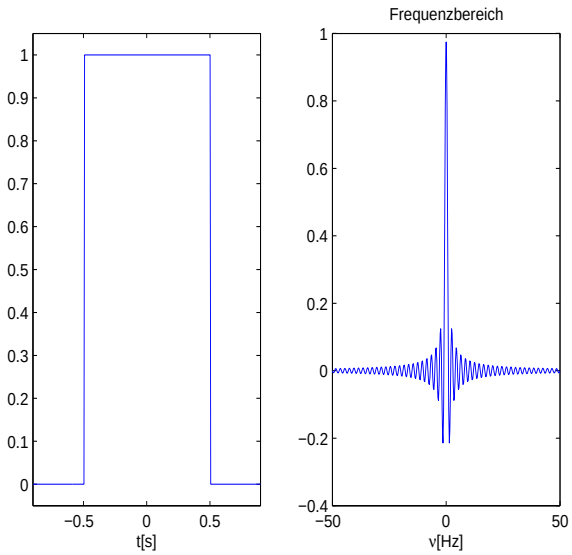


Abbildung 2.8. Reckteckimpuls im Zeit- und Frequenzbereich

Die Spektralfunktion des Rechteckimpulses (siehe Abb. 2.8) zeigt, dass es Signale gibt, deren exakte Übertragung ein unendliches Frequenzspektrum benötigt. Das nutzbare Frequenzspektrum eines Übertragungsmediums ist aber stets begrenzt. Daher definieren wir die Größe des nutzbaren Bereichs.

Definition 2.3. *Die* Frequenzbandbreite *oder kurz* Bandbreite *eines physikalischen Übertragungsmediums (in Hz) ist die Differenz zwischen der höchsten und der niedrigsten sinnvoll übertragbaren Frequenz. Diese Frequenzen heißen* obere Grenzfrequenz *bzw.* untere Grenzfrequenz. *Der Bereich zwischen oberer und unterer Grenzfrequenz heißt Durchlassbereich, der Bereich unterhalb der unteren und oberhalb der oberen Grenzfrequenz heißt* Sperrbereich.

Beispielsweise kann durch Frequenzmultiplexen die Bandbreite eines einzelnen Nutzers stark begrenzt sein. Liegt die untere Grenzfrequenz dicht bei Null, ist Basisbandübertragung (s.u.) möglich. Abbildung 2.9 veranschaulicht Definition 2.3.

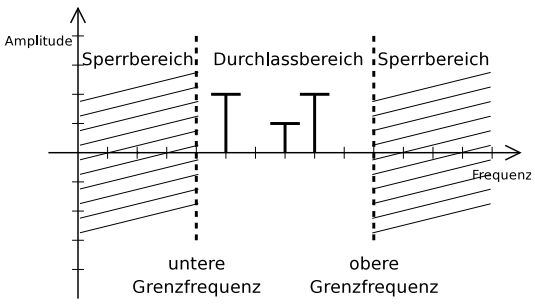


Abbildung 2.9. Konsequenzen einer begrenzten Bandbreite

Implikation einer endlichen Bandbreite für die Qualität der Signalapproximation

Im Folgenden wird als primäres Signal ein ASCII-codiertes `b`, also die Bitsequenz `01100010` angenommen. Dieses Signal ist endlich und kann als periodisch aufgefasst werden. Es kann also als

unendliche Fourierreihe dargestellt werden.

$$s(t) = \frac{1}{2}c + \sum_{n=1}^{\infty} a_n \cdot \sin(n\omega t) + \sum_{n=1}^{\infty} b_n \cdot \cos(n\omega t)$$

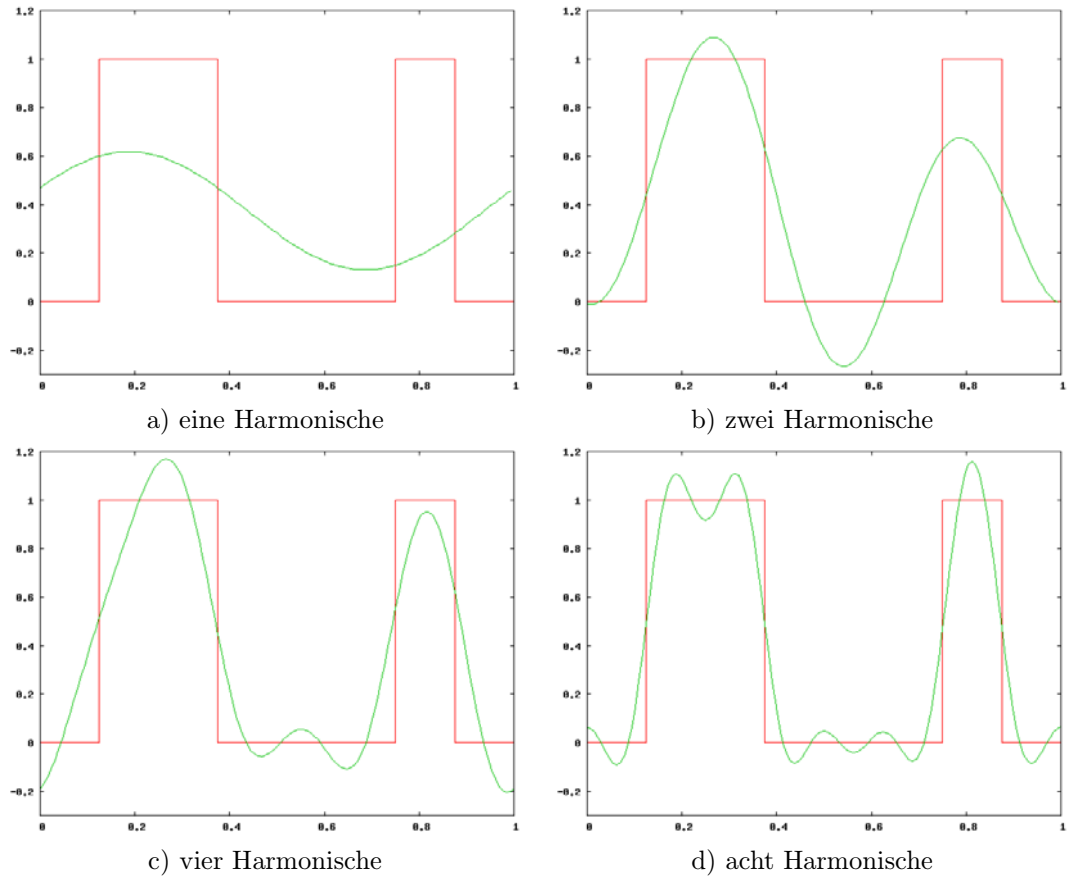
Man beachte, dass im allgemeinen (und auch in diesem Fall) unendlich hohe Frequenzen benötigt werden, um das Signal so darzustellen. Kanäle in der Praxis haben jedoch immer eine begrenzte Bandbreite, so dass eine bestimmte Maximalfrequenz vorgegeben ist. Das Signal kann also in der Praxis nur approximiert werden. Dabei werden nur endlich viele Vielfache der Kreisfrequenz verwendet. Diese nennen wir Harmonische.

$$s(t) = \frac{1}{2}c + \sum_{n=1}^{k} a_n \cdot \sin(n\omega t) + \sum_{n=1}^{k} b_n \cdot \cos(n\omega t)$$

Aus den oben hergeleiteten Formeln ergeben sich die folgenden Koeffizienten.

$$\begin{aligned} a_n &= \frac{1}{\pi n} \cdot \left[\cos\frac{\pi n}{4} - \cos\frac{3\pi n}{4} + \cos\frac{6\pi n}{4} - \cos\frac{7\pi n}{4} \right] \\ b_n &= \frac{1}{\pi n} \cdot \left[-\sin\frac{\pi n}{4} + \sin\frac{3\pi n}{4} - \sin\frac{6\pi n}{4} + \sin\frac{7\pi n}{4} \right] \\ c &= \frac{2}{T} \cdot \frac{3}{8} \cdot T = \frac{3}{4} \end{aligned}$$

Abbildung 2.10 zeigt, dass die Qualität der Signalapproximation stark von der Anzahl der übertragenen Harmonischen abhängig ist. Andererseits gilt, dass die Anzahl der übertragbaren Harmonischen von der Bandbreite abhängt (Tabelle 2.1).



a) eine Harmonische b) zwei Harmonische

c) vier Harmonische d) acht Harmonische

Abbildung 2.10. Approximation eines ASCII-codierten `b` bei endlicher Bandbreite

Datenrate [Bit/s]	benötigte Zeit pro Bit [ms]	benötigte Zeit pro Byte [ms]	Harmonische erste [Hz]	Harmonische übertragbar
300	3,33	26,67	37,5	80
1200	0,83	6,67	150	20
2400	0,42	3,33	300	10
4800	0,21	1,67	600	5
9600	0,1	0,83	1200	2
19200	0,05	0,42	2400	1
38400	0,026	0,21	4800	0

Tabelle 2.1. Abhängigkeit der Anzahl übertragbarer Harmonischer von der Datenrate (bei 3000 Hz Bandbreite)

Leistungs- und Zuverlässigkeitskenngrößen einer Datenübertragung

Definition 2.4. *Die Dauer T_S der Zeitintervalle, in denen ein Signal konstant ist, also Δt bei Signalen der Klasse 2 und 4 und die Minimaldauer t_0 bei Signalen der Klasse 3 nennen wir* Schrittdauer. *Der Kehrwert der Schrittdauer $V_S = \frac{1}{T_S}$ heißt* Schrittgeschwindigkeit. *Sie wird in der Einheit Baud*[2.3] *(abgekürzt Bd) angegeben.*

Definition 2.5. *Werden informationstragende Signalelemente zeitlich nacheinander übertragen, sprechen wir von* serieller Datenübertragung. *Werden m solcher Schritte gleichzeitig übertragen (etwa über vier getrennte Leitungen), sprechen wir von* paralleler Datenübertragung.

In der Praxis wird bei paralleler Datenübertragung für den Wert m meist eine Zweierpotenz (z.B. 8, 16, 32) gewählt. Abbildung 2.11 veranschaulicht eine parallele Datenübertragung mit $m = 4$.

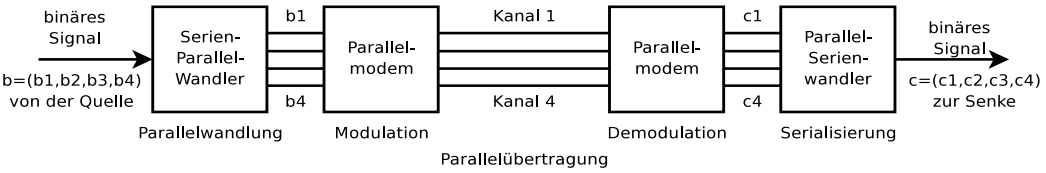


Abbildung 2.11. Parallele Datenübertragung mit $m = 4$

Grundlagen der Informationstheorie

Die Informationstheorie stellt einen Zweig der statistischen Kommunikationstheorie (und damit auch der Wahrscheinlichkeitstheorie) dar. Sie untersucht den Informationsgehalt von Nachrichten bzw. physikalischen Beobachtungen sowie den Zusammenhang zwischen Informationsgehalt und Übertragung dieser Information von einem Ort zum anderen. Der Begriff Information wird dabei über die Wahrscheinlichkeit für das Auftreten von Symbolen definiert. Der semantische und pragmatische Aspekt von Nachrichten bleibt unberücksichtigt. Im folgenden soll ein kurzes Resümee von einigen Inhalten des B.Sc.-Pflichtmoduls "Rechnerstrukturen" gegeben werden.

Wir betrachten eine Menge von Zeichen Z_i mit $i \epsilon \mathbb{N}$ und Auftrittswahrscheinlichkeiten p_i für ein Zeichen Z_i.

Definition 2.6. *Der* Informationsgehalt h_i *eines Zeichens in bit beträgt*

$$h_i = -\operatorname{ld}(p_i) = \operatorname{ld}\left(\frac{1}{p_i}\right)$$

Der mittlere Informationsgehalt h *einer Menge von n Zeichen in bit, auch* Entropie *genannt, beträgt*

$$h = \sum_{i=1}^{n} p_i \cdot h_i = \sum_{i=1}^{n} p_i \cdot (-\operatorname{ld}(p_i))$$

2.3. sprich [Bo:d]

Sind alle n Auftrittswahrscheinlichkeiten der Zeichen gleich (also folglich $p_i = 1/n$), so ist der mittlere Informationsgehalt maximal.

Definition 2.7. *Der* Entscheidungsgehalt *ist die Informationsmenge, die nötig ist, um ein Zeichen aus einem Zeichenvorrat von n gleich wahrscheinlichen Zeichen mit ja/nein-Entscheidungen auszuwählen. Es gilt:*

$$H_0 \;=\; \mathrm{ld}(n)\ [\mathrm{bit}]$$

Die Differenz aus Entscheidungsgehalt und Entropie wird auch als absolute Redundanz R bezeichnet. Die relative Redundanz r ergibt sich aus dem Quotient von R und H_0.

$$R = H_0 - h$$

$$r = \frac{R}{H_0} = \frac{H_0 - h}{H_0}$$

Besteht ein Zeichenvorrat aus nur zwei Zeichen, beispielsweise 0 und 1, so ist der Entscheidungsgehalt offensichtlich $H_0 = \mathrm{ld}(2) = 1$. Bei einem Zeichenvorrat von acht Zeichen ist $H_0 = \mathrm{ld}(8) = \mathrm{ld}(2^3) = 3$.

Definition 2.8. *Die* Übertragungsgeschwindigkeit *oder* Datenrate v_D *von Binärentscheidungen bei gegebener Schrittgeschwindigkeit v_S und Entscheidungsgehalt H_0 pro Schritt ergibt sich bei serieller Datenübertragung und n möglichen relevanten Signalwerten pro Schritt zu*

$$v_D = v_S \cdot H_0 = v_S \cdot \mathrm{ld}\, n$$

Die Einheit für die Übertragungsgeschwindigkeit (bzw. Datenrate) ist damit:

$$\left[\frac{\text{Schritte}}{\text{s}}\right] \cdot \left[\frac{\text{bit}}{\text{Schritt}}\right] = \left[\frac{\text{bit}}{\text{s}}\right]$$

Satz 2.9. *Die Übertragungsgeschwindigkeit oder Datenrate v_D von Binärentscheidungen bei paralleler Datenübertragung über m Kanäle mit gegebenen Schrittgeschwindigkeiten v_S^i und n^i möglichen relevanten Signalwerten pro Schritt in Kanal i ergibt sich durch:*

$$v_D = \sum_{i=1}^{m} v_S^i \cdot \mathrm{ld}\, n^i \left[\frac{\text{bit}}{\text{s}}\right]$$

Für den Sonderfall gleicher Schrittgeschwindigkeiten und möglicher relevanter Signalwerte in allen Kanälen ergibt sich

$$v_D = m \cdot v_S \cdot \mathrm{ld}\, n$$

Codes und Codierung

Codes werden unter anderem zur Verschlüsselung von Daten, zur rechnerinternen Repräsentation von Symbolen, zur digitalen Repräsentation analoger Daten und zur Komprimierung großer Datenmengen verwendet.

Definition 2.10. *Sind Z_1 und Z_2 Zeichenvorräte, so nennen wir eine Abbildung $f\colon Z_1 \to Z_2$ eine* Codierungsvorschrift *oder kurz einen* Code.

Teilweise wird auch die Bildmenge $f(Z_1)$ als Code bezeichnet. Die Abbildung muss dabei weder injektiv noch surjektiv sein.

Beispiel 2.11. Sei $Z_1 = \{\text{Hund}, \text{Katze}, \text{Goldfisch}\}$ und $Z_2 = \{00, 01, 10, 11\}$ so ist die Abbildung f mit

$$f(\text{Hund}) = 01, \quad f(\text{Katze}) = 10, \quad f(\text{Goldfisch}) = 11$$

ein Code bzw. eine Codierungsvorschrift.

Ist $Z = \{Z_1, Z_2, ..., Z_n\}$ eine zu codierende Zeichenmenge mit Auftrittswahrscheinlichkeiten p_i und bezeichnet $C(z)$ das zu z gehörende Codewort sowie $l(z)$ die Länge des Codewortes $C(z)$, so lässt sich folgende Eigenschaften für einen Code definieren.

Definition 2.12. *Die* mittlere Codewortlänge *$L(C)$ eines Codes C ist* $\sum_{i=1...n} l(Z_i) \cdot p_i$. *Die* Redundanz *$R_C$ des Codes ist definiert als $R_C = L(C) - h$, wobei h die Entropie bezeichnet.*

Dauer von Datenübertragungsvorgängen

Sei $\tau_I(d)$ der Zeitpunkt, zu dem eine Dateneinheit d über eine Schnittstelle I übergeben wird. Sofern die Übergabezeit nicht vernachlässigbar ist, wählt man entweder den Beginn oder das Ende der Übergabe. Wir können nun Verweil- bzw. Verzögerungszeiten definieren:

Definition 2.13. *Die* Verweilzeit *oder* Verzögerungszeit (delay) *einer Dateneinheit*

- *innerhalb einer Schicht auf Sendeseite A mit Schnittstellen I_1 "nach oben" und I_2 "nach unten" ist $\tau^A_{I_2}(d) - \tau^A_{I_1}(d)$.*
- *innerhalb einer Schicht auf Empfangsseite B mit Schnittstellen I_1 "nach oben" und I_2 "nach unten" ist $\tau^B_{I_1}(d) - \tau^B_{I_2}(d)$.*
- *innerhalb sämtlicher Schichten unterhalb der Schnittstelle I_1 ist $\tau^B_{I_1}(d) - \tau^A_{I_1}(d)$.*

Die Verzögerungsschwankung (delay jitter) *für eine Folge übertragener Dateneinheiten ist ein Maß für die Variation der Verzögerungszeiten (s.a. Echtzeitkommunikation).*[2.4]

Sofern sich innerhalb einer Schicht, die Verweilzeit in eine Warte- und eine Bedienphase unterteilen lässt, nennen wir die Dauer dieser Phasen Wartezeit (waiting time) *und* Bedienzeit (service time).

Zuverlässigkeit einer Datenübertragung

Wird ein sinusförmiges Signal s_1 über ein reales Übertragungsmedium übertragen, so unterscheidet sich das empfangene Signal s_2 in der Regel von s_1. So werden beispielsweise elektrische Signale bei zunehmender Länge der Leitung schwächer (Dämpfung). Dazu kommen Echos, Frequenzverwerfungen und Phasenschwankungen.

Wir sprechen von einer *verzerrungsfreien Datenübertragung*, wenn die folgenden Bedingungen erfüllt sind:

1. $\frac{s_2}{s_1}$ ist zeit- und amplitudenunabhängig, d.h. die Dämpfung des Signals ist zu jeder Zeit und bei jeder Amplitude dieselbe.
2. $\left|\frac{s_2}{s_1}\right|$ ist frequenzunabhängig, d.h. unterschiedliche Frequenzen erfahren dieselbe Dämpfung.
3. $\phi = \phi_2 - \phi_1$ wächst proportional zur Frequenz, d.h. die Phasenlaufzeit ist frequenzunabhängig.

Gilt nur die dritte Bedingung nicht, sprechen wir von *Laufzeitverzerrung*. Gilt nur die zweite Bedingung nicht, sprechen wir von *Dämpfungsverzerrung*. Beide Arten von Verzerrungen sind lineare Verzerrungen.

Begriffe zur Sicherheit einer Datenübertragung

Definition 2.14. *Ein* Fehler (error) *ist eine Abweichung des empfangenen Zeichens (bzw. der empfangenen Zeichenfolge) vom gesendeten Zeichen (bzw. von der gesendeten Zeichenfolge).*

2.4. Eine einheitliche, präzise Definition hat sich bislang nicht etabliert.

Ein Bitfehler *ist eine Abweichung eines empfangenen Bit vom entsprechenden gesendeten Bit. Analog werden Zeichen- und Blockfehler definiert.*

Unter einem Block verstehen wir hier eine zum Zwecke der Datenübertragung – insbesondere zur Fehlerüberwachung – zu einer logischen Einheit zusammengefasste, begrenzte Anzahl von Bits oder Zeichen.

Definition 2.15. *Ist während einer Messdauer T die Anzahl der fehlerhaft empfangenen Bits b_{error} und die Gesamtanzahl empfangener Bits b_{total}, dann ist die* Bitfehlerhäufigkeit *während der Dauer T das folgende Verhältnis:*

$$\mathrm{FH}_{\mathrm{Bit}}(T) = \frac{b_{\text{error}}}{b_{\text{total}}}$$

Die Bitfehlerwahrscheinlichkeit *ist die Wahrscheinlichkeit für das Auftreten eines Bitfehlers und ist gegeben durch:*

$$\mathrm{FW}_{\mathrm{Bit}}(T) = \lim_{T \to \infty} \mathrm{FH}_{\mathrm{Bit}}(T)$$

Der Kehrwert der Bitfehlerwahrscheinlichkeit heißt Bitfehlersicherheit*. Entsprechend lassen sich Zeichen- und Blockfehlerwahrscheinlichkeiten (und -sicherheiten) definieren. Wird die Fehlerwahrscheinlichkeit durch geeignete Schutzmaßnahmen verringert, so nennt man die dann verbleibende Fehlerwahrscheinlichkeit auch* Restfehlerwahrscheinlichkeit*.*

Die Bestimmung der Bitfehlerwahrscheinlichkeit für existierende Übertragungsmedien ist im allgemeinen schwierig, da die gemessenen Werte für Bitfehlerhäufigkeiten sehr stark vom gewählten Beobachtungsintervall abhängen können. Überdies sind Bitfehler in der Regel keineswegs stochastisch unabhängig, sondern treten typischerweise temporär gehäuft als sog. "Fehlerbüschel" (bursts) auf. Es ist offensichtlich, dass die mittlere Länge der Fehlerbüschel von der gewählten Datenrate abhängt, da während externer Störeinflüsse für eine gewisse Zeitdauer bei hoher Datenrate mehr Bits in diesem Intervall übertragen werden.

2.2 Elektrische Signalübertragung

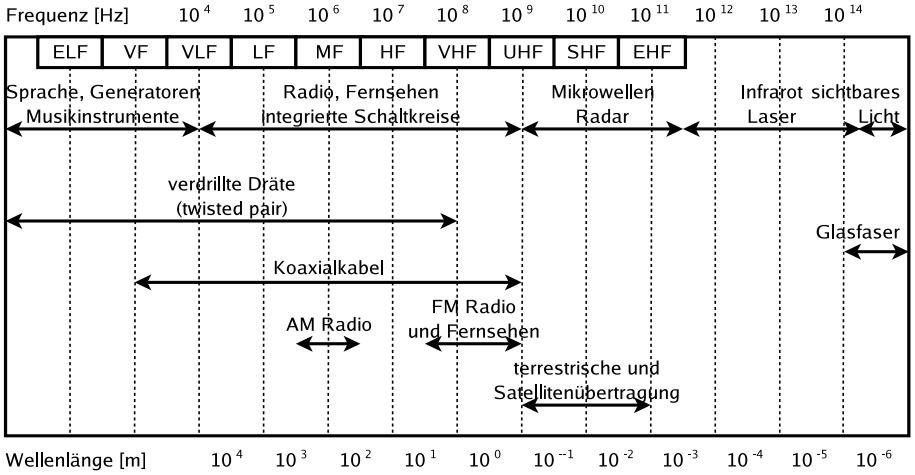


Abbildung 2.12. Nutzung der Frequenzbänder

Abbildung 2.12 zeigt die Nutzung des Frequenzspektrums elektromagnetischer Wellen. Daraus ergibt sich folgende Klassifikation physikalischer Medien zur Signalübertragung:

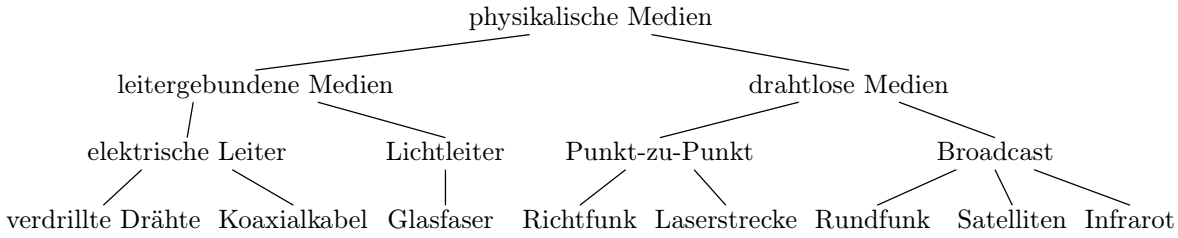


Abbildung 2.13 zeigt schematisch den Aufbau unterschiedlicher elektrischer Leitungen. Die billigste davon ist die Kupferdoppelader (a und b). Die Störanfälligkeit lässt sich reduzieren, wenn die beiden Adern verdrillt (engl.: twisted pair) werden (c). Sie kann frei verlegt werden oder mit anderen Paaren in einen schützenden und isolierenden Kunststoffmantel gebündelt werden (d). Eine weitere Verbesserung läßt sich durch Abschirmung erreichen (e). Die Bandbreite der verdrillten Kupferader liegt im KHz- bis MHz-Bereich und ist damit relativ gering. Koaxialkabel (f) sind noch relativ preiswert und besitzen eine weit größere Bandbreite (vgl. auch Tabelle 2.12)

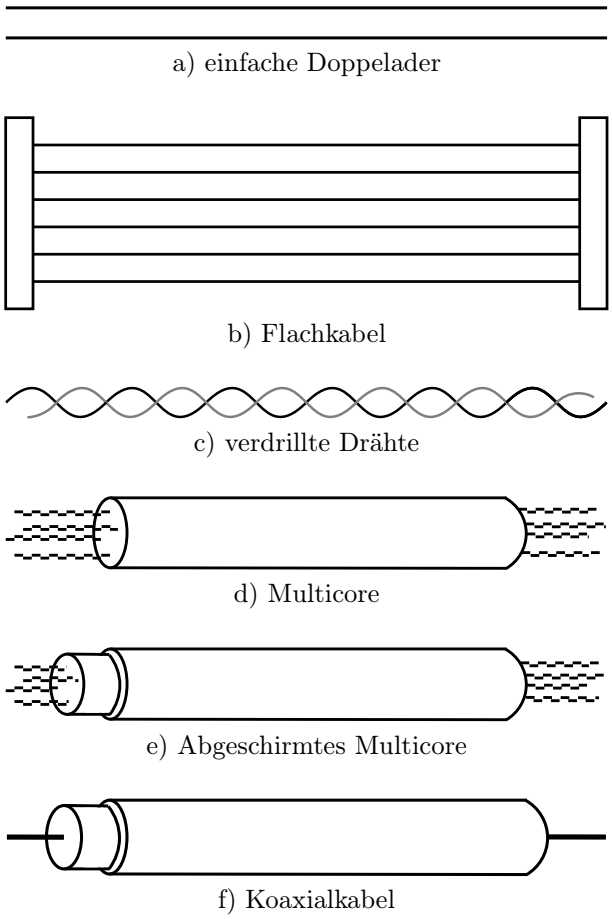


Abbildung 2.13. Unterschiedliche elektrische Leitungen

Charakteristika homogener Leitungen

Unter einer homogenen Leitung verstehen wir eine Leitung, deren elektrische Eigenschaften längs der gesamten Strecke konstant ist. Die Eigenschaften, die die Übertragung elektrischer Signale beeinflussen, sind

- Widerstand R
- Induktivität L

- Ableitung G
- Kapazität C

Die entsprechenden Werte pro Längeneinheit (oft Kilometer) heißen *Leistungskonstanten* $R´$, $L´$, $G´$ sowie $C´$. Der Widerstandsbelag $R´$ wächst mit zunehmender Frequenz, während der Induktivitätsbelag bzw. Kapazitätsbelag ($L´$ bzw. $C´$) kaum frequenzabhängig sind. Der Ableitungsbelag $G´$ ist bei Wechselstrom größer als bei Gleichstrom.

2.3 Optische Signalübertragung

Bei der optischen Signalübertragung dient Licht als elektromagnetische Welle. Wir unterteilen vier Bereiche (mit den angegebenen Wellenlängen):

- fernes Infrarot (ca. 0,05 mm bis 1 mm)
- nahes Infrarot (ca. 1 μm bis 50 μm)
- sichtbares Licht (ca. 400 nm bis 800 nm)
- Ultraviolett (ca. 1 nm bis 100 nm)

Als Lichtleiter dienen Glasfasern, als Lichtquelle Leuchtdioden (LED[2.5]) oder Laserdioden (ILD[2.6]), als Sensoren Photo- bzw. PIN-Dioden. In Abb. 2.14 ist der grundsätzliche Aufbau bei der Datenübertragung mit Hilfe von Lichtleitern dargestellt.

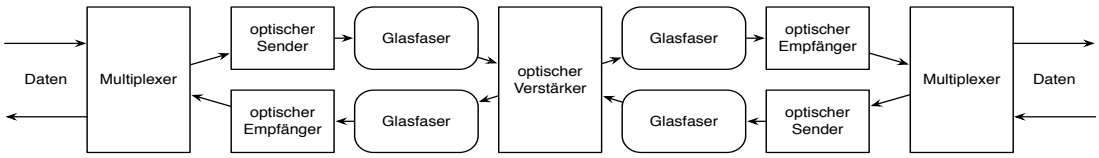


Abbildung 2.14. Aufbau eines Lichtleiters

Beurteilungskriterien für Lichtleiter

Zur Beurteilung von Lichtleitern hinsichtlich der Signalübertragung werden zwei Maße herangezogen. Die *Signaldämpfung* als Maß für die Abschwächung der übertragenen Lichtimpulse und die *Dispersion* als Maß für die Deformation der übertragenen Lichtimpulse.

Die Signaldämpfung ist abhängig von der Durchlässigkeit des verwendeten Materials und der Präzision der verwendeten Steckverbindungen. Dispersion kann aus unterschiedlichen Signallaufzeiten resultieren. Diese sind bei Lasern kleiner als bei Leuchtdioden.

Typen von Lichtleitern

Als Material für die Lichtleiter dienen im Inneren Glas höchster Reinheit und als Hülle wiederum Glas oder ein Kunststoff wie z.B. Teflon.

single-mode step-index. Dieser Typ (auch Monomode-Faser genannt) besitzt einen konstanten Brechungsindex auf seinem gesamten Querschnitt. Der Durchmesser des Kerns ist klein und das Material muss sehr rein sein. Die Herstellung ist deshalb entsprechend aufwändig. Die Kopplung von Lichtquelle (nur Laser) und Glasfaser ist ebenfalls aufwändig. Die Dispersion ist geringer als bei den anderen Typen.

2.5. light emitting diode

2.6. injection laser diode

multimode step-index. Bei diesem Typ besitzen die Fasern bzgl. des Brechungsindex ein Stufenprofil, also eine diskrete Abnahme des Brechungsindex nach außen. Der Kern besteht aus konzentrisch angeordneten Glasschichten. Die Dispersion ist beträchtlich, dafür bietet dieser Typ den größten tolerierbaren Einspeiswinkel.

multimode graded-index. Hier wird ein Gradientenprofil, also eine kontinuierliche Abnahme des Brechungsindex von innen nach außen verwendet. Sowohl Dispersion als auch der Aufwand der Kopplung sind mittelgroß.

Eigenschaften von Lichtleitern

Zusammenfassend lassen sich folgende Eigenschaften für Lichtleiter feststellen, deren Fazit lautet, dass Lichtleiter im lokalen, regionalen und überregionalen Bereich sehr niedrige Datenübertragungskosten pro übertragenem Bit besitzen.

- Dämpfungen über 100 dB/km bezeichnen wir als hoch, solche unterhalb von 20 dB/km als niedrig. Der Verstärkerabstand kann also im Kilometerbereich liegen.
- Lichtleiter sind unempfindlich gegenüber elektrischen Störungen. Eine elektrische Abschirmung ist nicht nötig.
- Es existiert keine galvanische (leitende) Verbindung zwischen Sender und Empfänger.
- Das Übertragungsmedium ist aus billigem Material und besitzt ein geringes Gewicht.
- Das Abhören von Lichtleitern ist deutlich komplizierter als bei elektrischer Signalübertragung oder gar bei Funkübertragung, was die Sicherheit der Datenübertragung erhöht.
- Lichtleiter sind empfindlicher gegenüber mechanischen Einflüssen, ihre Reparatur aufwändiger.
- Die Lebensdauer der optischen Sender (insbesondere der Laser) ist eher gering.

2.4 Drahtlose Signalübertragung

2.4.1 Funksysteme

Wir unterscheiden zunächst drei Varianten von Funkübertragungssystemen:

Rundfunk. Es wird der Radiowellenbereich (LF, MF, HF, VHF) genutzt. Sowohl Empfänger als auch Sender können mobil sein. Zellulare Netze sind möglich.

Terrestrischer Mikrowellen-Richtfunk. Hier besteht direkter Sichtkontakt zwischen Sender und Empfänger. Es wird das SHF-Band genutzt. Die maximale Entfernung zwischen Sender und Empfänger liegt bei ca. 50 km.

Satellitenübertragung. Hier werden die Bänder SHF und (seltener) UHF von umlaufenden (LEO[2.7]) oder geostationären Satelliten (GEO[2.8]) verwendet. Ein Beispiel für LEO-Satelliten ist das IRIDIUM-System. Bei geostationären Satelliten wird im wesentlichen das C-Band (4/6 GHz) und neuerdings das KU-Band (12/14 GHz) genutzt. Es existiert allerdings eine Tendenz zum Frequenzbereich 20/30 GHz. Ein Beispiel für die genutzten Frequenzbereiche im C-Band sind [5.925 GHz, 6.425 GHz] zum Satelliten und [3.7 GHz, 4.2 GHz] zur

2.7. low earth orbiting

2.8. geostationary/geosynchronouns

Erdstation. Da die Erzeugung höherer Frequenzen mehr Energie erfordert, erhält der Weg vom Satelliten zur Erde das niedrige Teilband.

Die verwendete Wellenlänge nimmt Einfluss auf die Struktur und Größe der verwendeten Antennen (z.B. Parabolspiegel) und die Art der Wellenausbreitung in der Atmosphäre. In der Atmosphäre existierende Ionisationsschichten können (je nach Wellenlänge) reflektierend wirken.

F-Schicht. Diese Schicht liegt in 200 km bis 300 km Höhe und besteht aus ionisierten Stickstoffmolekülen (in der Teilschicht F_1) und ionisierten Sauerstoffatomen, Wasserstoff und Helium (in der Teilschicht F_2).

E-Schicht. In 100 km Höhe besteht diese Schicht aus ionisierten Sauerstoffmolekülen.

D-Schicht. Diese in 50 km Höhe liegende Schicht ist nur tagsüber existent.

Ultrakurzwellen (mit einer Wellenlänge unter 10 m) werden unter anderem für Satellitenübertragungen – wegen nicht vorhandener Wellenreflexionen in den o.g. Schichten – verwendet.

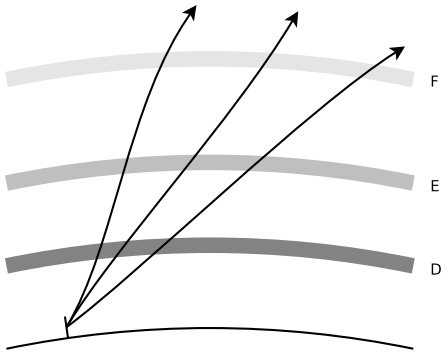


Abbildung 2.15. Ultrakurzwellen

Kurzwellen (mit Wellenlängen zwischen 10 m unter 200 m) werden in der F-Schicht reflektiert. Die Nahstrahlung hängt vom Ausstrahlungswinkel ab (z.B. 15 m bei 20° und 25 m bei 50°). Der Bereich, der dichter als die Nahstrahlung aber weiter als die Reichweite der Bodenwelle am Empfänger liegt, heißt *tote Zone*. Wenn sich die Fernstrahlung und die (am Boden reflektierte) Nahstrahlung ungünstig überlagern, spricht man von *Fernschwund*.

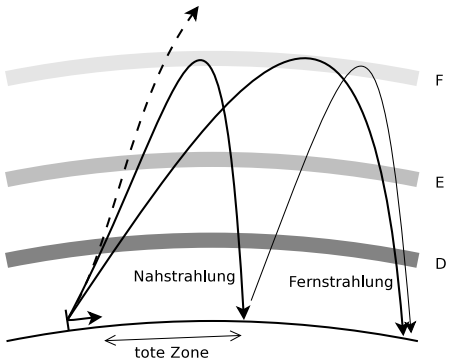


Abbildung 2.16. Kurzwellen

Mittel- und Langwellen (mit Wellenlängen zwischen 200 m und 2000 m) werden in der E-Schicht reflektiert. Die Reichweite der Bodenwelle ist deutlich größer. Hier tritt zusätzlich zum o.g. Fernschwund auch ein *Nahschwund* auf, wenn sich Bodenwelle und Nahstrahlung ungünstig überlagern.

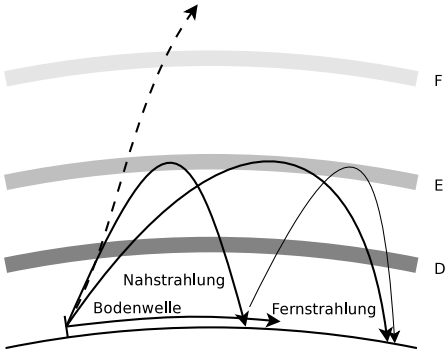


Abbildung 2.17. Mittel- und Langwellen

2.4.2 Mobilfunkübertragung

Das Grundprinzip der terrestrischen Mobilfunknetze ist die Organisation in *Zellen* mit je einer zugeordneten *Basisstation (BS)*. Dadurch lassen sich identische Frequenzen in hinreichend weit entfernten Zellen mehrfach verwenden. Die Kommunikation zwischen *Mobilstationen (MS)* und Basisstation verläuft drahtlos. Die Basisstationen kommunizieren miteinander leitungsgebunden oder per Richtfunk. Abbildung 2.18 zeigt eine Aufteilung eines Gebietes in Zellen (hier sechseckig gezeichnet).

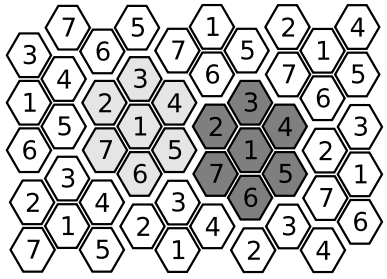


Abbildung 2.18. Aufteilung in Zellen

Die Systeme für zellularen Mobilfunk[2.9] unterteilt man in drei Generationen:

Generation 1. Analoge Übertragung. In Deutschland wurde das C-Netz im 450 MHz Band, in den USA das System AMPS[2.10] im 800 MHz Band verwendet.

Generation 2. Digitale Übertragung. Unter anderem in Europa wird das GSM[2.11] im D-Netz (GSM 900) mit dem $[890, 915]$ MHz-Band für den "uplink" und dem $[925, 960]$ MHz-Band für den "downlink" verwendet.

Generation 3. Ein weltweit einheitlicher Standard mit digitaler Übertragung. Die Datenraten liegen im Bereich von MBit/s . Die Bandbreitennutzung ist verbessert. Ein Beispiel für ein System der dritten Generation ist UMTS[2.12].

2.4.3 Infrarotübertragung

Bei der Infrarotübertragung werden elektromagnetische Wellen im Infrarotbereich genutzt, z.B. im Intervall $[850, 900]$ nm.[2.13] Sender und Empfänger müssen sich in direkter optischer Sicht ohne Hindernisse befinden.

2.9. amerikanisch: cellular radio

2.10. advanced mobile phone system

2.11. global system for mobile communications

2.12. universal mobile telecommunications system

2.13. vgl. IEEE 802.11

Diese Art der Übertragung ist durch die nur kurzen überbrückbaren Entfernungen (bis ca. 60 m), den notwendigen direkten Sichtkontakt und die Unmöglichkeit der Nutzung im Freien (u.a. wegen des Infrarotanteils des Sonnenlichts) recht eingeschränkt. Dafür ist für den Betrieb keine Lizenz erforderlich und die Realisierung sehr kostengünstig.

Es gibt sowohl Punkt-zu-Punkt Topologien (direct beam infrared), beispielsweise bei Fernbedienungen oder PC-Drucker-Verbindungen, als auch Broadcastsysteme (diffused infrared), wie ein "wireless LAN" mit Infrarotübertragung.

2.5 Allgemeine Charakteristika physikalischer Übertragungsmedien

Zur Bewertung physikalischer Übertragungmedien lassen sich eine Vielzahl von Eigenschaften nennen. Diese können physikalischer, ökonomischer, rechtlicher und pragmatischer Natur sein.

- maximale Entfernung zwischen Sender und Empfänger
- Signallaufzeit, die bei Satelliten bei ca. 250 ms liegt
- Art der Datenübertragung, digital oder analog
- Bitfehlerhäufigkeit, die bei schlechten Leitungen 10^{-4} betragen kann, bei Lichtleitern hingegen oft vernachlässigbar ist
- Verfügbarkeit des Übertragungsmediums, die gesetzlich geregelt sein kann
- Bandbreite, die vom kHz-Bereich bis zum GHz-Bereich reichen kann
- Übertragungsgeschwindigkeit/Datenrate, die u.a. auch von der Entfernung abhängt
- Kosten pro übertragenem Bit
- die Möglichkeit von Broadcast-Verbindungen, wie sie z.B. Satelliten bieten
- Randbedingungen, wie die Verlegung fester Leitungen oder die Vermeidung von Hindernissen in Richtfunkstrecken
- Unterstützung mobiler Kommunikation

	Datenrate	Fehlerraten	Reichweite	sonstiges
verdrillte Drähte	MBit/s	hoch	gering ohne Modem	preisgünstig
Koaxialkabel	MBit/s	gering	10 km	preisgünstig
Lichtleiter	TBit/s	sehr gering	50 km	abhörsicher
Rundfunk	MBit/s	mittel	überregional	mobile Empfänger
Richtfunk	MBit/s	gering	50 km	direkte Sicht
Satellitenübertragung	GBit/s	streuend	interkontinental	teuer, hohe Signallaufzeit
Infrarotübertragung	MBit/s	hoch	in Räumen	direkte Sicht

Tabelle 2.2. Grobbeurteilung physikalischer Übertragungsmedien

Digital Subscriber Line

Ausgangspunkt für den DSL-Dienst ist die Telefonleitung (insbesondere UTP-Kabel der Kategorie 3) mit einem nutzbaren Frequenzbereich von $[300, 3400]$ Hz, also einer Bandbreite von $3,1$ kHz. Der Grund für die niedrige Bandbreite ist eine Tiefpassfilterung zur Bandbreitenreduktion. Für den DSL-Dienst wird auf die Tiefpassfilterung verzichtet, um eine weit höhere Datenrate als 56 kBit/s ohne Beeinträchtigung der existierenden Telefon- und Faxgeräte zu erreichen.

Bei einer Bandbreite von $1,1$ MHz kann der Frequenzbereich beispielsweise in 256 Kanäle K_i mit je 4312, 5 Hz zerlegt werden. Der Kanal K_0 wird als Sprachkanal verwendet, die Kanäle K_1 bis K_5 bleiben ungenutzt, je ein Kanal wird für die Kontrolle des "upstream" und des "downstream" verwendet. Die restlichen 248 Kanäle können für Benutzerdaten verwendet werden. Die Datenraten werden bei DSL dynamisch an eine variierende Leitungsqualität angepasst.

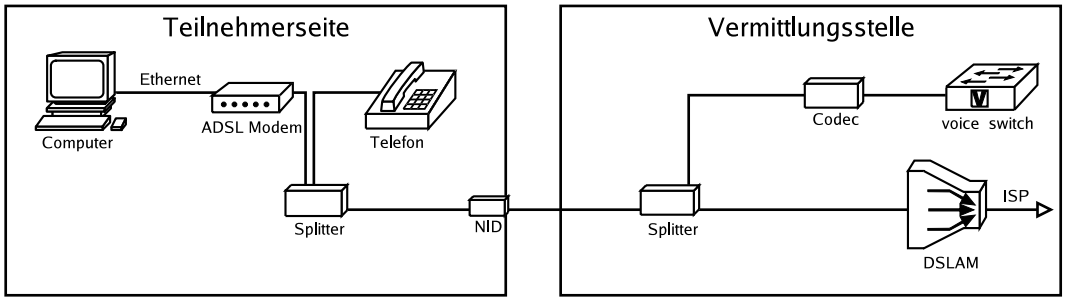


Abbildung 2.19. Typische ADSL-Konfiguration

Abbildung 2.19 zeigt eine typische ADSL-Konfiguration. Der Splitter trennt das eingehende Signal mittels eines Tief- und eines Hochpasses in zwei Signale auf. Das DSL-Modem demoduliert dann die analogen Signale, das Telefon kann wie zuvor verwendet werden. Der DSLAM[2.14] interpretiert die vom Teilnehmer ankommende Bitfolge und sendet entsprechende Pakete an den ISP.

Weitere Varianten von DSL sind HDSL (high bit rate) mit einer Datenrate von 2 Mbit/s und einer Reichweite bis ca. 3 km und VDSL (very high bit rate) mit Datenraten zwischen 12 Mbit/s und 52 Mbit/s. Die Reichweiten liegen hier zwischen 1500 m und 300 m.

2.6 Gesetze von Shannon und Nyquist

Nyquistbedingung 1

Gegeben sei eine durch eine Signalfunktion $s(t)$ beschriebene Folge von informationstragenden Signalelementen $I_1, I_2, ...$ Ist ν die Übertragungsgeschwindigkeit bezogen auf ein Signalelement und sind Abtastzeitpunkte t_n definiert als

$$t_n = \frac{n}{\nu} = n \cdot T \quad n \in \mathbb{N}, \quad T = \frac{1}{\nu}$$

dann gilt, dass wenn zum Abtastzeitpunkt t_j sämtliche Beiträge der Signalelemente I_k mit $k \neq j$ zum Signal $s(t)$ verschwinden, eine Interferenz der Signalelemente vermieden wird und die Fehlerwahrscheinlichkeit der Übertragung verringert wird.

Die Nyquistbedingung[2.15] ist trivialerweise erfüllt, wenn für einen Zeitpunkt t_k der Signalparameter für die Signalelemente I_j Nullstellen für alle $j \neq k$ besitzt.

Das Abtasten eines Signals zu den durch die Nyquistbedingung 1 gegebenen Zeitpunkten liefert eine Folge von Abtastwerten $s(n \cdot T)$ mit $n \in \mathbb{N}$.

Abtasttheorem von Shannon

Eine Signalfunktion $s(t)$ (evtl. durch Überlagerung von Signalfunktionen $s_1(t), s_2(t), ...$ entstanden) ist durch eine Folge von Abtastwerten $s(n \cdot T)$ mit $n \in \mathbb{Z}$ eindeutig bestimmt, falls für die Fouriertransformierte $S(\omega)$ von $s(t)$ gilt:

$$\forall \omega \quad |\omega| > \frac{\pi}{T} \Rightarrow S(\omega) = 0$$

Es ist dann

$$s(t) = \sum_{n=-\infty}^{\infty} s(n \cdot T) \cdot \frac{\sin\left(\frac{\pi}{T} \cdot (t - n \cdot T)\right)}{\frac{\pi}{T} \cdot (t - n \cdot T)}$$

und

$$S(\omega) = \begin{cases} T \cdot \sum_{n=-\infty}^{\infty} s(n \cdot T) \cdot e^{-i \cdot n \cdot T \cdot \omega} & , |\omega| \leq \frac{\pi}{T} \\ 0 & , \text{sonst} \end{cases}$$

2.14. digital subscriber line access multiplexer

2.15. auch Nyquist-Kriterium

Dabei gelten die folgenden Beziehungen: $|\omega| > \frac{\pi}{T} \Leftrightarrow 2 \cdot \pi \cdot |\nu| > \frac{\pi}{T} \Leftrightarrow |\nu| > \frac{1}{2T}$. Ist $s(t)$ nur für Zeitpunkte $t \geq 0$ definiert, kann $\mathbb{N}$ anstelle von $\mathbb{Z}$ treten.

Zusammenhang zwischen Datenrate und Bandbreite im idealen Kanal

In einem *idealen Kanal*, also einem Kanal ohne jegliche Störungen, lässt sich mit L Signalniveaus und einer Bandbreite B und $2 \cdot B$ Abtastungen pro Sekunde (vgl. Abtasttheorem) bestenfalls eine maximale Datenrate von $C_{\max}$ erreichen, für die gilt:

$$C_{\max} = 2 \cdot B \cdot \mathrm{ld}(L) \left[\frac{\mathrm{bit}}{\mathrm{s}} \right]$$

Bei einer Telefonleitung mit $B = 3\,\mathrm{kHz}$ (vergl. Fernsprechkanal mit 3,1 kHz) und $L = 2$, also einem zweistufigen binären Signal ist damit $C_{\max} = 2 \cdot 3\,[\mathrm{kHz}] \cdot \mathrm{ld}(2)\,[\mathrm{bit}] = 6 \cdot 1000 \left[\frac{1}{\mathrm{s}} \cdot \mathrm{bit} \right] = 6 \left[\frac{\mathrm{kbit}}{\mathrm{s}} \right]$.

$C_{\max}$ wird auch als Kanalkapazität bezeichnet. Der Wert ist eventuell bei weitem nicht erreichbar, sondern stellt lediglich eine obere Schranke dar. Insbesondere ist es praktisch nicht möglich, beliebig viele Signalniveaus zu verwenden.

Zusammenhang zwischen Datenrate und Bandbreite im realen Kanal

In einem *realen Kanal* mit Störungen ist bei der Errechnung der maximal erreichbaren Datenrate das Verhältnis zwischen Signalstärke und Störungen zu berücksichtigen. Dieser Wert läßt sich als Signal-Rausch-Leistungsverhältnis[2.16] $\frac{S}{N}$ darstellen. Die maximale Datenrate bei einer gegebenen Bandbreite B ist dann

$$C_{\max} = B \cdot \mathrm{ld}\left(1 + \frac{S}{N}\right) \left[\frac{\mathrm{bit}}{\mathrm{s}} \right]$$

Selbstverständlich gilt die Kapazitätsbegrenzung des idealen Kanals auch für jeden realen Kanal, da dieser nicht besser als ein idealer Kanal sein kann.

2.7 Datenübertragungsverfahren

Die Frage, ob eine Datenübertragung digital oder analog erfolgt, soll im folgenden genauer betrachtet werden. Wir unterscheiden:

Daten. Damit meinen wir die zu übertragenen Nutzdaten, die entweder analog (z.B. Sprache) oder digital (z.B. Text) sind.

Signalisierung. Darunter verstehen wir die Art des Signals. Nimmt der Signalparameter kontinuierliche Werte an, so sprechen wir von analoger Signalisierung.

Übertragung. Hiermit meinen wir die Art der Interpretation des empfangenen Signals. Bei Interpretation gemäß diskretem Wertebereich sprechen wir von digitaler Übertragung.

Beispiel 2.16. Als Beispiel soll die Übertragung von Sprache dienen. Die Daten sind also zunächst analog. Wird die Sprache durch PCM-Codierung gewandelt, sind die zu übertragenen Daten digital. Wird dann unter Nutzung von Amplitudenumtastung[2.17] (ASK, amplitude shift keying) ein Signal erzeugt, ist dieses kontinuierlich, die Signalisierung also analog. Beim Empfang des Signals wird aber nicht dessen kontinuierlicher Signalparameter interpretiert, sondern nur der quantisierte, damit ist der Wertebereich bei der Interpretation diskret, die Übertragung also digital.

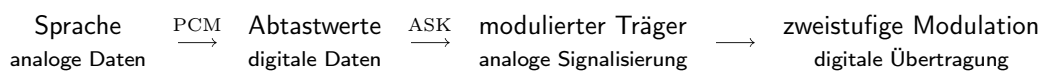


Abbildung 2.20. Digitale Übertragung analoger Sprachdaten

2.16. auch: signal-to-noise ration, SNR

2.17. auch: digitale Amplitudenmodulation

Beispiel 2.17. Ein weiteres Beispiel ist die Übertragung von Daten aus dem Hauptspeicher eines Rechners über ein Ethernet-basiertes LAN. Hier sind die Daten von Anfang an digital. Mit Hilfe der Manchester-Codierung (MCH in Abbildung 2.21) wird ein digitales Signal erzeugt und beim Empfänger gemäß eines diskreten (hier: zweiwertigen) Wertebereichs interpretiert. Signalisierung und Übertragung sind also beide digital.

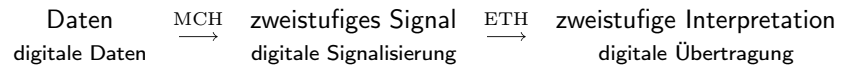


Abbildung 2.21. Übertragung von Daten aus dem Hauptspeicher über ein Ethernet

Medium	analog	digital
Kupferdrähte	ja	ja
Lichtleiter	ja	nein
Drahtlose Medien	ja	nein

Tabelle 2.3. Übertragungsmedien und Möglichkeiten der Signalisierung

2.7.1 Basisbandübertragung

Unter *Basisbandübertragung* verstehen wir eine Übertragung, bei der das Frequenzband des zu übertragenden Signals bis zur Frequenz 0 Hz reicht. Es wird keine Modulation eines Trägersignals verwendet (vgl. Abschnitt 2.7.2). Frequenzmultiplexen (FDMA) ist somit im Gegensatz zu Zeitmultiplexen nicht zur Mehrfachnutzung des Übertragungsmediums möglich (vgl. Abschnitt 2.9). Bei Basisbandübertragung ist digitale Signalisierung, z.B. durch Strom- oder Spannungsimpulse, möglich.

Codierung	Bitwert	Amplitude
Non-Return To Zero (NRZ)	0	0
	1	A
Return To Zero (RZ)	0	00
	1	A0
Bi-Phase-L (Manchester)	0	A0
	1	0A
Alternate Mark Inversion (AMI)	0	0
	1	A/-A (alternierend)

Tabelle 2.4. Einige Codierungen für Basisbandübertragungen

Anforderungen an und Bewertung von Basisbandübertragungsverfahren

In der folgenden Tabelle sind unterschiedliche Codierungsverfahren am Beispiel einer Bitsequenz dargestellt. Wir nennen kurz einige wichtige Kriterien für die Bewertung der Codierungen.

Selbsttaktende Codierung. Einige Codierungen unterstützen die Bitsychronisation, indem sie Pegelwechsel zu bestimmten Zeitpunkten erzwingen. So kann auch in einer Bitsequenz, die nur aus Nullen besteht, der Takt beim Empfänger ermittelt werden. Ein typisches Beispiel ist die Manchester-Codierung (g). Auch die Codierungen (a), (b) und (h) sind selbsttaktend.

Gleichstromanteil. Es ist oft vorteilhaft, eine Codierung so zu gestalten, dass sie keinen Gleichstrom produziert. Dazu muss die Differenz der positiven und negativen Impulse für beliebige Bitsequenzen mit einer Länge, die gegen unendlich strebt kleiner oder gleich 1 sein. Die Codierungen (g), (h) und (i) erfüllen diese Voraussetzung.

Bandbreitennutzung. Die Verteilung der Signalenergie auf den Frequenzbereich ist bei den Codierungen sehr unterschiedlich. Hierfür betrachtet man zufällige Bitmuster.

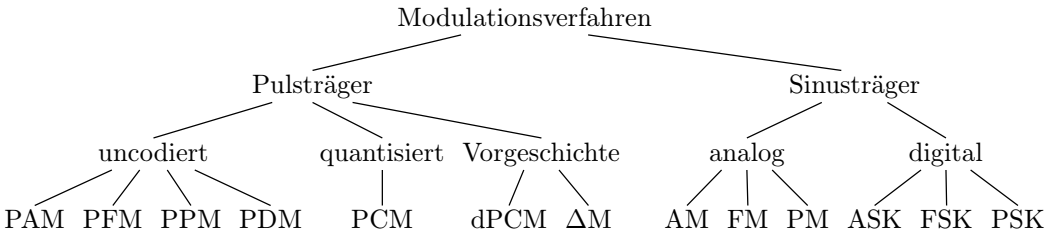
	1	0	1	0	0	1	1	1	0	Bezeichnung	Regel	
a	+0	00	+0	00	00	+0	+0	+0	00	RZ (unipolar)	$0 \rightarrow$ 00	$1 \rightarrow$ +0
b	+0	-0	+0	-0	-0	+0	+0	+0	-0	RZ (bipolar)	$0 \rightarrow$ -0	$1 \rightarrow$ +0
c	++	00	++	00	00	++	++	++	00	NRZ-Level (unipolar)	$0 \rightarrow$ 00	$1 \rightarrow$ ++
d	++	--	++	--	--	++	++	++	--	NRZ-Level (bipolar)	$0 \rightarrow$ --	$1 \rightarrow$ ++
e	++	++	--	--	--	++	--	++	++	NRZ-Mark	$0 \rightarrow$ W	$1 \rightarrow$ KW
f	++	--	--	++	--	--	--	--	++	NRZ-Space	$0 \rightarrow$ KW	$1 \rightarrow$ W
g	+-	-+	+-	-+	-+	+-	+-	+-	-+	Bi-Phase-Level	$0 \rightarrow$ -+	$1 \rightarrow$ +-
h	+-	++	-+	--	++	-+	-+	-+	-+	Bi-Phase-Mark	$0 \rightarrow$ W/KW	$1 \rightarrow$ W/W
i	++	00	--	00	00	++	--	++	00	AMI pseudoternär	$0 \rightarrow$ 00	$1 \rightarrow$ ++/--

Abbildung 2.22. Unterschiedliche Codierung der Bitsequenz 101001110 (W: Wechsel, KW: kein Wechsel des Pegels)

2.7.2 Modulationsverfahren durch Sinusträger

Ist ein Übertragungsmedium nur zur Übertragung von Frequenzen im Intervall $[\nu_0, \nu_1]$ mit $\nu_0 > 0$ geeignet, ist Basisbandübertragung unmöglich. Es ist aber möglich, das ursprüngliche Signal durch Variation eines Trägers[2.18] $f(t)$ zu codieren. Die folgenden Signalparameter können dabei variiert werden.

- Amplitude
- Frequenz
- Phase
- Impulsdauer (bei Pulsträger)



Klasse	Abk.	englische Bezeichnung
Pulsträger	PAM	Pulse Amplitude Modulation
	PFM	Pulse Frequency Modulation
	PPM	Pulse Phase Modulation
	PDM	Pulse Duration Modulation
	PCM	Pulse Code Modulation
	dPCM	Differential Pulse Code Modulation
	ΔM	Delta-Modulation
Sinusträger	AM	Amplitude Modulation
	FM	Frequency Modulation
	PM	Phase Modulation
	ASK	Amplitude Shift Keying
	FSK	Frequency Shift Keying
	PSK	Phase Shift Keying

Tabelle 2.5. Abkürzungen für Modulationsverfahren

Es folgt eine Übersicht über Modulationsverfahren. In diesem Abschnitt werden wir die Modulationsverfahren mit Sinusträger betrachten. Die folgende Abbildung zeigt anschaulich die unterschied-

2.18. Sinus- oder Pulsträger

lichen Modulationsverfahren.

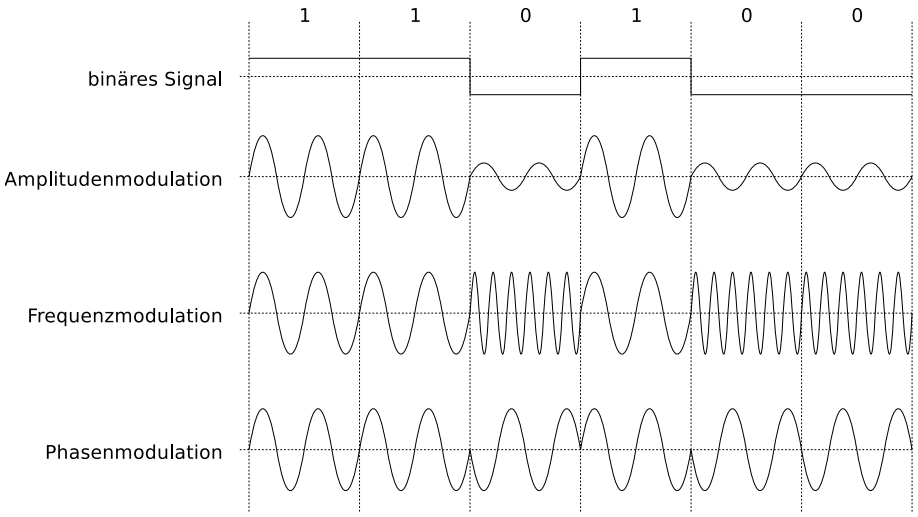


Abbildung 2.23. Ergebnis unterschiedlicher Modulationen mit Sinusträger

Amplitudenmodulation

Bei der *Amplitudenmodulation* wird ein zu übertragenes Signal $s(t)$ verwendet, um die Amplitude A_T eines Trägersignals $f(t)$ zu variieren. Es resultiert eine Signalfunktion $g(t)$.

$$g(t) = (A_T + s(t)) \cdot \cos(\omega_T \cdot t)$$

Abbildung 2.24 zeigt beispielhaft die Modulation eines Sprachsignals. Rechnerisch lässt sich zeigen, dass das Frequenzspektrum der so erzeugten Signalfunktion in drei Bereichen Werte ungleich Null besitzt: Die Trägerfrequenz, das untere und das obere Seitenband.

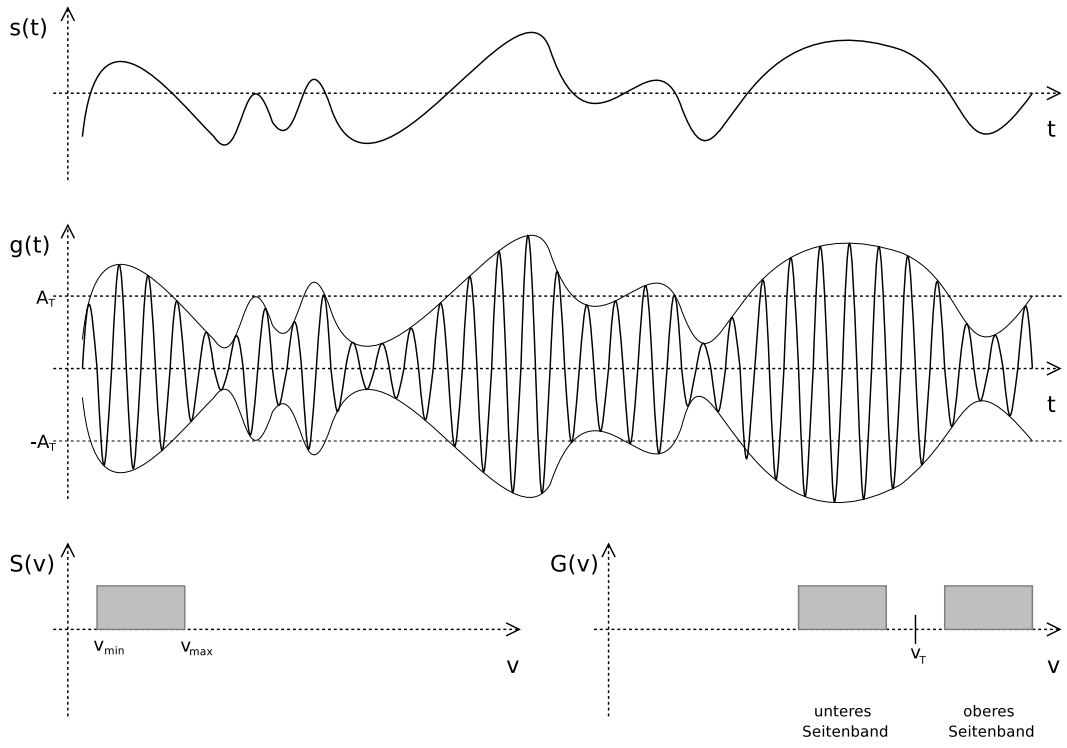


Abbildung 2.24. Amplitudenmodulation eines Sprachsignals

Es ist nicht nötig, beide Seitenbänder auch zu übertragen, obwohl die zusätzliche Redundanz zur Fehlerbehandlung genutzt werden kann. Es gibt folgende Möglichkeiten:

- Zweiseitenband-AM: Übertragung beider Seitenbänder und des Trägers
- Einseitenband-AM: Übertragung eines Seitenbandes ohne Träger
- Restseitenband-AM: Übertragung eines Seitenbandes sowie teilweise des zweiten und des Trägers

Ist das zu übertragene Signal digital, so spricht man von *Amplitudenumtastung* (amplitude shift keying, ASK). Im einfachsten Fall wird folgende Codierung vorgenommen: $0 \rightarrow 0$, $1 \rightarrow A$. Die Amplitude des Trägers wechselt also je nach Bit von 0 nach A . Allgemeiner ist die folgende Vorschrift, bei der unterschiedliche Amplituden A_0 und A_1 für die Codierung verwendet werden: $0 \rightarrow A_0, 1 \rightarrow A_1$.
Es ist auch möglich, mehr als zwei unterschiedliche Amplituden zu verwenden. Beispielsweise wird bei quaternärer Amplitudenumtastung aus zwei Bits eine Amplitude errechnet: $00 \rightarrow A_{00}$, $01 \rightarrow A_{01}$,$10 \rightarrow A_{10}, 11 \rightarrow A_{11}$.

Frequenzmodulation

Frequenzmodulation bietet geringe bis mittlere Übertragungsgeschwindigkeiten mit relativ geringem Aufwand. Sie ist weniger störempfindlich als Amplitudenmodulation.
Eine Trägerschwingung $f(t) = A_T \cdot \cos(\omega_T \cdot t + \phi_T)$ wird mit einem Signal $s(t)$ moduliert zu

$$g(t) = A_T \cdot \cos(\omega_T \cdot t + \phi_T + \phi_{\mathrm{mod}}(t))$$

Bei digitalen Signalen spricht man wiederum von Frequenzumtastung oder kurz Frequenztastung. Bei der binären Frequenzumtastung wird den Bitwerten eine Frequenz[2.19] zugeordnet: $0 \rightarrow \nu_0, 1 \rightarrow \nu_1$
Die Frequenzen ν_0 und ν_1 heißen *Kennfrequenzen* der binären Frequenzmodulation. Der Mittelwert der Kennfrequenzen $\nu_{\mathrm{VT}} = \frac{\nu_0 + \nu_1}{2}$ heißt *virtuelle Trägerfrequenz* oder *Mittenfrequenz.* Der Wert $\Delta\nu = \left|\frac{\nu_1 - \nu_0}{2}\right|$ heißt *Frequenzhub.*

Phasenmodulation

Phasenmodulation ist bei mittleren Datenraten weit verbreitet, da sie relativ unempfindlich gegenüber Störungen ist. Es gibt drei Varianten. Dabei dient als Informationsträger die Differenz zwischen den Phasen.

- *Phasendifferenzmodulation (PDM)* des übertragenen Signals zu zwei aufeinander folgenden Zeitpunkten t_1 und t_2.
- *Phasenmodulation (PM)* des Signals und des Trägersignals zu einem Zeitpunkt t.
- *Frequenzdifferentielle Phasenmodulation (FDPM)* zweier parallel übertragener Signale $g_1(t)$ und $g_2(t)$ mit unterschiedlichen Frequenzen.

Auch hier wird bei digitaler Übertragung von Phasenumtastung gesprochen und es ist ebenfalls möglich, längere Bitsequenzen mit einem Phasensprung darzustellen (siehe Tabelle 2.6). Außerdem sind diverse Kombinationen mit Frequenz- und Amplitudenumtastung möglich.

2.19. Gemäß Empfehlung des CCITT gilt $\nu_0 > \nu_1$.

Dibit	Phasensprung	alternativ
00	0°	45°
01	90°	135°
10	180°	225°
11	270°	315°

Tabelle 2.6. Quaternäre Phasenumtastung: 4-PSK

Kombinationen

Längere Bitsequenzen lassen sich in einem Schritt auch durch Kombination der oben genannten Modulationsverfahren darstellen. So können beispielsweise vier unterschiedliche Amplitudenzustände mit acht verschiedenen Phasenzuständen kombiniert werden. Die resultierenden 32 Kombinationen könnten die fünf Bit repräsentieren. Häufig wird zur Verringerung der Fehlerhäufigkeit Redundanz eingefügt. So könnten in unserem Beispiel nur 16 der 32 Zustände einem korrekten Codewort entsprechen. In diesem Fall könnten pro Schritt nur vier Bit übertragen werden. Die folgende Tabelle gibt einen Überblick über die Vielzahl der verwendeten Kombinationen der Modulationsverfahren.

Verfahren	Amplituden	Phasen	Kombinationen	bit/Symbol	typischer SNR
BPSK	1	2	2	1	13 db
QPSK	1	4	4	2	17 db
8-PSK	1	8	8	3	23 db
16-PSK	1	16	16	4	30 db
16-APK	3	8	16	4	28 db
16-QASK	3	12	16	4	
32-QASK	3	28	32	5	32 db
64-QASK	4	52	64	6	
256-QASK	8	228	256	8	

Tabelle 2.7. Kombinationen aus Phasen- und Amplitudenmodulation mit typischem Signal to Noise Ratio bei einer Bitfehlerhäufigkeit von 10^{-6}

2.7.3 Modulationsverfahren mit Pulsträger

Mit Pulsmodulation lassen sich Signale der Klasse 1 codieren. Ein *Puls* ist eine periodische Folge von Impulsen (z.B. Rechteckimpulsen), wobei die Schrittdauer (Länge) und die Pausen zwischen den Impulsen ungleich sind. Es können unterschiedliche Parameter moduliert werden:

- Pulsamplitudenmodulation (PAM)
- Pulslängenmodulation[2.20] (PLM)
- Pulsphasenmodulation (PPM)
- Pulscodemodulation (PCM)

Bei letztem werden ganze Impulsgruppen generiert. Sprache kann beispielsweise mit 8 kHz Abtastfrequenz und 8 Bit pro Abtastwert wie folgt codiert werden. Alle 125 μs wird ein Wert abgetastet und codiert. Die Abtastfrequenz von 8 kHz ergibt sich dabei aus dem Abtasttheorem von Shannon unter Berücksichtigung des für einen digitalen Sprachkanal typischen Frequenzbereichs von [300, 3400] Hz.[2.21]

2.20. auch: Pulsdauermodulation, PDM

2.21. $8000\,\text{Hz} > 2 \cdot 3400\,\text{Hz}$

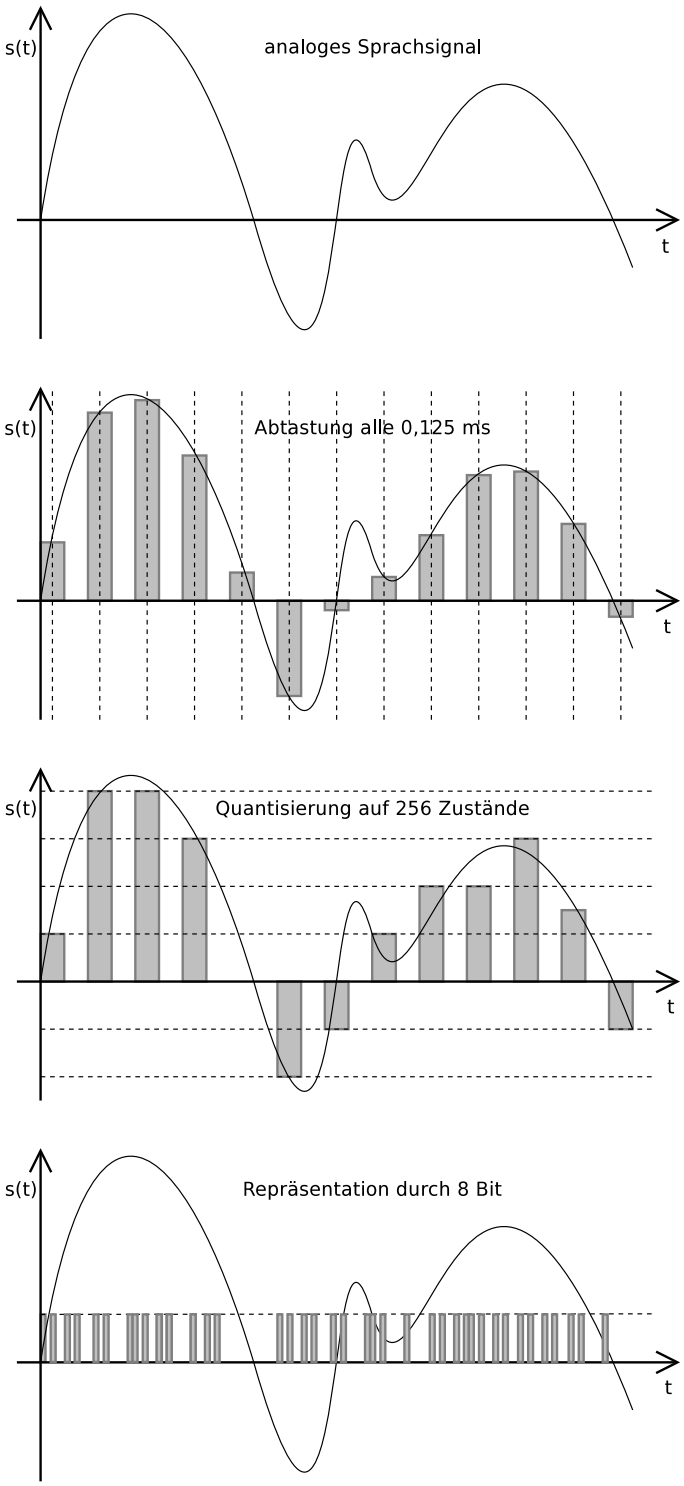


Abbildung 2.25. Beispiel einer Pulscodemodulation

Bei der Codierung der quantisierten Zustände gibt es allerdings eine Besonderheit. Der Wertebereich der Amplitude wird nicht (wie die Abbildung 2.25 eigentlich nahelegt) in gleich große Abschnitte zerlegt. Stattdessen werden zur Reduktion des Qualitätsverlustes der Sprachübertragung niedrige Amplituden genauer dargestellt als hohe. Dazu wird der Wertebereich zunächst in acht Segmente wachsender Größe unterteilt. In jedem Segment werden 16 Zustände unterschieden und mit vier Bit dargestellt. Abbildung 2.26 zeigt die Einteilung in Segmente und das Byteformat bei PCM.

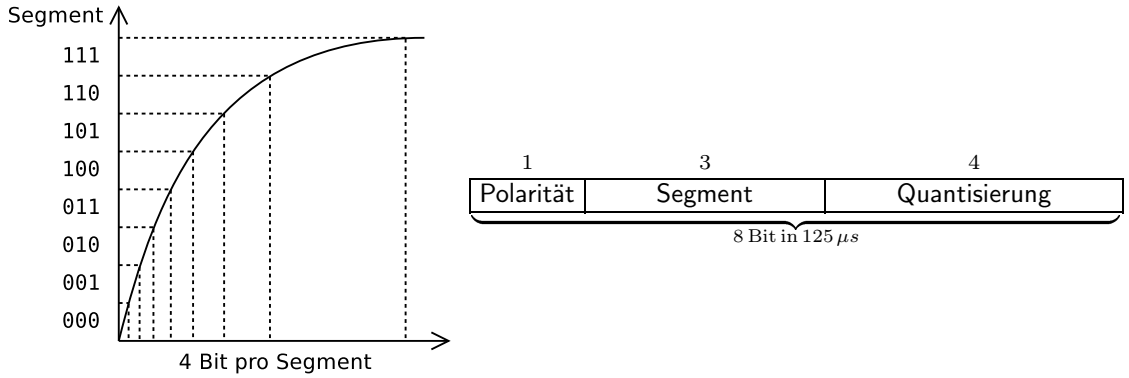


Abbildung 2.26. Einteilung des Wertebereichs bei PCM

2.8 Serielle versus parallele Datenübertragung

Serielle Übertragung ist bei größeren Entfernungen typisch. Bei kürzeren Entfernungen können verschiedene Leitungen gleichzeitig zur Parallelübertragung verwendet werden. Die Datenrate wird so um den Faktor der verwendeten Leitungen erhöht. Es treten allerdings zusätzliche Schwierigkeiten auf:

- Es werden mehrere Leitungen benötigt.
- Die empfangenen Bits müssen korrekt synchronisiert werden.
- Störungen wirken sich oft auf alle Leitungen aus, was die Fehlerkontrolle deutlich schwieriger macht.
- Wenn die Signallaufzeit die Gesamtverzögerung dominiert, bietet auch eine höhere Datenrate keinen deutlichen Geschwindigkeitsgewinn.

2.9 Mehrfachnutzung physikalischer Übertragungswege

Die Mehrfachnutzung eines physikalischen Übertragungsweges durch Datenendeinrichtungen A_1, $A_2, \ldots, A_n, B_1, B_2, \ldots, B_n$ wird Multiplexen genannt. Es sind unterschiedliche Varianten möglich.

Raummultiplexen.

Es werden räumlich getrennte Teile des Mediums verwendet. Das können beipielsweise mehrere Drähte sein, oder geographische Bereiche (Zellen) bei Funksystemen.

Zeitmultiplexen.

Es werden den Kommunikationspartnern Zeitbereiche (auch: Zeitscheiben) zugeordnet, in denen ihnen das Übertragungsmedium zur Verfügung steht. Haben diese eine konstante Länge, z.B. für die Übertragung eines PCM-Abtastwertes, spricht man von *synchronous time division multiplexing (STDM)*, variiert die Länge der Zeitbereiche, so liegt *asynchronous time division multiplexing (ATDM)* vor. Hier kann das Ende der Nutzungsdauer explizit im Kopf der Dateneinheit angezeigt werden.

Der Vorteil von synchronem Zeitmultiplexen ist, dass Echtzeitkommunikation insofern unterstützt wird, als Verzögerungen a priori bekannt sind. Dafür verfallen möglicherweise Zeitbereiche ungenutzt und der Übertragungsweg ist blockiert, wenn alle Zeitbereiche vergeben sind.

Wird die Vergabe der Zeitbereiche über einen speziellen separaten Kanal durchgeführt, so spricht man von *out-band-signalling*. Wenn diese Kontrollinformationen in die Nutzdaten eingebettet werden oder bestimmte Zeitscheiben dafür vorgesehen sind, spricht man von *in-band-signalling*.

Frequenzmultiplexen. [2.22]

Der zur Verfügung stehende Frequenzbereich wird unter den Kommunikationspartnern aufgeteilt. Sollen beispielsweise 60 Spachkanäle mit 3,1 kHz Bandbreite über ein Übertragungsmedium im Bereich $[312, 552]$ kHz (also einer Bandbreite von 240 kHz) übertragen werden, so ist aus zwei Gründen keine Basisbandübertragung möglich: Der zu übertragende Frequenzbereich liegt nicht im Frequenzbereich des Übertragungsmediums und die Frequenzbereiche der 60 Sprachkanäle überlappen (komplett).

Eine Lösung ist die Übertragung der Sprachkanäle mit Amplitudenmodulation, wobei die jeweiligen Trägerfrequenzen so gewählt sind, dass (zumindest) ein Seitenband komplett und ohne Überlappungen mit anderen im Frequenzbereich des Übertragungsmediums liegt.

2.9.1 Standardisierung von Zeitmultiplex-Verfahren in öffentlichen Netzen

Beim amerikanischen *"T1-Link"* werden Frames von 125 μs in 24 Zeitscheiben mit je acht Bit eingeteilt. Am Anfang des Frames wird zusätzlich ein "framing bit" eingefügt. Daraus ergibt sich eine Datenrate von 1,544 Mbit/s. Die Kontrollinformation wird in je einem Bit zweier dieser Zeitscheiben (6 und 12), für die dann nur noch 7 Bit für Nutzdaten zur Verfügung stehen, untergebracht. Die Datenrate aus Sicht eines Benutzers ist damit (je nach Zeitscheibe) 64 kbit/s oder 56 kbit/s.

In Europa wird der von der CCITT entwickelte *"E1-Link"* verwendet. Hier werden 32 Zeitscheiben verwendet, von denen zwei komplett für Kontrollinformationen genutzt werden. Die Gesamtdatenrate liegt bei 2,048 Mbit/s, aus Sicht eines Benutzers bei 64 kbit/s.

Mehrere T1- bzw. E1-Links können per Zeitmultiplexen auf Verbindungen mit höherer Datenrate untergebracht werden. Es ergibt sich also eine Hierarchie aus Trägern, die *plesiochrone digitale hierarchie (PDH)*. In der Regel erhöht sich die Anzahl der Sprachkanäle pro Hierarchieebene (Stufe) um den Faktor 4 gegenüber der nächsttieferen Stufe.

2.9.2 Standardisierung von Frequenzmultiplex-Verfahren in öffentlichen Netzen

Auch für Frequenzmultiplex-Verfahren gibt es in ähnlicher Weise hierarchisch organisierte Standards. Eine Primärgruppe umfasst 12 Kanäle mit je 4 kHz Bandbreite. Sekundär-, Tertiär-, Quartär- und Quintärgruppen fassen jeweils 3, 4 oder 5 der Untergruppen zusammen.

2.10 Synchronisation zur Datenübertragung

Eine wichtige Teilaufgabe bei der Datenübertragung ist es, den "Gleichlauf" zwischen Sender und Empfänger herzustellen. Unter *Bitsynchronisation* verstehen wir das Erzielen von hinreichend exakten Zeitpunkten zur korrekten Interpretation des empfangenen Signals. *Zeichen-* und *Blocksynchronisation* lassen sich analog definieren.

2.22. bei optischer Datenübertragung: wave length division multiplexing (WDM)

Bitsynchronisation

Es kann eine spezielle Codierung wie z.B. die Manchester-Codierung eingesetzt werden. Der Sender kann neben den Nutzdaten ein spezielles Taktsignal zur Verfügung stellen. Die dritte Möglichkeit besteht darin, sich bei hinreichend genauen Uhren für eine bestimmte Zahl von Bits schlicht auf die internen Uhren zu verlassen.

Zeichensynchronisation

Für Zeichensynchronisation können spezielle Bitmuster eingesetzt werden. Bei BSC ist es beispielsweise 01101000 (das SYN-Zeichen in ASCII-Codierung). Dieses Muster kann mehrmals hintereinander gesendet werden, ohne dass es auch "versetzt" in dieser Sequenz erkennbar ist.

Blocksynchronisation

Ist der Beginn des ersten Blocks korrekt festgestellt, kann Blocksynchronisation durch eine Längenangabe im Block oder ein spezielles Ende-Symbol erreicht werden. Bei HDLC wird beispielsweise als Beginn- und Endesymbol das sogenannte "Flag" `01111110` verwendet. Diese Sequenz kann für Nutzdaten dann nicht mehr verwendet werden. Ein Ausweg zur Vermeidung von a priori reservierten Kontrollzeichen in den Nutzdaten bietet ein "Escape"-Bitmuster. Ein anderer Ausweg (der von HDLC) ist es, nach jedem Vorkommen von fünf Einsen eine Null einzufügen, die beim Empfänger dann wieder entfernt wird (bit stuffing/bit stripping).

Eine andere Möglichkeit ist es, die speziellen Beginn-/Endsymbole durch "Verletzungen" (violations) der Codierung darzustellen. So entstehen z.B. bei der Manchester-Codierung, bei der in jedem Bittakt ein Wechsel des Signalparameters vorgesehen ist, zusätzliche Symbole, wenn man diese Regel mißachtet. Da beispielsweise die Sequenz `LH HL LL HH HL LL HH HL HL LH` von Nutzdaten nicht verwendet wird, kann sie zur Synchronisation eingesetzt werden.

2.11 Datenübertragungseinrichtungen und zugehörige Schnittstellen zu physikalischen Übertragungswegen sowie Zugang zu öffentlichen Netzen

Für die Kommunikation zwischen Endsystem und Modem muss eine Schnittstelle definiert sein. Zu normen ist dabei insbesondere

- physikalische Eigenschaften: Pegel und Leitungen (z.B. Erd-, Daten-, Takt-, Steuer- und Meldeleitungen)
- Semantik der Signale für Steuer- und Meldeleitungen
- Timing-Aspekte: maximale Wartezeiten, zulässige Sequenzen
- mechanische Eigenschaften: Form der Steckverbindungen, Belegung der Pins.

2.12 Fehlerkontrolle bei Datenübertragungen

Die Aufgaben der Fehlerkontrolle sind *Fehlererkennung* und *Fehlerkorrektur*. Fehlerkontrolle wird auf unterschiedlichen Schichten einer Protokollhierarchie durchgeführt. Zu den Fehlern gehören:

1. Verfälschung
2. Verlust
3. Verdopplung

4. Reihenfolgenverfälschung

Fehler des Typs 2 können durch Bestätigungen, Fehler des Typs 3 und 4 durch Sequenznummern erkannt werden. Die Fehlerkorrektur wird schlicht durch erneutes Senden bzw. Verwerfen der Duplikate realisiert. Fehler des Typs 1 können mit fehlererkennenden Codes erkannt werden. Die Fehlerkorrektur ist durch fehlerkorrigierende Codes oder Vorwärtsfehlerkontrolle möglich.

2.12.1 Codierungen zur Fehlerkontrolle

Wird den Nutzdaten durch eine Codierung redundante Zusatzinformation hinzugefügt, können (bestimmte) Fehler erkannt werden. Die einfachste Form ist ein Paritätsbit am Ende eines Blockes. Stimmt die Parität beim Empfang nicht mit dem Block überein, muss ein Fehler vorliegen. So werden alle n-Bit-Fehler erkannt, bei denen n ungerade ist. Ist n gerade kann der Fehler nicht erkannt werden.

Da Bit-Fehler häufig gebündelt auftreten (Büschelfehler), ist das Risiko, den Fehler nicht zu erkennen recht groß. Eine Verbesserung erreicht man, indem man einen Block von Daten nicht nur "zeilenweise" mit einem Paritätsbit absichert, sondern auch "spaltenweise". Damit ein Fehler unentdeckt bleibt, müssen mindestens vier Bit gerade so kippen, dass sie in genau zwei Spalten und zwei Zeilen liegen (siehe Abbildung 2.27).

0	0	0	0	0	1	0	0	1
0	1	***1***	0	1	1	***0***	1	1
1	1	0	0	0	1	0	0	1
0	0	0	1	1	0	1	1	0
1	0	***1***	1	0	1	***0***	0	0
1	1	1	0	1	1	0	0	1
1	0	0	1	0	0	1	1	0
0	1	1	0	1	1	1	0	1
0	0	0	1	0	0	1	1	1

Abbildung 2.27. Zeilen-/Spaltenparität mit Beispiel für unentdeckte Fehler

Hamming-Codes

Definition 2.18. *Der* Hamming-Abstand *oder die* Hamming-Distanz *zweier Codewörter ist die Zahl der Bits, in denen sie sich unterscheiden. So haben beispielsweise die beiden Codewörter* `01100100` *und* `10101100` *den Hamming-Abstand drei, denn sie unterscheiden sich im ersten, im zweiten und im fünften Bit (von links). Der Hamming-Abstand eines Codes ist der minimale Hamming-Abstand zwischen zwei Codewörtern dieses Codes.*

Es lässt sich folgendes zeigen: Besitzt ein Code einen Hamming-Abstand $d+1$, so können alle d-Bit-Fehler erkannt werden. Ist der Hamming-Abstand $2d+1$, so können alle d-Bit-Fehler korrigiert werden.[2.23] Ist beispielsweise der Hamming-Abstand eines Codes 5, so können alle 1-, 2,- 3,- und 4-Bit-Fehler erkannt, und alle 1- und 2-Bit-Fehler korrigiert werden.

Cyclic Redundancy Check (CRC)

Zu den komplizierteren Verfahren zur Prüfsummenbildung gehört die zyklische Codierung. Bei einer (n,k)-zyklischen Codierung wird aus Nutzdaten der Länge k Bit ein Codewort der Länge n Bit generiert. Es werden also $n-k$ Bit redundante Zusatzinformation zugefügt. Ist $(b_{n-1}, b_{n-2}, \ldots, b_1, b_0)$ ein Codewort einer zyklischen Codierung, so ist auch $(b_0, b_{n-1}, b_{n-2}, \ldots, b_2, b_1)$ ein Codewort dieser Codierung.[2.24]

2.23. Fehler mit mehr als d Bit können allerdings unerkannt bleiben oder fehlerhaft korrigiert werden.

2.24. Daher die Bezeichnung "zyklisch".

Die Bitsequenzen werden bei der zyklischen Codierung als Polynome aufgefasst. So wird etwa die Bitsequenz `01001011` als $0 \cdot x^7 + 1 \cdot x^6 + 0 \cdot x^5 + 0 \cdot x^4 + 1 \cdot x^3 + 0 \cdot x^2 + 1 \cdot x^1 + 1 \cdot x^0$ oder kurz als $x^6 + x^3 + x + 1$ aufgefasst. Die Berechnung des Codewortes geschieht durch Polynomdivision. Die Division von Polynomen, die als Koeffizienten nur null und eins besitzen, läßt sich durch einfaches bitweises XOR durchführen. CRC wird unter anderem so häufig verwendet[2.25], weil die Prüfsummenberechnung sehr einfach in Hardware implementiert werden kann.

Für gegebene n und k mit $k < n$ existiert (mindestens) ein Polynom $P(X)$ vom Grad $n - k$, das eine (n, k)-zyklische Codierung erzeugt. Das Polynom $P(X)$ wird auch als Generatorpolynom bezeichnet.

Im folgenden wird die genaue Vorgehensweise für die Berechnung der CRC-Prüfsumme dargestellt. Die Nutzdaten werden zunächst um $n - k$ Stellen bitweise nach links verschoben, dann wird der Rest der Division durch das Generatorpolynom berechnet und an die verschobenen Nutzdaten angehängt. Sämtliche Rechenoperationen basieren dabei auf der Modulo 2-Arithmetik.

1. Multiplikation der Nutzdaten $M(X)$ mit X^{n-k}. Das entspricht einem bitweisen "shift" um $n - k$ Stellen.
$$Y(X) = M(X) \cdot X^{n-k}$$
2. Division von $Y(X)$ durch das Generatorpolynom $P(X)$ mit $Q(X)$ als Quotienten und $R(X)$ als Rest der Division.
$$\frac{Y(X)}{P(X)} = Q(X) + \frac{R(X)}{P(X)}$$
oder anders notiert:
$$Y(X) = Q(X) \cdot P(X) + R(X)$$
3. Bildung des Codewortes $U(X)$ durch anhängen des Restes $R(X)$ an $Y(X)$.
$$U(X) = Y(X) + R(X) = \left[M(X) \cdot X^{n-k} \right] + R(X)$$

Beispiel 2.19. In der folgenden Abbildung wird eine Beispielrechung für die Bildung der Prüfsumme durchgeführt. Der Ablauf entspricht der schriftlichen Division für Dezimalzahlen. Der Quotient der Division wird hierbei ignoriert, da es nur auf den Rest ankommt.

Nutzdaten	`11100110`	$M(X)$
Generatorpolynom	*11001*	$P(X)$
Nutzdaten multipliziert	`111001100000`	$Y(X)$
	`11001↓↓`	$P(X)$
	`0010111`	
	`11001↓`	$P(X)$
	`011100`	
	`11001↓↓`	$P(X)$
	`0010100`	
	`11001↓`	$P(X)$
	`011010`	
	`11001↓`	$P(X)$
	`0110`	$R(X)$
Zu übertragende Nachricht	`11100110 0110`	$Y(X) + R(X)$

Abbildung 2.28. Beispielrechung für die Prüfsummenberechnung mit CRC

Qualität der Fehlererkennung bei zyklischer Codierung

2.25. unter anderem: Ethernet, TCP, IP, Kerberos, HDLC, ISDN, ATM, USB

Der folgende Satz lässt sich gut für die Bewertung der Qualität einer zyklischen Codierung verwenden, da er eine Aussage darüber erlaubt, welche Fehler erkannt werden können und welche unentdeckt bleiben.

Die möglicherweise fehlerhaft empfangene Dateneinheit $E(X)$ lässt sich als folgende Summe[2.26] darstellen, wobei $U(X)$ die abgesendete Dateneinheit ist und $F(X)$ das Fehlerpolynom – also eine Bitsequenz, die in jeder fehlerhaft empfangenen Stelle eine `1` aufweist:

$$E(X) = U(X) + F(X)$$

Satz 2.20. *$E(X)$ ist genau dann ein gültiges Codewort, wenn das Fehlerpolynom $F(X)$ durch das Generatorpolynom $P(X)$ teilbar ist. Also kann eine Nachricht dann und nur dann als fehlerhaft erkannt werden, wenn das Fehlerpolynom nicht durch das Generatorpolynom teilbar ist.*

Beweis. Die empfangene Dateneinheit lässt sich also – wie oben – als Summe von abgesendeter Dateneinheit und Fehlerpolynom darstellen. Wegen der bitweisen Addition können wir schreiben:

$$F(X) = U(X) + E(X) \tag{2.1}$$

Außerdem können wir die Dateneinheit als Vielfaches des Generatorpolynoms mit einem Rest darstellen.

$$E(X) = T(X) \cdot P(X) + S(X) \tag{2.2}$$

Wir setzen (2.2) in (2.1) ein.

$$F(X) = U(X) + T(X) \cdot P(X) + S(X) \tag{2.3}$$

Die abgesendete Dateneinheit ist immer ohne Rest durch das Generatorpolynom teilbar.

$$U(X) = Q(X) \cdot P(X) \tag{2.4}$$

Daher können wir (2.3) auch wie folgt schreiben.

$$F(X) = Q(X) \cdot P(X) + T(X) \cdot P(X) + S(X) = [Q(X) + T(X)] \cdot P(X) + S(X) \tag{2.5}$$

Wir beweisen nun den Satz.

- Ist $E(X)$ ein Codewort, so ist es ohne Rest durch $P(X)$ teilbar, also $S(X) = 0$ in (2.2). Damit ist $F(X) = [Q(X) + T(X)] \cdot P(X)$ und somit $F(X)$ durch $P(X)$ teilbar.
- Ist $F(X)$ durch $P(X)$ teilbar, so folgt $S(X) = 0$ in (2.2) und damit $E(X) = T(X) \cdot P(X)$. Also ist $E(X)$ ein gültiges Codewort

□

Wahl des Generatorpolynoms

Die Wahl des Generatorpolynoms hängt von der Art der Fehler ab, die erkannt werden sollen. Sollen beispielsweise alle 1-Bit-Fehler erkannt werden, kann man so argumentieren:

1-Bit-Fehler lassen sich als Fehlerpolynom $F(X) = X^i$ darstellen, wobei i die Position des fehlerhaften Bits in der Bitsequenz ist. Enthält das Generatorpolynom mehr als einen Term, folgt dass $F(X) = X^i$ nicht durch $P(X)$ teilbar ist (ansonsten müsste ein $C(X)$ mit $F(X) = C(X) \cdot P(X)$ existieren). Da also $F(X)$ nicht durch $P(X)$ teilbar ist, kann $E(X)$ kein Codewort sein und wird als fehlerhaft erkannt. Also werden alle 1-Bit-Fehler erkannt, wenn das Generatorpolynom aus mehr als einem Term besteht.

2-Bit-Fehler lassen sich als Fehlerpolynom $F(X) = X^i + X^j$ mit $i \neq j$ darstellen. Ohne Beschränkung der Allgemeinheit sei $i < j$. Dann ist $F(X) = X^i + X^j = X^i \cdot (1 + X^{j-i})$. Um alle 2-Bit-Fehler zu erkennen, ist es hinreichend, wenn $P(X)$ nicht ohne Rest durch X teilbar ist und überdies $1 + X^{j-i} = 1 + X^k$ mit beliebigen k nicht ohne Rest durch P(X) teilbar ist.

2.26. auch hier ist die Summe der bitweisen Addition gemeint

Jede (n,k)-zyklische Codierung erkennt Fehlerbüschel bis zu einer Länge von r Bits, wenn $n-k \geq r$ ist und das Generatorpolynom die beiden Terme X^r und 1 enthält. Im einfachsten Fall hat das Fehlerpolynom die Form $X^0 + \cdots + X^{r-1}$ (das entspricht einem Fehlerbüschel von r Bit). In diesem Fall ist $F(X) < P(X)$, also nicht ohne Rest durch $P(X)$ teilbar. Ist das Fehlerpolynom um n Bits verschoben, also $X^{0+n} + \cdots + X^{r-1+n}$, so können wir es als $X^n \cdot (X^0 + \cdots + X^{r-1})$ umformen. Keiner der Faktoren ist durch $P(X)$ teilbar.

Standard	$P(X)$ als Bitsequenz	Anwendung
CRC-12	1100000001111	Sicherung von 6-Bit-Zeichen
CRC-16	11000000000000101	
CRC-CCITT	10001000000100001	HDLC
CRC-32	100000100110000010001110110110111	Ethernet

Tabelle 2.8. Verbreitete, standardisierte Generatorpolynome

Vorwärtsfehlerkontrolle

Bei Simplexübertragungen (z.B. digitales Fernsehen), Echtzeitkommunikation oder Gruppenkommunikation (mit einem ausgelasteten Sender) ist eine Fehlerkorrektur durch erneutes Senden unpraktikabel (oder unmöglich). Hier wählt man den Ansatz der *Vorwärtsfehlerkontrolle*. Ziel ist es hier, die Nutzdaten in Form von mehreren Paketen, so zu senden, dass die Nutzdaten auch dann noch korrekt empfangen werden können, wenn ein festgelegter Anteil der Pakete verloren geht.

Wie bei CRC werden die Nutzdaten als Polynom des Grades g aufgefasst. Es werden dann $s > g$ Stützstellen in separaten Paketen übertragen. Solange mindestens g Stützstellen beim Empfänger ankommen, kann er das Polynom daraus errechnen.

Dieser Mechanismus läßt sich dynamisch an die beobachtete Paketverlusthäufigkeit anpassen. Ein Nachteil ist der relativ hohe Rechenaufwand für den Empfänger.

Quittierungsstrategien

Um Verfälschungen der Reihenfolge zu erkennen und um dem Sender (im Falle eines Fehlers) die erneut zu schickende Dateneinheit mitzuteilen, werden meist Sequenznummern verwendet. Wir unterscheiden zwei Arten der Quittierung von Dateneinheiten:

positive Quittung. Die Kopien der Dateneinheiten (beim Sender) werden gelöscht, wenn eine positive Quittung eintrifft. Entweder wird einzeln quittiert, oder es werden mit der Quittung einer Dateneinheit auch alle davorliegenden Dateneinheiten quittiert.

negative Quittung. Die Dateneinheit wird erneut gesendet, wenn eine negative Quittung eintrifft. Auch hier kann man jede Dateneinheiten einzeln quittieren. Alternativ kann die negative Quittung einer Dateneinheit eine positive Quittung aller davorliegenden Dateneinheiten implizieren. Zusätzlich *kann* dadurch auch eine negative Quittung aller danach gesendeten Dateneinheiten impliziert werden.

Kapitel 3
Rechnerinterne Kommunikationssysteme

Bei der Betrachtung rechnerinterner Kommunikationssysteme liegt der Fokus auf der Kommunikation zwischen Zentral-, E/A- und Leitungsprozessoren untereinander und mit Speichermoduln wie Cache, Hauptspeicher und Sekundärspeicher. In diesem Kapitel werden grundsätzliche Probleme, unterschiedliche Topologieformen sowie Realisierungsmöglichkeiten betrachtet. Die rechnerinterne Datenübertragung ist für alle Rechensysteme, besonders aber für Multiprozessorsysteme bzw. Parallelrechner und Vermittlungsrechner relevant.

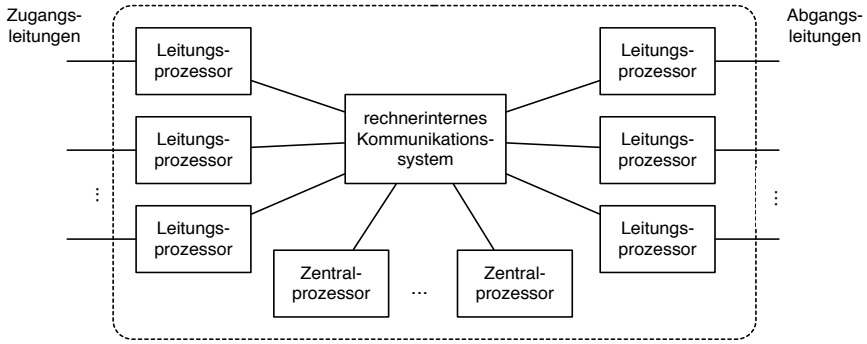


Abbildung 3.1. Vermittlungsrechner (typische Grobstruktur)

3.1 Einsatzgebiete, grundsätzliche Probleme und Lösungsansätze

Rechnerinterne Kommunikationssysteme profitieren von einigen Vereinfachungen gegenüber der Datenübertragung in Rechnernetzen. Aufgrund der geringen Entfernungen sind die *Signalverfälschungen*[3.1] und *Signallaufzeiten* sehr gering (häufig vernachlässigbar). Die zusätzlichen Kosten, die durch *parallele Datenübertragung* entstehen, sind gering, so dass sich der Durchsatz auf diese Weise relativ einfach erhöhen läßt. Desweiteren ist eine *zentralisierte Zugriffskontrolle* auf ein gemeinsames Übertragungsmedium eher akzeptabel, als in Rechnernetzen. Oft wird hier für die Zugriffskontrolle ein eigenes Kommunikationssystem verwendet (Steuerbus).

Diesen Vereinfachungen stehen spezielle Anforderungen gegenüber: Der geforderte *Durchsatz* ist – insbesondere bei der Kommunikation zwischen Zentralprozessoren und Hauptspeicher, dem so genannten *von Neumann-Flaschenhals* – in der Regel sehr hoch. Die akzeptierte *Verzögerungszeit* beim Zugriff auf ein gemeinsames Übertragungsmedium ist sehr gering und es wird eine besonders hohe *Verfügbarkeit* des rechnerinternen Kommunikationssystems erwartet.

3.1. Diese sind allerdings bei optischer Übertragung im LAN und WAN ebenfalls sehr gering.

Darüber hinaus ist meist eine sehr geringe *technische Komplexität* (etwa VLSI-Realisierbarkeit) gefordert. Auch die *algorithmische Komplexität*, etwa bei der Wegeermittlung, muss gering sein. Das kann u.a. durch statisches Routing erreicht werden (siehe dazu Kapitel 5). Zuletzt macht das Ziel hoher *Kompatibilität* oft den Einsatz von Standardlösungen nötig.

Kommunizierende Komponenten

- Prozessoren (Zentralprozessor, E/A-Prozessor, Leitungsprozessor)
- Speichermoduln (Hauptspeicher, Cache)
- LAN-Adapter o.ä.

Topologieformen

Folgende Topologieformen sind bei der rechnerinternen Datenübertragung üblich:

- individuelle Leitungen: Punkt-zu-Punkt-Verbindungen, vollständige Vermaschung
- Stern (Schaltstern), Banyan-Netz, Kreuzschienenverteiler
- Bus: Systembus
- Ring: Anschluss von Peripherie
- gemeinsamer Speicher: Multiprozessorsysteme
- Hypercube[3.2]: Parallelrechner

3.2 Infrastrukturen für rechnerinterne Kommunikation

Im folgenden werden Kreuzschienenverteiler und Banyan-Netze betrachtet. Diese sind sogenannte Schaltsterne, die Raummultiplexen in sehr einfacher Art und Weise realisieren. Vorteile von Schaltsternen sind eine weitgehend triviale Zugriffskontrolle und Wegeermittlung sowie eine einfache Adressierung. Grundsätzlich besteht ein Schaltstern aus einer Menge von Kommunikationspartnern, die über individuelle Leitungen mit einem zentralen Schaltelement verbunden sind. Abbildung 3.2 zeigt den grundsätzlichen Aufbau eines Schaltsterns.

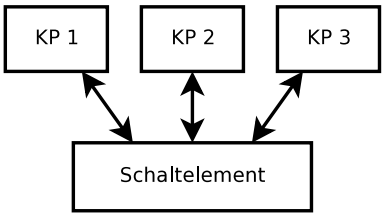


Abbildung 3.2. Schaltstern

3.2.1 Kreuzschienenverteiler

Kreuzschienenverteiler werden u.a. als Schaltstern in Multiprozessorsystemen eingesetzt. Abbildung 3.3 zeigt den grundsätzlichen Aufbau. Jeder Prozessor (P_i) und jedes Speichermodul (M_i) besitzt eine individuelle Leitung in das Netz aus *Koppeleinrichtungen* (hier als Rauten dargestellt). Durch unterschiedliches Verhalten der Koppeleinrichtung lassen sich so mehrere Kommunikationsbeziehungen zwischen je zwei Kommunikationspartnern herstellen.[3.3]

3.2. auch: Hyperwürfel

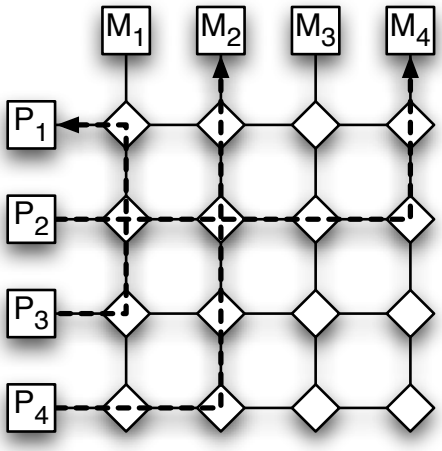


Abbildung 3.3. Kreuzschienenverteiler als Beispiel für einen Schaltstern

In Abbildung 3.4 ist der Aufbau einer einzelnen Koppeleinrichtung dargestellt. Sie besteht aus vier *Zugangspunkten,* die den Datenstrom auf unterschiedliche interne Leitungen weiterleiten können. Über einen Zugangspunkt darf nur maximal ein Datenstrom fließen. Die Koppelstufe könnte so realisiert werden, dass sie drei Zustände besitzen kann: Verwendung der internen Leitungen zwischen

1. Z_1 und Z_2 sowie Z_3 und Z_4
2. Z_1 und Z_4 sowie Z_2 und Z_3
3. Z_1 und Z_3 sowie Z_2 und Z_4

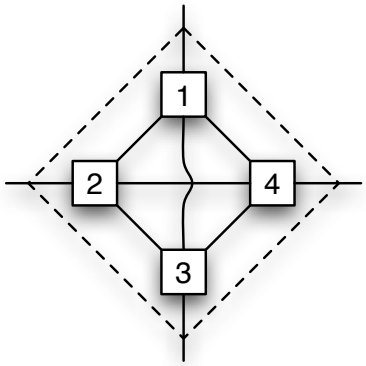


Abbildung 3.4. Detailansicht einer Koppeleinrichtung (modellhaft)

Mit "Verwendung" ist hier gemeint, dass ein Datenstrom auf dieser internen Leitung - *sofern vorhanden* - fließen würde. Natürlich muss die Leitung nicht notwendigerweise genutzt werden.

Der Kreuzschienenverteiler eignet sich besonders für länger andauernde Verbindungen. Wegeermittlung ist nur beim Einrichten eines Kommunikationskanals nötig. Der Durchsatz ist insgesamt hoch. Dem steht gegenüber, dass bei einem Kreuzschienenverteiler jeder Kommunikationspartner nur eine Kommunikationsbeziehung zur Zeit besitzen kann. Pfade können relativ schnell blockiert werden und Ausfalltoleranz ist nur sehr eingeschränkt gegeben.

3.3. Es sind auch Kommunikationsbeziehungen zwischen je zwei Prozessoren möglich.

Komplexere Koppeleinrichtungen

Im folgenden wird gezeigt, wie aus Koppeleinrichtungen mit vier Zugangspunkten zu größeren Koppeleinrichtungen zusammengesetzt werden können. Diese sogenannten *Raumkoppelstufen* werden zur Realisierung von Leitungs- bzw. Durchschaltevermittlung (z.B. in postalischen Vermittlungsrechnern) eingesetzt. Abbildung 3.5 zeigt neue Symbole für Koppeleinrichtungen, wie wir sie im Folgenden verwenden werden.

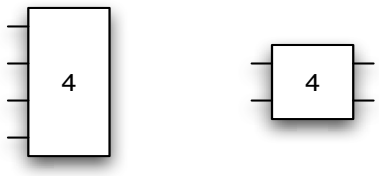
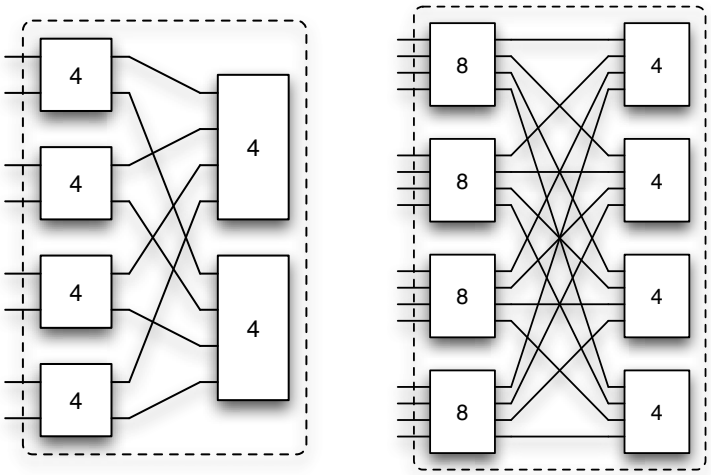


Abbildung 3.5. Neue Symbole für Koppeleinrichtungen

Aus sechs Koppeleinrichtungen mit vier Zugangspunkten (wie sie beim Kreuzschienenverteiler verwendet wurden) lässt sich eine Koppeleinrichtung mit acht Zugangspunkten realisieren (siehe Abbildung 3.6 a). Hier werden zwei Stufen aus elementaren Koppelstufen verwendet.

Aus je vier Koppeleinrichtungen mit vier bzw. acht Zugangspunkten läßt sich eine Koppeleinrichtung mit 16 Zugangspunkten realisieren (siehe Abbildung 3.6 b). In diesem Fall werden drei Stufen verwendet[3.4]. Je mehr Zugangspunkte realisiert werden sollen, umso mehr Stufen sind nötig.



a) b)

Abbildung 3.6. Komplexere Koppeleinrichtungen

3.2.2 Banyan-Netz

Ein weiterer Typ von Schaltstern ist das *Banyan-Netz*. Auch hier werden einzelne Koppeleinrichtungen in mehreren Stufen hintereinander geschaltet. Die Koppeleinrichtungen sind so geschaltet, dass sie Wegeermittlungsaufgaben in einfachster Weise (verteilt) übernehmen können.

Abbildung 3.7 zeigt den Aufbau eines Banyan-Netzes mit drei Stufen. Jede Koppeleinrichtung verfügt über zwei Eingangsleitungen und zwei Ausgangsleitungen. Die Ausgangsleitungen sind mit 0 bzw. 1 beschriftet. Für jede Dateneinheit wird die Ausgangsleitung verwendet, die mit dem ersten Bit der Zieladresse übereinstimmt. Bei jeder Weiterleitung wird das erste Bit der Adresse gelöscht.

3.4. Die zwei Stufen der Koppeleinrichtungen mit acht Zugangspunkten sind hier nicht explizit dargestellt.

Im Beispiel wird eine Dateneinheit an die Ausgangsleitung 110 gesendet. Die ersten beiden Stufen wählen die Ausgangsleitung 1, die dritte die Ausgansleitung 0.

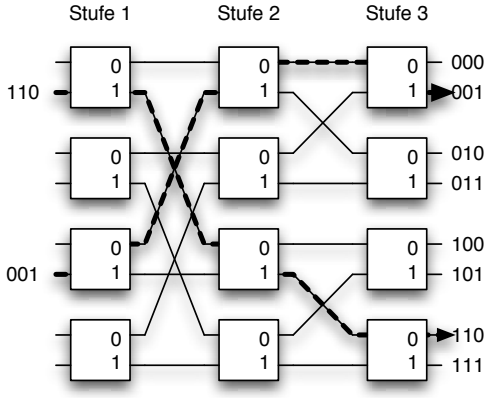


Abbildung 3.7. Banyan-Netz mit drei Stufen und zwei Kommunikationsbeziehungen

Ein Banyan-Netz mit $n = 2^k$ und $k \in \mathbb{N}$ Zugangspunkten benötigt $\mathrm{ld}(n) = k$ Stufen mit jeweils $\frac{n}{2}$ Knoten pro Stufe. Im Vergleich zum Kreuzschienenverteiler werden also deutlich weniger (allerdings kompliziertere) Knoten benötigt. Sie können synchron oder asynchron betrieben werden. Der wichtigste Nachteil ist die mögliche Blockierung von Leitungen. Abbildung 3.8 zeigt, wie Blockierungen entstehen können.

Größe	Knoten	Kreuzungspunkte
8x8	12	64
64x64	192	4096
128x128	448	16384

Tabelle 3.1. Benötigten Knoten beim Banyan-Netz und Kreuzungspunkte beim Kreuzschienenverteiler

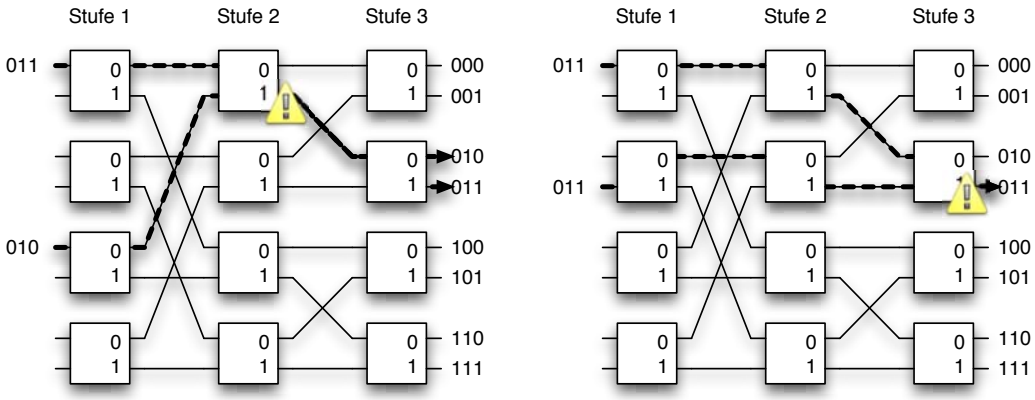


Abbildung 3.8. Blockierungen von Leitungen bei Banyan-Netzen

Um das Auftreten von Blockierungen zu vermeiden oder zu reduzieren, können einige Vorkehrungen getroffen werden:

- In allen Knoten existiert ein Paketpuffer (*buffered Banyan*). Klassisch existiert zusätzlicher Speicher zur Pufferung genau eines Pakets pro Zugangsleitung.

- Es werden mehrere Banyan-Netze parallel betrieben.
- Die interne Geschwindigkeit für die Datenweiterleitung wird erhöht.
- Der Zufluss von Daten wird durch vorgeschaltete Warteschlangen gedrosselt.
- Vor das Banyan-Netz wird ein *Verteilnetz (distribution network)* geschaltet.

Sort/Banyan-Netze

Die letzte genannte Vorkehrung zum Umgang mit Blockierungen wird nun am Beispiel von *Sort/Banyan-Netzen* näher betrachtet. Die grundlegende Idee für diese Netze ist, dass Blockierungen unter den folgenden Bedingungen vermieden werden:

- Die Pakete werden gemäß ihrer Zieladresse vorsortiert.
- Die Pakete werden nach dem Sortiervorgang in geeigneter Weise an das Banyan-Netz übergeben (*shuffle exchange*).
- Keine zwei Pakete sind für die gleiche Ausgangsleitung bestimmt.

Abbildung 3.9 zeigt, wie die Vorsortierung durch ein *Batcher-Netz* realisiert werden kann. Die einzelnen Koppeleinrichtungen geben je zwei Pakete nach Zieladresse sortiert an den Ausgangsleitungen aus. Der Pfeil gibt in dieser Darstellung an, ob die Sortierung auf- oder absteigend erfolgt. Man beachte, dass unter den o.g. Voraussetzungen im Batcher-Netz keine Blockierungen auftreten können. Nachdem die Pakete das Batcher-Netz durchlaufen haben, liegen sie sortiert am Ausgang vor und werden nach einem bestimmten Schema (shuffle exchange) an das Banyan-Netz übergeben.
Für Sort/Banyan-Netze werden zusätzlich zu den $\lceil \mathrm{ld}(n) \rceil$ Banyan-Stufen noch

$$\left\lceil \frac{1}{2} \cdot \mathrm{ld}(n)^2 + \mathrm{ld}(n) \right\rceil$$

Batcher-Stufen benötigt. Auch bei den Batcher-Stufen werden jeweils $\frac{n}{2}$ Knoten pro Stufe benötigt.

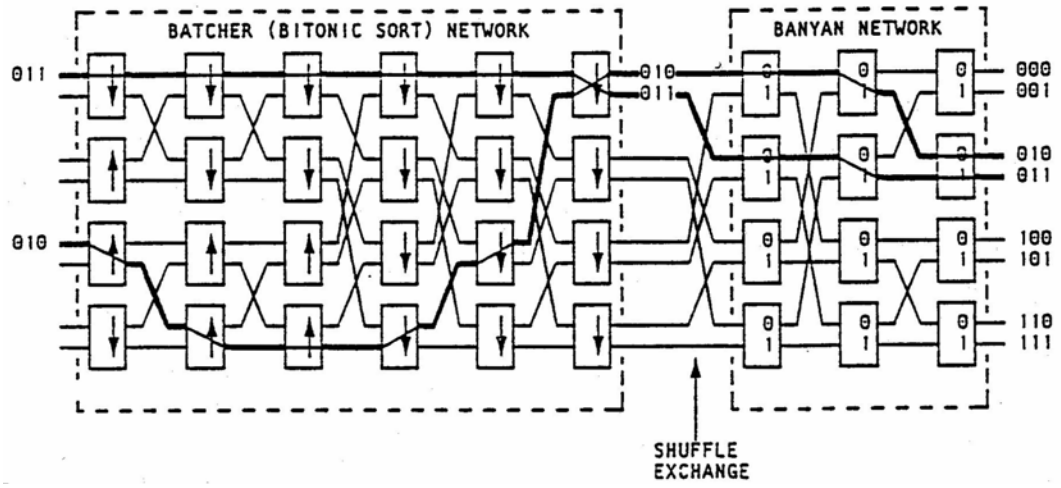


Abbildung 3.9. Struktur eines Sort/Banyan-Netzes

3.2.3 Bus

Bussysteme werden besonders häufig für die rechnerinterne Datenübertragung genutzt. Einsatzgebiete sind die Interkonnektion von Prozessoren und Hauptspeichermoduln (auch für Multiprozessorsysteme), die Kommunikation mit anderer Hardware (z.B. PCI, peripheral component

interconnect) und die Kommunikation mit Peripheriegeräten (z.B. USB, universal serial bus). Da bei Bussystemen die Nachrichtenvermittlung wegen des gemeinsamen Mediums trivial ist, stellt die Zugriffskontrolle ein zentrales Problem dar (vgl. Kapitel 4). Die folgenden Kriterien dienen der Klassifikation von Bussystemen:

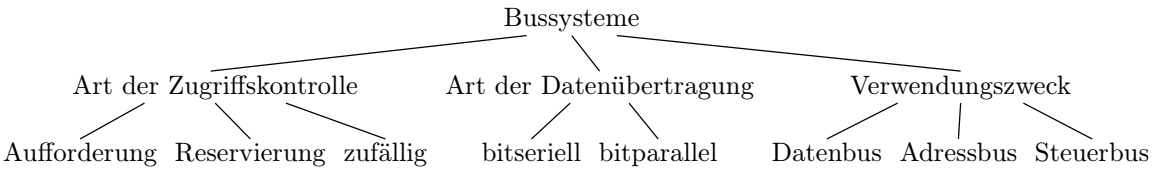


Abbildung 3.10 zeigt Darstellungen für Bussysteme mit unterschiedlichem Detaillierungsgrad. In der zweiten Verfeinerungsstufe wird (weitgehend) von der Zahl der parallelen Leitungen abstrahiert, in der dritten Stufe zusätzlich vom Verwendungszweck der Leitungen.

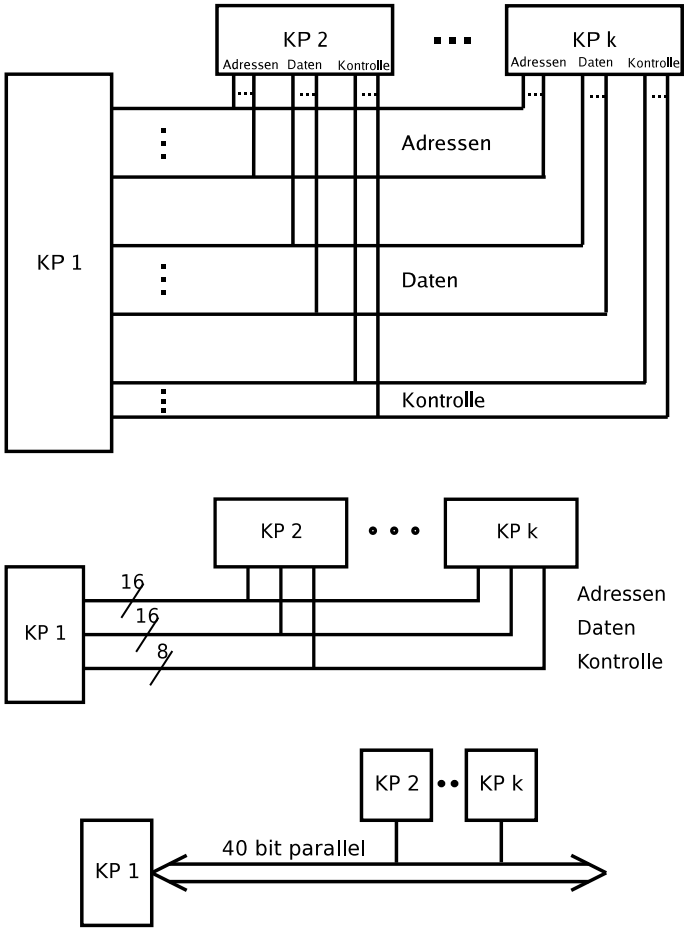
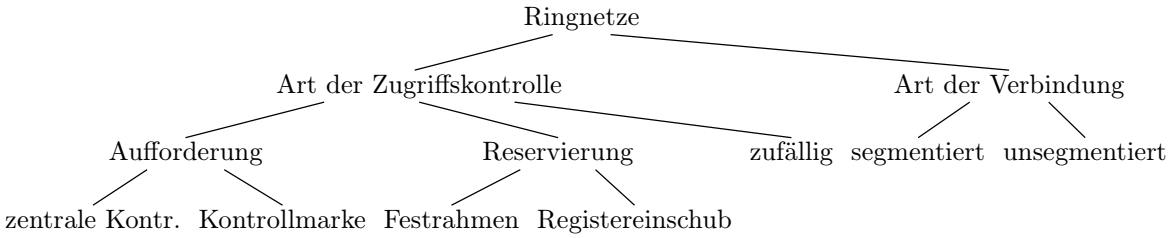


Abbildung 3.10. Unterschiedliche Darstellung von Bussystemen

3.2.4 Ring

Auch Ringe werden zur rechnerinternen Kommunikation verwendet. Die Zugriffskontrolle ist auch hier das zentrale Problem. Zur Klassifikation von Ringnetzen betrachten wir daher die Zugriffskontrolle:



Die Arten der Zugriffskontrolle werden im Kapitel 4 genauer erklärt. Bei einem segmentierten Ring existieren Punkt-zu-Punkt-Verbindungen zwischen benachbarten Stationen. Die einzelnen Stationen geben die empfangenen Daten aktiv weiter. Beim unsegmentierten Ring stellt das Medium (z.B. die Kupferleitung) tatsächlich einen Ring dar. Die Kopplung ist hier passiv. Meist wird in diesem Fall eine zentrale Steuerung benötigt, um beispielsweise Dateneinheiten wieder vom Ring zu entfernen. Abbildung 3.11 illustriert den Unterschied zwischen segmentierten und unsegmentierten Ringen.

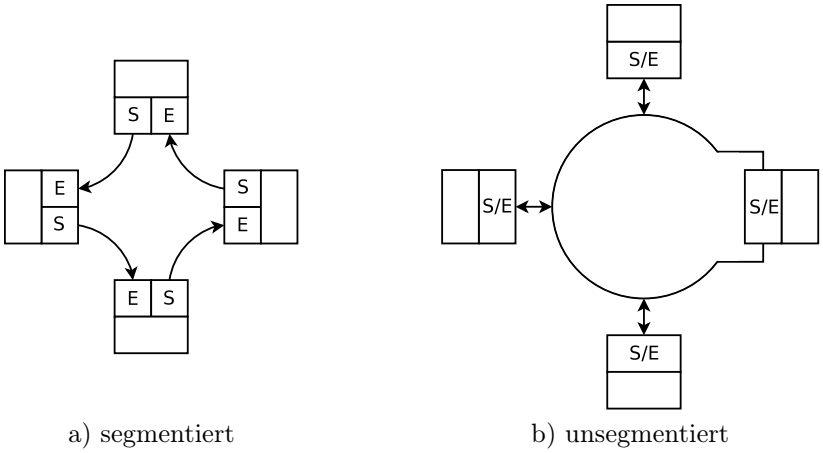


Abbildung 3.11. Segmentierter und unsegmentierter Ring

3.2.5 Gemeinsamer Speicher

Besonders bei Multiprozessorsystemen (siehe Abb. 3.12), aber auch bei Peripheriegeräten (DMA, direct memory access) ist die Kommunikation über gemeinsamen Speicher verbreitet. Zentrale Probleme sind hierbei die Realisierung eines geeigneten Kommunikationssystems und die Anwendung von temporären Zugriffssperren zum wechselseitigen Ausschluss (mutual exclusion) von Schreibvorgängen unterschiedlicher Kommunikationspartner. Als Kommunikationssystem können individuelle Leitungen, Kreuzschienenverteiler oder ein Bus verwendet werden.

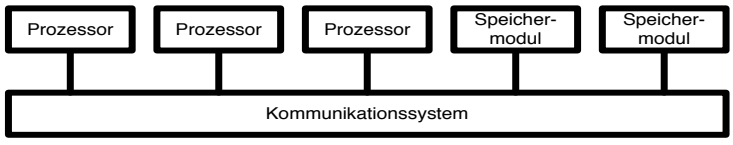


Abbildung 3.12. Multiprozessorsystem (typische Grobstruktur)

Zur Reduktion der benötigten Zugriffssperren kann der gemeinsame Speicher stärker modularisiert werden. So wird der Speicher in Abbildung 3.13 in Bereiche für unterschiedliche Zielrechner aufgeteilt. Außerdem wird ein Zustandsflag gesetzt, das das Auslesen von Nachrichten, die gerade geschrieben werden, verhindern kann.

Beispiel 3.1. Der Prozess Q_4 in Rechner R_2 hat eine Nachricht MSG_1 an den Prozess P_7 in Rechner R_1 gesendet. Die Antwort MSG_2 von Prozess P_7 wurde in die zweite Zeile des Bereichs für Nachrichten an R_2 geschrieben.

	Zustand	Ziel	Nachricht	Absender	
1	leer				an R_1
2	leer				↓
3	voll	P_7	MSG_1	Q_4, R_2	
⋮					
n	leer				
1	leer				an R_2
2	voll	Q_4	MSG_2	P_7, R_1	↓
3	leer				
⋮					
n	leer				
⋮					

Abbildung 3.13. Beispiel einer geeigneten Strukturierung des gemeinsamen Speichers

3.2.6 Hypercube

Ein Hypercube ist eine Topologieform, die einen "Würfel" n-ter Dimension darstellt. Er besteht aus 2^n Knoten, die jeweils n ausgehende Verbindungen besitzen. Die Gesamtzahl der Verbindungen beträgt damit $\frac{n \cdot 2^n}{2} = n \cdot 2^{n-1}$. Abbildung 3.14 zeigt Hypercubes unterschiedlicher Dimensionen. Man sieht, dass ein Hypercube der Dimension $n + 1$ entsteht, wenn man einen Hypercube der Dimension n dupliziert und die korrespondierenden Knoten miteinander verbindet. Die Vorteile des Hypercubes sind

- eine relativ einfache Wegeermittlung,
- eine relativ gute Zuverlässigkeit durch Alternativpfade,
- eine geringe Anzahl[3.5] von Knotendurchquerungen (hops) und
- eine relativ geringe Anzahl benötigter Verbindungen.

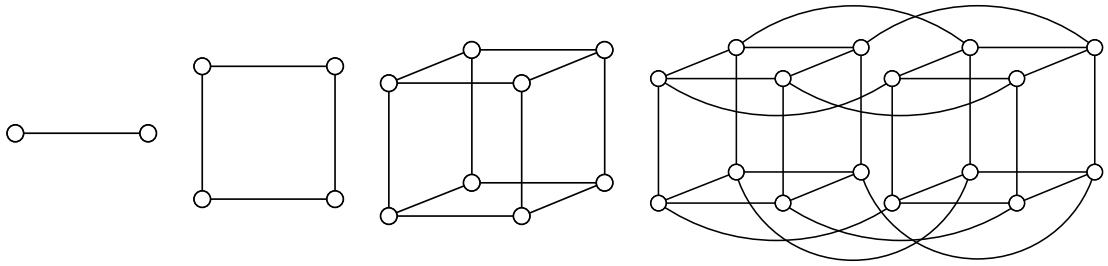


Abbildung 3.14. Hypercubes der Dimensionen n

3.3 Architekturen von Vermittlungsrechnern

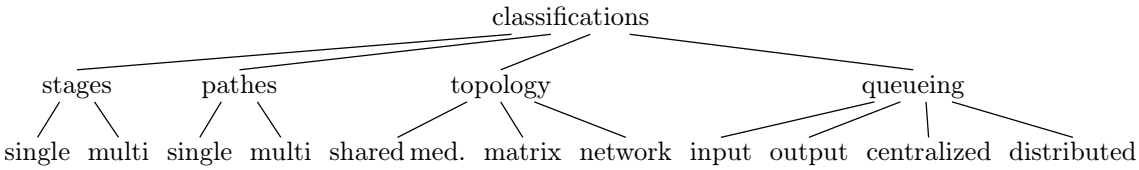
Als Vermittlungsrechner werden Spezialrechner eingesetzt, um die hoch effiziente und zuverlässige Weiterleitung der Dateneinheiten zu gewährleisten. Zu ihnen gehören u.a. IP-Router, ATM-Vermittlungsrechner, Fast Ethernet Switches und Hubs. In diesem Abschnitt wird die interne Architektur beschrieben. Die Aspekte der Wegeermittlung werden in Kapitel 5 betrachtet.

3.5. maximal n bei einem Hypercube der Dimension n

Bei Vermittlungsrechnern stehen sich zunächst die Leitungsvermittlungstechnik (auch Durchschaltetechnik) und die Nachrichtenvermittlung (auch Datagrammtechnik) gegenüber. Bei der Leitungsvermittlung kann im einfachsten Fall (nach Einrichtung der physikalischen Verbindung) der gesamte Datenverkehr der Verbindung an die selbe Ausgangsleitung geleitet werden. Bei der Nachrichtenvermittlung kann im schwierigsten Fall die Ausführung eines adaptiven Algorithmus für jedes einzelne Datagramm nötig sein.

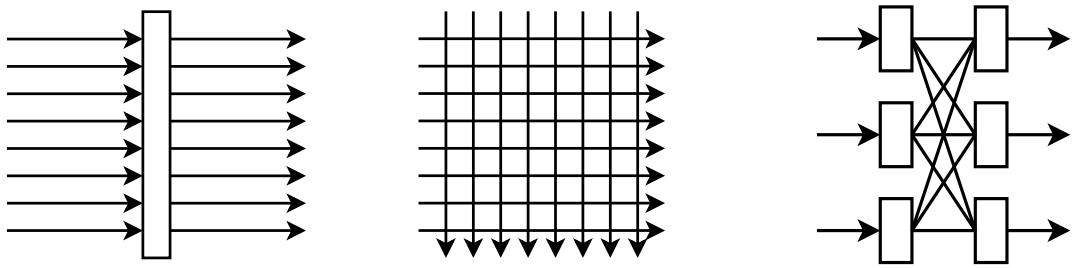
Eine neue Tendenz ist die *optische Vermittlung*, bei der eine optische Datenübertragung sowohl an den Schnittstellen des Vermittlungsrechner als auch intern verwendet wird. Mit dieser Technik wird der Durchsatz solcher Vermittlungsrechner die Größenordnung von TBit/s erreichen. Der Fokus liegt hier allerdings auf den heute üblichen "elektrischen" Vermittlungsrechnern.

3.3.1 Klassifikation von Vermittlungsrechnern



Als erstes bietet es sich an, Vermittlungsrechner nach der Zahl der *verwendeten Stufen* (single stage, multi stage) und nach der Zahl der *zeitgleich möglichen Übertragungen* (single path, multi path) zu klassifizieren. So arbeiten Bus- und Ringsysteme einstufig und bieten nur gleichzeitig eine Übertragung. Kreuzschienenverteiler arbeiten einstufig und unterstützen dabei mehrere gleichzeitige Übertragungen. Auch die besprochenen Banyan-Netze bieten mehrere gleichzeitige Übertragungen, allerdings werden mehrere Stufen verwendet.

Eine weitere Unterscheidung kann anhand der Topologie des Systems getroffen werden. Wir unterscheiden *gemeinsames Medium*, *Kreuzschienen-Matrix* und *Netze aus Vermittlungsknoten*. Schematische Darstellungen dieser Topologie sind in Abbildung 3.15 angegeben.



a) gemeinsames Medium b) Kreuzschienen-Matrix c) Netz aus Vermittlungsknoten

Abbildung 3.15. Topologieformen für Schaltsterne

Sofern das System Mechanismen zur Zwischenspeicherung von Dateneinheiten vorsieht, lässt sich die Positionierung der Warteschlagen als Kriterium für eine Klassifikation nutzen. In einem Banyan-Netz mit Zwischenspeicher können die Warteschlangen entweder vor den Koppeleinrichtungen oder danach positioniert werden. Letztere Variante hat den Vorteil, dass die Dateneinheit sofort abgesendet werden kann, wenn ihre Ausgangsleitung zur Verfügung steht. Außerdem können Warteschlagen auf das System verteilt sein (so wie im vorangegangenen Beispiel) oder zentral gehalten werden. Beispielsweise könnte einem Banyan-Netz ein zentraler Speicher vorgeschaltet werden, der die Dateneinheiten mit einer begrenzten Rate an das Banyan-Netz leitet.

Kapitel 4
Kommunikation in lokalen Rechnernetzen

In diesem Kapitel werden unterschiedliche Aspekte lokaler Rechnernetze beleuchtet. Der Fokus liegt hier auf der Zugriffskontrolle – also den Mechanismen zur Organisation des Zugriffs unterschiedlicher Rechner auf ein gemeinsames Medium, sowie den Vermittlungstechniken. Im Abschnitt 1.2 wurde die Klasse der lokalen Rechnernetze (LANs) durch eine geographische Ausdehnung der Größenordnung von Gebäuden und Grundstücken charakterisiert. Außerdem wird ein lokales Rechnernetz von einer einzelnen Institution betrieben.

Die Datenrate liegt bei dieser Klasse von Netzen typischerweise zwischen 10 MBit/s und 10 GBit/s. Die Datenraten bei lokalen Rechnernetzen mit drahtloser Übertragung sind meist deutlich niedriger. Als Topologien werden Bus, Ring, Stern und Baum verwendet (siehe Abbildung 1.8). Als Übertragungsmedien dienen insbesondere:

- verdrillte Drähte
- Koaxialkabel
- Glasfaser
- Rundfunk
- Infrarotübertragung
- Laserstrecken
- Richtfunk

Wegen der unterschiedlichen Übertragungsmedien und Topologien existieren auch diverse unterschiedliche Mechanismen der Zugriffskontrolle, darunter zirkulierende Kontrollmarke (Token Ring, FDDI, Token Bus) und CSMA/CD (Ethernet, Fast Ethernet, Gigabit Ethernet) die in den Abschnitten 4.2 und 4.3 betrachtet werden.

Topologie	Domäne	Standard	Zugriffskontrolle	Medium
Stern	Universität	ISO	CSMA/CD	Glasfaser
Baum	Büro	IEEE	Kontrollmarke	Twisted Pair
Ring	Industrie	NBS	Festrahmen	Koaxialkabel
Bus		EIA		
		ECMA		

Abbildung 4.1. Aspekte der Realisierung leitungsgebundener LANs

4.1 Standards für lokale Rechnernetze

Die folgenden Abbildungen geben einen Überblick über die Standardisierungsorganisationen und

deren hierarchische Organisation. Insbesondere der Aufbau der Projektgruppe 802 der IEEE ist von großer Bedeutung für die Standardisierung lokaler Rechnernetze.

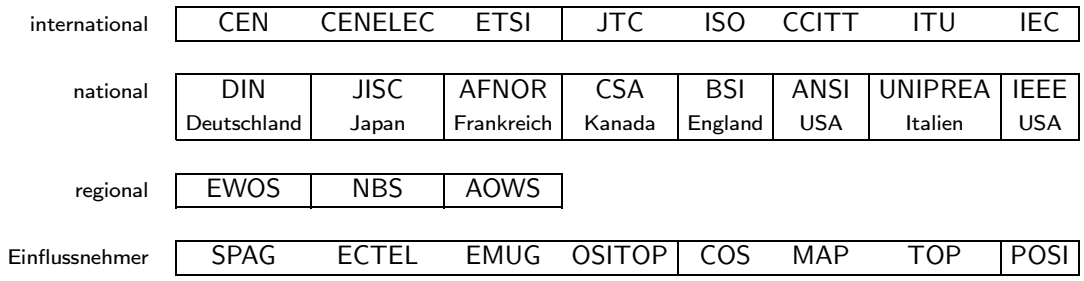


Abbildung 4.2. Standardisierungsorganisationen und Einflussnehmer

Nummer	Arbeitsgruppen	Aktuelle Tätigkeit
802.1	Higher Layer LAN Protocols	
802.2	Logical Link Control (LLC)	*zur Zeit inaktiv*
802.3	CSMA/CD und 100BaseT	Standardisierung des Gigabit Ethernet für IEEE 802.3z abgeschlossen. Standard 802.3ad – Link aggregation
802.4	Token Bus	zur Zeit inaktiv
802.5	Token Ring	Standardisierung des 100 Mbit/s High Speed Token Ring (HSTR) als IEEE 802.5t abgeschlossen (IEEE 802.5u wurde gelöscht).
802.6	Metropolitan Area Network (MAN)	*zur Zeit inaktiv*
802.7	Broadband Tag (BBTAG)	*zur Zeit inaktiv*
802.8	Fiber Optic Tag (FOTAG)	erarbeitet praktische Empfehlungen zur Glasfasertechnik
802.9	Integrated Services LAN (ISLAN)	
802.10	Standard for Inoperative LAN Security	Abschlussaktivitäten
802.11	Wireless LAN (WLAN)	erarbeitet Vorschläge zur Verbesserung der Übertragungsgeschwindigkeit
802.12	Demand Priority Access Method	
802.13	*nicht verwendet*	
802.14	Kabelmodems – Datenkommunikation über TV-Kabelnetze	Basis-Standard fertiggestellt.
802.15	Wireless Personal Area Network (WPAN)	Im Frühjahr 1999 ins Leben gerufen. Kommunikation verschiedener Funk-Endgeräte (PC, Handy, Pager usw.) untereinander.
802.16	Broadband Wireless Access	Im Frühjahr 1999 ins Leben gerufen. Spezifikation des physikalischen und des MAC-Layers zur Definition von Standards für die Kabellose Breitbandkommunikation.

Tabelle 4.1. Übersicht über die Projektgruppe 802 des IEEE (Stand: 2000)

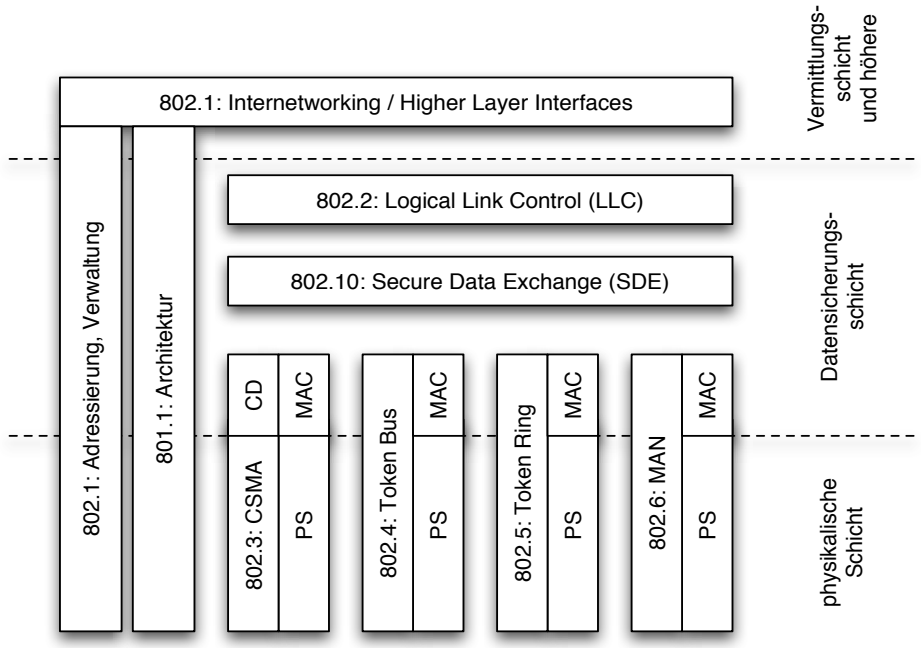


Abbildung 4.3. IEEE 802 –Architekturmodell für LANs (Inzwischen sind noch weitere wichtige Standards wie 802.11 auf der physikalischen und der Datensicherungsschicht hinzugekommen)

4.2 Zugriffskontrolle in Ringnetzen

4.2.1 Aufbau, Zweck und Grobbeurteilung von Ringnetzen

Bei einer *Ringtopologie* wird ein gemeinsames Übertragungsmedium benutzt, das einen geschlossenen (meist unidirektionalen) Ring bildet. Jede der kommunizierenden Instanzen ist dabei über das Übertragungsmedium mit genau zwei Kommunikationspartnern direkt verbunden. Die Menge von Verarbeitungselementen (Terminals, PCs, Workstations, periphere Geräte usw.), die in Form einer Ringtopologie untereinander verbunden sind, heißt *Ringnetz*.

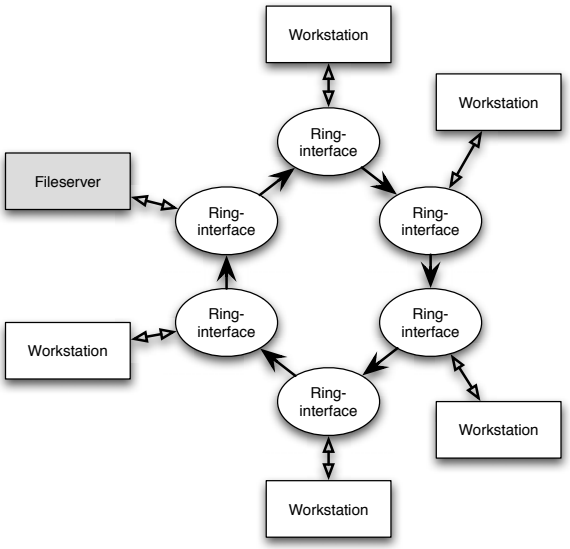


Abbildung 4.4. Beispiel eines Ringnetzes

Abbildung 4.4 zeigt die schematische Übersicht eines Ringnetzes. Als *Stationen* werden Workstations, PCs und Server eingesetzt. Als Übertragungsmedien können beispielsweise Koaxialkabel oder Lichtleiter dienen. Die Datenrate liegt typischerweise zwischen 1 MBit/s und 100 MBit/s.

In der Abbildung werden die angeschlossenen Stationen entlastet, indem spezielle *Ringinterfaces* verwendet werden. Die Ringinterfaces übernehmen Aufgaben wie Empfang, Versand, Adressierung, Weiterleitung und Löschung der Dateneinheiten, sowie die Verstärkung des die Daten repräsentierenden Signals.

Das Hauptproblem bei Ringnetzen stellt die Sychnchronistation der konkurrierenden Zugriffswünsche dar. Im Abschnitt 3.2 wurden grundsätzlich zwei Lösungsvarianten unterschieden: Ringnetze mit zentraler Kontrollinstanz und Ringnetze mit gleichberechtigten Stationen. Diese werden in den Abschnitten 4.2 genauer betrachtet.

Vorteile von Ringnetzen Ein wesentlicher Vorteil von Ringnetzen ist die unproblematische Wegeermittlung (Es existieren keine alternativen Pfade). Auch das Rundsenden (broadcast) ist innerhalb eines Ringes einfach zu realisieren. Da die Dateneinheiten zwischen Sender und Empfänger nicht zwischengespeichert werden ist keine Sättigungskontrolle notwendig. Darüber hinaus sind i.a. hohe Datenübertragungsraten und eine digitale Übertragung möglich. Überdies stellen Ringnetze meist eine kostengünstige und einfach erweiterbare Lösung dar.

Nachteile von Ringnetzen Die größten Nachteile sind eine mangelhafte Zuverlässigkeit, da der Ausfall eines Knotens weniger einfach zu handhaben ist. Hinzu kommt bei manchen Ringnetztypen das spezielle Problem, das bei einem Verlust des Zugriffsrechts (z.B. in Form einer Kontrollinstanz oder Kontrollmarke) entsteht. Das kann entweder ein Ausfall der zentralen Kontrollinstanz oder ein Verlust der Kontrollmarke sein.

4.2.2 Ring mit zentraler Kontrollinstanz

Bei dieser Form der Realisierung übernimmt (genau) eine Station die Rolle einer zentralen Kontrollinstanz (master). Der Master vergibt dann Zugriffsrechte an die anderen Stationen (slaves). Ein Datenaustausch findet ausschließlich zwischen Master und Slaves statt. Die Kommuikation zwischen den Slaves untereinander findet indirekt über den Master als Vermittler statt.

Vorteile Dieses Konzept ist sehr einfach zu realisieren. Insbesondere wird die Adressierung vereinfacht, da der Master nur den Empfänger adressieren muss und die Slaves nur den Absender (die eigene Adresse) angeben müssen.

Nachteile Da bei einem Ausfall der zentralen Kontrollinstanz jede weitere Kommunikation unmöglich ist, lässt die Zuverlässigkeit dieses Konzepts zu wünschen übrig.

Varianten Die Reservierung von *Zeitscheiben (time slots)* kann auf zwei Arten geschehen: statisch oder dynamisch. Bei *statischer Reservierung (static assignment)* werden Zeitscheiben a priori für die Stationen reserviert (synchronous time division multiplexing, siehe Abschnitt 2.9). Diese Variante ist sehr einfach. Da die Zuordnung von Zeitscheiben auf Stationen global bekannt ist, entfällt die Adressierung. Statische Reservierung ist allerdings wenig effizient, da Zeitscheiben ungenutzt bleiben, wenn die entsprechende Station nicht sendewillig ist. Dieser Effekt wird bei *dynamischer Reservierung (demand assignment)* vermieden. In diesem Fall signalisieren die Slaves dem Master ihre Sendewünsche. Dies kann aktiv aus Sicht der Slaves geschehen oder passiv, indem der Master deren Sendebereitschaft erfragt.[4.1]

In den folgenden Abschnitten werden Ringnetze mit gleichberechtigten Stationen behandelt. Die Mechanismen der Zugriffskontrolle werden dadurch aufwändiger.

4.2.3 Token Ring

Der *Token Ring* stellt zur Zeit die wichtigste Realisierungsform von Ringnetzen dar. Zirkulierende Kontrollmarken wurden bereits 1969 durch Farmer/Newhall eingeführt. Im Jahr 1997 basierten noch ca 30% aller LANs auf Token Ring.

Als Zugriffskontrollmechanismus wird eine *zirkulierende Kontrollmarke* verwendet. Die Kontrollmarke berechtigt den Besitzer zur Nutzung des gemeinsamen Mediums. Eine Weitergabe der Kontrollmarke erfolgt z.B. wenn ein Knoten nicht sendewillig ist oder schon eine (oder sämtliche) Dateneinheiten abgesendet hat. Für die Weitergabe des Zugriffsrechts exitstieren zwei Varianten:

1. über eine *dedizierte Steuerleitung*, also einem zusätzlichen Ring
2. unter Verwendung einer *Kontrollmarke*, also durch ein spezielles Bitmuster

Vorteile Der wesentliche Vorteil eines zirkulierenden Zugriffsrechts ist die Möglichkeit des Sendens von Dateneinheiten beliebiger Länge. Oft ist es allerdings sinnvoll, eine obere Grenze für die Länge festzulegen.

Nachteile Insbesondere bei großen Ringen und hohen Datenraten ist es oft möglich, dass mehrere Stationen gleichzeitig senden (ohne dass sich die Signale auf dem Ring überlagern). In dieser Realisierungsvariante ist allerdings zu jedem Zeitpunkt nur maximal eine Station sendeberechtigt. Der Verlust und die Duplizierung der Kontrollmarke (z.B. durch Ausfall der markenbesitzenden Station) muss gehandhabt werden. Außerdem kann ein Fairness-Problem entstehen, wenn ein Ringinterface die Kontrollmarke festhält.

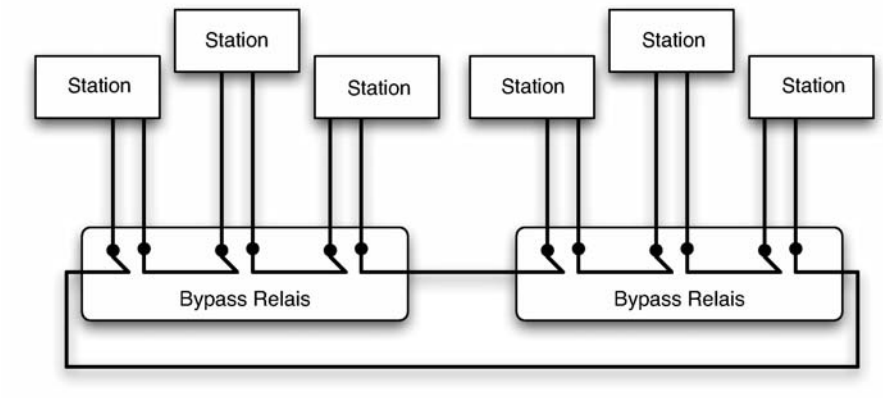


Abbildung 4.5. Aufbau eines IBM Token Ring

4.1. Der Unterschied zu Multidrop-Verbindungen mit Polling ist gering.

Zustände des Ringinterfaces

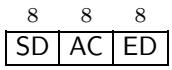
Im fehlerfreien Fall befindet sich ein Ringinterface in einem von zwei Zuständen:

- Sendemodus (im Besitz der Kontrollmarke)
- Abhörmodus (auf Suche nach einer Kontrollmarke)

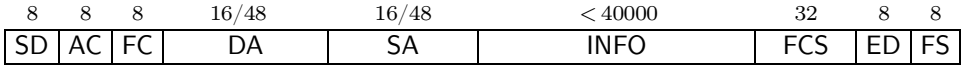
Im *Sendemodus* wird eine Dateneinheit gesendet und ein Timer gestartet. Danach wird gewartet, bis die eigene Dateneinheit (nach vollständiger Umrundung des Rings) eintrifft. Wenn die Dateneinheit unverfälscht eintrifft, wird die Kontrollmarke weitergegeben. Andernfalls erfolgt eine Sonderbehandlung.

Im *Abhörmodus* werden sämtliche Dateneinheiten analysiert. Ist die Station sendewillig und wird eine freie Kontrollmarke empfangen, so wird die Kontrollmarke (durch Kippen eines Bits) als "belegt" gekennzeichnet und die Station geht in den Sendemodus über.

Zusätzlich kann man Prioritäten einführen. Nach einer Sendung umrundet die Kontrollmarke den Ring dann einmal mit gesetztem Prioritätsbit. In diesem Zustand "verwendet" eine Station die Kontrollmarke nur, wenn sie hoch priorisierte Sendewünsche hat. Wird die Kontrollmarke innerhalb von einer Runde nicht verwendet, wird das Prioritätsbit gelöscht.



a) Kontrollmarke



b) Datenrahmen

Acronym	Bezeichnung
SD	start delimiter
ED	end delimiter
AC	access control
FC	frame control
SA/DA	source/destination address
FS	frame status

c) spezielle Symbole

Abbildung 4.6. Aufbau der Protokolldateneinheiten bei Token Ring

Signalverzögerung im Ringinterface Passiert eine freie Kontrollmarke eine sendewillige Station, so muss diese dafür sorgen, dass die Kontrollmarke so verändert wird, dass die folgenden Stationen sie nicht mehr als eine freie Kontrollmarke interpretieren. Dazu ist eine ständige Verzögerung des Signals um mindestens 1 Bit nötig. Andernfalls würde im Moment des Erkennens der freien Kontrollmarke (mit dem letzten Bit), diese schon an die Nachbarstation weitergesendet werden. Aus diesem Grund kommt der Mechanismus der zirkulierenden Kontrollmarke auch nur segmentierte Ringe in Frage.

Entnahme der Dateneinheiten vom Ring Damit eine Dateneinheit nicht endlos im Ring zirkuliert, muss sie (nachdem sie ihren Empfänger erreicht hat) vom Ring entfernt werden. Dies kann vom Sender der Dateneinheit übernommen werden. In diesem Fall erhält der Sender durch die Dateneinheit (die er mit der abgesendeten vergleichen kann) auch eine Quittung. Alternativ kann

die Dateneinheit auch vom Empfänger oder durch den darauffolgenden Sender entfernt werden.

Überwachungsknoten Zur Initialisierung, Synchronisation und insbesondere zur Regeneration einer verlorenen Kontrollmarke kann eine Station die Rolle eines Überwachungsknotens übernehmen. In diesem Fall verzichtet man auf vollständig gleichberechtigte Stationen.

Leistungsfähigkeit lokaler Rechnernetze auf der Basis von Token Ring

Für die Beurteilung der Effizienz von Rechnernetzen auf der Basis von Token Ring ist die Zahl der Dateneinheiten, die sich gleichzeitig "auf dem Ring befinden" können, interessant. Diese Anzahl wird durch den *Parameter a* repräsentiert. Sie läßt sich errechnen, indem die Dauer der vollständigen Umrundung eines Signals durch die Dauer der Sendung einer Dateneinheit geteilt wird. Alternativ kann auch die Gesamtlänge des Rings durch die "Länge" des Signals einer Dateneinheit geteilt werden. In beiden Fällen hängt a von den folgenden Größen ab.

- der Datenrate $v_D \left[\frac{\text{bit}}{\text{s}}\right]$
- der Länge des Rings $l\,[\text{m}]$
- der Länge einer Dateneinheit $L\,[\text{bit}]$
- der Signalausbreitungsgeschwindigkeit $c\left[\frac{\text{m}}{\text{s}}\right]$

Wir wählen als Symbol für die Signalausbreitungsgeschwindigkeit hier c, weil sie in den betrachteten Systemen zumindest sehr grob der Lichtgeschwindigkeit entspricht. Der Parameter a errechnet sich wie folgt.

$$a \;=\; \frac{l}{c} \Big/ \frac{L}{v_D} \;=\; \frac{l \cdot v_D}{c \cdot L}$$

Der Quotient $\frac{l}{c}$ entspricht der Dauer einer Ringumrundung. Multipliziert mit der Datenrate v_D ergibt sich die Zahl der Bits, die sich gleichzeitig auf den Ring befinden können. Teilt man diesen Wert durch die Länge der Dateneinheiten in Bit, ergibt sich die Anzahl der Dateneinheiten, die sich gleichzeitig auf dem Ring "befinden" können.

In kleinen und langsamen Ringnetzen liegt der Parameter a weit unter eins. Tabelle 4.2 zeigt einige Beispiele (als Signalausbreitungsgeschwindigkeit werden 2/3 der Lichtgeschwindigkeit angenommen).

Datenrate	Ringlänge	Rahmengröße	Parameter a
4 MBit/s	20 km	1 kByte	0,05
100 MBit/s	400 m	0,5 kByte	0,05
1 GBit/s	10 km	1 kByte	6,25

Tabelle 4.2. Der Parameter a für unterschiedliche Netzgrößen

Liegt der Parameter a weit über eins, ist die Effizienz eines Token Rings niedrig, da nur maximal eine Station sendeberechtigt ist. Bei höheren Datenraten und wachsenden Ringlängen kann die Rahmengröße erhöht werden. Daraus folgt natürlich nicht in jedem Fall, dass die größeren Rahmen auch von den Stationen ausgenutzt werden können.

Stattdessen kann auch die Kontrollmarke sofort nach Absenden der Dateneinheit weitergegeben werden (early token release). Auf diese Weise können mehrere Dateneinheiten gleichzeitig auf dem Ring "liegen". Abbildung 4.7 zeigt, wie die Ausnutzung des Ringnetzes mit wachsendem Parameter a sinkt. Die Ausnutzung erhöht sich wegen der frühen Weitergabe der Kontrollmarke bei einer wachsenden Anzahl von Stationen.

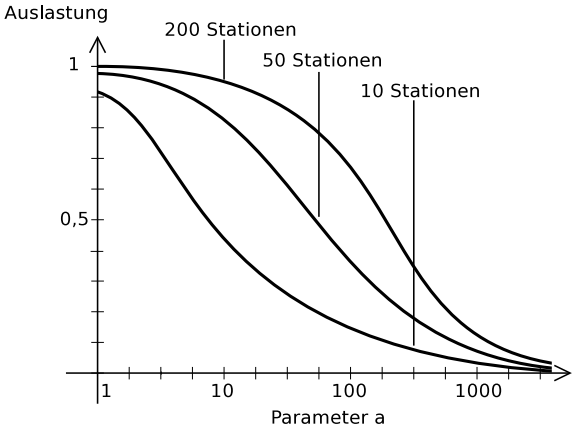


Abbildung 4.7. Ausnutzung eines Ringnetzes

Der Parameter a ist auch für Bussysteme relevant. Damit die Stationen auftretende Konflikte noch während der Übertragung feststellen können, muss a kleiner als $\frac{1}{2}$ sein.

4.2.4 Ring mit zufälligem Zugriff

Im vorigen Abschnitt wurde schon angedeutet, dass (besonders für große und schnelle Ringnetze) die Kontrollmarke direkt nach dem Absenden einer Dateneinheit weitergegeben werden kann. Beim *Ring mit zufälligem Zugriff (contention ring)*[4.2] ist es einer Station außerdem erlaubt, sofort zu senden, wenn das Medium nicht belegt ist. Das sendewillige Ringinterface hört also das Medium ab.

- Wenn das Medium frei ist
  - wird sofort eine Dateneinheit gesendet und
  - eine Kontrollmarke angehängt.
- Wenn das Medium als belegt erkannt wurde
  - wird auf eine Kontrollmarke gewartet,
  - diese in eine Verbindungsmarke (connector) umgewandelt,
  - eine Dateneinheit angehängt und
  - eine neue Kontrollmarke angehängt.

Die Dateneinheit wird nach einer vollständigen Ringumrundung vom Sender absorbiert.

Konflikte Auch durch das Abhören des Mediums kann nicht ausgeschlossen werden, dass sich die Sendungen von zwei Stationen überlagern. Solche *Zugriffskonflikte* können allerdings von den Absendern erkannt werden, wenn die gesendete Dateneinheit mit der zu absorbierenden verglichen wird. Im Falle eines Zugriffskonflikts bietet es sich an, dass beide Stationen ein zufälliges Zeitintervall warten und die Sendung erneut versuchen.

4.2.5 Ring mit Festrahmenzirkulation

Beim Ring mit Festrahmenzirkulation[4.3] werden bei der Initialisierung *Rahmen* fester Größe auf den Ring gebracht, die diesen kontinuierlich umrunden. Eine sendewillige Station wartet auf einen freien Rahmen, markiert diesen als benutzt, verwendet ihn zum Senden der Dateneinheit und markiert ihn nach einer Umrundung wieder als frei (anstatt das Signal zu absorbieren).

4.2. nach CLARK

4.3. eingeführt durch PIERCE, ca. 1972, BELL LABS

Ist T_0 die Signallaufzeit einer vollständigen Ringumrundung, werden n Zeitscheiben der Länge T mit $n \cdot T \leq T_0$ gebildet. Für diese Zeitscheiben werden bei der Initialisierung Rahmen generiert und als frei markiert. Diese umlaufen den Ring im Idealfall endlos nach dem "Waggon-Prinzip" – wie hintereinander fahrende Waggons auf einer kreisförmigen Eisenbahnstrecke.

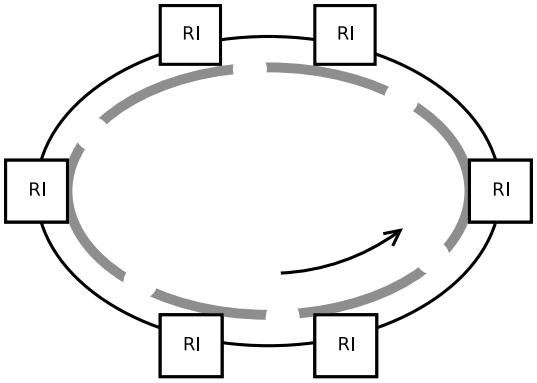


Abbildung 4.8. Momentanaufnahme eines Rings mit Festrahmenzirkulation

Bewertung Im Gegensatz zu anderen Varianten von Ringnetzen wird das gemeinsame Übertragungsmedium fair aufgeteilt. Durch die feste Größe der Rahmen werden Stationen daran gehindert, das Medium mit sehr langen Dateneinheiten zu blockieren. Die konstante Rahmengröße impliziert allerdings auch die Notwendigkeit der Fragmentierung und Reassemblierung der Dateneinheiten. Außerdem kann die Effizienz durch nur teilweise gefüllte Rahmen sinken.

4.2.6 Ring mit Registereinschub

Die grundsätzliche Idee des Registereinschubs (register insertion, delay insertion) ist es, den Ring dynamisch zu verlängern und zu verkürzen. Soll eine Dateneinheit gesendet werden, wird der Ring verlängert und die Dateneinheit auf das neue Teilstück gelegt. Hat die Dateneinheit den Ring komplett umrundet, wird sie absorbiert, indem der Ring verkürzt wird. Auf diese Weise sollen die Vorteile des Token Ring mit denen der Festrahmenzirkulation kombiniert werden.

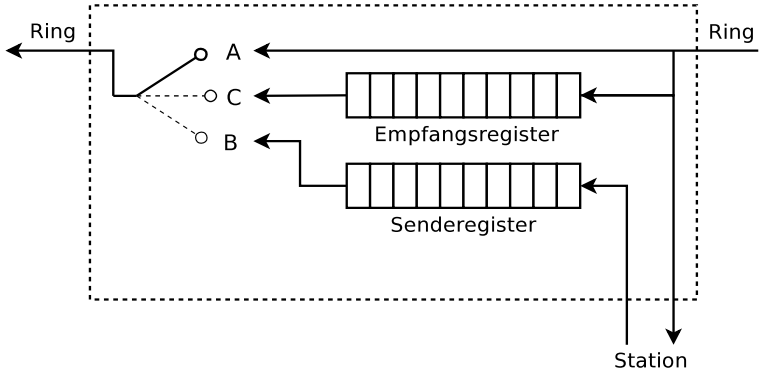


Abbildung 4.9. Ringinterface mit Registereinschub

Abbildung 4.9 zeigt den möglichen Aufbau eines Ringinterfaces. Es besitzt ein Sende- und ein Empfangsregister zur Pufferung von Daten. Im folgenden werden die drei Zustände und deren Übergänge beschrieben. Andere Übergänge als $A \rightarrow B$, $B \rightarrow C$ und $C \rightarrow A$ sind nicht erlaubt.

A) Im Zustand A werden die Signale (wie beim unsegmentierten Ring) direkt weitergeleitet (und außerdem von der Station empfangen). Eine abzusendende Dateneinheit wird zunächst im Senderegister gepuffert.

B) Ist das Senderegister voll, wechselt das Ringinterface in den Zustand B und beginnt seine Sendung. Die Daten anderer Stationen werden nicht mehr weitergeleitet, sondern im Empfangsregister gepuffert. So wird der Ring um das Empfangsregister verlängert.

C) Nachdem die Daten aus dem Senderegister gesendet wurden, wechselt das Ringinterface in den Zustand C. Die Daten des Empfangsregisters werden nun so lange weitergeleitet, bis sich die eigene Nachricht nach einer vollständigen Ringumrundung komplett im Empfangsregister befindet. In diesem Moment wird der Zustand A eingenommen. Auf diese Weise wird die eigene Sendung absorbiert und der Ring wieder verkürzt.

Neben der beschriebenen Variante existieren anspruchsvollere Varianten mit der Möglichkeit der Akkumulation der Lücken zwischen Dateneinheiten. So kann die gesendete Dateneinheit größer als das Empfangsregister sein.

Bewertung Die Vorteile des Rings mit Registereinschub sind eine faire Aufteilung und eine gute Auslastung des Übertragungsmediums sowie geringe Verweilzeiten für Dateneinheiten zwischen Sender und Empfänger. Die Länge der Dateneinheiten ist zwar variabel, jedoch durch die Größe der Register limitiert. In der hier beschriebenen Variante ist pro Station immer nur maximal eine Dateneinheit auf dem Ring. Bei einer kleinen Zahl von Stationen und einem großen Wert des Parameters a führt das zu einer mangelhaften Ausnutzung.

4.2.7 Aufbau von Ringinterfaces

Der Aufbau eines Ringinterfaces soll nun genauer betrachtet werden. Wie bereits erwähnt wird das Ringinterface zur Entlastung der Station von dieser "entkoppelt". Das Ringinterface selbst teilt sich nun in eine *netzorientierte Komponente (NOK)* und eine *stationsorientierte Komponente (SOK)*. Die Aufgaben der netzorientierten Komponente sind

- Sendekontrolle: Mechanismen zur Aufbereitung der Dateneinheiten, Markierung von Festrahmen, Erkennen und Umwandeln von Kontrollmarken, Erzeugung von Prüfsummen
- Empfangskontrolle: Mechanismen zur Annahme von Dateneinheiten, Überprüfung von Prüfsummen, Absorbierung von Dateneinheiten
- Adressanalyse: Mechanismen zur Auswertung von Adressen im Kopf der Dateneinheiten
- Signalformung: Mechanismen zur Umwandlung von digitalen in analoge Signale, Gewinnung des Bit-Taktes

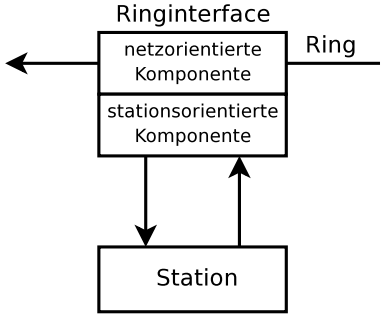


Abbildung 4.10. Schematischer Aufbau eines Ringinterfaces

Die stationsorientierte Komponente stellt eine Möglichkeit zur Kommunikation zwischen netzorientierter Komponente und angeschlossener Station bereit. Der Datenaustausch erfolgt hier mit hoher Datenrate (meist mindestens 1 MBit/s). Häufig wird vollduplex-DMA von den Ringinterfaces eingesetzt.

4.3 Zugriffskontrolle in Bus- und Broadcast-Systemen

In diesem Abschnitt werden Bus- und Broadcast-Systeme behandelt. Für drahtlose Kommunikation, bei der mehr als zwei Stationen direkt miteinander kommunizieren können (im Gegensatz zu Punkt-zu-Punkt-Verbindungen beim Richtfunk), wird meist die Bezeichnung Broadcast *gewählt*. Da diese Systeme bezüglich der Zugriffskontrolle den leitungsgebundenen Bussystemen sehr ähnlich sind, werden sie hier gemeinsam behandelt. Wenn im folgenden von Bussystemen oder Bustopologien gesprochen wird, sind damit beide Arten von Systemen gemeint.

4.3.1 Aufbau, Zweck und Grobbeurteilung von Bussystemen

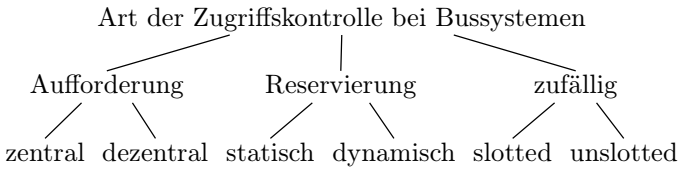
Bei einem Kommunikationssystem mit *Bustopologie* kommunizieren mindestens drei Kommunikationspartner über ein gemeinsames Übertragungsmedium. Jeder Kommunikationspartner ist dabei mit sämtlichen anderen Kommunikationspartnern direkt verbunden. Eine Menge von Stationen, die in Form einer Bustopologie miteinander verbunden sind, heißt *Bussystem*. Wie bei Ringnetzen erfolgt auch hier der Anschluss an das Übertragungsmedium meist indirekt über dedizierte *Businterfaces*.

Die Datenraten bei Bussystemen liegen meist über 1 MBit/s. Als Übertragungsmedien werden u.a. Koaxialkabel und Lichtleiter verwendet. Außerdem werden terrestrische Funksysteme und Satellitenverbindungen genutzt. Wie bei Ringnetzen stellt die Zugriffskontrolle das Hauptproblem dar.

Vorteile von Bussystemen Die Vorteile von Bussystemen decken sich größtenteils mit denen der Ringnetze (siehe Abschnitt 4.2). Die Erweiterbarkeit ist äußerst einfach und die Kosten steigen i.a. weniger als proportional zur Zahl der angeschlossenen Stationen. Im Gegensatz zu den Ringnetzen stellt der Ausfall einer Station bei Bussystemen kein Problem dar.

Nachteile von Bussystemen Bei Bussystemen gibt es keine alternativen Übertragungswege, so dass sie beim Ausfall des Übertragungsmediums (Beschädigung der Leitung, längerfristige Störung bei Funksystemen) nicht mehr funktionsfähig sind. Außerdem kann das Übertragungsmedium bei hohem Verkehrsaufkommen zum Engpass der Kommunikation werden. Eine Erhöhung der Übertragungskapazität ist oft schwieriger zu erreichen, als bei anderen Systemen.

In Abschnitt 3.2 haben wir drei Arten der Zugriffskontrolle in Bussystemen genannt. Diese werden hier nun noch genauer unterteilt[4.4] und beschrieben:



4.3.2 Bus mit Aufforderungsverfahren

Bei Aufforderungsverfahren unterscheiden wir drei Realisierungsvarianten, die in Abbildung 4.11 illustriert werden. Sie werden hier für den Fall einer zentralen Kontrolle beschrieben. Abbildung 4.12 zeigt, wie dieselben Varianten auch im dezentralen Fall umgesetzt werden.

Daisy Chain. Die Stationen sind in einer Kette miteinander verbunden (daisy chain). Die erste Station erhält von der zentralen Kontrolle eine Aufforderung. Sie kann diese Aufforderung für eine Übertragung nutzen oder sie an ihren Nachfolger weitergeben. Jede weitere Station, die eine Aufforderung erhält, wählt zwischen Übertragung oder Weitergabe der

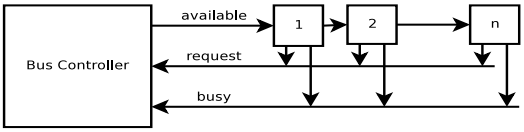
4.4. nach Luczak

Aufforderung. Es ist offensichtlich, dass Stationen, die weit hinten in der Kette positioniert sind, deutlich benachteiligt werden.

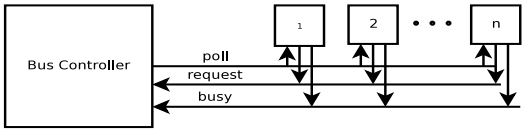
Polling. Bei einer passiven Form des Polling wird von einer Zentrale periodisch ein Signal generiert. Dieses Signal wird über einen Bus (clock) an alle Stationen gesendet. Die erste Station die sendewillig ist, äußert ihren Wunsch, indem sie ein Signal auf einen weiteren Bus (busy) legt. Jede weitere sendewillige Station erkennt durch Abhören des zweiten Bus, dass die Aufforderung (clock) schon vergeben ist.

Bei einer aktiven Variante des Polling senden die Stationen auf einem Bus (request) ein Signal an die Zentrale. Über einen zweiten Bus (poll count) sendet die Zentrale die Anzahl der noch unerfüllten Sendewünsche. Die Stationen erkennen die Erfüllung eines Sendewunsches an einem dritten Bus (busy). Sie speichern die Anzahl der unerfüllten Wünsche zum Zeitpunkt ihrer Anfrage. Sind alle fremden Sendewünsche erfüllt, kann eine Station mit ihrer eigenen Sendung beginnen.

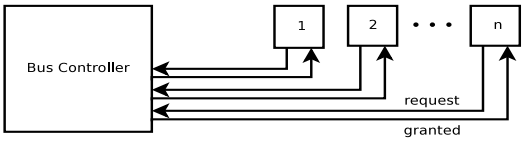
individuelle Leitungen. Die Stationen besitzen individuelle Leitungen, um der Zentrale ihren Übertragungswunsch mitzuteilen und um von der Zentrale eine Aufforderung zu erhalten (independent request). Diese Variante erfordert viele Leitungen.



a) Daisy Chain



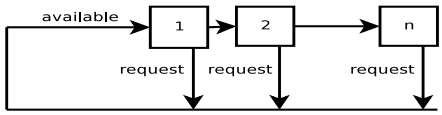
b) Polling



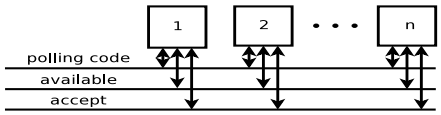
c) individuelle Leitungen

Abbildung 4.11. Aufforderungsverfahren mit zentraler Kontrolle

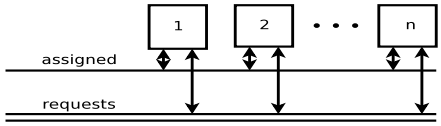
Es folgen die Abbildungen für den dezentralen Fall. Man beachte, dass die hier beschriebene Unterscheidung in der Regel nicht eindeutig auf existierende Verfahren anwendbar ist. Kombinationen sind möglich und oft sind unterschiedliche Interpretationen gerechtfertigt.



a) Daisy Chain



b) Polling



c) individuelle Leitungen

Abbildung 4.12. Aufforderungsverfahren ohne zentrale Kontrolle

4.3.3 Bus mit zufälligem Zugriff

Beim Bus mit zufälligen Zugriff erfolgt keine Vergabe von Zugriffsrechten. Stattdessen findet ein zufälliger Zugriff der Stationen auf das Übertragungsmedium statt. Da in diesem Fall Konflikte beim gleichzeitigen Zugriff durch zwei Stationen entstehen, besteht die Hauptaufgabe der Zugriffskontrolle in der *Erkennung* und der *Auflösung* solcher Konflikte.

Reduktion von Zugriffskonflikten

Ähnlich wie bei Ringnetzen werden oft feste Zeitscheiben definiert, indem die Gesamtzeit in Intervalle (slots) der Dauer T unterteilt wird[4.5]. Die sendende Station fragmentiert dann die Dateneinheiten in Teile, die während einer Zeitscheibe gesendet werden können. Die Übertragung eines Teils beginnt immer am Anfang einer Zeitscheibe. Auf diese Weise wird die Häufigkeit von Konfliktsituationen reduziert.

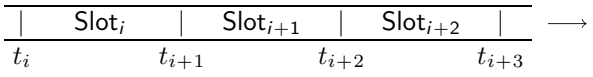
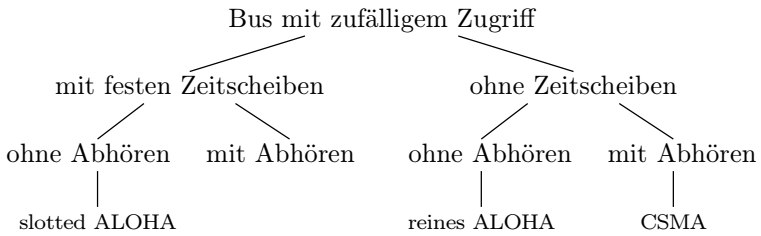


Abbildung 4.13. Aufteilung der Gesamtzeit in Intervalle

Ein weiterer naheliegener Mechanismus zur Reduktion der Konfliktsituationen besteht im Abhören des Übertragungsmediums vor Beginn einer Übertragung. Ist das Übertragungsmedium belegt, wird die eigene Übertragung verzögert. Dieser Mechanismus ist im Bereich von Funknetzen oft nicht einsetzbar, weil fremde Übertragungen weit entfernter Stationen nicht wahrnehmbar sind (und sich dennoch überlagern). Wir verfeinern nun die Klassifikation für Bussysteme mit zufälligem Zugriff:



Auflösung von Zugriffskonflikten

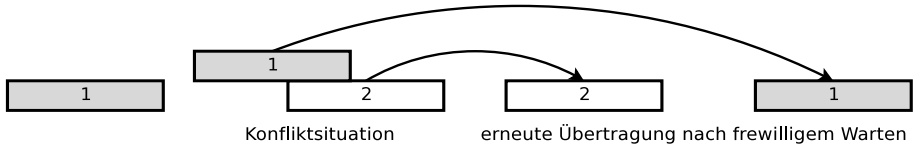
Tritt ein Zugriffskonflikt auf, wird dieser in der Regel durch Warten und erneutes Senden aufgelöst. Offensichtlich kommt es zu einem erneuten Konflikt, wenn die Wartezeit der am Konflikt beteiligten Stationen identisch ist. Wir betrachten vier Möglichkeiten der Auflösung von Konflikten

1. Verzögerung um eine zufällig gewählte Wartezeit (RTI, retransmission time interval).
2. Adaptive Verzögerung um eine Wartezeit, die sich bei erneutem Auftreten eines Konfliktes erhöht. Die Adaption kann auf lokalen oder netzweiten Beobachtungen beruhen.
3. Verzögerung gemäß einer gesetzten Knotenpriorität.
4. Wahl eines anderen Mechanismus der Zugriffskontrolle (wie Reservierung).

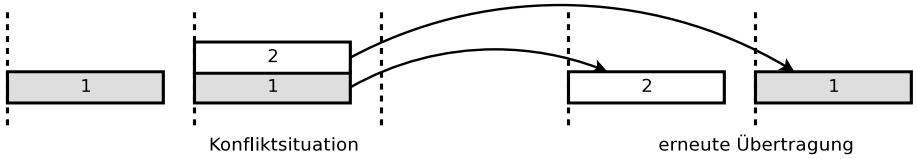
ALOHA-Verfahren

Im folgenden werden unterschiedliche Varianten des ALOHA-Verfahrens betrachtet. Es handelt sich hierbei immer um zufälligen Zugriff ohne Abhören des Mediums. Zur Konfliktauflösung wird die erste Variante (zufällige Wartezeit) verwendet. Es soll verdeutlicht werden, warum die Konflikthäufigkeit bei der Verwendung fester Zeitscheiben sinkt und damit der Durchsatz steigt.

4.5. Bei gegebener Datenrate kann die Dauer einer Zeitscheibe auch in Bit angegeben werden.



a) reines Aloha



b) slotted Aloha

Abbildung 4.14. Konfliktsituation bei reinem ALOHA und slotted ALOHA

Abbildung 4.14 zeigt das Entstehen und die Auflösung eines Konflikts. In beiden Fällen wird eine erneute Übertragung nach einer zufälligen Wartezeit durchgeführt. Im Fall von slotted ALOHA beginnt allerdings jede Übertragung am Anfang einer Zeitscheibe.

Wir möchten nun den maximal erreichbaren Durchsatz von ALOHA-Verfahren berechnen. Dazu nehmen wir an, dass das Auftreten eines Sendewunsches ein stochastischer Prozess ist. Außerdem nehmen wir an, dass die Sendewilligkeit einer Station stochastisch unabhängig von der Sendewilligkeit anderer Stationen ist. G_i bezeichne die Wahrscheinlichkeit eines Sendewunsches der Station B_i.

Offensichtlich hängt der maximal erreichbare Durchsatz von den Wahrscheinlichkeiten G_i ab: Ist $G_x = 1$ und $G_y = 0$ für alle $y \neq x$, so ist eine Station permanent sendewillig – alle anderen nie. Mit diesen Wahrscheinlichkeiten läßt sich ein Durchsatz von 1 erreichen. Es ist ebenfalls nachvollziehbar, dass der Durchsatz sinkt, wenn zu viel oder zu wenig Sendewilligkeit besteht.

Wir berechnen nun die Wahrscheinlichkeit S_i für einen erfolgreichen Zugriff durch die Station B_i. Ein Zugriff ist erfolgreich, wenn die Station B_i sendewillig ist und alle Stationen B_j mit $j \neq i$ nicht sendewillig sind, also gilt im Falle der Unabhängigkeit der Stationen im Zugriffsverhalten:

$$S_i = G_i \cdot \prod_{j \neq i} (1 - G_j)$$

Satz 4.1. *Seien B_1 und B_2 Stationen mit deutlich unterschiedlichen Sendewahrscheinlichkeiten G_1 und G_2.*

$$G_1 = x \cdot G_2 \qquad x \gg 1$$

Die Wahrscheinlichkeit S für eine erfolgreiche Sendung wächst monoton mit x.

Beweis. Es ist $S_1 = G_1 \cdot (1 - G_2)$ und $S_2 = G_2 \cdot (1 - G_1)$. Damit ist

$$\begin{aligned} S &= S_1 + S_2 \\ &= G_1 \cdot (1 - G_2) + G_2 \cdot (1 - G_1) \\ &= (x \cdot G_2) \cdot (1 - G_2) + G_2 \cdot (1 - (x \cdot G_2)) \\ &= x \cdot G_2 - x \cdot G_2 \cdot G_2 + G_2 - G_2 \cdot x \cdot G_2 \\ &= x \cdot \left[G_2 - 2 \cdot (G_2)^2 \right] + G_2 \end{aligned}$$

□

Satz 4.2. *Sind die Wahrscheinlichkeiten für einen Sendewunsch für alle n Stationen mit $n \to \infty$ identisch und voneinander unabhängig, so liegt der maximal erreichbare Durchsatz beim* slotted ALOHA *bei*

$$\frac{1}{\mathrm{e}} \text{ mit } \mathrm{e} = 2,7182...$$

Beweis. Nach Voraussetzung ist $G_i = \frac{G}{n}$ und $S_i = \frac{S}{n}$ für alle Stationen B_i.
und damit

$$\begin{aligned} S &= S_i \cdot n \\ &= G_i \cdot \left(1 - \frac{G}{n}\right)^{n-1} \cdot n \\ &= G \cdot \left(1 - \frac{G}{n}\right)^{n-1} \end{aligned}$$

Wir betrachten nun den Grenzfall $n \to \infty$, also den Fall unendlich vieler Stationen. Es ist dann

$$\begin{aligned} S &= G \cdot \lim_{n\to\infty} \left(1 - \frac{G}{n}\right)^{n-1} \\ &= G \cdot \lim_{n\to\infty} \left(1 + \frac{-G}{n}\right)^{n} \\ &= G \cdot \mathrm{e}^{-G} \end{aligned}$$

Wir suchen nun das Maximum von S, indem wir die Ableitung S' gleich null setzen.

$$S' = -G \cdot \mathrm{e}^{-G} + \mathrm{e}^{-G} = 0 \implies G = 1$$

Die Erfolgswahrscheinlichkeit S ist also maximal für $G = 1$ und der maximale Durchsatz $S_{\max}$ ist:

$$S_{\max} = G \cdot \mathrm{e}^{-G} = 1 \cdot \mathrm{e}^{-1} = \frac{1}{\mathrm{e}}$$

□

Aus dem Ergebnis von Satz 4.2 schließen wir nun auf den maximal erreichbaren Durchsatz beim reinen ALOHA. Hier argumentieren wir, dass die Zahl der störenden Zugriffe während einer Übertragung doppelt so hoch ist, wie beim slotted ALOHA. Der Grund dafür ist, dass das *Verwundbarkeitsintervall* – also der Zeitraum in dem eine weitere Sendung zu einem Konflikt führen würde – doppelt so groß ist.

Die Verwundbarkeitsintervalle werden in Abbildung 4.15 und 4.16 veranschaulicht. Solche Zugriffswünsche, die zu einem Konflikt führen würden, sind fett gedruckt. Offenbar führt jeder Zugriffswunsch während der Übertragung zu einem Konflikt (es wird schließlich sofort gesendet). Auch einige Zugriffswünsche, die vor der von uns betrachteten Übertragung auftreten führen zu Konflikten.

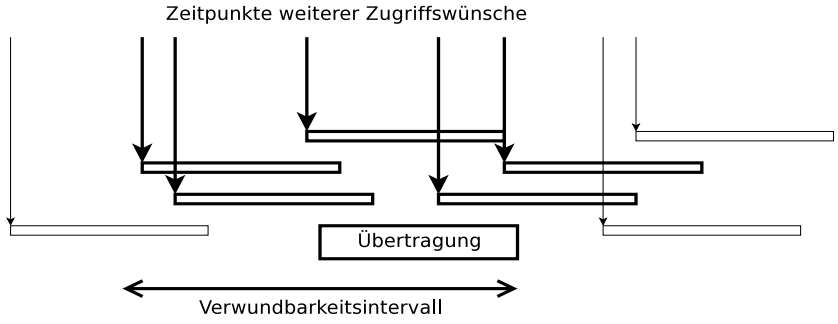


Abbildung 4.15. Verwundbarkeitsintervalle beim reinen ALOHA

Anders sieht es beim slotted ALOHA aus. Jeder Zugriffswunsch wird auf den Anfang des nächsten Zeitscheibe verschoben (in der Abbildung durch horizontale Pfeile dargestellt). Offensichtlich führen nur diejenigen Zugriffswünsche zu einem Konflikt, die im Zeitslot vor der betrachteten Übertragung entstehen. Alle Zugriffswünsche, die während der Übertragung auftreten, stören diese nicht.

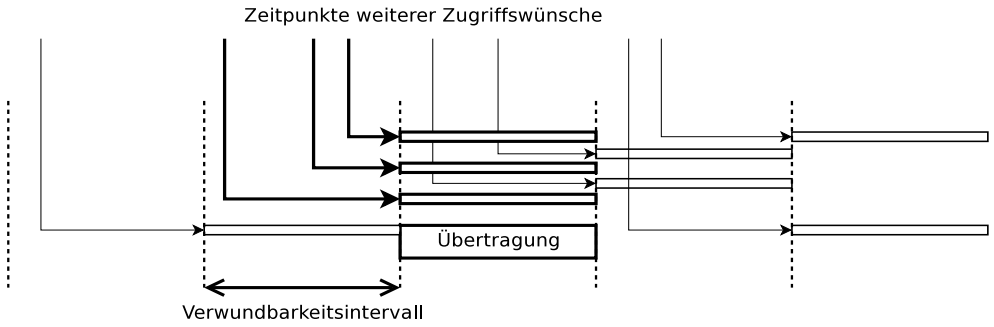


Abbildung 4.16. Verwundbarkeitsintervalle beim slotted ALOHA

Satz 4.3. *Sind die Wahrscheinlichkeiten für einen Sendewunsch für alle n Stationen mit* $n \to \infty$ *identisch, so liegt der maximal erreichbare Durchsatz beim* reinen ALOHA *bei*

$$\frac{1}{2\,\mathrm{e}} \text{ mit } \mathrm{e} = 2,7182...$$

Er ist damit halb so groß, wie bei slotted ALOHA.

Beweis. Da die Zugriffswahrscheinlichkeit wegen des doppelt so großen Verwundbarkeitsintervalls zweimal so groß ist, gilt analog zum Beweis von Satz 4.2

$$S \;=\; G \cdot \left(1 - \frac{2G}{n}\right)^{n-1}$$

Für den Grenzfall unendlich vieler Stationen gilt ebenfalls analog

$$S \;=\; G \cdot \mathrm{e}^{-2G}$$

Die Erfolgswahrscheinlichkeit S ist also maximal für $G = \frac{1}{2}$ und der maximale Durchsatz $S_{\max}$ ist

$$S_{\max} \;=\; G \cdot \mathrm{e}^{-2G} \;=\; \frac{1}{2} \cdot \mathrm{e}^{-2 \cdot \frac{1}{2}} \;=\; \frac{1}{2\,\mathrm{e}}$$

□

Mobilfunk

Einen ganz anderen Weg zur Vermeidung von Zugriffskonflikten wird u.a. bei der Mobilkommunikation mit Funktelefonen gegangen. Hier wird das Übertragungsmedium durch Multiplexing für die Kommunikationspartner konfliktfrei nutzbar gemacht.

- Das Gesamtgebiet wird in *Zellen* eingeteilt. Diese werden oft schematisch als Sechsecke dargestellt. Es findet also eine Form von *Raummultiplexen* statt.
- Jede Zelle Z_i verwendet einen Frequenzbereich F_i. Keine zwei benachbarten Zellen benutzen denselben Frequenzbereich: $\forall_{i,j} F_i \cap F_j = \emptyset$, wenn Z_i und Z_j benachbart sind. Innerhalb einer Zelle wird der Frequenzbereich durch *Frequenzmultiplexen* in kleinere Bereiche aufgeteilt.
- Zusätzlich werden die kleineren Frequenzbereiche durch *Zeitmultiplexen* auf die Kommunikationspartner aufgeteilt.

Bewegen sich die Kommunikationspartner über Zellgrenzen hinaus, ist ein Wechsel des Frequenzbereichs nötig. Dieser Vorgang wird "handover" genannt. Im einfachen Fall erfragt die Mobilstation bei der neuen Basisstation einen neuen Frequenzbereich. Ist diese Basisstation voll ausgelastet bricht die Verbindung ab (hard handover). Um das zu verhindern, kann die Mobilstation die Verbindung zur vorherigen Basisstation erst abbauen, wenn die Verbindung zur neuen Basisstation hergestellt wurde (soft handover). Dazu muss die Mobilstation allerdings in zwei Frequenzbereichen gleichzeitig senden können.

Abbildung 4.17 zeigt die Einteilung eines Gebietes in Zellen. Der Sendebereich der mittleren Zelle geht über die Grenzen zu den Nachbarzellen hinaus. Daher ist der Frequenzbereich 1 dort nicht nutzbar. Die hier gewählte Zuordnung von Frequenzbereichen auf Zellen genügt der Bedingung, dass keine zwei benachbarten Zellen denselben Frequenzbereich nutzen. Weitere Details zur Mobilkommunikation folgen in Kapitel 6.

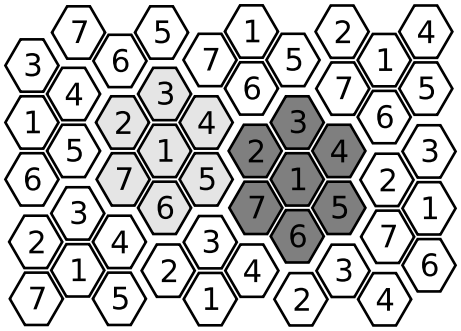


Abbildung 4.17. Einteilung eines Gebietes in Zellen

CSMA-Verfahren

Das Prinzip von CSMA-Verfahren besteht darin, dass jede Station die Möglichkeit besitzt, die Belegung des Übertragungsmediums festzustellen. Bei einem belegten Medium wird die Sendung verzögert. Varianten zur Auflösung von Zugriffskonflikten und zur Wahl der Verzögerungszeit wurden schon am Anfang dieses Abschnittes diskutiert. Wir betrachten drei Varianten noch einmal genauer.

non-persistent CSMA. Diese Variante ist sehr einfach und führt zu vielen Konflikten: Ist das Medium frei, so wird sofort mit der Sendung begonnen. Bei einem belegten Medium wird die Sendung um eine zufällig gewählte Zeit verzögert und dann ohne weiteres Abhören begonnen.

***p*-persistent CSMA.** Im Gegensatz zur ersten Variante werden hier einige Konflikte durch permanentes Abhören des Mediums vermieden: Ist das Medium frei, wird a) durchgeführt, ist es belegt, wird mit b) fortgefahren.

a) Es wird mit Wahrscheinlichkeit p mit der Sendung begonnen. Anderenfalls (mit der Wahrscheinlichkeit $1 - p$) wird die Sendung für die Dauer eines *Minislots* verzögert. Ist das Medium noch frei, wird a) wiederholt. Ist es inzwischen belegt, wird mit b) fortgefahren.

b) Es wird gewartet, bis das Medium frei ist und dann a) durchgeführt.

Stationsprioritäten. Eine dritte Möglichkeit besteht darin, die Wartezeiten der einzelnen Stationen von deren Priorität abhängig zu machen. Hoch priorisierte Stationen warten dann kürzer als niedrig priorisierte. Beim WLAN-Standard 802.11 werden beispielsweise Kontrollinformationen wie Bestätigungen höher priorisiert. Die Prioritäten der Stationen können auch im "Round-Robin"-Verfahren zyklisch vertauscht werden, wenn eine feste Priorisierung der Stationen nicht gewünscht ist.

Auch für die *Konflikterkennung* existieren unterschiedliche Möglichkeiten.

- Der Empfänger einer Sendung kann deren korrekte Übertragung mit Hilfe von Prüfsummen feststellen und deren Erhalt quittieren. Eine ausbleibende Quittung deutet dann auf einen Zugriffskonflikt hin. Diese Variante wird bei HYPERchannel verwendet.

- Der Sender kann während seiner Sendung das Übertragungsmedium abhören und Überlagerungen mit seiner eigenen Sendung feststellen. Dieser Mechanismus wird CSMA/CD (collision detection) genannt und bei Ethernet verwendet.

Ethernet

Ethernet wurde bereits 1975 von der XEROX CORP. entworfen. Ausgehend von diesem Entwurf wurde von einer Kooperation[4.6] zwischen DEC, INTEL und XEROX eine Neuspezifikation erarbeitet. 1997 basierten ca. 45% aller LANs auf Ethernet und dessen Anteil steigt seitdem weiter. Im folgenden werden einige der Entwurfsziele für Ethernet aufgelistet.

- Ethernet sollte ein *einfaches* und *konstengünstiges* Kommunikationssystem darstellen.
- Wegen des Wunsches nach Kompatibilität unterschiedlicher Implementierungen wurde auf *optionale Funktionen verzichtet.*
- Eine *flexible Adressierung* sollte es ermöglichen, einzelne Knoten, Gruppen von Knoten und alle Knoten zu adressieren.
- Es sollte ein *fairer Zugriff* der gleichberechtigten Knoten auf das Medium stattfinden.
- Es sollte eine *hohe Datenrate* bei einer *geringen Gesamtverzögerung* von Dateneinheiten zwischen Sender und Empfänger erreicht werden.
- Das System sollte auch bei hoher Belastung ein *stabiles Verhalten* zeigen.
- Eine *einfache Wartbarkeit* und ein *unkomplizierter Betrieb* sollten realisiert werden.
- Eine *Schichtenarchitektur* gemäß des ISO/OSI-Referenzmodells sollte verwendet werden.

Es folgen einigen Restriktionen, die beim Entwurf von Ethernet entstanden sind.

- Wegen des gemeinsamen Übertragungsmediums ist kein vollduplex-Betrieb möglich. Allerdings ist dies bei Stern- oder Punkt-zu-Punkt-Topologien in späteren Ethernet-Standards möglich.
- Es existiert nur eine stark eingeschränkte Fehlerkontrolle. So existiert ausschließlich eine Bitfehlererkennung, jedoch keine Fehlerkorrektur.
- Zunächst war ausschließlich die Verwendung eines Ü-Mediums mit fester Datenrate von 10 Mb/s möglich. Inzwischen werden auch 100 Mb/s (Fast Ethernet) und 1 Gbit/s (Gigabit Ethernet) verwendet. Der Trend geht zum 10 Gbit/s-Ethernet und möglicherweise darüber hinaus.
- Ursprünglich war keine Möglichkeit zur Vergabe von Prioritäten für Stationen vorgesehen. Diverse Erweiterungen des Ethernet-Standards zur Realisierung von Dienstgüte-Garantien wurden aber inzwischen erarbeitet.
- Es existiert kein Schutz gegen permanent sendende Stationen.
- Die maximale Entfernung zwischen zwei Stationen beträgt 2,5 km.
- Es können maximal 1024 Stationen betrieben werden.

Abbildung 4.18 zeigt den konventionellen Aufbau eines Ethernet-Segments. Als Übertragungsmedium dient ein Koaxialkabel, das an beiden Enden durch einen Widerstand terminiert wird. Die Stationen sind über einen *Transceiver* und ein *Ethernet Controller Board* mit dem Bus verbunden (siehe Abb. 4.19). Soll ein komplexeres LAN auf der Basis von Ethernet realisiert werden, können mehrere Segmente über *Repeater*, *Hubs* oder *Switches* miteinander verbunden werden. Inzwischen wurde dieser konventionelle Aufbau größtenteils durch Hub- oder Switch-Lösungen verdrängt.

4.6. Diese Kooperation wurde entsprechend der Anfangsbuchstaben der Unternehmen "DIX-Gruppe" genannt.

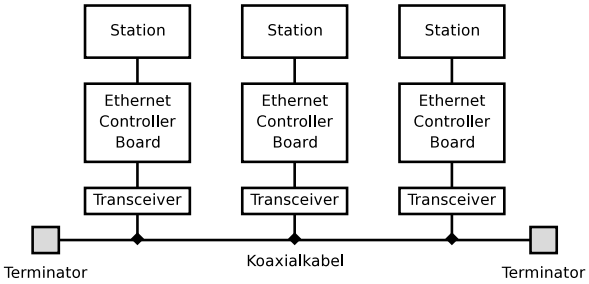


Abbildung 4.18. Konventioneller Aufbau eines Ethernet-Segments

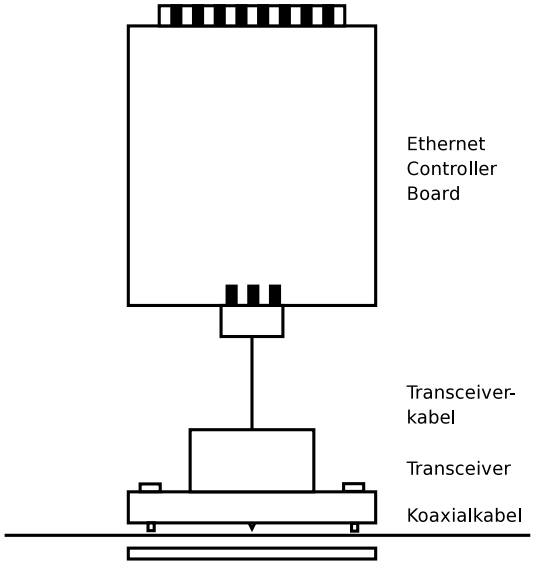


Abbildung 4.19. Komponenten zur Verbindung von Stationen mit dem Übertragungsmedium

Der heute übliche Aufbau eines Ethernet als Baumtopologie (switched ethernet) wird in Abbildung 4.20 dargestellt. Abbildung 4.21 zeigt das Format einer Dateneinheit bei Ethernet (ethernet frame). Die Präambel dient zur Synchronisation, die Absenderadresse und der Typ werden in der Datensicherungsschicht nicht interpretiert.

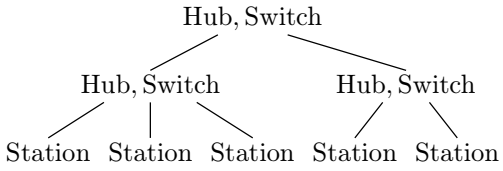


Abbildung 4.20. Hierarchischer Aufbau eines Ethernet

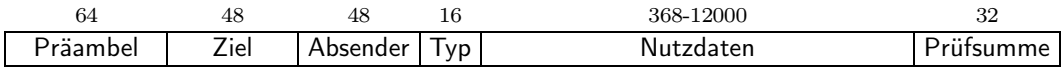
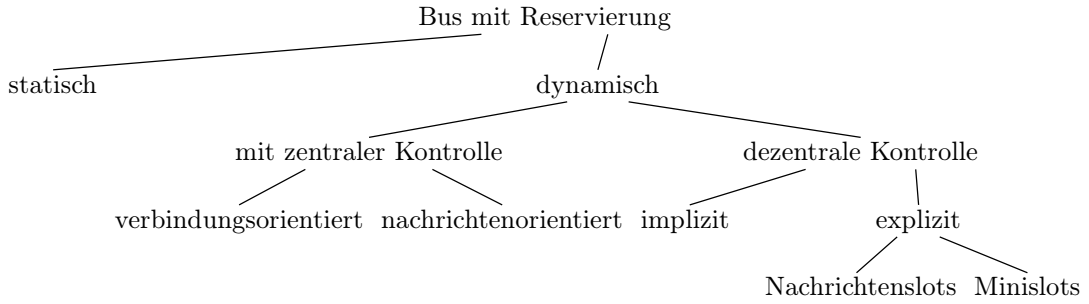


Abbildung 4.21. Format der Dateneinheiten bei Ethernet

4.3.4 Bus mit Reservierung

Nach Bussystemen mit Aufforderung und Bussystemen mit zufälligem Zugriff betrachten wir nun zuletzt Bussysteme, bei denen die Berechtigung zum Zugriff auf das gemeinsame Medium durch *Reservierung* eingeholt wird. Wir verfeinern zunächst die Klassifikation vom Anfang dieses Abschnittes.



Der *statische* Fall liegt beim schon angesprochenen TDMA bzw. FDMA (time bzw. frequency division multiple access) vor. Hier werden Zeitscheiben oder Frequenzbereiche für relativ lange Zeiträume (z.B. die Dauer eines Telefongesprächs) fest reserviert. Meist grenzen diese nicht direkt aneinander, sondern werden zur Vermeidung von Interferenzen von kleinen "Lücken" unterbrochen (siehe Abbildung 4.22). Statische Reservierung hat den großen Vorteil, dass den Kommunikationspartnern für die Dauer ihrer Reservierung ein festgelegter Durchsatz (eher) garantiert werden kann. Allerdings werden Reservierungen abgelehnt, wenn sämtliche Zeitscheiben, Frequenzbereiche o.ä. reserviert sind. Außerdem kann die Auslastung schlechter sein, wenn Stationen ihre Reservierungen nicht ausnutzen. Statische Reservierung benötigt im allgemeinen weniger aufwändige Mechanismen.

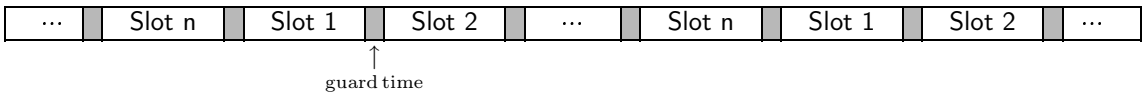


Abbildung 4.22. Statische Reservierung mit Zeitscheiben und *guard times* (grau dargestellt)

Wir betrachten deshalb nun insbesondere die *dynamische Reservierung* des Mediums. Hier unterscheiden wir noch weiter zwischen zentraler und dezentraler Kontrolle. Im zentralen Fall können die Stationen entweder ein gewünschtes Zeitintervall reservieren, um eine *Verbindung* zu etablieren oder sie reservieren ein festes Zeitintervall zur Übertragung einer *Nachricht*.

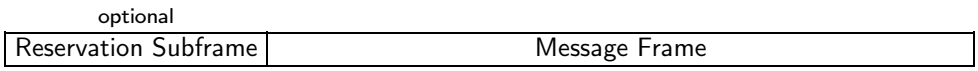


Abbildung 4.23. Grobaufbau eines Rahmens bei expliziter Reservierung

Abbildung 4.23 zeigt einen Rahmen bei *expliziter Reservierung*. Der Beginn des Rahmens wird für Reservierungen verwendet (reservation subframe). Darauf folgt der Teil der für Nutzdaten verwendet wird (message frame). Ist der Reservierungsteil optional, würde man im Sinne der obigen Klassifikation von *speziellen Minislots* sprechen. Ist er fester Bestandteil jedes Rahmens, würde man von Reservierung *innerhalb der Nachrichtenslots* sprechen.

Zuletzt betrachten wir den Fall dezentraler und impliziter Reservierung. Diese Kombination wird bei *reservation ALOHA* verwendet. Dabei handelt es sich um das in Abschnitt 4.3.3 beschriebene slotted ALHOA mit der folgenden Erweiterung: Die Stationen S_i befinden sich jeweils in einem von drei Zuständen. Eine Station befindet sich im Zustand

- Z_1, wenn sie selbst den letzten Slot erfolgreich benutzt hat,
- Z_2, wenn eine andere Station den letzten Slot erfolgreich benutzt hat,
- Z_3, wenn der letzte Slot nicht erfolgreich genutzt wurde.

Ist eine Station sendewillig, erfolgt der Zugriff zustandsabhängig:

- In Z_1 sendet die Station während die anderen Stationen warten.

– In Z_2 wartet die Station. Eine andere Station sendet oder der Slot bleibt ungenutzt.

– In Z_3 sendet die Station. Ein Zugriffskonflikt mit anderen Stationen ist möglich.

Wir bezeichnen diesen Mechanismus als implizit, weil die Stationen den folgenden Slot indirekt durch ihre Übertragung reservieren.

4.4 Lokale Netze im Hochgeschwindigkeitsbereich

Unter einem *Hochgeschwindigkeitsnetz* verstehen wir Rechnernetze mit einer Datenrate ab ca. 100 MBit/s bis 1 GBit/s.[4.7] Im lokalen Bereich werden diese oft als *HSLANs (high speed local area networks)* bezeichnet. Für Hochgeschwindigkeitskommunikation wurden drei wesentliche Standards[4.8] geschaffen, die im lokalen, regionalen und überregionalen Bereich eingesetzt werden. Diese Standards werden in diesem Abschnitt betrachtet.

- *FDDI (fiber distributed digital interface)* als Weiterentwicklung von Token Ring
- *Fast Ethernet* und *Gigabit Ethernet* als Weiterentwicklung von Ethernet
- *DQDB (distributed queue dual bus)*

Hochgeschwindigkeitsnetze besitzen meist folgende neue Merkmale:

- Bei kleiner Netzausdehnung kann die Übertragungsverzögerung auf dem Übertragungsmedium zunehmend vernachlässigt werden. Eine große Netzausdehnung läßt sich nicht unbegrenzt durch Erhöhung der Datenrate kompensieren, da die Signallaufzeit trivialerweise nicht reduzierbar ist.
- Auf physikalischer Schicht und Datensicherungsschicht existieren häufig keine Durchsatzengpässe. Die Kapazität ist typischerweise nur bei zahlreichen gleichzeitigen Bewegtbildübertragungen erschöpft.
- Echtzeitkommunikation (vgl. Kapitel 7) ist bei niedriger Netzauslastung (bezogen auf kurze Zeitintervalle im Bereich von 100 ms) relativ problemlos realisierbar.

Im folgenden werden nun einige Probleme aufgezeigt, die sich durch die erhöhte Datenrate ergeben. Dadurch soll verdeutlicht werden, dass sich die bisher beschriebenen Mechanismen nicht ohne weiteres auf Hochgeschwindigkeitsnetze übertragen lassen.

Probleme bei langsamen Stationen Bei sehr hohen Datenraten können die angeschlossenen Stationen zu Verarbeitungsengpässen werden. Beispielsweise ist es denkbar, dass ein Datei-Server zwar problemlos Daten aus dem Hauptspeicher mit 1 GBit/s senden kann, diese aber bei weitem nicht mit derselben Datenrate von den Festplatten laden kann.

Probleme bei zirkulierender Kontrollmarke Der Wert des Parameters a (vgl. Abschnitt 4.2) erhöht sich, sofern nicht die Netzausdehnung verringert oder die Paketgröße erhöht wird, was zu einer schlechteren Effizienz führen kann.

Probleme bei CSMA/CD Bei der Kollisionserkennung von CSMA/CD für Bussysteme tritt ein ähnliches Problem auf. Dazu betrachten wir den "worst case" einer Kollision. In einem Bus mit Signallaufzeit t_ρ sendet eine Station S zum Zeitpunkt t_0 einen Datenblock. Zum Zeitpunkt $t_0 + t_\rho$ erreicht dieser Datenblock die entfernteste Station E. Da der Bus kurz zuvor von E als frei erkannt wurde, beginnt E im selben Moment eine Sendung, so dass ein Zugriffskonflikt entsteht. Zum Zeitpunkt $t_0 + 2 \cdot t_\rho$ erreicht die zweite Sendung die Station S, die so den Konflikt erkennt. Sollen die Stationen sämtliche auftretenden Konflikte noch *während* der Sendung feststellen können, so

4.7. Stand: 2005

4.8. Zusätzlich sind ATM-Vermittlungsrechner für die schnelle Paketvermittlung einsetzbar.

müssen diese offensichtlich mindestens für die doppelte Dauer der Signallaufzeit senden. Aus der Datenrate v_D und der (doppelten) Signallaufzeit $2 \cdot t_\rho$ ergibt sich so die minimale Blockgröße $L_{\min}$ gemäß:

$$L_{\min} > t_\rho \cdot v_D$$

Für die beiden letztgenannten Probleme gibt es zwei mögliche Reaktionen:

- Die Blockgröße kann erhöht werden. Sehr große Blöcke können allerdings oft gar nicht komplett ausgenutzt werden, wenn die zu versendenden Dateneinheiten klein sind.
- Die Signallaufzeit kann durch Verkleinerung der Netzausdehnung reduziert werden, wie es der Fast Ethernet Standard vorsieht. Bei existierenden Netzinfrastrukturen ist dieser Weg allerdings oft mit hohen Kosten verbunden.

Beispiel 4.4. Ein Bus mit 5 km Ausdehnung hat (bei einer Signalausbreitungsgeschwindigkeit von zwei Dritteln der Lichtgeschwindigkeit) eine Signallaufzeit von

$$t_\rho = \frac{5\,\text{km}}{\frac{2}{3} \cdot 300000\,\text{km/s}} = 25\,\mu s$$

Wir nehmen eine Blocklänge von $L = 10000$ Bit und eine Datenrate von 10 MBit/s an. Die Übertragung eines Blockes dauert dann

$$t_X(L) = \frac{L}{v_D} = \frac{10^4\,\text{Bit}}{10 \cdot 10^6\,\text{Bit/s}} = 1\,\text{ms}$$

Teilt man die Signallaufzeit t_ρ durch die Übertragungsdauer eines Blockes $t_X(L)$, so erhält man einen Wert:

$$a = \frac{t_\rho}{t_X(L)} = \frac{25\,\mu\text{s}}{1\,\text{ms}} = \frac{1}{40}$$

Eine Kollision wird also nach spätestens nach $2 \cdot \frac{1}{40} \cdot 10000\,\text{Bit} = 500\,\text{Bit}$ erkannt. Erhöht man die Datenrate auf mehr als 200 MBit/s, so wird der Wert a größer als $\frac{1}{2}$. Kollisionen werden dann möglicherweise erst *nach* dem $2 \cdot \frac{1}{2} \cdot 10000\,\text{Bit} = 10000\,\text{Bit}$ erkannt, also nach Abschluss der Sendung.

4.4.1 Der FDDI-Standard

FDDI wurde als Standard[4.9] für Ringnetze zur Interkonnektion (im "backbone"-Bereich) von heterogenen LANs entwickelt. Verwendet wird ein Glasfaserdoppelring mit einer Datenrate von 100 MBit/s. FDDI unterstützt

- eine relativ hohe Datenrate
- eine unterschiedliche Behandlung unterschiedlicher Verkehrsarten (Daten, Sprache, Video)
- relativ große Entfernungen, von 2 km bzw. 60 km zwischen den einzelnen Stationen (bei multimode bzw. single mode Glasfasern) und einer maximalen[4.10] Ringlänge von 200 km.
- eine gewisse Ausfalltoleranz durch die Verwendung von zwei Ringen (Primär- und Sekundär-Ring)

Auf der Datensicherungsschicht wird bei FDDI eine 4B/5B-Codierung verwendet, um zusätzliche Kontrollsymbole zu gewinnen. Dabei wird ein 4-Bit Codewort zu einem 5-Bit Codewort erweitert.

4.9. FDDI ist ein ANSI-Standard des Kommitees X 3T9.5.

4.10. Durch zusätzliche Signalverstärker läßt sich die Ausdehnung weiter vergrößern.

Von den 16 zusätzlichen Symbolen werden acht als Kontrollsymbole verwendet. Tabelle 4.3 zeigt die 4B/5B-Codierung. Abbildung 4.25 zeigt einen Nutzdaten-Rahmen und ein Token. Abbildung 4.24 zeigt den schematischen Aufbau des Ringinterfaces bei FDDI.

4-Bit-/Kontrollsymbol	5-Bit-Symbol
0000	11110
0001	01001
0010	10100
0011	10101
0100	01010
0101	01011
0110	01110
0111	01111
1000	10010
1001	10011
1010	10110
1011	10111
1100	11010
1101	11011
1110	11100
1111	11101
IDLE	11111
J	11000
K	10001
T	01101
R	00111
S	11001
QUIET	00000
HALT	00100

Tabelle 4.3. 4B/5B-Codierung bei FDDI

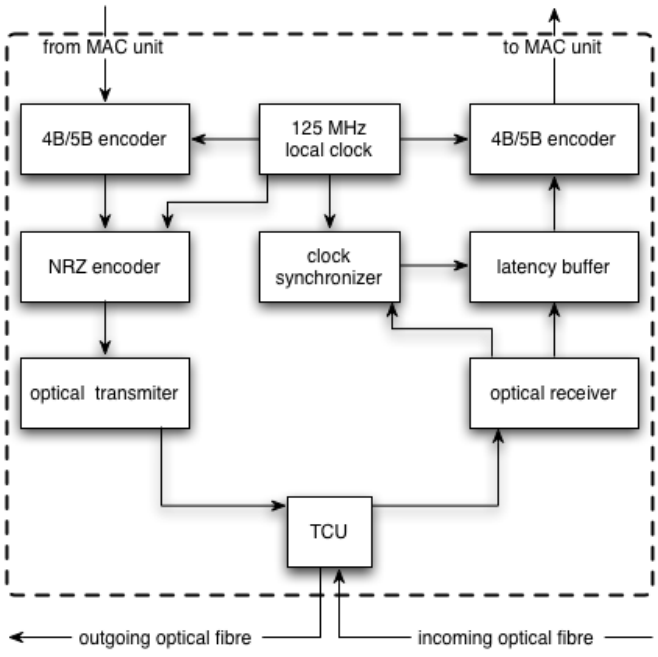
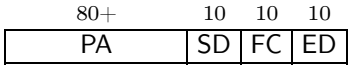
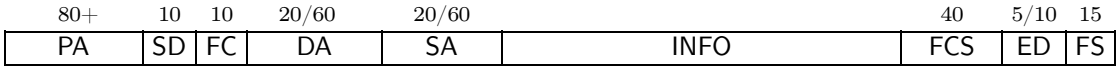


Abbildung 4.24. Ringinterface für FDDI



a) Kontrollmarke



b) Datenrahmen

Abbildung 4.25. Aufbau einer Kontrollmarke und eines Rahmens für Nutzdaten (Längen in Bit, vgl. Tabelle 4.3)

Abk.	Bedeutung
PA	preamble
SD	start delimiter
FC	frame control
DA	destination address
SA	source address
FCS	frame check sequence
ED	end delimiter
FS	frame status

Tabelle 4.4. Bedeutung der Abkürzungen in Abbildung 4.25

Am Anfang dieses Abschnitts wurde schon darauf hingewiesen, dass FDDI zwei Glasfaserringe verwendet, einen *Primärring* und einen *Sekundärring*. Die naheliegende Vermutung, dass bei Ausfall des Primärrings der Sekundärring verwendet wird, ist allerdings falsch.[4.11] Stattdessen sind die Stationen in der Lage, den Ring im Falle eines Kabelbruchs umzukonfigurieren. Abbildung 4.27 zeigt, wie der Ring nach dem Ausfall beider Fasern zwischen zwei Stationen weiterarbeiten kann. Mit jedem weiteren Ausfall zerfällt der Ring in Einzelringe.

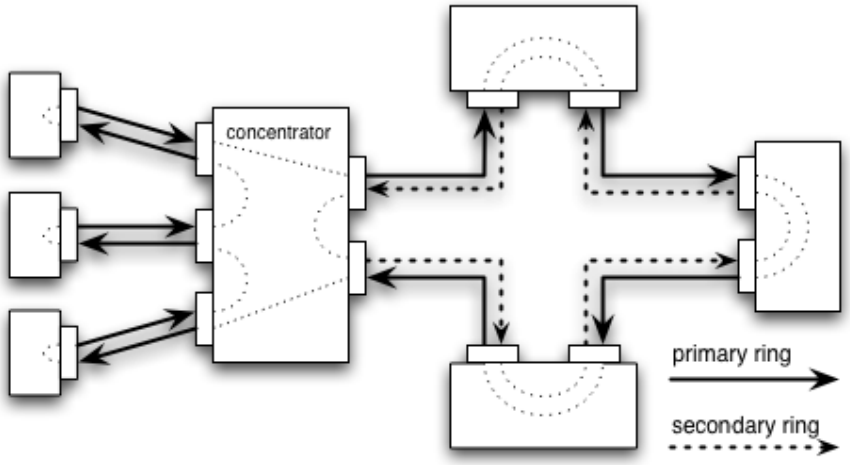


Abbildung 4.26. FDDI-Ring im Primärbetrieb

4.11. Da die beiden Fasern nebeneinander verlegt werden ist es sehr unwahrscheinlich, dass nur eine von beiden beschädigt wird.

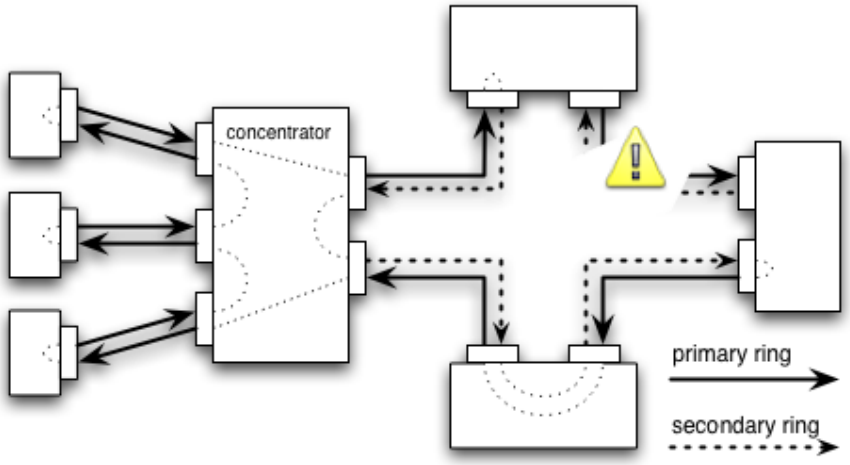


Abbildung 4.27. FDDI-Ring bei Ausfall eines Segments

Leistungsfähigkeit von FDDI

In FDDI-Netzen wird die Kontrollmarke direkt nach Versenden der Dateneinheiten weitergegeben (early token release). Um Stationen daran zu hindern, den Ring für längere Zeit zu blockieren, wird eine *angestrebte Umlaufzeit* t_{OPR} *(operative target token rotation time)* für die Kontrollmarke definiert. Die Stationen sind dafür zuständig, diese Umlaufzeit nicht zu überschreiten. Auf diese Weise kann eine gewisse Fairness gewährleistet werden, ohne dass für nicht sendewillige Stationen Ressourcen reserviert werden.

Bei geringer Last ist die Wahrscheinlichkeit hoch, dass eine Station ihre Dateneinheiten komplett versenden kann. Bei hoher Last wird meist schon ein Teil der Umlaufzeit von anderen Stationen "verbraucht", so dass die Station ihre Dateneinheit fragmentiert versenden muss und Fragmente zwischenspeichern muss. Abbildung 4.28 zeigt die mittlere Anzahl gepufferter Nachrichten in Abhängigkeit von der angebotenen Last und der Umlaufzeit t_{OPR}. Es wird deutlich, dass höhere Umlaufzeiten zu niedrigeren Warteschlangenbelegungen führen. Allerdings muss eine sendewillige Station im Mittel länger warten, um in den Besitz einer Kontrollmarke zu gelangen. Die Wahl von t_{OPR} hängt also stark von der Verkehrsart ab.[4.12]

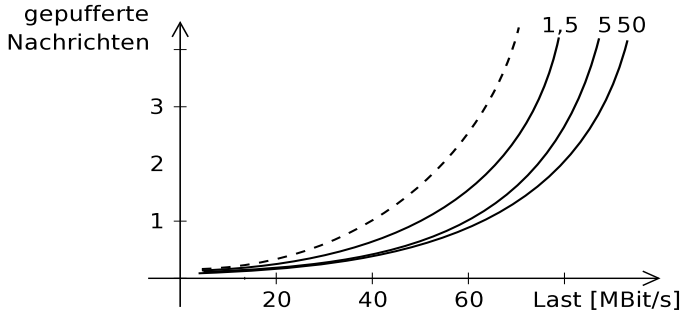


Abbildung 4.28. Mittlere Anzahl gepufferter Nachrichten in einem 25 km langen FDDI-Ring mit unterschiedlichem $t_{\mathrm{OPR}} \in \{1.5\,\mathrm{ms}, 5\,\mathrm{ms}, 50\,\mathrm{ms}\}$ bzw. Einzelabfertigung der Dateneinheiten (gestrichelt)

Die Bedeutung von FDDI ist stark rückläufig. Stattdessen sind Fast/Gigabit Ethernet – und

4.12. Hier zeigt sich die Notwendigkeit einer realitätsnahen Lastmodellierung.

bedingt auch noch ATM – deutlich wichtiger geworden.

4.4.2 Fast Ethernet

Der Standard für Fast Ethernet ist aus dem Ethernet-Standard hervorgegangen.[4.13] Um eine maximale Datenrate von 100 MBit/s zu gewährleisten, wurde der Standard in einigen Punkten angepasst:

- Als physikalische Topologieform wurde ein Stern (mit Hubs im Zentrum) anstelle des Multidrop-Kabels gewählt.
- Die maximale Entfernung zwischen zwei Stationen wurde auf 200 m reduziert. Die maximale Entfernung von einer Station zu einem Hub beträgt also 100 m.
- Als Leitungen dienen Kabel der Kategorie 3 (UTP) oder 5 (STP) mit vier Drähten. Zwei davon werden bidirektional verwendet, zwei unidirektional. Somit kann eine sendende Station drei Drähte mit einer Datenrate von jeweils 33,3 MBit/s verwenden. Alternativ können auch zwei Glasfasern verwendet werden.
- Die Taktfrequenz wurde auf 25 MHz erhöht. In je sechs Takten werden acht Bit übertragen (8B6T-Codierung), so dass sich die benötigten 33,3 MBit/s pro Draht ergeben. Werden Glasfasern zur Übertragung verwendet, wird eine 4B5B-Codierung eingesetzt, so dass eine Taktfrequenz von 125 MHz nötig ist.

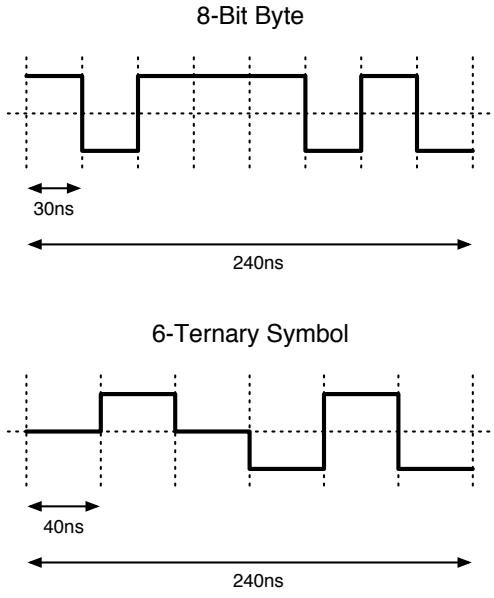


Abbildung 4.29. Codierung eines Bytes durch sechs Ternärsymbole

Abbildung 4.29 illustriert, wie bei der 8B6T-Codierung acht Bit als sechs Ternärsymbole übertragen werden. Mit sechs Ternärsymbolen ließen sich $3^6 = 729$ unterschiedliche Symbole bilden, also deutlich mehr als die $2^8 = 256$ Möglichkeiten mit acht Binärsymbolen.[4.14] Es stellt sich also

4.13. Auch der VG-AnyLAN-Standard der IEEE 802.12 ist daraus hervorgegangen.

4.14. Allerdings würden fünf Ternärsymbole nicht ausreichen: $3^5 = 243$

die Frage, welche der Ternärsymbole für die Codierung verwendet werden.

Betrachtet man Tabelle 4.5 so fällt auf, dass die Summe w eines 8B6T-Codewortes immer entweder 0 oder 1 ist. Mit der Summe w meinen wir hier die Summe der einzelnen Amplituden. Die Summe von `++0-0-` ist also 0, die von `+--0++` ist 1.

Byte	Codewort	Byte	Codewort	Byte	Codewort	
00	-+00-+	20	-++-00	40	-00+0+	···
01	0-+-+0	21	+00+--	41	0-00++	···
02	0-+0-+	22	+00+--	42	0-0+0+	···
03	0-++0-	23	+-0-++	43	0-0++0	···
04	-+0+0-	24	+-0+00	44	-00++0	···
05	+0--+0	25	-+0+00	45	00-0++	···
06	+0-0-+	26	+00-00	46	00-+0+	···
07	+0-+0-	27	-+++--	47	00-++0	···
08	-+00+-	28	0++-0-	48	00+000	···
09	0-++-0	29	+0+0--	49	++-000	···
0A	0-+0+-	2A	+0+-0-	4A	+-+000	···
⋮	⋮	⋮	⋮	⋮	⋮	⋱

Tabelle 4.5. Codewörter der 8B6T-Codierung (Ausschnitt)

Diese Auswahl wurde getroffen, um den Gleichstromanteil auszugleichen, der sonst auf dem Draht entstehen würde. Um diesen auf ein Minimum zu reduzieren, geht man wie folgt vor: Der sendende Adapter befindet sich in einem von zwei Zuständen. Diese Zustände stehen für die Gesamtsumme S, aller bisher übertragenen Codewörter (auch diese Summe wird immer entweder 0 oder 1 sein.) Ist die $S = 0$, so werden alle Codewörter normal übertragen. Nach der Übertragung des ersten Codewortes mit $w = 1$, ist auch die Gesamtsumme $S = 1$. In diesem (zweiten) Zustand, werden nun alle Codewörter mit $w = 0$ normal übertragen, das erste Codewort mit $w = 1$ wird allerdings invertiert gesendet, so dass die Gesamtsumme nach dieser Übertragung wieder 0 ist und der erste Zustand eingenommen wird. In Abbildung 4.30 ist ein entsprechender Automat dargestellt.

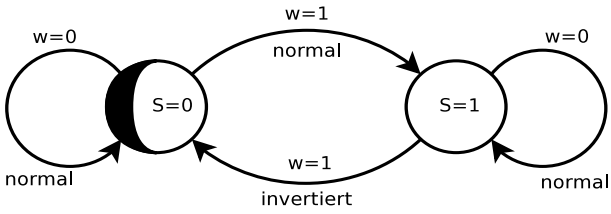


Abbildung 4.30. Ausgleich des Gleichstromanteils bei Fast Ethernet

Abbildung 4.31 zeigt die Verwendung der Drähte. Das Abhören (CS, carrier sense) und die Kollisionserkennung (CD, collision detection) wird jeweils an einem unidirektional verwendeten Draht durchgeführt. In Abbildung 4.32 wird dargestellt, wie der mit 100 MBit/s am Ethernet-Adapter eingehende Datenstrom auf die Drähte verteilt wird. Das erste Byte wird über den ersten Draht gesendet. Da mit dem Draht nur 33, 3 MBit/s (bzw. 25 MT/s) gesendet werden können, ist dieser Draht für die Dauer von drei eingehenden Bytes blockiert. Die nächsten zwei Bytes werden über die anderen beiden Drähte übertragen, das vierte Byte kann dann wieder über den ersten Draht versendet werden.

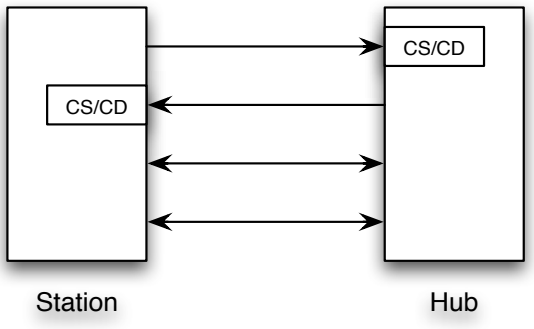


Abbildung 4.31. Verwendung der Drähte bei Fast Ethernet

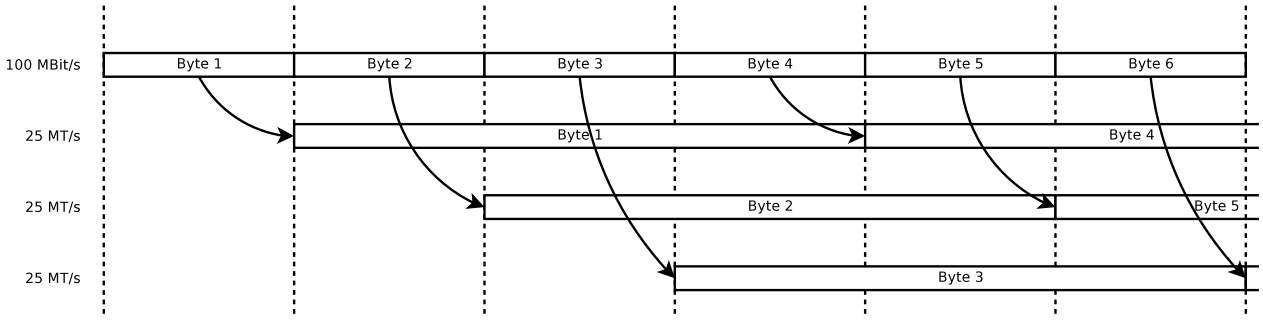


Abbildung 4.32. Aufteilung der zu sendenen Bytes auf die Drähte

4.4.3 Der DQDB-Standard

Neben FDDI wurde mit DQDB (distributed queue dual bus) ein weiterer Standard für Netze zur Interkonnektion von heterogenen Netzen definiert. Auch bei DQDB ist eine relativ große Netzausdehnung (mehr als 50km) möglich und es werden unterschiedliche Verkehrsarten (Daten, Sprache, Video) bei Datenraten in der Größenordnung von 100 MBit/s unterstützt.[4.15] DQDB verwendet im Gegensatz zu FDDI allerdings eine Bustopologie. Durch die Verwendung von zwei separaten Leitungen (Koaxialkabel oder Glasfaser) wird eine gewisse Ausfalltoleranz erreicht. Es werden drei unterschiedliche Dienstarten angeboten:

- verbindungslose Datenübertragung mit Rahmen bis zu 9188 Byte
- verbindungsorientierte Datenübertragung mit 52 Byte großen Segmenten[4.16]
- isochrone Datenübertragung für die byte-weise Übertragung in vorab reservierten Slots

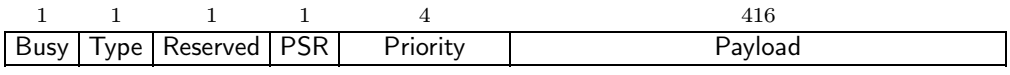


Abbildung 4.33. Aufbau eines 52 Byte langen DQDB-Segments

Die folgenden Abbildungen zeigen den Aufbau eines DQDB-Netzes. Es werden zwei separate Leitungen verwendet, die in gegenläufige Richtungen betrieben werden. Für jede Leitung existiert am Anfang ein *Slotgenerator* und am Ende ein *Terminator*. Grundsätzlich sind zwei unterschiedliche

4.15. Höhere Datenraten sind geplant.

4.16. Ein Segment ergibt zusammen mit einem 1 Byte langen Kontrollfeld einen 53 Byte großen *Slot*, der genau in eine ATM-Zelle passt

Konfigurationen möglich: Eine offene, bei der die äußersten Stationen jeweils einen Slotgenerator und einen Terminator besitzen (open bus configuration) und eine geschlossene, bei der eine Station mit beiden Enden des Busses verbunden ist (looped bus configuration).

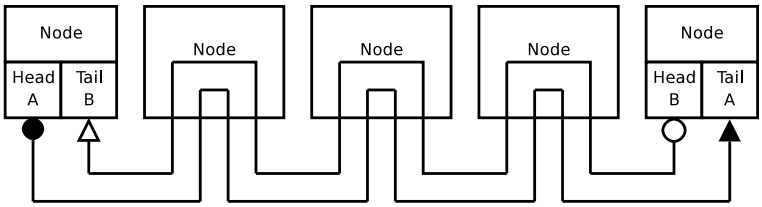


Abbildung 4.34. DQDB-Netz: open bus configuration

Bei der geschlossenen Konfiguration genügt ein Slotgenerator für beide Leitungen. In diesem Fall liegt eine Interpretation der Netztopologie als physikalischer Ring zwar nahe, in Hinblick auf die Datensicherungsschicht ist aber eine Interpretation als Bus sinnvoll.

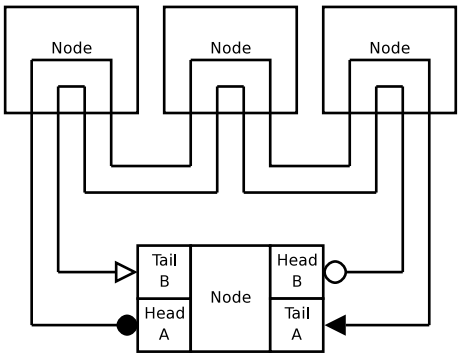


Abbildung 4.35. DQBB-Netz: looped bus configuration

In beiden Konfigurationen zerfällt das Netz beim Ausfall einer Doppelleitung in zwei Subnetze, die jeweils auf einem Doppelbus weiterarbeiten können. Dafür muss allerdings jeder Knoten in der Lage sein, Rahmen zu generieren und zu absorbieren.

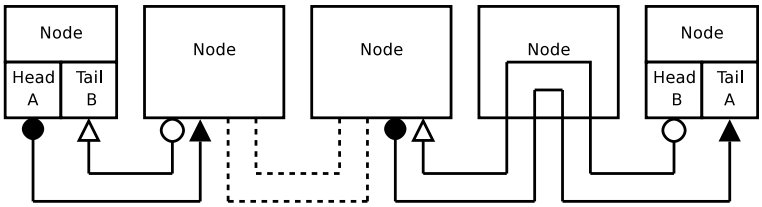


Abbildung 4.36. Rekonfiguration eines DQDB-Netzes im Fehlerfall

Standard	Datenrate und Medium
ANSI DS3	44,736 Mb/s, 75Ω Koax oder Glasfaser
ANSI SONET STS-3c	155,520 Mb/s, single mode Glasfaser
CCITT G.703	34,368 Mb/s bzw 139,26 Mb/s, elektrische Signalübertragtung

Tabelle 4.6. Physikalische Attribute

Zugriffskontrolle bei DQDB

Die Zugriffskontrolle ist zunächst vergleichbar mit der Festrahmenzirkulation in Ringnetzen. Allerdings findet ein verteilter Reservierungsmechanismus statt, so dass eine Station einen freien Slot nur dann verwendet, wenn er ihr gemäß der Reservierungsreihenfolge auch zusteht.

Zur Reservierung eines Slots auf einer Busleitung verwenden die Stationen grundsätzlich die jeweils andere Leitung. Auf diese Weise informiert eine reservierende Station genau die Stationen über ihren Sendewunsch, die (auf der Leitung für die tatsächliche Übertragung) *vor* ihr liegen, damit diese einen eintreffenden freien Slot nicht für sich verwenden. Eine Station muss sich also für beide Richtungen jeweils merken:

1. die Zahl der Sendewünsche von Stationen, die physikalisch hinter ihr liegen, jedoch vor dem eigenen Sendewunsch angemeldet wurden, und noch nicht erfüllt wurden. Dafür wird der Zähler CD (countdown counter) verwendet.
2. die Zahl der Sendewünsche, die nach dem eigenen Sendewunsch angemeldet wurden. Hierfür dient der Zähler RQ (request counter).

Die Stationen speichern also nur *wieviele* Stationen vor und hinter ihnen liegen, nicht jedoch um *welche* Stationen es sich dabei handelt.

$$\overbrace{\text{andere} \rightarrow \text{andere} \rightarrow \text{andere} \rightarrow \text{andere}}^{\text{RQ=4}} \rightarrow \text{eigene} \rightarrow \overbrace{\text{andere} \rightarrow \text{andere}}^{\text{CD=2}}$$

Jede Station geht nun wie folgt vor. Man beachte, dass der Algorithmus hier nur für eine Richtung beschrieben ist. Insgesamt benötigen die Stationen zwei Request Counter und zwei Countdown Counter und müssen den Algorithmus für beide Richtungen durchführen.

- Bei jedem passierenden Request wird RQ um eins erhöht.
- Bei Versendung eines eigenen Request wird CD auf RQ gesetzt und RQ auf 0.
- Wurde ein Sendewunsch angemeldet und steht CD auf 0, so wird der nächste freie Slot für die eigene Sendung verwendet.
- Wenn ein freier Slot die Station passiert und diese nicht sendewillig ist, wird RQ um eins reduziert, wenn er nicht schon 0 ist.
- Wenn ein freier Slot die Station passiert und diese eine Sendung angemeldet hat, wird CD um eins reduziert, wenn er nicht schon 0 ist.

Beispiel 4.5. Die folgende Abbildung soll den Algorithmus anhand eines Beispiels mit fünf Stationen veranschaulichen. Wir betrachten dabei nur Sendungen auf dem Bus A (nach rechts) und damit Reservierungen auf dem Bus B (nach links). Eine Erklärung der einzelnen Schritte folgt.

$$\text{Station}_1 \underset{\leftarrow B}{\overset{A \rightarrow}{}} \text{Station}_2 \underset{\leftarrow B}{\overset{A \rightarrow}{}} \text{Station}_3 \underset{\leftarrow B}{\overset{A \rightarrow}{}} \text{Station}_4 \underset{\leftarrow B}{\overset{A \rightarrow}{}} \text{Station}_5$$

a) Topologie

	Vorgang	RQ_1	CD_1	RQ_2	CD_2	RQ_3	CD_3	RQ_4	CD_4	RQ_5	CD_5
1	Station 4 reserviert	1	–	1	–	1	–	0	0	0	–
2	Station 2 reserviert	2	–	0	1	1	–	0	0	0	–
3	Station 3 reserviert	3	–	1	1	0	1	0	0	0	–
4	Station 4 sendet	2	–	1	0	0	0	0	–	0	–
5	Station 2 sendet	1	–	1	–	0	0	0	–	0	–
6	Station 5 reserviert	2	–	2	–	1	0	1	–	0	0
7	Station 3 sendet	1	–	1	–	1	–	1	–	0	0
8	Station 5 sendet	0	–	0	–	0	–	0	–	0	–

b) Ablauf

Abbildung 4.37. Beispielhafter Ablauf der Reservierung bei DQDB

1. Die Station 4 macht eine Reservierung auf dem Bus B. Diese passiert die Stationen 1 bis 3, die ihren RQ um eins erhöhen. Station 4 setzt CD auf RQ und RQ auf 0.

2. Station 2 reserviert als nächstes. Die Reservierung wird nur von Station 1 wahrgenommen. Diese erhöht RQ um eins auf 2. Station zwei schreibt die 1 aus RQ nach CD und setzt RQ auf 0.

3. Station 3 schickt eine Reservierung. Die Stationen 1 und 2 erhöhen RQ. Station 3 setzt CD auf RQ und RQ auf 0.

4. Ein freier Slot erreicht die Station 1 auf dem Bus *A*. Sie ist nicht sendewillig, senkt also RQ um eins. Station 2 ist sendewillig, muss allerdings noch warten, da CD nicht 0 ist. CD wird um eins gesenkt. Dasselbe gilt für Station 3. Station 4 kann senden, denn ihr CD ist gleich null. Die Station 5 erreicht kein freier Slot.

5. Ein weiterer freier Slot erreicht die Stationen. Station 1 ist nicht sendewillig, senkt also RQ um eins. Der CD der Station 2 ist 0, sie kann also senden. Die Stationen 3 bis 5 nehmen keinen freien Slot wahr, da er von Station 2 bereits verwendet wurde.

6. Station 5 reserviert und setzt CD auf RQ, RQ auf 0. Alle anderen Stationen erhöhen RQ um eins.

7. Ein freier Slot erreicht die Stationen. Station 1 und 2 senken RQ. Station 3 kann senden. Die Stationen 4 und 5 sehen keinen freien Slot.

8. Der freie Slot läuft an den Stationen 1 bis 4 vorbei, die RQ um eins senken. Station 5 sendet.

4.5 Intranets

Die wachsende Bedeutung des Internets sowie dessen spezifischer Dienste und Protokolle, hat in vielen Unternehmen den Wunsch erzeugt, für das interne und das externe Firmennetz dieselbe Technologie zu verwenden. Für solche Netze hat sich der Begriff *Intranet* etabliert, der den unternehmensinternen Charakter und die technologische Nähe zum Internet verbindet.

Definition 4.6. *Ein Unternehmensnetz, das auf den standardisierten Internetprotokollen basiert, nennen wir* Intranet. *Diese Netze besitzen in der Regel die folgenden Charakteristika:*

- *Sie basieren auf den Protokollen TCP/IP und unterstützen die typischen anwendungsorientierten Dienste des Internet.*
- *Als Betreiber fungiert ein einzelnes Unternehmen.*
- *Die physikalische Ausdehnung entspricht häufig (aber nicht notwendigerweise) der eines LANs.*
- *Eventuell besteht eine Kopplung mit dem globalen Internet. Diese Kopplung ist dann allerdings durch Sicherheitsmechanismen wie Firewalls stark begrenzt.*

Protokolle	Schicht
HTTP, FTP, Telnet	Anwendungsschicht
TCP, UDP	Transportschicht
IP, ICMP, IGMP	Vermittlungsschicht
Fast Ethernet	Datensicherungsschicht physikalische Schicht

Tabelle 4.7. Typische Protokollhierarchie in einem Intranet

Tabelle 4.7 zeigt eine typische Protokollhierarchie in einem Intranet. In diesem Fall werden die Protokolle des Internet ab einschließlich der Vermittlungsschicht eingesetzt. Unterhalb der Vermittlungsschicht wird das für LANs sehr verbreitete Fast Ethernet verwendet. Unsere Definition des Intranets macht keine genaue Aussage darüber, ab welcher Schicht die Protokolle des Internets verwendet werden. Deshalb betrachten wir nun vier Varianten, die sich zunehmend von unserer Definition entfernen.

1. Wie in Tabelle 4.7 besteht *Flexibilität unterhalb der Vermittlungsschicht*. Darunter kann etwa Ethernet, FDDI, ATM oder S-ISDN zum Einsatz kommen. Diese Protokolle können allerdings in bestimmten Einsatzgebieten zu ineffizienten oder geographisch limitierten Lösungen führen.

2. Bei dieser Variante werden die *Internetprotokolle erst aber der Transportschicht* eingesetzt. Anwendungsorientierte Dienste (deren Implementation auf der Transportschicht basieren) können so weitgehend unverändert beibehalten werden. Allerdings sind für einen Übergang zum globalen Internet komplexe Gateways nötig, da IP als gemeinsame Basis fehlt.

3. Die dritte Variante besteht in einem *Verzicht auf IP, TCP und UDP*, wobei aber weiterhin eine entsprechende Transportdienst-Schnittstelle verfügbar ist, auf der die anwendungsorientierten Dienste aufsetzen. Die eigene Implementation eines Transport- und Vermittlungsdienstes ist allerdings sehr aufwändig. Ein solches Intranet bezeichnen wir nicht mehr als *offen*.

4. Auch bei einem *generellen Verzicht auf den Internet-Protokollstapel* ist es möglich, anwendungsorientierte Dienste mit ähnlicher Funktionalität und Benutzungsschnittstelle zu realisieren. Die Kosten für die Implementation sind meist sehr hoch. Dafür kann den Betreiber die Architektur flexibel an seine Bedürfnisse anpassen. Spätestens bei dieser Variante sprechen wir nicht mehr von einem Intranet.

Wir fassen noch einmal einige der Gründe für die Tendenz zum Intranet zusammen.

- Die Bedeutung des Internets und dessen anwendungsorientierte Dienste wächst stark.
- Zahlreiche Hersteller bieten Internet-Technologien an.
- Unternehmen möchten ihre Netze für die Kommunikation nach außen öffnen.
- Internetdienste sollen auch im Unternehmensnetz aufwandsarm nutzbar sein.
- Die Internet-Technologie stellt eine gut erprobte Lösung dar.
- Die Wartung wird erleichtert, da Wissen über die Technologie meist vorhanden ist.

Die Nutzung der Internet-Technologie in Unternehmensnetzen führt allerdings auch zu einigen Problemen. Zwei Ursachen für diese Probleme sind dabei besonders hervorzuheben. Zum einen wurden die Dienste des Internet ursprünglich für WAN- und GAN-Kommunikation im Forschungsbereich geschaffen. Zum anderen führen die unterschiedlichen Paradigma der Paketvermittelung (packet switching) und der Leitungsvermittlung (circuit switching) zu Problemen.

- Das Internetprotokoll IP in der Version stößt gegenwärtig in zweierlei Hinsicht an seine Grenzen. Zum einen entwickelt sich durch die weltweite Nutzung ein *Adressenengpass*, zum anderen besteht eine *mangelhafte Unterstützung von Mobilkommunikation*.[4.17]
- Die Dienste des Internet haben einen "best effort"-Charakter und geben *keine Dienstgütegarantien* ab. Die Vermittlungsrechner unterscheiden im allgemeinen nicht zwischen unterschiedlichen Kommunikationsformen (Daten, Sprache, etc.).[4.18] Um solche Garantien in Zukunft bieten zu können, wurden Vorschläge für Reservierungsprotokolle (RSVP, CoS) gemacht. Deren Einsatz erfordert allerdings eine Modifikation der bestehenden Vermittlungsrechner.

4.17. Diese Probleme könnten durch die Version 6 von IP bzw. durch Mobile IP gelöst werden. Vgl. RFC 1883-1887

4.18. obwohl dafür im Header eines IP-Pakets ein Feld vorgesehen ist (TOS, type of service)

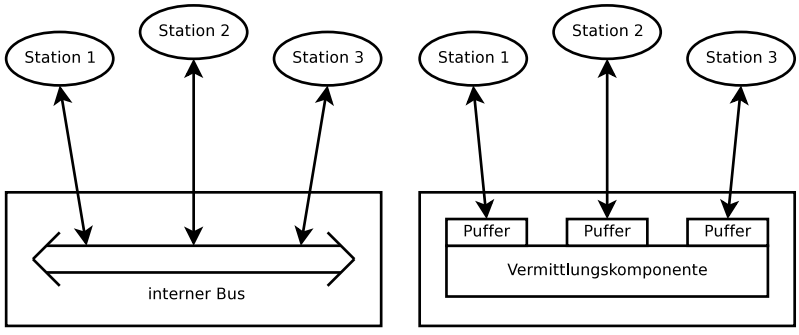
- Die *Netzwerksicherheit* stellt eines der größten Probleme von Intranets mit Kopplung an das globale Internet dar. Zur Lösung dieses Problems existiert eine Vielzahl von Produkten und Mechansimesn wie Firewalls, Datenverschlüsselung und -signierung und virtuelle private Netzwerke (VPN, virtual private network).
- Auch die *Kombination des paketvermittelnden IP mit verbindungsorientierten Basisdiensten* (z.B. IP over ATM) kann schwierig sein. Insbesondere stellt sich die Frage, wieviel Dienstgüte ein "best effort"-Dienst auf der Basis eines QoS-orientieren Dienstes bieten kann.

4.6 Vermittlungsrechner in lokalen Rechnernetzen

Vermittlungsrechner spielen in lokalen Rechnernetzen eine immer größere Rolle. Das liegt einerseits an der zunehmenden Bedeutung der (im vorigen Abschnitt beschriebenen) Intranets, andererseits am Trend zu sternförmigen Topologien im LAN-Bereich, der vor allem durch Fast Ethernet entstand. Wir betrachten deshalb hier drei unterschiedliche Typen von Vermittlungsrechnern für die sich die drei Begriffe *Hub*, *Switch* und *Router* eingebürgert haben. Beispiele für solche Vermittlungsrechner sind

- paketvermittelnd: Ethernet-Switches, IP-Router und ATM-Switches
- leitungsvermittelnd: Nebenstellenanlagen (PBX[4.19], private branch exchange), ISDN-Telefonanlagen und -vermittlungsrechner

Hub Ein Hub ist die einfachste Form eines Vermittlungsrechners. Ein Hub liegt im Zentrum einer sternförmigen Topologie und besitzt Punkt-zu-Punkt-Verbindungen zu den angeschlossenen Stationen. Der Hub verbindet die Stationen über einen eigenen *internen Bus*. Die Wirkungsweise eines Hubs ähnelt einem Bus, denn Kollisionen werden nicht durch den Vermittlungsrechner vermieden und jede gesendete Nachricht erreicht alle angeschlossenen Stationen. Hubs regenerieren wie Repeater das physikalische Signal. Die Vermittlung findet hier also auf der Schicht 1 der ISO/OSI-Protokollhierarchie statt.



a) Hub b) Switch

Abbildung 4.38. Hub und Switch mit angeschlossenen Stationen

Switch Im Gegensatz zu Hubs besitzen Switches für jede angeschlossene Station einen *Eingangs-* und einen *Ausgangspuffer* sowie eine *Vermittlungskomponente*. Die Vermittlungskomponente stellt anhand der Adressen in den Paket-Headern den Empfänger fest und verschiebt die Dateneinheit in den entsprechenden Ausgangspuffer. Diese Vorgehensweise hat den Hubs gegenüber zwei große Vorteile.

- Der Verkehr im Netz kann besser kontrolliert werden, da unzulässige Sendungen vom Switch gefiltert werden können. Dadurch kann die Sicherheit eines Netzes verbessert werden.

4.19. auch PABX, private automatic branch exchange

- Bei der Nutzung von Switches entstehen keine Zugriffskonflikte in Form von Kollisionen, da kein gemeinsames physikalisches Übertragungsmedium verwendet wird. Darüber hinaus werden (außer beim Rundsenden) nur die dafür benötigten Leitungen verwendet. Daraus resultiert in der Regel eine höhere Effizienz.

Bei Switches entsteht allerdings eine neue Art von Zugriffskonflikt, wenn dessen Puffer bei starker Last voll sind und Pakete nicht mehr aufgenommen bzw. weitergeleitet werden können. Außerdem kann die Verzögerungszeit durch den Vermittlungsprozess erhöht werden. Die Vermittlung kann hier auf der Schicht 2 (z.B. anhand von MAC-Adressen) oder Schicht 3 (z.B. IP-Adressen) stattfinden.

Router Im globalen Internet und in größeren Intranets aber auch in leitungsvermittelnden Netzen wie dem öffentlichen Telefonnetz wird oft eine hierarchische Topologie verwendet. Insbesondere wenn die Last im lokalen Bereich (z.B. innerhalb einer Büroetage oder einer Stadt) hoch und in größeren Bereichen (z.B. zwischen Filialen oder Ländern) gering ist, bietet sich diese Topologie an. Für diese Topologie werden spezielle Vermittlungsrechner benötigt, die ihre untergeordneten Stationen wie ein Switch bedienen und zusätzliche Kommunikationsbeziehungen zu den übergeordneten Stationen besitzen. Diese Vermittlungsrechner benötigen eine zusätzliche Wegeermittlung, um für die Pakete, die sie nicht direkt vermitteln können einen geeigneten anderen Vermittlungsrechner zu finden. Switches, die zusätzlich Wegeermittlungsaufgaben übernehmen, nennen wir Router[4.20].

Für lokale Rechnernetze können auch leitungsvermittelnde Techniken eingesetzt werden. Bei PBX (vgl. Kapitel 5) werden Kanäle mit einer Datenrate von 64 kb/s vermittelt. Dabei wird eine Sterntopologie verwendet, bei der eine limitierte Anzahl von Kommunikationspartnern über verdrillte Drähte mit dem Vermittlungsrechner verbunden werden. Die Anzahl der gleichzeitig aktiven Benutzer ist begrenzt, daher kann es beim Verbindungsaufbau zu einer Ablehnung kommen. Beim Verbindungsaufbau muss eine signifikante Verzögerung in Kauf genommen werden. Danach ist die Verzögerung jedoch konstant und der Durchsatz garantiert.

4.20. sprich [Ru:ter] oder [Rauter]

Kapitel 5
Kommunikation in Weitverkehrsnetzen und im globalen Internet

5.1 Übersicht über und Klassifikation für überregionale Rechnernetze

In diesem Abschnitt wollen wir Möglichkeiten der Klassifikation von Weitverkehrsnetzen diskutieren. Dafür werden wir Kriterien wie Übertragungsgeschwindigkeit, Netzausdehnung, Vermittlungstechnik u.a. verwenden. Zunächst klassifizieren wir Kommunikationsnetze anhand ihrer Übertragungsgeschwindigkeit und ihrer Netzausdehnung. Abbildung 5.1 stellt die beiden Dimensionen dar.

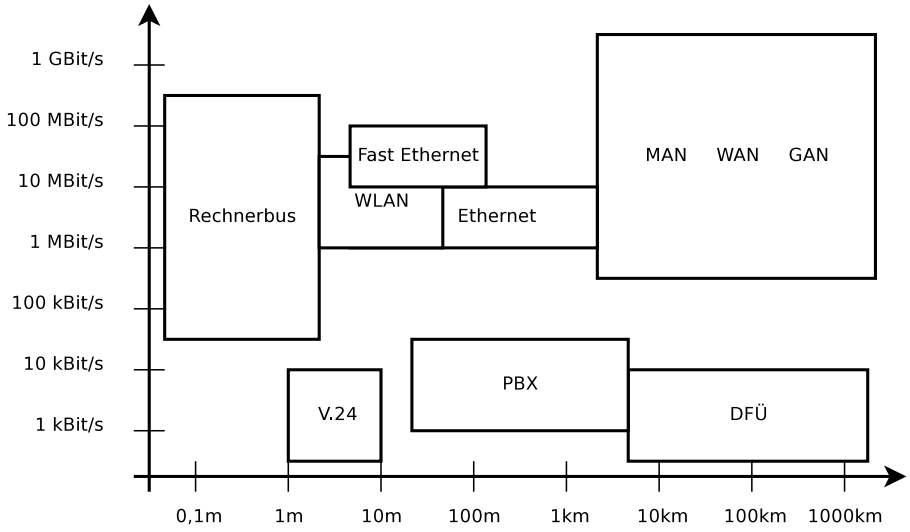


Abbildung 5.1. Klassifikation anhand von Übertragungsgeschwindigkeit und Netzausdehnung

Wir betrachten nun die Art, in der Informationen in einem Kommunikationsnetz ausgetauscht werden und unterscheiden dabei zwei Klassen:

Verteilnetz. Bei diesen Netzen wird einer großen Zahl von Empfängern durch einen (oder durch wenige) Sender dieselbe Information unidirektional bereitgestellt. Beispiele hierfür sind Fernsehen und Rundfunk.

Vermittlungsnetz. Im Gegensatz zum Verteilnetz werden hier individuelle Informationen bidirektional zwischen einer (meist) kleineren Anzahl von Kommunikationspartnern ausgetauscht. Beispiele sind das öffentliche Telefonnetz und das Netz des klassischen Brief- und Paketverkehrs.

Außerdem unterscheiden wir Kommunikationnetze nach ihrem Benutzerkreis.

privates Netz. Die Betreiber solcher Netze sind Privatpersonen oder Unternehmen, die ein Netz für einen bestimmten Personenkreis verfügbar machen. Beispiele sind Firmennetze zur Verbindung der Arbeitsplatzrechner.

öffentliches Netz. Betreiber dieser Netze sind beispielsweise Telekom, Internet Service Provider (ISP) oder der DFN-Verein in Deutschland (im Wissenschaftsbereich), die ihre Dienste einem breiten Benutzerkreis (meist gegen Bezahlung) anbieten.

5.1.1 Topologieformen

Für regionale und überregionale Kommunikationsnetze haben sich teilweise andere Begriffe zur Beschreibung der Topologie etabliert. Abbildung 5.2 zeigt häufige Topologieformen in Weitverkehrsnetzen und deren Bezeichnungen.

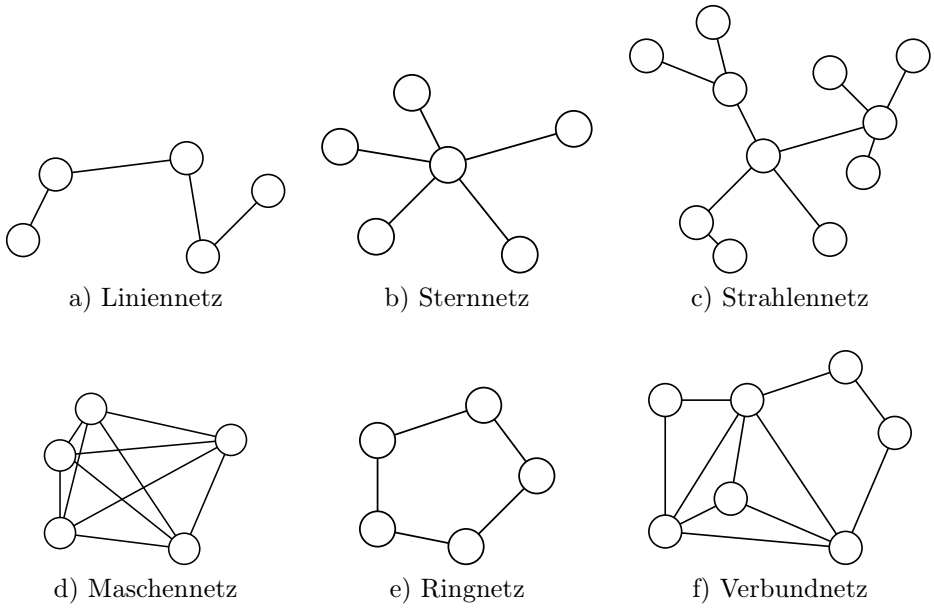


Abbildung 5.2. Häufige Topologieformen in Weitverkehrsnetzen

Diese Topologieformen werden im folgenden genauer betrachtet und bewertet. Als Kriterien zur Bewertung dienen dabei insbesondere die geographische Verteilung, die Art der zu übertragenden Daten (Größe der Dateneinheiten, Verkehrsmatrix, etc.), Robustheit gegenüber Störungen und Ausfällen, sowie die Art der Vermittlungstechnik innerhalb der Knoten.

Liniennetz. Diese Topologie besitzt eine starke Ähnlichkeit zum Bus. Oft werden mehrere redundante Übertragungswege realisiert, um Leitungsausfälle zu kompensieren. Die Gesamtzahl der erforderlichen Leitungen (Leitungsbündeln) ist relativ gering. Fällt eine Leitung oder ein Knoten aus, existieren keine alternativen Wege.

Sternnetz. Auch diese Topologieform kommt mit einer geringen Zahl von Leitungen pro Knoten aus. Die Gestaltung ist übersichtlich und die Wegeermittlung trivial. Der Ausfall einer Leitung führt zur Isolation des entsprechenden Knotens, während bei einem Ausfall des zentralen Vermittlungsknotens keine weitere Kommunikation möglich ist.

Strahlennetz. Netze mit dieser Topologie weisen eine Nähe zur hierarchischen Netztopologie auf. Besonders wenn der Großteil des Verkehrsaufkommens im lokalen Bereich liegt, bietet sich diese Topologieform an. Die Wegeermittlung ist komplizierter als beim Linien- und Sternnetz. Da es aber für keine Kommunikationsbeziehung alternative Pfade gibt, erfolgt die Wegeermittlung statisch. Beim Ausfall einer Leitung zerfällt das Netz in zwei neue Strahlennetze. Fällt ein Knoten aus, so zerfällt das Netz (möglicherweise) in mehrere neue Sternnetze.

Maschennetz. Diese Topologieform wird auch als *vollständige Vermaschung* bezeichnet. Im nicht-lokalen Bereich ist sie nur bei einer geringen Knotenzahl relevant.[5.1] Der Ausfall eines Knotens beeinträchtigt nie die Kommunikation zwischen anderen Knoten. Beim Ausfall einer Leitung stehen alternative Pfade zur Verfügung.

5.1. Wegen der großen Anzahl von $\sum_{i=1}^{n-1} i = \frac{(n-1)\cdot n}{2}$ benötigten Leitungen bei n Knoten.

Ringnetz. Bei dieser Topologieform ist die Zahl der benötigten Leitungen sehr gering. Aus dem Bereich der lokalen Rechnernetze sind viele Mechanismen der Datensicherungsschicht anwendbar. Sofern kein mehrfacher Ring (wie bei FDDI) vorliegt, ist keine Kommunikation nach dem Ausfall eines Knotens oder einer Leitung mehr möglich.

Verbundnetz. Diese Topologieform wird auch als *irreguläre Vermaschung* bezeichnet. Leitungen zwischen Knoten können bedarfsorientiert gewählt werden. Oft wird auch die Leitungskapazität und damit die Datenrate bedarfsorientiert gewählt. Netzengpässe können gezielt durch zusätzliche Leitungen eliminiert werden. Ausfalltoleranz kann individuell an die äußeren Gegebenheiten angepasst werden. Der hohen Flexibilität folgt allerdings auch eine hohe Komplexität, die sich in einer aufwändigen Wegeermittlung, Flusskontrolle und Sättigungskontrolle, sowie einer anspruchsvollen Netzkonfiguration äußert.

Strukturierung der Topologie sehr großer Netze

Bei sehr großen Netzen kann deren Topologie weiter strukturiert werden, indem auch die Art der Vermaschung der *Teilnetze* betrachtet wird. In Abbildung 5.3 sind vier Teilnetze in einer sternförmigen Topologie miteinander verbunden. Die äußeren Knoten sind lokale Rechnernetze (z.B. einer Firma mit drei Filialen) als zentraler Knoten fungiert ein Weitverkehrsnetz (z.B. der Telekom). Solche Netze werden oft als *backbone (Rückgrat)* bezeichnet.

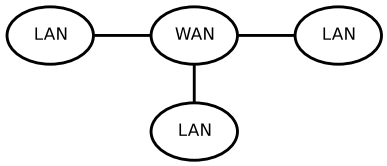


Abbildung 5.3. Ein Netz aus Netzen mit Sterntoppologie

Ein weiteres Beispiel liefert Abbildung 5.4. Hier dienen zur Kopplung der beiden lokalen Netze mehrere Weitverkehrsnetze (z.B. von unterschiedlichen ISPs), die untereinander irregulär vermascht sind.

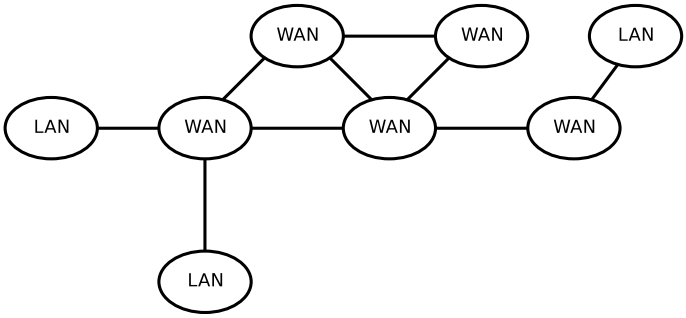


Abbildung 5.4. Ein Netz aus Netzen mit Verbundtopologie

Betreiberhierarchie für öffentliche Kommunikationsnetze

Im vorigen Abschnitt wurde eine Strukturierung auf der Ebene von Teilnetzen angesprochen. Es stellt sich allerdings die Frage, wie aus der Menge aller Knoten eines Netzes solche Teilnetze hervorgehen. Ein naheliegendes Kriterium hierfür stellen die Organisationen, die die Knoten und Leitungen betreiben, dar. Bei öffentlichen Kommunikationsnetzen findet man dabei oft eine hierarchische Struktur vor:

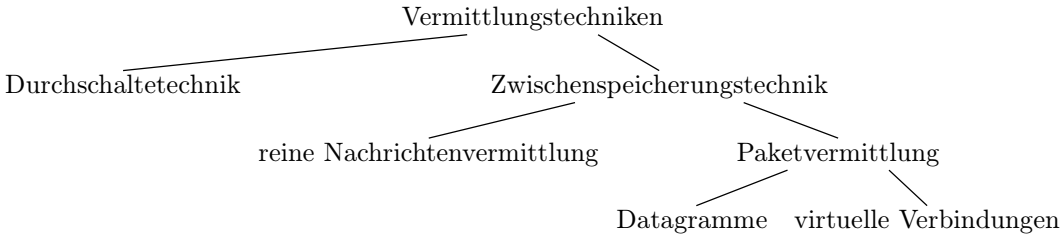
- Anbieter von *Übertragungsdiensten* (sogenannte *carrier*), die z.B. optische Leitungen, Satellitenverbindungen oder Funkstrecken anbieten
- Anbieter von *elementaren Daten- oder Sprachübertragungsdiensten*, wie dem X.25-Dienst

- Anbieter von *Internet-Zugangsdiensten* (sogenannte *internet service provider*), wie z.B. AOL, T-Online, oder HanseNet
- Anbieter von *höherwertigen Individualkommunikationsdiensten* über dienstintegrierte Netze, wie das ISDN-Netz
- Anbieter von Informations-, Auskunfts-, Suchdiensten u.ä., wie Fachinformationsdienste und Suchmaschinen im Internet (sogenannte *content provider*).

Internet Service Provider nutzen häufig die Telekommunikationsdienste einer Telefongesellschaft. Diese nutzt wiederum oft die Übertragungsdienste eines anderen Carriers, um einen Datenübertragungsdienst zu etablieren. Zu den Konsequenzen dieser Betreiberhierarchie gehören ein erschwertes Netzmanagement, eine komplizierte Abrechung der Kosten (accounting) und ein Konkurrenzkampf zwischen den Betreibern, der zu Preisvorteilen für den Verbraucher führt.

5.2 Vermittlungstechniken

Bei großen öffentlichen Netzen ist eine vollständige Vermaschung aller Kommunikationspartner unmöglich. Informationen, die zwischen zwei Kommunikationspartnern ausgetauscht werden, durchlaufen in solchen Netzen also in der Regel mehrere Knoten. Es werden also Techniken benötigt, um die Übertragung der Informationen über diese Knoten hinweg zu realisieren. Diese Techniken fassen wir unter dem Begriff *Vermittlungstechniken* zusammen.



Durchschaltetechnik Dieses Prinzip hat seinen Ursprung in den ersten Telefonnetzen. Bei der Durchschaltetechnik (auch: *Leitungsvermittlung, circuit switching*) wird für die gesamte Dauer der Kommunikation eine fest durchgeschaltete physikalische Verbindung aufgebaut (und danach abgebaut). Abbildung 5.5 zeigt, wie die einzelnen Verbindungen beim Verbindungsaufbau nach und nach etabliert werden. Man beachte, dass es sich hier um eine statische Reservierung der Betriebsmittel handelt und dass ein voll ausgelasteter Vermittlungrechner weitere Verbindungswünsche ablehnen muss.

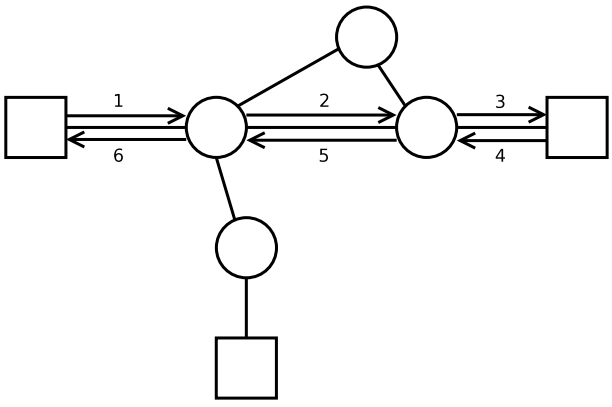


Abbildung 5.5. Schrittweiser Verbindungsaufbau

Zwischenspeicherungstechnik Dieses Prinzip wurde zuerst in wissenschaftlichen Netzen wie dem ARPAnet eingesetzt. Im Gegensatz zur Durchschaltetechnik werden bei der Zwischenspeicherungstechnik (auch: *store and forward*) keine dedizierten physikalischen Verbindungen etabliert. Stattdessen werden die Dateneinheiten schrittweise von einem Knoten zum nächsten übertragen. Hierbei speichern die Vermittlungsrechner die Dateneinheiten solange, bis sie an den nächsten Vermittlungsrechner oder bereits an das als Empfänger fungierende Endsystem weitergeleitet werden können. Die Dateneinheiten können bei diesem Prinzip auf unterschiedlichen Pfaden durch das Netz geleitet werden. Man beachte, dass bei diesem Prinzip die Betriebsmittel dynamisch alloziiert werden.

Wir fassen in Tabelle 5.1 noch einmal die Vor- und Nachteile beider Prinzipien zusammen. Im folgenden betrachten wir zwei unterschiedliche Arten der Zwischenspeicherungstechnik

Aspekt	Durchschaltetechnik	Zwischenspeichertechnik
Verzögerungszeit	weitgehend konstant	stark schwankend
Wegeermittlung	nur beim Aufbau	bei jeder Dateneinheit
Verbindungsaufbau	lohnt bei kurzen Übertragungen oft nicht	nicht erforderlich*
Effizienz	Verschwendung durch statische Reservierung	meist besser
Komplexität	im allgemeinen geringer	im allgemeinen höher

*) außer bei Nutzung virtueller Verbindungen

Tabelle 5.1. Gegenüberstellung von Durchschaltetechnik und Zwischenspeichertechnik

Reine Nachrichtenvermittlung Die Dateneinheiten werden bei der reinen Nachrichtenvermittlung (message switching) von einem Vermittlungsrechner frühestens dann weitergeleitet, wenn sie komplett empfangen wurden. Große Dateneinheiten können für die Übertragung in Blöcke fragmentiert[5.2] werden. Diese Blöcke werden von jedem Vermittlungsrechner reassembliert. Man sieht, dass diese Technik bei sehr großen Dateneinheiten (etwa beim Transfer einer großen Datei) drei grundsätzliche Probleme besitzt:

- Die Vermittlungsrechner müssen über einen großen Speicher verfügen. Bei sehr großen Dateneinheiten müssten diese eventuell auf Sekundärspeichern zwischengespeichert werden.
- Die Verzögerungszeit beträgt bei n Vermittlungsrechnern mindestens die n-fache Übertragungsdauer.
- Kurze Dateneinheiten (z.B. aus interaktiven Anwendungen) können durch große Dateneinheiten sehr lange blockiert werden.

Paketvermittlung Da die reine Nachrichtenvermittlung bei großen Dateneinheiten offensichtlich unpraktikabel ist, setzt man meist Paketvermittlung (packet switching) ein. Dabei werden die (aus der Fragmentierung der Dateneinheit) hervorgegangenen Blöcke unabhängig voneinander übertragen. Ein Vermittlungsrechner kann mit der Übertragung eines Blockes beginnen, wenn dieser komplett empfangen wurde – nicht erst, wenn die gesamte Dateneinheit empfangen wurde. Jeder Block trägt die Zieladresse in sich, so dass die Blöcke auf unterschiedlichen Pfaden übertragen werden können. Die Reassemblierung findet erst am Ziel statt. Die Verzögerungszeit wird auf diese Weise stark reduziert. Allerdings müssen Informationen wie Zieladresse und Position des Blockes in der Dateneinheit in jedem Block gespeichert werden. Bei der Wahl der (maximalen) Blockgröße stehen sich also zwei Faktoren gegenüber:

- Ist die Blockgröße zu klein, kann der Anteil der Zusatzinformationen an den zu übertragenden Daten unerwünscht groß werden.
- Ist die Blockgröße zu groß, treten die bei der reinen Nachrichtenvermittlung angegebenen Probleme auf.

5.2. Nur der erste Block muss die Zieladresse der Nachricht enthalten.

Für die Übertragung von digitalen Daten wird häufig Paketvermittlung durchgesetzt. Das Internetprotokoll IP ist in heutigen Rechnernetzen zweifellos das wichtigste Protokoll für Paketvermittlung.

Datagrammtechnik Bei dieser einfachsten Form der Paketvermittlung bewegen sich die Dateneinheiten vollkommen unabhängig voneinander vom Sender zum Empfänger. Auf viele wünschenswerte Eigenschaften einer dedizierten Leitung muss verzichtet werden. Stattdessen können die folgenden Effekte auftreten:

- Die Reihenfolge der empfangenen Pakete stimmt nicht mit der gesendeten überein.
- Einzelne Pakete können dupliziert werden oder verloren gehen.
- Übertragungsfehler werden in der Regel nicht korrigiert.
- Es existiert keine Flusskontrolle zwischen Sender und Empfänger.[5.3]

Dennoch wird die Datagrammtechnik häufig verwendet – insbesondere bei nur kurz andauernden Kommunikationsbeziehungen (z.B. Abfrage der Uhrzeit bei einem Timeserver) oder bei Echtzeitkommunikation (z.B. Voice over IP). Ein wichtiges Protokoll für die Datagrammvermittlung ist UDP (user datagram protocol).

Virtuelle Verbindung Um einige der Eigenschaften einer dedizierten Leitung auch bei der Verwendung von Paketvermittlung zu bieten, werden virtuelle Verbindungen (virtual circuit, VC) verwendet. Diese virtuellen Verbindungen stellen die ursprüngliche Reihenfolge der Pakete wieder her, fordern fehlerhaft empfangene oder verlorene Pakete erneut an und bieten häufig auch eine Flusskontrolle (bezogen auf die virtuelle Verbindung).

Virtuelle Verbindungen können unterschiedlich realisiert werden und der Charakter einer "Verbindung" kann unterschiedlich stark ausgeprägt sein. Wir unterscheiden deshalb die Varianten der Paketvermittlung noch detaillierter. Die Abbildungen 5.6 bis 5.8 zeigen die Varianten mit jeweils zwei virtuellen Verbindungen.

reines Datagramm. Datagramme werden zwischen kommunizierenden Prozessen versendet. Virtuelle Verbindungen existieren nicht. Als Beispiel hierfür dient UDP/IP.

geschichtete virtuelle Verbindung auf Datagrammbasis. Die Prozesse kommunizieren über *sockets*. Die von den kommunizierenden Anwendungsprozessen benutzten Betriebsystemdienste zur abgesetzten Interprozesskommunikation tauschen Datagramme aus. Die virtuelle Verbindung ist nur in den Arbeitsrechnern (und nicht innerhalb des Vermittlungsnetzes) bekannt. TCP/IP ist ein Beispiel für diese Variante.

Realisierung der virtuellen Verbindung in den Randknoten. Das Vermittlungsnetz bietet nach außen einen verbindungsorientierten Dienst an. Diese Verbindungen sind nur in den Randknoten des Vermittlungsnetzes bekannt. Innerhalb des Vermittlungsnetzes werden Datagramme ausgetauscht. Der X.25-Dienst in der Realisierungsvariante des früheren TRANSPAC-Netzes (Frankreich) entspricht dieser Variante.

ungeschichtete virtuelle Verbindung. Auf dem gesamten Pfad innerhalb des Vermittlungsnetzes sind die virtuellen Verbindungen der kommunizierenden Prozesse bekannt. Beispiele hierfür sind ATM sowie der X.25-Dienst in der Realisierungsvariante des früheren DATAPAC-Netzes (Kanada).

Abbildung 5.6 zeigt die zeitlichen Abläufe einer Datenübertragung bei (a) Leitungsvermittlung, (b) Nachrichtenvermittlung, (c) Datagrammvermittlung und (d) einer virtuellen Verbindung.

a) Beim Verbindungsaufbau tritt in jedem Knoten eine Verzögerungszeit für das Etablieren der Verbindung auf. Danach tritt allerdings keine Verzögerung in den Knoten auf. Für lange Dateneinheiten lohnt sich der Aufwand einer Verbindung also.

5.3. auch wenn eine Flusskontrolle zwischen den Vermittlungsrechnern existiert

b) Bei der Nachrichtenvermittlung wird die Sendung in jedem Knoten für die Sendedauer der gesamten Dateneinheit verzögert. Bei großen Dateneinheiten und vielen Knoten ist das in der Regel inakzeptabel.

c) Bei der Datagrammvermittlung tritt pro Knoten mindestens eine Verzögerung der Sendedauer eines Datagramms auf, da jeder Knoten die Daten erst weitersendet, wenn er die gesamte Dateneinheit empfangen hat.

d) Virtuelle Verbindungen verbinden bezüglich der Verzögerungszeit die Schwächen der vorigen Varianten. Der Verbindungsaufbau kostet Zeit und alle Datagramme werden in jedem Knoten verzögert. Dennoch stellen sie häufig eine sinnvolle Lösung dar.

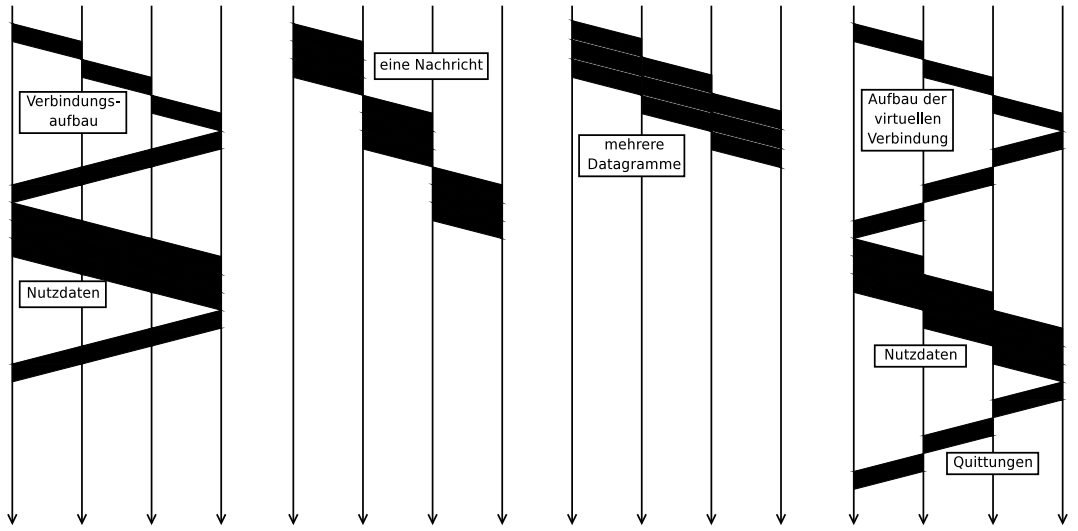


Abbildung 5.6. Zeitlicher Ablauf bei unterschiedlichen Vermittlungstechniken

5.2.1 Bewertung der Techniken

Wir fassen noch einmal die Vor- und Nachteile der Datagrammtechnik und der virtuellen Verbindung zusammen. Die einfachere Datagrammtechnik hat gegenüber der komplizierteren virtuellen Verbindung einige Vorteile.

- Die Datagrammtechnik eignet sich besonders bei Anwendungen mit stark fluktuierendem Verkehrsaufkommen.
- Sie eignet sich ebenfalls, wenn die Fehlerrate nicht vernachlässigbar gering sein muss.
- Es können alternative Pfade für jedes Paket gewählt werden.
- Die Schnittstelle eines Datagrammdienstes und dessen Implementierung sind einfacher als bei virtuellen Verbindungen.
- Die Kopplung unterschiedlicher Netze auf Datagrammbasis ist einfacher als bei virtuellen Verbindungen.

Dem gegenüber stehen die Vorteile der virtuellen Verbindung.

- Für die Pakete sind weniger Kontrollinformationen notwendig.
- Anwendungen mit Punkt-zu-Punkt-Verbindungen zwischen kommunizierenden Prozessen sind in natürlicher Weise abbildbar.
- Es ist eine einfachere Sättigungskontrolle durchführbar.

5.2.2 Zellenvermittlung

Bei einer speziellen Form der Paketvermittlung werden sehr kleine Pakete mit einer festen Größe[5.4] verwendet. Die Pakete werden in diesem Fall auch *Zellen (cells)* genannt, die Vermittlungstechnik Zellenvermittlung *(cell switching)*. ATM verwendet Zellenvermittlung mit 53 Byte großen Zellen. Dabei entfallen auf 48 Byte Nutzdaten weitere 5 Byte Kontrollinformationen. Der Durchsatz-Overhead von $\frac{5}{53} \approx 10\%$ ist damit beträchtlich. Die Vermittlung sehr kleiner Zellen mit konstanter Länge ist allerdings aus folgenden Gründen sehr schnell:

- Die konstante Länge vereinfacht die Interpretation und die Betriebsmittelverwaltung (für Speicherung und Übertragung) durch den Vermittlungsrechner.
- Die geringe Paketgröße führt zu einer kurzen Verweilzeit im Speicher des Vermittlungsrechners. Bei einer Datenrate von 1 GBit/s und einer Zellgröße von 53 Byte beträgt die Übertragungszeit einer Zelle nur noch $\frac{53 \cdot 8}{10^9}\,\mathrm{s} = 0,424\,\mu s$.

Zellen werden gewonnen, indem große Dateneinheiten wie Dateien fragmentiert werden oder kleine Dateneinheiten wie PCM-Abstastwerte verkettet werden. Man beachte, dass die letzte Zelle wegen der festen Größe in der Regel "aufgefüllt" werden muss.

5.3 Wegeermittlung

Wenn es in einem Kommunikationssystem alternative Pfade gibt, wird ein Mechanismus benötigt, der einen (möglichst günstigen) Pfad ermittelt. Bei der Durchschaltetechnik wird dieser Mechanismus beim Verbindungsaufbau benötigt, bei der Zwischenspeicherungstechnik möglicherweise für jede einzelne Dateneinheit. Mechanismen zur Wegeermittlung (engl.: routing[5.5]) nennen wir *Routing-Verfahren* oder *Routing-Algorithmen*.

Wir betrachten zunächst den Fall der Zwischenspeicherungstechnik. Dafür verwenden wir für die Vermittlungsrechner des Kommunikationssystems ein vereinfachtes Warteschlangenmodell (siehe Abbildung 5.7).

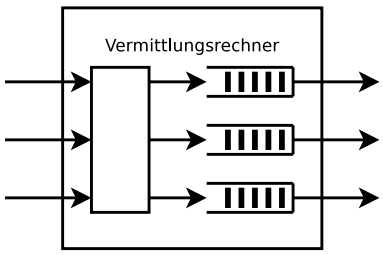


Abbildung 5.7. Vereinfachtes Warteschlangenmodell eines Vermittlungsrechners

Definition 5.1. *Unter* Wegeermittlung (routing) *verstehen wir die Zuordnung jeder empfangenen Dateneinheit zu einer der Ausgangswarteschlangen. Eine Sequenz von hintereinander liegenden Übertragungsleitungen, die einen Sendeknoten S mit einem Empfangsknoten E verbindet, inclusive der dabei zu passierenden Transitknoten, nennen wir* Pfad (path).

Definition 5.2. *Wird ein einzelner Pfad zwischen S und E für die Übertragung einer Menge von Dateneinheiten verwendet, sprechen wir von* Routing auf einfachem Pfad (single path). *Wird mehr als ein Pfad für die Übertragung einer Menge von Dateneinheiten benutzt, von* Routing auf multiplen Pfaden (multiple paths).

5.4. und nicht nur einer maximalen Größe

5.5. sprich: [ru:ting] oder [rauting]

Routing auf einfachem Pfad besitzt grundsätzlich zwei Vorteile. Zum einen muss die Wegeermittlung für die Menge der Dateneinheiten nur einmal durchgeführt werden. Zum anderen wird die Sequenz der Dateneinheiten auf dem Weg vom Sender zum Empfänger beibehalten, so dass keine aufwändige Umsortierung beim Empfänger nötig ist.

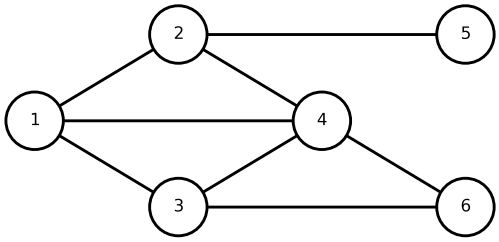
5.3.1 Ziele, Beispiele und Klassifikation

Wird ein Pfad durch ein Routing-Verfahren ermittelt, so wird (abgesehen von sehr einfachen Verfahren) versucht, aus der Menge der möglichen Pfade einen möglichst *günstigen* auszuwählen. Zur Beurteilung der Pfade kann eine *Kostenfunktion* verwendet werden. Mögliche Kostenfunktionen für einen Pfad sind:

- die Anzahl der auf dem Pfad liegenden Knoten (number of hops);
- die Summe der Kosten aller auf dem Pfad befindlichen Leitungen; die Kosten für eine Leitung können umgekehrt proportional zu deren Kapazität sein oder durch die aktuell gemessene Auslastung und Fehlerrate geprägt sein;
- die in der Vergangenheit gemessene Verzögerung beim Transport von Dateneinheiten auf diesem Pfad.

Die gewählte Kostenfunktion kann darüber hinaus vom Typ des zu übertragenden Verkehrs abhängen. So wird für Dialogverkehr hauptsächlich eine sehr niedrige Verzögerungszeit gewünscht, während beim Dateitransfer eher die Datenrate von Bedeutung ist.

Beispiel 5.3. Bei diesem einfachen Beispiel (siehe Abbildung 5.8) besitzt jeder Knoten eine Tabelle mit Wegeermittlungsinformationen, eine sogenannte *Routing-Tabelle*. In dieser Tabelle ist für jedes mögliche Ziel der nächste Knoten auf dem optimalen Pfad angegeben. Teilweise ist auch ein alternativer Nachfolger verzeichnet (in der Abbildung in Klammern dargestellt). Beim Erhalt einer weiterzuleitenden Dateneinheit leitet der Vermittlungsrechner diese an den in der Tabelle verzeichneten Knoten weiter. Ist dieser Knoten ausgefallen, kann der alternative Pfad eingeschlagen werden.



Router 1		Router 2	
1	–	1	1
2	2	2	–
3	3	3	1 (4)
4	4	4	4
5	2 (4)	5	5
6	3 (4)	6	4 (1)

Abbildung 5.8. Routing-Verfahren mit Routing-Tabellen (alternative Pfade in Klammern)

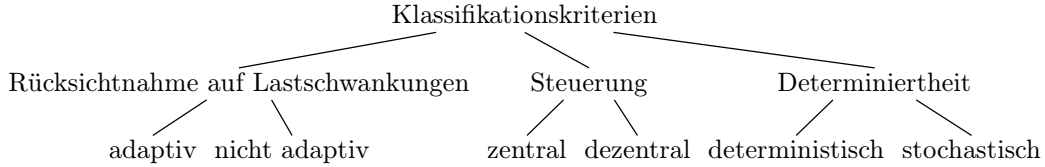
Ziele der Wegeermittlung

Wir fassen einige Ziele bzw. Qualitätsmerkmale für Wegeermittlungsverfahren zusammen:

- geringe Verweilzeit der Dateneinheiten im Kommunikationssystem
- kurze Speicherbelegung in den Vermittlungsrechnern
- geringer Implementationsaufwand
- möglichst geringer Overhead durch Kontrollinformationen
- gute Reaktionsfähigkeit auf Lastschwankungen und Topologieänderungen
- sehr geringe Wahrscheinlichkeit für Fehlersituationen wie Deadlocks und Zyklen[5.6]

Klassifikation

Zur Klassifikation von Routing-Verfahren verwendet wir zunächst drei Kriterien. So können Routing-Verfahren die Auslastung von Teilen des Kommunikationsnetzes messen (oder mitgeteilt bekommen) und diese in die Wegeermittlung mit einbeziehen. Solche Verfahren nennen wir *adaptiv*. Außerdem kann die zur Wegeermittlung verwendete Information (und damit auch die Verantwortung) bei einer einzelnen zentralen Instanz oder auf mehrere (oder alle) Knoten verteit sein. Zuletzt werden bei der Wegeermittlung entweder deterministische Entscheidungsregeln oder probabilistische Verfahren verwendet.



Der Algorithmus aus Beispiel 5.3 ist also adaptiv, deterministisch und dezentral. Als adaptiv bezeichnen wir ihn, weil er den Ausfall von Knoten (teilweise) berücksichtigt. Die "Wahrnehmung" der Knoten beschränkt sich allerdings darauf, dass sie den Ausfall eines Knotens feststellen können. Im folgenden beschreiben wir weitere Möglichkeiten zur Realisierung von adaptiven Verfahren.

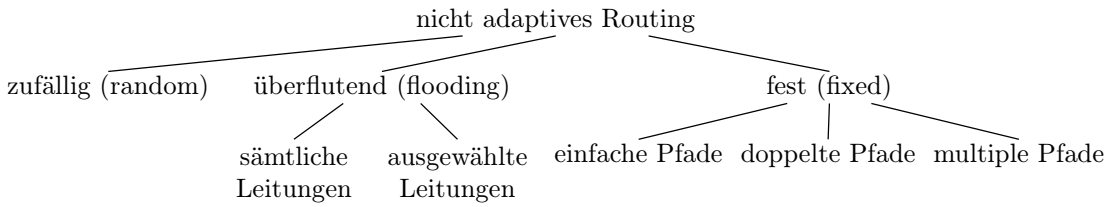
Realisierung von adaptiven Verfahren

Wir gehen nun davon aus, dass der Zustand des Kommunikationssystems detaillierter erfasst und berücksichtigt wird. Beispielsweise kann die Auslastung der einzelnen Knoten und Leitungen oder deren Fehlerhäufigkeit gemessen werden.

- Der Zustand des Kommunikationssystem (oder dessen Subsysteme) wird durch einen, mehrere oder alle Knoten gemessen.
- Der gemessene Zustand wird an alle, mehrere oder einen zentralen Knoten gesendet.
- Die neuen "optimalen Wege" werden berechnet und im zentralen Fall verteilt.
- Die Routing-Tabellen der Knoten werden aktualisiert.

5.3.2 Nicht adaptive Routingverfahren

Betrachten wir zunächst die Klassifikation für nicht-adaptive Routing-Verfahren. Wir unterscheiden zwischen drei Klassen:



Beim *zufälligen Routing* wird die Dateneinheit ohne Berücksichtigung des Empfängers oder des Netzzustandes an eine zufällig gewählte Ausgangsleitung gegeben. Die Ausgangsleitungen können identische oder unterschiedliche Wahrscheinlichkeiten besitzen. Zyklen können vermieden werden, indem das wiederholte Durchlaufen eines Knotens verboten wird. Auch mit vermiedenen Zyklen ist die Effizienz der Algorithmen dieser Klasse sehr schlecht.

5.6. wie der Ping-Pong-Effekt, bei dem eine Dateneinheit endlos zwischen zwei Vermittlungsrechnern hin- und hergereicht wird

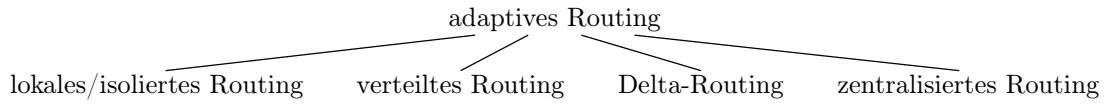
Beim *überflutenden Routing* werden Dateneinheiten an sämtliche (oder viele) Nachbarknoten weitergeleitet. Davon ausgenommen ist der Knoten, von dem die Dateneinheit erhalten wurde. Auch hier ist es sinnvoll, Zyklen (wie oben beschrieben) zu vermeiden. Der wesentliche Nachteil ist das beträchtliche zusätzliche Verkehrsaufkommen.

Beim *festen Routing* werden die Dateneinheiten nach festen Regeln (z.B. durch Routing-Tabellen repräsentiert) weitergegeben. Dabei werden einfache, doppelte oder multiple Pfade verwendet.

- Bei *einfachen Pfaden* ist für jedes Ziel genau ein Nachfolgerknoten festgelegt. Durch korrekt gesetze Nachfolgerknoten werden Zyklen vermieden. Allerdings findet keine Anpassung an Lastschwankungen und Ausfälle von Knoten statt.
- Bei *doppelten Pfaden* sind für jedes Ziel zwei Ausgangsleitungen festgelegt. Die Wahl der Ausgangsleitung findet entweder deterministisch oder stochastisch[5.7] statt. Auch hier können Zyklen durch korrekte Routing-Tabellen vermieden werden. Eine Anpassung an Lastschwankungen und Ausfälle kann (begrenzt) stattfinden. Besonders bei permanent starkem Verkehr und geringen Änderungen des Verkehrs (bezogen auf die Verkehrsmatrix) ist dieses Verfahren in der Praxis recht gut brauchbar.
- *Multiple Pfade* stellen eine Verallgemeinerung der doppelten Pfade dar, bei denen jeweils mehr als zwei Alternativen bestehen.

5.3.3 Adaptive Routingverfahren

Wieder betrachten wir zunächst die Klassifikation der adaptiven Routing-Verfahren die vier grundsätzliche Klassen unterscheidet.



Lokales Routing

Beim lokalen oder isolierten Routing findet das Erfassen des Netzzustandes durch jeden Knoten einzeln statt. Informationen über den Netzzustand werden nicht ausgetauscht. Wir unterscheiden zwei Unterklassen nach der Messung des Netzzustandes.

- Beim *"hot potato"-Routing* wird die Belegung der Warteschlangen der Ausgangsleitungen als Maß für die Auslastung der Nachfolgeknoten verwendet. Dateneinheiten werden ohne Berücksichtigung des Ziels an die Ausgangsleitung mit der geringsten Warteschlangenbelegung gesendet. So wird eine geringe Speicherbelegung in den Vermittlungsrechnern gewährleistet, das Leistungsverhalten ist jedoch sehr schlecht und das Verfahren daher nicht praktikabel.
- Bei *lokaler Abschätzung der Übertragungsverzögerung* wird eine Dateneinheit mit Ziel z im Knoten k an denjenigen Nachfolgerknoten n aus der Menge N weitergeleitet, für den gilt:

  $$T_k(n, z) = \min_{m \in N} T_k(m, z)$$

  wobei $T_k(m, z)$ für die von k beobachtete Verzögerung der von z gesendeten und über den Nachbarknoten m bei ihm eintreffenden Dateneinheiten steht. Die zu erwartende Übertragungsverzögerung der Dateneinheit für z wird also abgeschätzt, indem die Verzögerung der Dateneinheiten von z betrachtet werden. Dazu wird angenommen, dass die Verzögerung auf einem Pfad richtungsunabhängig ist. Außerdem muss ein regelmäßiger Verkehr zwischen k und z vorausgesetzt werden. Ist der Pfad zwischen k und z sehr lang, kann die gemessene Verzögerung veraltet sein und nicht mehr der momentan zu erwartenden Verzögerung entsprechen.

5.7. Wahl des ersten Pfades mit Wahrscheinlichkeit p und der Alternative mit Wahrscheinlichkeit $1 - p$

Verteiltes Routing

Zusätzlich zu den lokal gemessenen Informationen über den Netzzustand können Informationen von weiteren Knoten hinzugezogen werden. Diese Informationen werden entweder periodisch oder ereignisgesteuert ausgetauscht. In diesem Fall spricht man von verteiltem Routing. Eine Implementierung dieses Verfahrens diente ursprünglich als Routing-Verfahren im ARPA-Netz.

Werden nur direkte Nachbarknoten berücksichtigt, spricht man von *Nachbarknoten erster Ordnung*. Werden nur Knoten berücksichtigt, die über $n-1$ zwischenliegende Knoten (hops) erreicht werden können, von *Nachbarknoten n-ter Ordnung*. Je mehr Nachbarknoten mit einbezogen werden, desto größer ist der Overhead und die Komplexität des Algorithmus. Im allgemeinen verbessert sich dadurch aber die Abschätzung. Wir betrachten einige offene Fragen genauer:

Es stellt sich wieder die Frage, wie der Netzzustand gemessen wird. Als Parameter kann beispielsweise die Knotenauslastung, die Leitungsauslastung oder die Fehlerhäufigkeit einer Leitung gemessen werden. Im folgenden verwenden wir die Auslastung der Leitungen.

Dabei stellt sich die Frage, wie die Auslastung einer Leitung gemessen wird. Da eine Leitung zu einem Zeitpunkt immer entweder belegt oder frei ist (siehe Abbildung 5.9), ist die Definition einer Momentanauslastung ρ nur eingeschränkt sinnvoll. Wir können allerdings die Auslastung der Leitung in der nahen Vergangenheit für eine Abschätzung ρ^* verwenden, wenn wir annehmen, dass eine belegte Leitung eher belegt bleibt und eine freie Leitung eher frei.

Wir nehmen also an, dass für äquidistante Zeitpunkte t_i der Vergangenheit Messungen $a(t_i) \in [0,1]$ vorliegen. Diese Messungen beziehen sich jeweils auf das vorangegangene Intervall der Dauer Δt. Im einfachsten Fall könnte man den Mittelwert der letzten h Messungen bilden und als Abschätzung verwendet. Wir verbessern die Abschätzung ρ^* für den Zeitpunkt t_n, indem wir jüngere Messungen stärker gewichten als ältere, beispielsweise

$$\rho^*(t_n) = 0,5 \cdot a(t_{n-1}) + 0,3 \cdot a(t_{n-2}) + 0,15 \cdot a(t_{n-3}) + 0,05 \cdot a(t_{n-4})$$

oder allgemein mit k Koeffizienten: $\rho^*(t_n) = \sum_{i=1}^{k} w_i \cdot a(t_{n-i})$.

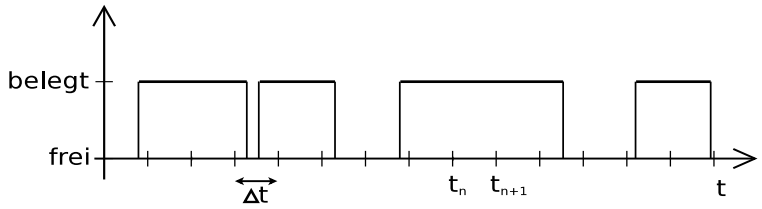


Abbildung 5.9. Abschätzung der Momentanauslastung einer Leitung

Eine weitere offene Frage betrifft den Zeitpunkt für die Aktualisierung. Am Anfang dieses Abschnittes haben wir schon periodische und ereignisgesteuerte Aktualisierung angesprochen. Als Ereignis kann dabei eine Änderung der gemessenen Auslastung dienen. Werden die Nachbarknoten jedes mal benachrichtigt, wenn sich die Auslastung auch nur geringfügig ändert, entsteht in der Regel ein nicht tolerabler Overhead. Stattdessen können Schwellwerte definiert werden. Wird einer der Schwellwerte unter- oder überschritten, erfolgt eine Benachrichtungung der Nachbarknoten. Dieser einfache Mechanismus hat einen großen Nachteil: Schwingt die approximierte Auslastung um einen der Schwellwerte, so werden dennoch sehr viele Benachrichtigungen versendet. Zwei Vorgehensweisen bieten sich an, um diesen Effekt zu vermeiden:

1. Die oben genannten Koeffizienten w_i zur Approximation der Momentanauslastung werden so gewählt, dass die ältere Vergangenheit stärker gewichtet wird. Dadurch wird die Approximation ρ^* geglättet.

2. Es werden stets Paare von Schwellwerten (ρ_o, ρ_u) definiert. Eine Aktualisierung erfolgt nur dann, wenn ein Schwellwert durchlaufen wird und die letzte Aktualisierung durch einen Durchlauf des jeweils anderen Schwellwerts verursacht wurde (siehe Abbildung 5.10). Die

Folge von Aktualisierungen wird also alternierend durch ρ_o und ρ_u erzeugt. Die Differenz $\rho_o - \rho_u$ ist dabei größer als die Oszillation, die unterdrückt werden soll.

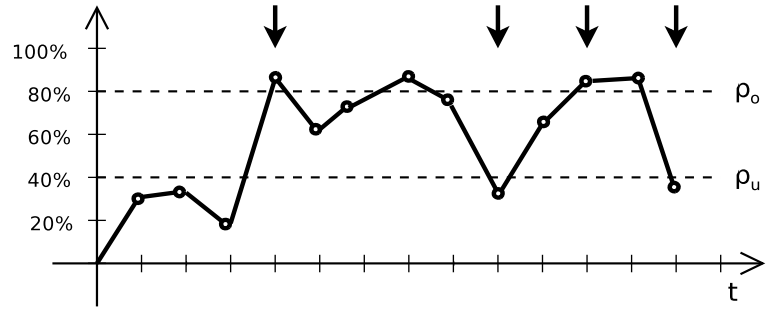


Abbildung 5.10. Schwellwertpaar als Aktualisierungskriterium (Aktualisierungszeitpunkte als Pfeile dargestellt)

Zuletzt sollte auch die absolute Kapazität der Leitungen in Betracht gezogen werden. Die verbleibende Datenrate einer zu 80% ausgelasteten Leitung mit 1 GBit/s Datenrate ist weit höher als die einer zu 10% ausgelasteten Leitung mit 100 MBit/s. Für die Wahl eines Pfades aus mehreren Alternativen kann also eine Gewichtung $g(l)$ der Leitung l nach ihrer Kapazität festgelegt werden. Das Gewicht eines Pfades P über die Leitungen $l_1 \ldots l_n$ ist dann

$$g(P) = \sum_{i=1}^{n} g(l_i)$$

Eine Möglichkeit zur Festlegung der Gewichtung einer Leitung als der zu minimierenden Größe auf dem gewählten Pfad ist

$$g(l_i) = \frac{1}{v_D} \cdot \frac{1}{(1-\rho_i^*)^2}$$

wobei ρ^* die oben definierte Abschätzung der Momentanauslastung und v_D die Datenrate der Leitung darstellt. Durch das Quadrieren des zweiten Faktors werden hohe Auslastungen besonders stark gewertet.

Delta-Routing

Eine Mischform zwischen zentralem und dezentralem Routing stellt das Delta-Routing dar. Hierbei sammelt eine zentrale Instanz Statusinformationen des gesamten Kommunikationssystems. Diese Informationen werden (eventuell relativ selten) an die Knoten verteilt. Die Knoten verwenden zur Wegeermittlung sowohl die von der Zentrale übermittelten Informationen als auch eigene lokale Beobachtungen.

Zentralisiertes Routing

Beim zentralisierten Routing werden sämtliche Routing-Entscheidungen von einer zentralen Instanz getroffen. Beispielsweise könnte der Absender für ein bestimmtes Ziel bei der Zentrale einen Pfad erfragen und diesen Pfad im Kopf der Dateneinheit speichern. Hierdurch entsteht aber ein beträchtlicher Overhead an zusätzlich zu transportierenden Dateneinheiten. Außerdem können Verfügbarkeitsprobleme entstehen, wenn die zentrale Instanz überlastet ist oder ausfällt.

5.4 Namensgebung und Adressierung

In Rechnernetzen gibt es eine Reihe von Objekten, denen Namen zugewiesen werden, um sie zu unterscheiden und benennen zu können. Dazu gehören Dateien, Prozesse, Speicherbereiche, periphere Geräte, Benutzer, etc. Tabelle 5.2 enthält einige Beispiele für Namen.

5.4.1 Begriffe und Beispiele

Definition 5.4. *Unter einem* Namen *verstehen wir eine Bezeichnung (in Form einer Kette von Symbolen) mit Bezug auf ein Objekt. Unter einer* Adresse *verstehen wir eine Angabe zur Lokalisierung eines Objektes.*

	Beispiel	Domäne
Rechner	`rzdspc3`, `tkrn182`	Rechner am Informatikum
Prozess	`2345`	Prozess-ID in UNIX-Systemen
Gerät	`eth2`, `hda1`	Gerätenamen unter UNIX
Datei	`readme.txt`	Dateiname unter MS-DOS
Benutzer	`1mueller`, `3schmidt`	Benutzerkennungen am Informatikum

Tabelle 5.2. Beispiele für Namen in Rechnernetzen

Wir unterscheiden also zwischen Namen und Adressen derart, dass Namen zur Bezeichnung und Unterscheidung und Adressen zum Auffinden von Objekten dienen. Für Namen und Adressen werden häufig bestimmte Konventionen festgelegt. Diese können die Länge der Kette (variabel oder fest) sowie den Symbolvorrat betreffen (Bits, Zeichen, numerisch, alpanumerisch, etc.).

5.4.2 Namensgebung

Namen dienen, wie schon erwähnt, zur Benennung und Unterscheidung von Objekten. Hierfür ist es offensichtlich notwendig, dass die Namen eindeutig sind. Beispielsweise kann ein Ordner unter MS-DOS keine zwei Dateien mit demselben Namen enthalten und keine zwei Ethernet-Karten haben die gleiche MAC-Adresse. Offensichtlich kann sich die Eindeutigkeit auf sehr unterschiedliche Bereiche (Ordner bzw. Planeten) beziehen. Wir verwenden u.a. die folgenden Begriffe um die Eindeutigkeit eines Namens zu beschreiben.

- benutzereindeutig
- rechnereindeutig
- netzeindeutig
- domaineindeutig
- weltweit eindeutig

Die eindeutige Zuordnung von Namen erfordert oft Koordination. So sind unterschiedliche Instanzen für unterschiedliche Bereiche zuständig. Eindeutige Namen in einem größeren Bereich lassen sich oft durch Kombination von Namen erlangen. So kann beispielsweise die Kombination von Prozess-ID und Rechnername (z.B. `1234-pc12`) zu einem netzeindeutigen Namen für den Prozess führen. Setzt man diese Vorgehensweise fort, kommt man zu den hierarchischen Verknüpfungen (hierarchical concatenation), wie sie im globalen Internet oder im Fernsprechnetz üblich sind: `3meier@swtpc2.informatik.uni-hamburg.de` bzw. `0049-4-101-83477`.

Hierarchische Verknüpfung

Diese hierarchische Vergabe der Namen hat den Vorteil, dass die Zuständigkeiten verteilt werden. Internationale Organisationen vergeben beispielsweise Ländervorwahlen oder *top level domains*, regionale Organisationen brauchen nur regional eindeutige Namen zu vergeben. Außerdem tragen solche Namen oft schon Informationen zum Auffinden des Objektes in sich (z.B. ist der Wohnort im Kreis Pinneberg aus der o.g. Telefonnummer ersichtlich), was ihnen zusätzlich den Charakter einer Adresse gibt.

Die heterogene Namensvergabe kann aber auch problematisch sein, wenn Konventionen fehlen oder nicht eingehalten werden. Darüber hinaus kann es schwierig sein, Knoten mit einer neuer "Art" der Namensgebung in die bestehende Hierarchie einzugliedern. Im Bereich Internet Domain Naming steht beispielsweise die top level domain entweder für eine Nation (`de`, `fr`, `it`) oder für eine Organisationsart (`edu`, `gov`, `com`, `org`).

Dynamische Namenszuteilung

Manchmal ist es sinnvoll, Namen nur für ein bestimmtes Zeitintervall zu vergeben. Das ist beispielsweise dann der Fall, wenn Kommunikationsbeziehungen nur kurz andauern und die Menge der möglichen Kommunikationspartner sehr groß ist. Wird die Zahl der zu einem Zeitpunkt vergebenen Namen klein gehalten, können kürzere Namen verwendet werden.

Wir betrachten nun den Fall, in dem eine Menge von Prozessen P, die auf Rechnerknoten K laufen miteinander kommunizieren wollen. Für die Kommunikation stehen eine Menge von Namen N zur Verfügung. Die Prozesse besitzen zunächst nur rechnereindeutige Namen.

Beispiel 5.5. Auf jedem Rechnerknoten läuft ein Prozess Q_i, der für die Namensverwaltung zuständig ist. Diesen Prozessen ist a priori ein Name $n_Q \in N$ zugewiesen. N_Q bezeichnet die Menge der a priori zugewiesenen Namen. Die restlichen Namen $N \backslash N_Q$ werden auf die Namensverwaltungsprozesse aufgeteilt (Partitionierung).

Wir betrachten nun den Fall, dass der Prozess P_{134} im Knoten K_{71} mit dem Prozess P_{352} in Knoten K_{28} kommunizieren möchte.

1. P_{132} teilt dem Namensverwaltungsprozess Q_{28} im Knoten K_{28} seinen Wunsch mit: `P352@K28`
2. Q_{28} ermittelt den freien netzeindeutigen Namen `27` und nimmt eine dynamische Zuordnung von `27` zu `P352` vor.
3. Q_{28} teilt P_{132} den Namen `27` mit.
4. P_{132} kann nun mit P_{352} unter Verwendung des kürzeren[5.8] Namens `27` kommunizieren.
5. Ist die Kommunikation zwischen P_{132} und P_{352} beendet, wird die dynamische Zuordnung aufgehoben und der Name wieder freigegeben.

Es folgt eine stichwortartige Grobbeurteilung der dynamischen Namenszuteilung.

- Die Namensverwaltung ist stark zentralisiert.
- Die Menge der pro Knoten zu verwaltenden netzeindeutigen Namen ist relativ klein.
- Es ist eine starre Zuordung von Prozessen zu Knoten notwendig.
- Der Initiator einer Kommunikationsbeziehung benötigt umfangreiche Informationen über den Kommunikationspartner.
- Die Implementation ist recht komplex.
- Die Auflösung einer dynamischen Zuordung muss allen Kommunikationspartnern mitgeteilt werden.

Statische Namenszuteilung

Bei der statischen Namenszuteilung verwaltet das Netz eine a priori definierte Menge von netzeindeutigen Namen N. Es erfolgt eine statische Zuordung von N zu der Menge der Prozesse, die für Kommunikationsbeziehungen in Frage kommen. Kommunizierende Prozesse kennen gegenseitig ihre netzeindeutigen Namen, die Abbildung ("mapping") von netzeindeutigen Namen auf die lokalen Namen der Prozesse erfolgt in den Knoten.

Ein Beispiel für eine derartige Namenszuteilung stellt die Rufnummernvergabe bei öffentlichen Mobilfunknetzen dar. Die Teilnehmer besitzen netzeindeutige Namen (ihre Telefonnummern), die zur Adressierung und Lokalisierung der mobilen Teilnehmer verwendet werden.

Die lokalen Namen, die Objekte zusätzlich zu ihren netzeindeutigen Namen besitzen können, sind in jedem Knoten frei wählbar, da sie nach außen nicht sichtbar sind und somit auch keine Konflikte entstehen können. Darüber hinaus ist die Verlagerung eines Objektes in einen anderen Knoten möglich, ohne das dessen Kommunikationspartner davon wissen müssen. Beispielsweise kann ein Teilnehmer im Mobilfunknetz durch seinen netzeindeutigen Namen unabhängig von dessen Position (Mobilfunkzelle) von anderen Teilnehmern adressiert werden.

5.8. kürzer als `P352@K28`

Zu Beginn dieses Abschnittes haben wir zwischen Namen und Adressen unterschieden. In unseren Beispielen wurden Objekte aber dennoch häufig über ihren Namen adressiert. Es existieren aber auch andere Möglichkeiten der Adressierung von Objekten:

- durch Adressierung nach Inhalt oder Wert des Objektes
- durch Angabe des Absenders, sofern nur ein Empfänger in Frage kommt
- durch Angabe des Namens einer adressierten Gruppe
- durch Angabe des Weges zum gewünschten Objekt
- durch Angabe einer Beziehung des gewünschten Objektes zum Absender (z.B. `own_adr+1`).

Darüber hinaus wird häufig die Adressierung mehrerer Objekte benötigt. Hierfür haben sich die englischen Begriffe *multicast* (bei Adressierung mehrerer Objekte) und *broadcast* (bei Adressierung aller möglichen Objekte) eingebürgert. Beispiele für *multiple Adressierung* sind:

- die explizite Angabe der Zielobjekte (z.B. als Liste)
- die Angabe einer Gruppe
- die implizite Adressierung aller Objekte (z.B. durch eine dafür reservierte Adresse).

Vom Namen zur Adresse

Eine häufige Problemstellung ist es, zu einem bekannten Namen eine zugehörige Adresse zu finden. Beispiele für solche Dienste sind die Telefonbücher und Telefonauskünfte im öffentlichen Telefonnetz sowie *name server* oder *directories* in Rechnernetzen. Diese Beispiele führen uns zur Definition des Namensverwalters.

Definition 5.6. *Ein* Namensverwalter (name server) *ist eine Instanz in einem verteilten System oder Rechnernetz, die (ggf. in Kooperation mit anderen Namensverwaltern) eine Abbildung von einem Namensraum*[5.9] R_1 *in einen anderen Namensraum* R_2 *vornimmt. Der von einem Namensverwalter erbrachte Dienst heißt* Namensverwaltungsdienst (name service).

Beispiel 5.7. Im ersten Beispiel sei R_1 eine Menge von netzeindeutigen Namen, R_2 die Menge der Adressen der Objekte in einem verteilten System. Der Namensverwaltungsdienst besteht darin, zu einem Namen die entsprechende Adresse zu liefern.

Im zweiten Beispiel steht R_1 für eine Menge von Personen (charakterisiert durch Name, Strasse und Wohnort) und R_2 für die Menge der zu den Personen gehörenden Telefonnummern. Ein Dienst, der diese Abbildung vornimmt, ist u.a. die Telefonauskunft.

Weitere Beispiele sind Anfragen der Art

- Wo ist der Aufenthaltsort des Mobilfunkbenutzers mit folgender Telefonnummer?
- Welche IP-Adresse gehört zur folgenden ATM-Adresse?
- An welchem Rechner befindet sich der folgende Benutzer?

Namensverwaltungsdienste können zentral durch eine einzelne Instanz oder verteilt, von mehreren Instanzen erbracht werden. Für den Fall der verteilten Diensterbringung unterscheiden wir zum einen, ob die Informationen durch

- Replikation oder
- Partitionierung

auf die Instanzen verteilt werden. Außerdem gibt es unterschiedliche Möglichkeiten, für die Wahl des Namensverwalters durch den Nutzer des Dienstes:

- ein einzelner Verwalter wird gezielt angesprochen
- alle Verwalter werden angesprochen (broadcast)

5.9. eine Menge von Namen

- ein bevorzugter Verwalter, der möglicherweise Unteranfragen abschickt, wird angesprochen.

Von der Adresse zum Pfad

Zuständig für die Abbildung von Adressen auf Pfade sind die im vorigen Abschnitt angesprochenen *Routing-Komponenten*. Häufig wird die Adresse (oder ein Teil davon) herangezogen, um sich möglichst schnell und einfach dem gewünschten Objekt zu nähern. Hierfür eignen sich besonders Adressen, die durch hierarchische Verknüpfung gebildet wurden.

Grundsätzlich gibt es zwei Möglichkeiten für den Zeitpunkt der Zuordnung des Pfades zu einer Adresse. Sie kann als *statische Routenzuordnung* durch den Absender erledigt werden (vgl. source routing) oder als *flexible, dynamische Routenzuordnung* erst auf dem Weg zum Empfänger vervollständigt werden. In diesem Fall liegt eine verteilte Entscheidungsfindung durch mehrere Routing-Komponenten vor. Details der Wegeermittlung wurden bereits im Abschnitt 5.3 besprochen.

5.5 Interkonnektion von Netzen

Eine große Bedeutung kommt der Interkonnektion (Kopplung) bestehender Rechnernetzen zu. Der Wunsch nach netzübergreifender Kommunikation und der Nutzung der Dienste anderer Netze sind die wichtigsten Ursachen dafür. Die Verbindung unterschiedlicher Rechnernetze (mit oftmals unterschiedlichen Architekturen) führt allerdings häufig zu Problemen. Einige Aspekte, die Schwierigkeiten verursachen können, sind:

- Heterogenität der Architektur
- Netz- und Datensicherheit
- Effizienzeinbußen
- Kostenabrechung (accounting)
- übergreifendes Netzmanagement
- aufwändige Fehlersuche
- unklare Zuständigkeiten

Die Verbindung unterschiedlicher Rechnernetze erfolgt in der Regel durch Spezialrechner, sogenannte *Gateway-Rechner*. Diese müssen in der Lage sein, auf einer Schicht der Protokollhierarchie, eine Konvertierung der Protokolldateneinheiten der in den Netzen verwendeten Protokolle vorzunehmen bzw. die Netzdienste einer Schicht geeignet aufeinander abzubilden. Unterhalb dieser Schicht müssen sie die Protokolle aller Netze implementieren, die sie miteinander verbinden. Aspekte, die bei der Interkonnektion von Rechnernetzen berücksichtigt werden müssen, sind:

- Adressierung
- Wegeermittlung (routing) – nicht nur netzintern sondern auch zwischen Netzen
- Fehlerbehandlung
- Fragmentierung und Reassemblierung von Dateneinheiten
- Flusskontrolle zwischen Netzen
- Kostenabrechnung (accouting)
- Netzsicherheit[5.10]

Wird ein Gateway-Rechner eingesetzt, um mehrere Netze zu verbinden, spricht man von *direkter Kopplung*. Ist die Zahl der zu koppelnden Netze n groß, kann diese Aufgabe sehr komplex sein, wie man Tabelle 5.3 für den Fall $n = 7$ entnehmen kann. Ein einzelner Gateway-Rechner muss $\frac{n \cdot (n-1)}{2}$ Transformationsvorschriften beherrschen.

5.10. z.B. in Form einer *firewall*

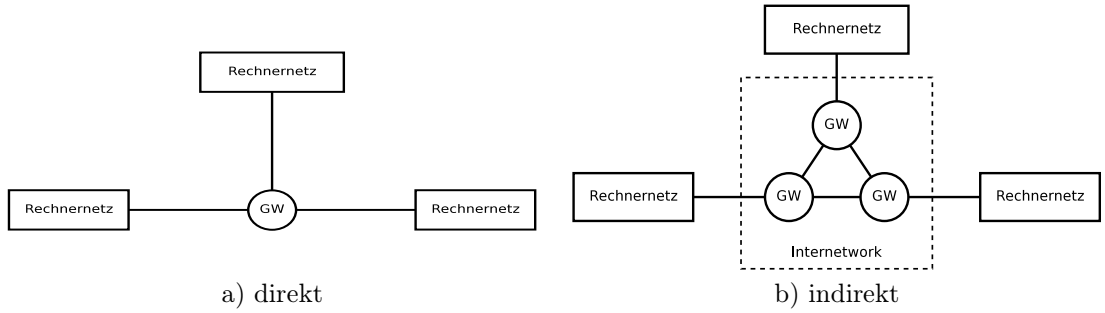


Abbildung 5.11. Direkte und indirekte Kopplung

Ein Ausweg stellt die *indirekte Kopplung* dar. Bei ihr wird jedem Netz ein eigener Gateway-Rechner zugeordnet, der die Protokolle dieses Netzes beherrscht. Die Gateway-Rechner bilden ein eigenes Netz (internetwork) und kommunizieren untereinander mit einem eigenen Protokoll, dem *internetwork protocol*. In diesem Fall ist die Zahl der benötigten Gateway-Rechner deutlich größer, deren Komplexität dagegen reduziert, da sie nur die Transformation zwischen dem in ihrem Netz verwendeten Protokoll und dem internetwork protocol beherrschen müssen.

	direkt	indirekt
benötigte Gateway-Rechner	1	7
Transformationen insgesamt	21	7
Transformationen pro Gateway-Rechner	21	1

Tabelle 5.3. Benötigte Gateway-Rechner und Transformationen bei der Kopplung von 7 Netzen

Wir haben schon festgestellt, dass der Gateway-Rechner die Transformation der Dateneinheiten auf einer bestimmten Schicht der Protokollhierarchie vornehmen muss. Meist wird dafür eine Schicht ausgewählt, auf der die Protokolle möglichst ähnlich sind. Häufig (aber nicht notwendigerweise) sind die höheren Protokolle der zu koppelnden Netze identisch. Beispielsweise wird TCP/IP sowohl bei Token Ring als auch bei Ethernet eingesetzt. Wir betrachten nun ein generisches Architekturmodell für einen Gateway-Rechner bei der direkten Kopplung zweier Netze (Abbildung 5.12).

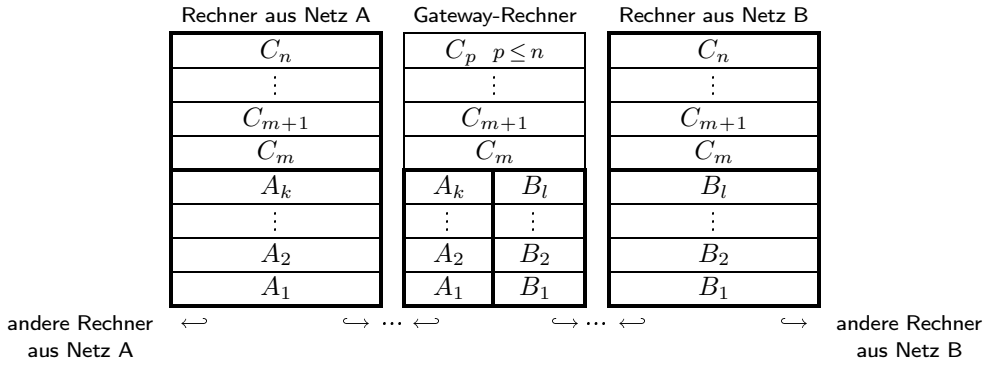


Abbildung 5.12. Generisches Architekturmodell für direkte Kopplung

Ein Gateway-Rechner verbindet zwei Netze A und B. Diese Netze arbeiten ab der Schicht m mit denselben Protokollen. Unterhalb dieser Schicht verwenden sie jedoch unterschiedliche Protokolle: $A_1 ... A_k$ im Netz A sowie $B_1 ... B_l$ im Netz B. Diese Protokolle werden vom Gateway-Rechner implementiert. Außerdem implementiert er eine Transformation der Dateneinheiten zwischen A_k und B_l. Darüber hinaus kann auch der Gateway-Rechner höhere Protokolle bis zu einer Schicht p implementieren, um weitere Dienste anzubieten. Mit diesem generischen Modell lassen sich viele übliche spezielle Gateway-Rechner beschreiben:

Bezeichnung	k	l	p
Application Level Gateway	7	7	7
Switch	3	3	3
Switch mit Hostfunktionalität	3	3	7
Bridge/Hub	2	2	2
Repeater	1	1	1

Tabelle 5.4. Typische Sonderfälle des Architekturmodells

Abbildung 5.13 zeigt das entsprechende Architekturmodell für den Fall der indirekten Kopplung. Hier stellt sich die Frage nach einer geeigneten Zwischenarchitektur J. Es folgen einige Beispiele für Zwischenarchitekturen.

- auf TCP/IP basierende Protokollhierarchien mit Kopplung auf der Vermittlungs- oder Transportschicht ($j=3$ oder $j=4$)
- Kopplung auf Schicht 3 im postalischen Vermittlungsnetz wie bei X.25
- Verwendung des Standard-Architekturmodells OSI mit $j=7$
- SNA in früheren Netzen mit DECnet-SNA-Gateways (aus den Zeiten herstellerspezifischer Netzarchitekturen)

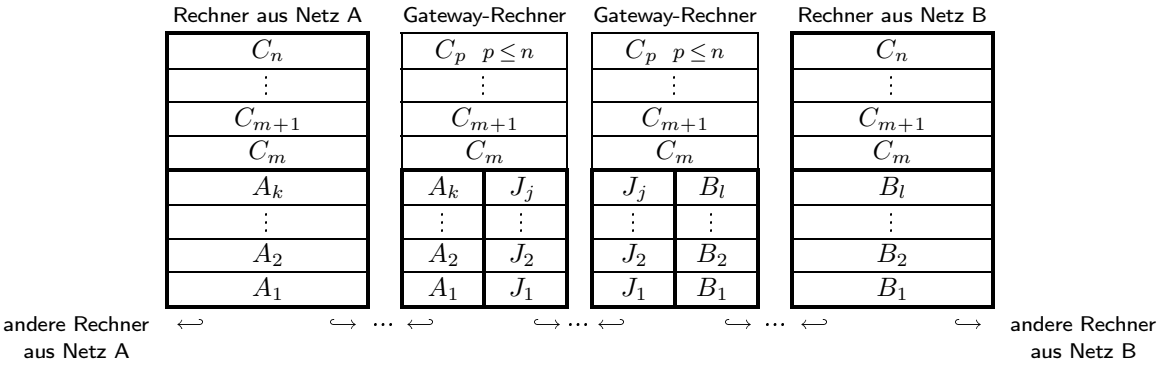


Abbildung 5.13. Generisches Architekturmodell für indirekte Kopplung mit den Internetwork-Protokollen $J_1 ... J_j$

Dienst- und Protokollkopplung

Im vorigen Abschnitt haben wir stets eine Kopplung über die Protokolle betrachtet. Dies bietet sich an, wenn die zu koppelnden Protokolle weitgehend identisch sind, so dass die Dateneinheiten des einen Protokolls recht einfach in die Dateneinheiten das anderen Protokolls überführt werden können. Abbildung 5.14 zeigt ein einfaches Beispiel, für den Fall identischer Protokolle mit der Ausnahme, dass die maximale Paketgröße unterschiedlich ist. Die Transformation besteht in diesem Fall ausschließlich in der Fragmentierung bzw. Reassemblierung der Dateneinheiten.

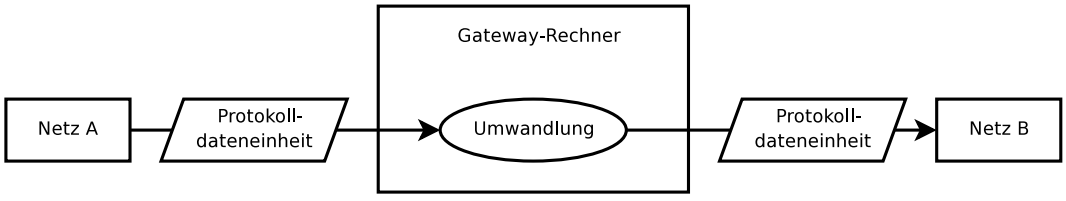


Abbildung 5.14. Beispiel für Kopplung auf Protokollebene

Es existiert allerdings auch ein anderes Paradigma für die Kopplung von Netzen, die *Dienstkopplung*. Sie bietet sich an, wenn weitgehend identische Dienste in beiden Netzen vorhanden sind. Als Beispiel betrachten wir die Nutzung des Dateitransfer-Dienstes eines Servers in Netz B durch einen Client in Netz A. Wir gehen davon aus, das die Dienste für Dateitransfer in beiden Netzen sehr ähnlich sind, die dafür verwendeten Protokolle aber sehr unterschiedlich sind und sich schlecht für die Protokollkopplung eignen.

Die Lösung besteht nun darin, dass der Gateway-Rechner sowohl den Server-Dienst des Netzes A als auch den Client-Dienst des Netzes B implementiert. Möchte der Client eine Datei vom Server in Netz B laden, wendet er sich mit dem in seinem Netz üblichen Protokoll an den Server-Dienst auf dem Gateway-Rechner. Dieser verhält sich dem Client gegenüber, als verfüge er über die angeforderte Datei. Tatsächlich lädt er die Datei aber mit Hilfe des in Netz B üblichen Protokolls vom Server in Netz B und stellt sie nach Erhalt dem Client zur Verfügung.

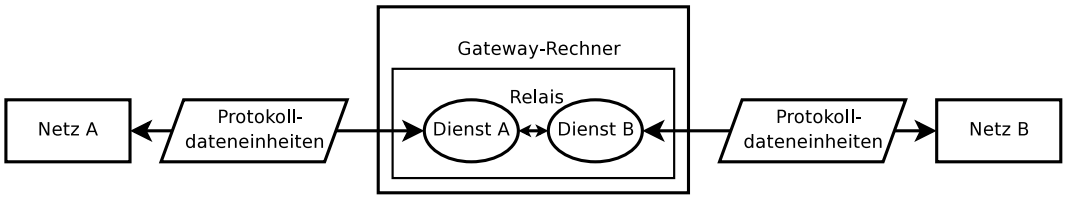


Abbildung 5.15. Beispiel für Kopplung auf Dienstebene

Wir stellen noch einmal Dienstkopplung und Protokollkopplung gegenüber. In beiden Fällen wird eine Sequenz von Protokolldateneinheiten des einen Protokolls in eine Sequenz von Protokolldaten des anderen überführt. Bei der Protokollkopplung geschieht dies anhand relativ einfacher Regeln, während bei der Dienstkopplung in der Regel beide Dienste implementiert werden müssen. Protokollkopplung ist daher im allgemeinen einfacher, jedoch nur möglich, wenn die Protokolle ähnlich genug sind.

5.6 Schmalband-ISDN

Das Integrated Service Digital Network (ISDN) ist ein internationaler Standard für leitungsgebundene Telekommunikation. Dieser Standard verfolgte u.a. das Ziel, unterschiedliche Dienste wie Telefon, leitungsvermittelnde Datenübertragung (Datex-L), paketvermittelnde Datenübertragung (Datex-P) und Telefax in einem gemeinsamen Netz zu integrieren. Die für die Teilnehmer wohl wichtigste Neuerung war die Möglichkeit, die bestehende physikalische Leitung durch Multiplexen wie zwei unabhängige Leitungen nutzen zu können. Dadurch ist es möglich, gleichzeitig zwei Telefongespräche zu führen, während eines Telefongesprächs eine Verbindung zu einem Internet Service Provider zu nutzen sowie durch *Kanalbündelung* eine relativ hohe Datenrate von 128 kBit/s zu erreichen. Für Großkunden wurden spezielle *Primärmultiplexanschlüsse* angeboten, die 30 Nutzdatenkanäle durch Multiplexen realisieren.

In Deutschland hat die Deutsche Post in den frühen 1980er Jahren damit begonnen, das gesamte Telefonnetz auf die digitale Technik umzustellen. Besitzern eines ISDN-Anschluss wird ein spezielles Gerät (*Network Termination, NT*) zur Verfügung gestellt, das die bestehende Telefonleitung für eine digitale Übertragung mit der Ortsvermittlungsstelle benutzt.[5.11] Bei Kunden, die weiterhin den analogen Anschluss wünschen, findet Datenübertragung bis zur Vermittlungsstelle analog statt. Erst dort wird die Umwandlung in ein digitales Signal vorgenommen.

Abbildung 5.16 zeigt die unterschiedlichen Schnittstellen bei ISDN mit den jeweils typischen Geräten. Alle digitalen Geräte im Haushalt des Kunden kommunizieren mit dem NT über eine S-Schnittstelle. Analoge Geräte werden indirekt über einen *Terminal Adapter* (z.B. eine Telefonanlage) mit dem NT verbunden.

5.11. Bei einem einfachen Anschluss mit zwei Nutzdatenkanälen wird dieses Gerät "Network Termination for ISDN Basic Rate Access" (NTBA) genannt.

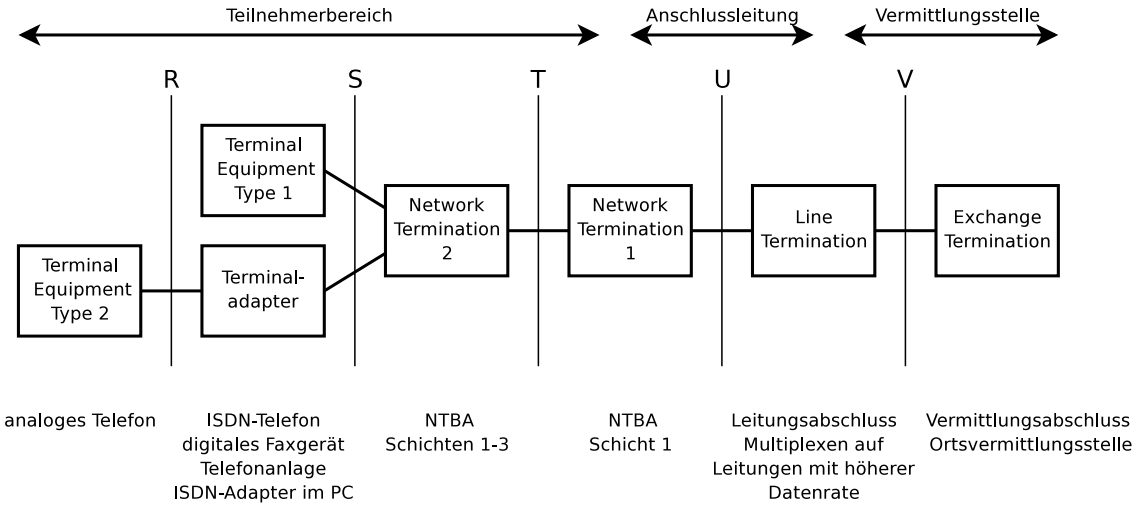


Abbildung 5.16. Schnittstellen und Bezugspunkte bei ISDN

5.6.1 Die Schnittstellen S_0 und S_{2M}

Die folgenden Abbildungen zeigen die Funktionen der beiden Schnittstellen S_0 und S_{2M}. Die S_0-Schnittstelle wird bei einfachen Anschlüssen mit zwei Nutzdatenkanälen verwendet. Die Kanäle für Nutzdaten werden *B-Kanäle* genannt und besitzen eine Datenrate von jeweils 64 kBit/s (vgl. digitaler Sprachkanal). Darüber hinaus werden ein *D-Kanal* und ein *YY-Kanal* mit 16 kBit/s bzw. 48 kBit/s für Aufgaben wie Steuerung, Synchronisation, Überwachung etc. genutzt. Daraus ergibt sich eine Bruttodatenrate von 192 kBit/s und eine Nettodatenrate von 144 kBit/s.

Terminal Eqipment	S_0	Network Termination
2 B-Kanäle (je 64 kBit/s)	$\longleftrightarrow$	
D-Kanal (16 kBit/s)	$\longleftrightarrow$	
Bittakt	$\longleftrightarrow$	
Worttakt	$\longleftrightarrow$	
Synchronisation	$\longleftarrow$	
Echokanal	$\longleftarrow$	
Fernspeisung	$\longleftarrow$	
Aktivierung	$\longrightarrow$	
Deaktivierung	$\longleftarrow$	
TE von S_0 getrennt	$\longleftarrow$	

Abbildung 5.17. Aufbau der S_0-Schnittstelle bei ISDN

Bei der S_{2M}-Schnittstelle können 30 solcher B-Kanäle genutzt werden. Für Steuerung, Synchronisation, Überwachung etc. existieren ein D-Kanal und ein *XX-Kanal* mit jeweils 64 kBit/s. Die Datenrate des gesamten Anschlusses liegt damit bei 2 MBit/s (brutto) bzw. 1984 kBit/s (netto).

Terminal Eqipment	S_{2M}	Network Termination
30 B-Kanäle (je 64 kBit/s)	$\longleftrightarrow$	
D-Kanal (64 kBit/s)	$\longleftrightarrow$	
Bittakt	$\longleftrightarrow$	
Worttakt	$\longleftrightarrow$	
Rahmensynchronsiation	$\longleftrightarrow$	

Abbildung 5.18. Aufbau der S_{2M}-Schnittstelle bei ISDN

Man beachte, dass die Pfeile in den obigen Abbildungen Funktionen und nicht etwa physikalische Leitungen darstellen. Bei der S_0-Schnittstelle werden zwei Kupferdoppeladern verwendet, so dass Vollduplex-Betrieb möglich ist. Als Codierung wird ein AMIe-Variante (siehe Tabelle 2.4) verwendet, wobei spezielle Symbole zur Synchronisation durch Coderegelverletzungen (violations) repräsentiert werden. Die einzelnen Kanäle werden durch Zeitmultiplexen realisiert. Abbildung 5.19 zeigt das Rahmenformat der S_0-Schnittstelle. In 250 μs werden 32 Bit für die beiden B-Kanäle und vier Bit für den D-Kanal übertragen. Das Bit F kennzeichnet den Beginn eines Rahmens, L dient als Paritätsbit zum Wechselstromausgleich. Der D-Kanal wird vom NT gespiegelt und über den "Echo-Kanal" (innerhalb der E-Bits) an das Endgerät zurückgesendet. Die S-Bits werden für unterschiedliche Steuerungsaufgaben verwendet.

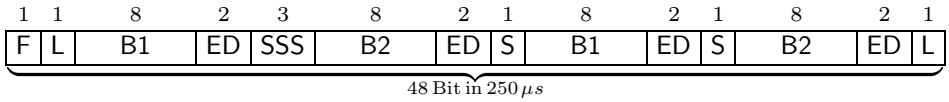
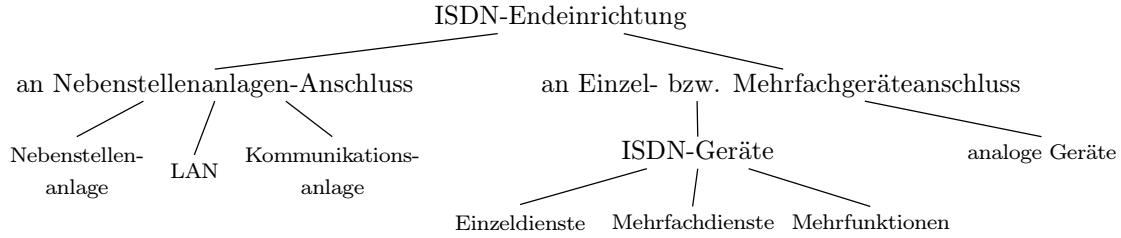


Abbildung 5.19. Rahmenstruktur der S_0-Schnittstelle

5.6.2 Endeinrichtungen

Die Endeinrichtungen (Terminal Equipment, TE) lassen sich wie folgt klassifizieren. Sie können in Stern-, Bus-, Ring- und Punkt-zu-Punkt-Topologien mit dem NT verbunden werden.



5.7 ATM-Protokolle und Netze

Im Kapitel über Wegeermittlung sind wir kurz auf die Zellenvermittlung zu sprechen gekommen. Ein wichtiger Standard für Zellenvermittlung ist ATM (asynchronous transfer mode). ATM wird häufig als Basisdienst für verbindungsorientierte Dienste wie B-ISDN und verbindungslose Dienste wie IP verwendet. ATM-Netze sind also in der Regel Backbone-Netze.

Als Vermittlungstechnik kommt Zellenvermittlung zum Einsatz, die wegen der kurzen und festen Paketgröße eine schnelle Vermittlung und kurze Verweilzeiten der Pakete realisiert. ATM realisiert ungeschichtete virtuelle Verbindungen, um einen verbindungsorientierten Dienst zu bieten. Die Funktionsweise der Wegeermittlung für Zellen anhand der virtuellen Verbindungen betrachten wir noch genauer.

ATM-Netze realisieren unterschiedliche Kategorien der Dienstgüte (quality of service, QoS). Dafür wählt der Benutzer des Netzes beim Aufbau einer Verbindung eine Kategorie. Um die Dienstgüte-Garantien einzuhalten, muss das Netz einerseits Betriebsmittel für den Nutzer reservieren, andererseits muss auch das Benutzerverhalten kontrolliert werden.[5.12] Das Netz verwendet dafür sowohl deterministische als auch stochastische Mechanismen.

5.7.1 Virtuelle Kanäle

ATM vermittelt 53 Byte lange Zellen. Davon entfallen 5 Byte auf den Zellenkopf (header). Es ist offensichtlich, dass es ineffizient wäre, die Routing-Entscheidungen für jede Zelle einzeln zu treffen. Abgesehen davon wäre der Paketkopf in der Regel zu klein, um die vollständige Zieladresse zu beinhalten. Daher werden die *virtuellen Verbindungen* auch für die Wegeermittlung genutzt.

5.12. Auch der Benutzer muss sich an bestimmte Regeln halten (wie z.B. eine benutzerspezifische Begrenzung der generierten Maximallast).

Beim Aufbau einer virtuellen Verbindung wird deshalb der Weg für sämtliche Zellen dieser Verbindung festgelegt und in den Vermittlungsrechnern gespeichert. Dafür werden die virtuellen Verbindungen nummeriert, so dass die Zellen ausschließlich eine Nummer ihrer Verbindung – den *Virtual Channel Identifier (VCI)* – beinhalten müssen, um von den Vermittlungsrechnern korrekt weitergeleitet zu werden. Die genaue Funktionsweise hierfür beschreiben wir am folgenden Beispiel.

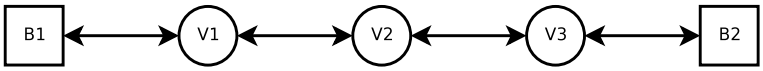


Abbildung 5.20. Verbindung über drei Vermittlungsrechner

Beispiel 5.8. Der Benutzer B_1 eines ATM-Dienstes möchte eine Verbindung mit Benutzer B_2 herstellen. Für diese Verbindung wählt er die Nummer 12 (da vom selbem Rechner schon elf andere Verbindungen in Benutzung sind). Der erste Vermittlungsrechner V_1 entscheidet, dass die Zellen über V_2 geleitet werden und trägt die Zeile $(B_1, 12, V_2)$ in seine Tabelle ein.

Als nächstes entscheidet V_2, dass er Zellen von V_1 an B_2 über V_3 weiterleitet und trägt das Paar $(V_1, 12, V_3)$ in seine (teilweise gefüllte) Tabelle ein. Hier tritt folgendes Problem ein: Zwar kann V_2 zwischen Zellen der beiden virtuellen Kanäle mit der Nummer 12 unterscheiden, da sie von unterschiedlichen Vermittlungsrechnern eingehen. V_3 könnte diese Unterscheidung aber nicht vornehmen, da alle Zellen von V_2 bei ihm eintreffen. Daher nimmt V_2 eine Umbenennung des virtuellen Kanals von 12 nach 13 vor. V_3 ist direkt mit B_2 verbunden, kann also die Zellen direkt zum Ziel leiten. Sämtliche Zellen, die von B_1 nach B_2 gesendet werden sollen, können jetzt sehr einfach anhand der aufgestellten Tabellen geleitet werden. Lediglich in V_2 muss die Nummer der virtuellen Verbindung für den Verkehr $V_1 \to V_2 \to V_3$ im Kopf aller Zellen geändert werden.

V_1

FROM	VCI	TO
B_1	12	V_2

V_2

FROM	VCI	TO
V_9	4	V_6
V_7	6	V_{11}
V_4	12	V_3
V_1	$12 \to 13$	V_3

V_3

FROM	VCI	TO
V_3	13	B_2

Tabelle 5.5. Routing-Tabellen der Vermittlungsrecher aus Beispiel 5.8

5.7.2 Virtuelle Pfade

Backbone-Netze besitzen oft hierarchische Topologien, bei denen einige Vermittlungsrechner eine sehr große Last bewältigen müssen. So ist es denkbar, dass in einer Stadt der gesamte Verkehr vom Osten in den Westen von zwei Vermittlungsrechnern und einer sehr leistungsfähigen Leitung übernommen wird.

Obwohl die Zellenvermittlung, nachdem eine Verbindung aufgebaut wurde, sehr einfach erfolgen kann, würden diese Vermittlungsrechner eine sehr große Zahl von Verbindungen organisieren müssen. Die Routing-Tabellen würden entsprechend wachsen, wodurch die Effizienz sinken würde. Aus diesem Grund sieht ATM einen weiteren Mechanismus vor, um die Vermittlung der Zellen weiter zu vereinfachen, die *virtuellen Pfade*. Hierfür werden die Vermittlungsrechner in zwei Kategorien unterteilt:

VC-Switch. Vermittlungsrechner die niedrig in der Hierarchie stehen und eine kleine Zahl von Verbindungen besitzen. Sie verhalten sich genau, wie im obigen Beispiel beschrieben.

VP-Switch. Vermittlungsrechner die hoch in der Hierarchie stehen und eine große Zahl von Verbindungen realisieren. Sie besitzen nicht für jede Verbindung einen Eintrag in der Routing-Tabelle, sondern leiten die Pakete nur anhand einer gröberen Richtungsangabe weiter, dem *Virtual Path Identifier (VPI)*. Hierzu betrachten wir noch ein Beispiel.

Beispiel 5.9. Wieder baut B_1 eine Verbindung zu B_2 auf. Der erste Vermittlungsrechner VC_1 ist ein VC-Switch mit wenig Last im selben Stadtteil wie B_1. Dieser hält VP_1, den Hamburger VP-Switch, für einen geeigneten Nachfolger. Dieser erkennt, dass sich B_2 in Süddeutschland befindet. Da der gesamte Verkehr von Hamburg nach Süddeutschland über VP_2 in Frankfurt geht, reicht er die Zelle an VP_2 weiter und wählt den virtuellen Pfad Nummer 4, der nach München führt.[5.13]

Der Frankfurter VP-Switch wird nun deutlich entlastet. Er besitzt für alle Verbindungen von Hamburg nach München nur einen Eintrag in seiner Routing-Tabelle: den virtuellen Pfad 4. Erst beim Müchener VP_3 endet der Pfad. Hier werden die virtuellen Verbindungen wieder einzeln betrachtet.

Stellingen: VC_1

FROM	VCI	TO
B_1	12	VP_1

Hamburg: VP_1

FROM	VCI	VPI	TO
VC_1	12	4	VP_2

Frankfurt: VP_2

FROM	VCI	VPI	TO
VP_1	alle	4	VP_3

Tabelle 5.6. Routing-Tabellen der Vermittlungsrecher

Zusammenfassend betrachten wir noch einmal anhand von Abbildung 5.21, wie die Zellen einer Verbindung von den unterschiedlichen Vermittlungsrechnern weitergeleitet werden. Dabei achten wir besonders darauf, wie der VPI und der VCI im Kopf der Zelle verändert werden. Der VCI wird auf dem Weg zum Ziel mehrmals umbenannt (12, 13 und 27). Die Zelle wird auf zwei virtuelle Pfade (4 und 6) geleitet. Die Vermittlungsrechner VP_2, VP_3, VP_4 und VP_6 leiten die Zellen ausschließlich anhand ihres VPI weiter, der VCI wird von ihnen weder betrachtet noch verändert.

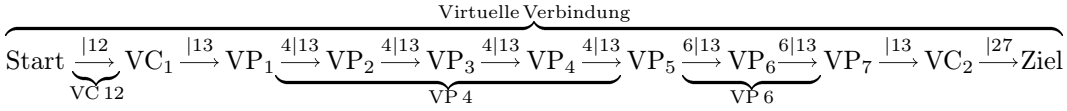


Abbildung 5.21. Der Weg einer ATM-Zelle von Sender zum Empfänger

Wir betrachten nun den vollständigen Aufbau einer ATM-Zelle. Sie beginnt mit einem 4 Bit großen Feld GFC[5.14] (generic flow control), gefolgt von dem uns bekannten 8 Bit langen VPI und dem 16 Bit langen VCI. Das Feld PT (payload type) gibt an, um welchen Typ von Zelle es sich handelt, während das Flag CLP (cell loss priority) zwischen hoher und niedriger Priorität beim Zellenverlust unterscheidet. Im letzten Byte, HEC (header error control), wird eine CRC-Prüfsumme des Headers gespeichert. Eine Fehlerkontrolle für die Nutzdaten (payload) wird durch ATM nicht geleistet – dies ist somit Aufgabe höherer Schichten der Protokollhierarchie.

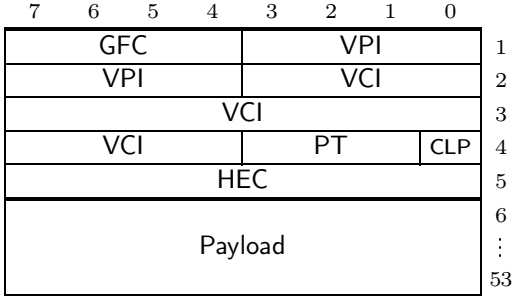


Abbildung 5.22. Aufbau einer ATM-Zelle (mit UNI-Format)

5.13. Virtuelle Pfade werden relativ statisch für ATM-Netze definiert.

5.14. Dieses Feld ist für eine lokale Flusskontrolle zwischen Netz und Benutzer reserviert, jedoch bis heute undefiniert.

5.7.3 Verkehrsvertrag

Wie schon erwähnt unterscheidet ATM unterschiedliche Verkehrstypen und bietet unterschiedliche Dienstgütegarantien. Der Verkehrstyp wird beim Aufbau der Verbindung festgelegt. Sowohl das Netz als auch der Benutzer geben dabei gewisse Garantien über ihr Verhalten ab. Dafür hat sich der Begriff Verkehrsvertrag (traffic contract) etabliert. Die unterschiedlichen Typen sind:

Unspecified Bit Rate (UBR). Dies ist der Standardtyp für ATM-Verbindungen. Eine geforderte oder gebotene Datenrate wird nicht vereinbart. Diesen Verbindungen wird soviel Datenrate gewährt, wie nach Abwicklung der anderen Typen übrig bleibt. Ein Beispiel für geeigneten Verkehr ist ein großer Dateitransfer, der im Hintergrund läuft.

Constant Bit Rate (CBR). Bei diesem Typ wird eine feste Datenrate zwischen Benutzer und Netz vereinbart. Der Benutzer darf diese nicht überschreiten, das Netz darf sie nicht unterschreiten. Ein typischer Einsatzbereich ist der durch Zeitmultiplexen mehrerer Telefongespräche zustande kommende T1-Link (vgl. Abschnitt 2.9.1).

Available Bit Rate (ABR). Die Datenrate wird anhand der aktuellen Auslastung des Netzes festgelegt. Beispielsweise eignet sich dieser Typ gut für das Browsen im World Wide Web.

Variable Bit Rate (VBR). Bei diesem Typ wird eine durchschnittliche Datenrate vom Benutzer verlangt. Er verpflichtet sich, diese Datenrate im Mittel nicht zu überschreiten und die Abweichungen davon in Grenzen zu halten. Es wird unterschieden zwischen Echtzeit- und Nicht-Echtzeitverkehr (real time bzw. non real time). Für Echtzeitverkehr wird die Übertragungsverzögerung durch das Netz gering gehalten. Als Beispiel dient hier eine Videoübertragung. Handelt es sich um eine Videokonferenz, so ist RT-VBR sinnvoll. Wird ein Spielfilm zum Betrachten an einen Benutzer gesendet, ist eine Verzögerung im Bereich von Sekunden meist unproblematisch. Hier würde sich NRT-VBR anbieten.

Charakteristik	UBR	CBR	ABR	RT-VBR	NRT-VBR
garantierte Datenrate	nein	ja	optional	ja	ja
geeignet für Echtzeitverkehr	nein	ja	nein	ja	nein
geeignet für stark schwankende Last	ja	nein	ja	nein	bedingt
Informationen über die Netzauslastung	nein	nein	ja	nein	nein

Tabelle 5.7. Verkehrstypen bei ATM

5.8 Das globale Internet

In Abschnitt 5.5 haben wir über die Interkonnektion von Netzen gesprochen. Die prominenteste Form einer solchen Interkonnektion stellt das globale Internet dar. Das Internet entstand in den 1960er Jahren aus dem vom US-amerikanischen Verteidigungsministerium entwickelten ARPAnet, einem dezentralen Netz zur Übertragung von Daten. Zunächst wurden Universitäten untereinander verbunden. Erst in den 1990er Jahren entstand durch das wachsende Angebot des World Wide Web das große Interesse in der Bevölkerung.

Das Internet stellt ein Netzwerk aus heterogenen Rechnernetzen dar (mit dem in Abschnitt 5.5 genannten Schwierigkeiten). Rechner im Internet sind entweder Arbeitsrechner (hosts) oder Vermittlungsrechner wie Router, Gateway-Rechner und Bridges. Als Zwischenarchitektur bei der Verbindung der Teilnetze wird die Internet-Protokollhierarchie mit den wichtigen Protokollen TCP und IP verwendet. Die Kopplung findet in der Regel auf der Vermittlungsschicht statt, da Datagrammdienste einfacher zu koppeln sind als verbindungsorientierte Dienste.

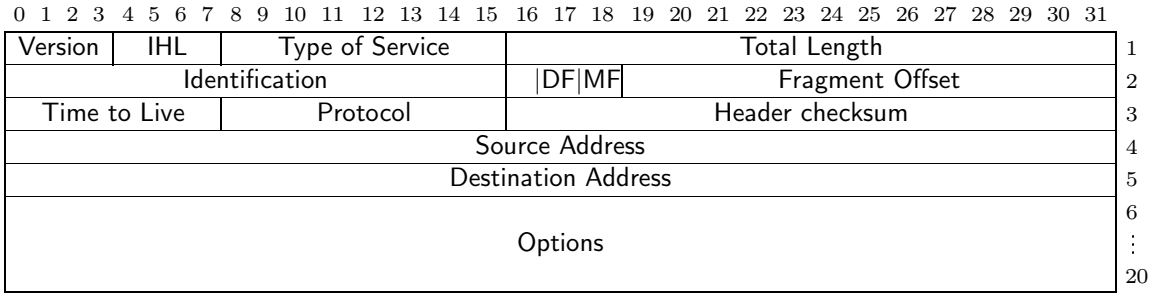


Abbildung 5.23. Aufbau des Paketkopfes beim IP-Datagramm

Bezeichnung	Bit	Erklärung
Version	4	Versionsnummer des Protokolls
IP Header Length	4	Länge des Paketkopfes (in 32-Bit-Worten)
Type of Service	8	Verkehrsart (wird i.d.R. nicht verwendet)
Total Length	16	Gesamtlänge des Paketes (bis 64 kByte)
Identification	16	Sequenznummer des Pakets
DF	1	keine weitere Fragmentierung erlaubt (don't fragment)
MF	1	es folgen weitere Fragmente (more fragments)
Fragment Offset	13	Position des Fragments im ursprünglichen Paket
Time to Live	8	Lebensdauer des Pakets in hops
Protocol	8	verwendetes Transportprotokoll (z.B. TCP oder UDP)
Header Checksum	16	CRC-Prüfsumme des Paketkopfes
Source Address	32	IP-Adresse des Absenders
Destination Address	32	IP-Adresse des Empfängers
Options	0-480	weitere Optionen (werden selten verwendet)

Tabelle 5.8. Legende zu Abbildung 5.23

Der "best effort"-Charakter des Internet

Wir rufen uns noch einmal die Variationen der Vermittlungtechniken ins Gedächtnis. Diese reichen von einer sehr statischen Reservierung der Betriebsmittel und daher oft schlechten Auslastung wie bei Standleitungen oder ISDN über Mittelwege wie ATM und X.25 bis hin zu sehr flexiblen, häufig effizienteren datagrammorientierten Techniken. Je flexibler die Betriebsmittelverwaltung durchgeführt wird, desto weniger Garantien können bezüglich der Dienstgüte gegeben werden. Im traditionellen Internet trägt dieser Charakter auch den Namen "best effort", etwa "nach bestem Bestreben". Dieser Mangel an Garantien ist für Echtzeitkommunikation (siehe Kapitel 7) meist inakzeptabel. Hierfür fordert man Netzqualitäten wie

- limitierte Paketverlustrate
- limitierte Verzögerungsschwankungen (delay jitter)
- garantierter Mindestdurchsatz[5.15]

Um diese Merkmale auch im Internet zu realisieren gibt es unterschiedliche Herangehensweisen. Diese unterscheiden sich in ihrer Komplexität und den Kosten.

- schlichte Überdimensionierung ("dumb fat pipe")
- Reservierung von Ressourcen (z.B. ISA und RSVP)
- Priorisierung des Verkehrs (z.B. DiffServ)

5.15. auch für kurze Beoobachtungsintervalle

Der "black box"-Charakter des Internet

Die Beobachtbarkeit des Internet ist sehr eingeschränkt. Dadurch sind Verhaltensprognosen und Reaktionen auf den gegenwärtigen Netzzustand schwierig oder unmöglich. Allerdings stellt sich die Frage, ob Mechanismen, denen der gesamte Netzzustand bekannt ist, nicht ohnehin unbeherrschbar (da zu komplex) wären. Zudem bietet der "black box"-Charakter teilweise ein gewisses Maß an Sicherheit. Es folgen einige Gründe für die eingeschränkte Beobachtbarkeit des Internet.

- Das Internet wird von zahlreichen Organisationen betrieben. Diese schränken die Beobachtbarkeit ihrer Netze bewusst aus Gründen des Konkurrenzkampfes und wegen Sicherheitsanforderungen ein.
- Die Zahl der Benutzer und Endsysteme im Internet ist enorm hoch. Eine Datenerfassung würde zu einer riesigen, und schwer zu verwaltenden, Datenmenge führen.
- Das Verhalten der Benutzer ist sehr dynamisch und kaum vorhersehbar.
- Eng kontrollierte Randbedingungen sind für Messungen und Experimente normalerweise kaum herzustellen.

Dennoch gibt es gewisse Möglichkeiten, um Teile des Netzzustandes zu erfassen. So sind Lastmessungen an anwendungsnahen Schnittstellen relativ unproblematisch. Auch einzelne Verbindungen (z.B. TCP-Verbindungen) können gemessen werden. Zuletzt können Informationen über das Netz- und Benutzerverhalten durch langfristiges Messen eines Endgerätes gewonnen werden.

Sicherheit im Internet

Das Internet bietet von sich aus quasi keine Mechanismen zum Datenschutz. Auch in vielen anwendungsorientierten Diensten fehlt es an Konzepten, die die Vertraulichkeit und Integrität der Daten gewährleisten. Wir wollen nur einige der vielen Angriffsmöglichkeiten nennen.

- Die *Modifikation der Absenderadresse* in IP-Paketen ist in der Regel einfach möglich. Internet Service Provider (ISPs) prüfen die Absenderadressen der Pakete ihrer Kunden in der Regel nicht. Hierdurch ist das Vortäuschen eines falschen Absenders sehr einfach.
- Viele anwendungsorientierte Dienste wie Telnet, FTP, POP u.a. übertragen Passwörter unverschlüsselt. Dadurch ist es sehr leicht möglich, *Passwörter abzuhören* und später unbefugt zu benutzen.
- Die Verfügbarkeit eines Dienstes im Internet zu gewährleisten, kann sehr schwierig sein. So kann ein Dienst durch gezieltes Verbrauchen der Ressourcen durch einen (und besonders durch mehrere) Benutzer schnell außer Betrieb gesetzt werden (*denial of service*). Ein Schutz hierfür ist auf der Basis der Internet-Protokollhierarchie schwer zu realisieren.
- Bei datagrammorientierten Diensten findet häufig keine Bestätigung für den korrekten Erhalt eines Paketes statt. Auf dem Weg zum Ziel böswillig *veränderte oder gelöschte Pakete* fallen den Kommunikationspartnern möglicherweise nicht auf.

Die Liste der Angriffsmöglichkeiten im Internet ließe sich noch weiter führen. Um die Sicherheit im Internet zu erhöhen, werden unterschiedliche Schutzmechanismen eingesetzt. Diese basieren häufig auf symmetrischen und asymmetrischen Verschlüsselungsverfahren, die die Integrität und Vertraulichkeit der Daten in der Anwendungsschicht realisieren.

Firewall. Als Firewall bezeichnet man ein Gerät oder Programm, das den Netzverkehr meist auf Vermittlungsschicht anhand bestimmter Regeln filtert. Unter anderem können so bestimmte Kommunikationsbeziehungen ausgeschlossen werden. Auch können verdächtige Datagramme (ping of death) von einer Firewall gelöscht werden.

Verschlüsselung. Um vertrauliche Daten über ein ungeschütztes Netz zu senden, wird Verschlüsselung eingesetzt. Es gibt eine Vielzahl von Algorithmen zur Verschlüsselung, Signierung und zum geheimen Schlüsselaustausch. Viele Protokolle der Anwendungsschicht nutzen Verschlüsselung (z.B. HTTPS, SSH).

Tunnel. Eine besonders einfache Möglichkeit, um zwei Kommunikationspartner (oder auch zwei Netze) über ein unsicheres Netz zu verbinden, stellen Tunnel dar. Hier werden die Datagramme durch Gateways auf der Vermittlungsschicht ver- und entschlüsselt. Für die Endgeräte geschieht dies völlig transparent.

Das Internet entwickelt sich momentan sehr schnell weiter. Die Zahl der Benutzer steigt exponentiell (Anfang 2000 betrug sie etwa 100 Mio.). Wir nennen nur einige der aktuellen Trends im Bereich Internet:

- Optische Kommunikationsinfrastrukturen und optische Vermittlungsrechner versprechen deutlich höhere Datenraten.
- Mobilkommunikation über WLAN, GPRS und UMTS mit den IP-Protokoll wird häufig verwendet. Die Vergabe von IP-Adressen an mobile Geräte ist allerdings wegen der hierarchischen Adressierung problematisch.
- Echtzeitanwendungen wie Internet-Telefonie (Voice over IP, VoIP) und Videokonferenzen, werden durch steigende Datenraten zunehmend interessant.
- Die Version 6 des Internet-Protokolls (IPv6) löst die Version 4 langsam ab. Die Version 6 bietet riesige Adressfelder (128 Bit), mehr Sicherheit und bessere Dienstgüte.
- Multicast-Backbones werden eingesetzt, um das Rundsenden von Multimedia-Daten effizienter zu gestalten. Hierfür werden innerhalb der Subnetze Multicast-Router eingesetzt, die den Multicast-Verkehr über einen Tunnel erhalten und innerhalb ihres Subnetzes verbreiten. Der resultierende Datenverkehr ist so deutlich geringer, als wenn der Multicast an alle Stationen einzeln gesendet würde.

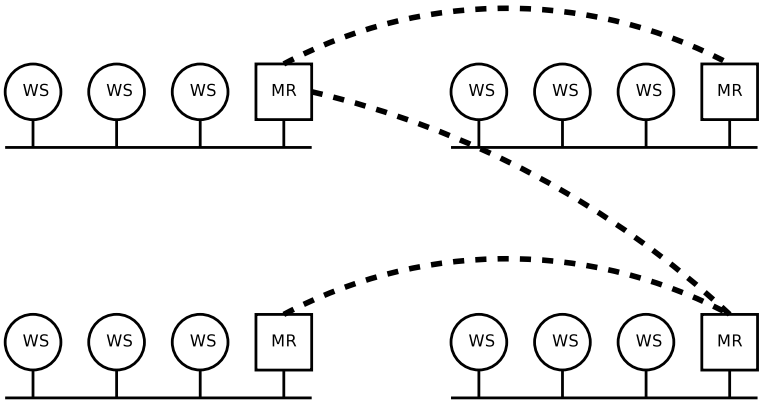


Abbildung 5.24. Mulitcast-Backbone mit Workstations (WS) und Multicast-Routern (MR)

Kapitel 6
Drahtlose Datenübertragung und Mobilkommunikation

Einen besonders stark wachsenden Bereich der Datenkommunikation stellen die drahtlose Datenübertragung und die Mobilkommunikation dar. So gibt es zur Zeit enorme Wachstumsraten beim WLAN und der Mobiltelefonie. Die Anzahl der Mobilfunkteilnehmer hat in Deutschland die Anzahl der Festnetzanschlüsse überstiegen. Es folgt eine Liste von Anwendungsgebieten.

- Fernsprechen mit ISDN-Merkmalen
- Zahlungsverkehr über mobile Stationen
- Steuerung von Fahrzeugen (z.B. Taxis, Schiffe, LKW)
- Kommunikation mit Datenbanken
- Austausch von Kurznachrichten
- Messdatenerfassung
- Zugriff auf mobilen WAP-Dienst
- Austausch kleinerer Dateien
- Mobile Computing

6.1 Grundlegende Eigenschaften

Obwohl die Begriffe "drahtlose Übertragung" und "Mobilkommunikation" miteinander zusammenhängen, wollen wir sie zunächst voneinander abgrenzen:

Definition 6.1. *Unter* Mobilkommunikation *verstehen wir Kommunikation, bei der die Kommunikationspartner (in gewissem Maße) ortsunabhängig kommunizieren können. Wir unterscheiden zwischen Benutzer- und Gerätemobilität.* Drahtlose Datenübertragung *bedeutet, dass die Übertragung der Daten nicht leitungsgebunden ist. Ortsungebundenheit ist keine Voraussetzung für drahtlose Datenübertragung, drahtlose Datenübertragung keine Voraussetzung für Mobilkommunikation.*

Jahr	Netz	Beschreibung
1958	A-Netz	Erstes Mobilfunknetz mit Verbindung zum Telefonnetz
1972	B-Netz	Weiterentwicklung des A-Netzes
1981	C-Netz	Mobilfunknetz mit digitaler Signalisierung
1987	GSM	Globaler Standard für digitale Mobilkommunikation
1992	DECT	Europäischer Standard für digitales schnurloses Telefonieren
1993	Modacom	Paketorientierter Datenfunkdienst der Telekom (bis 2002)
1998	Iridium	Weltumspannendes Satellitenkommunikationsnetz
2000	UMTS	Versteigerung der UMTS-Lizenzen

Tabelle 6.1. Wichtige Entwicklungen der Mobilkommunikation (Signalisierung ist hier nicht im Sinne unserer Unterscheidung Daten/Signalisierung/Übertragung gemeint, sondern steht für jenen Teil der Mobilkommunikation, der für den Verbindungsaufbau durchgeführt wird!)

Wir fassen einige wichtige Eigenschaften der drahtlosen und Mobilkommunikation zusammen.

- Bei terrestrischen Systemen erfolgt die Kommunikation meist per Funk über eine Basisstation. Die Basisstationen verfügen häufig über ein leitungsgebundenes Netz (Festnetz).
- Außer bei älteren Netzen wird in der Regel digital übertragen. Die meisten Systeme bieten daher auch Datendienste, wie Dateitransfer und Kurzmitteilungen.
- Die Netzsicherheit ist wegen der typischerweise benutzten Funkübertragung besonders problematisch. Daher werden in der Regel Verschlüsselungsalgorithmen verwendet.
- In öffentlichen Netzen sind die Datenraten pro Teilnehmer meist wegen der limitierten Bandbreite eher gering. Allerdings könnten bei UMTS in Ballungsräumen noch relativ hohe Datenraten erreicht werden.
- Wegen der Funkübertragung muss meist eine störanfällige Übertragung hingenommen werden.

6.2 Lokale Mobilkommunikation

In diesem Abschnitt betrachten wir zwei wichtige Standards für Mobilkommunikation im lokalen Bereich, DECT und WLAN. Lokale Mobilkommunikation ist durch eine relativ geringe Reichweite der Funkverbindungen charakterisiert. Durch die geringe Reichweite kann ein Frequenzbereich in ausreichender Entfernung erneut verwendet werden. Daher erfordert der Betrieb von lokalen Mobilkommunikationseinrichtungen wie schnurlosen Telefonen oder drahtlosen Netzwerken keine Lizenz.

6.2.1 DECT

Als Nachfolger der Standards CT1 und CT2 hat sich für schnurloses Telefonieren in Europa der DECT-Standard (digital enhanced cordless communication[6.1]) durchgesetzt. DECT bietet insbesondere eine verbesserte Abhörsicherheit, da sämtliche Nutzdaten verschlüsselt übertragen werden. Der dafür verwendete Schlüssel wird nie per Funk zwischen Basisstation und Mobilgerät ausgetauscht. Allerdings ist die Verschlüsselung ein optionaler Teil der Spezifikation und wird nicht von allen Geräten realisiert.

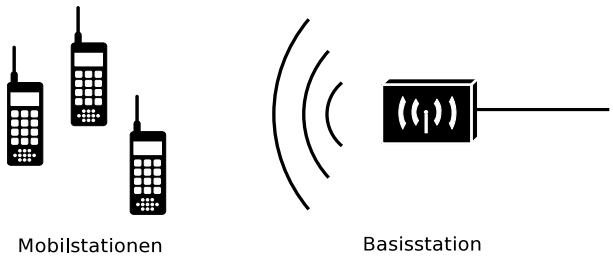


Abbildung 6.1. DECT-Konfiguration mit Basistation und drei Mobilgeräten

Die Bandbreite von 20 MHz wird auf 120 Teilnehmer per Frequenz- und Zeitmultiplexen aufgeteilt. Zur Signalisierung wird Frequenzmodulation verwendet. Dabei werden, je nach Qualität der

6.1. früher: digital european cordless telephone

Funkverbindung, zwei, vier oder acht Signalniveaus verwendet. In einem Schritt werden so also 1, 2 oder 3 Bit übertragen.

Merkmal	Wert
Frequenzbereich	$[1880, 1900]$ MHz
Bandbreite	20 MHz
Teilnehmer	bis 120
Bandbreite pro Teilnehemer	$20\,\text{MHz}/120 \approx 166\,\text{kHz}$
Übertragung	digital
Signalisierung	analog (2-FSK, 4-FSK oder 8-FSK)
Datenrate pro Teilnehmer	32 kbit/s (1152 kbit/s mit Kanalbündelung)
Reichweite	ca. 300 m im Freien, 200 m in Gebäuden
Multiplex-Verfahren	TDMA und FDMA

Tabelle 6.2. Technische Daten von DECT

6.2.2 WLAN

Unter einem WLAN (wireless local area network) versteht man ein drahtloses lokales Rechnernetz. Wegen seiner großen Bedeutung wird WLAN häufig mit dem IEEE-Standard 802.11, den wir in diesem Abschnitt betrachten werden, gleichgesetzt. Auch wir meinen im folgenden, wenn mit von WLAN sprechen, den IEEE-Standard. Auch das Kürzel *Wi-Fi*[6.2] wird häufig mit dem Begriff WLAN oder dem Standard verwechselt.

Zunächst wollen einige typische Anforderungen an WLANs nennen, die bei der Entwicklung der Standards berücksichtigt werden:

- ein möglichst hoher Durchsatz
- eine große Anzahl von Mobilstationen
- die Möglichkeit zur Interkonnektion mit leitungsgebundenen Netzen
- ein Versorgungsbereich in der Größenordnung von 100m bis 300m
- ein geringer Energieverbrauch in den Mobilstationen
- eine zuverlässige und sichere Datenübertragung
- ein lizenzfreier Betrieb
- ein zuverlässiger handoff/handover beim Wechsel in eine andere Funkzelle
- ein dynamisches Konfigurieren neu zu integrierender oder auszugliedernder Mobilstationen.

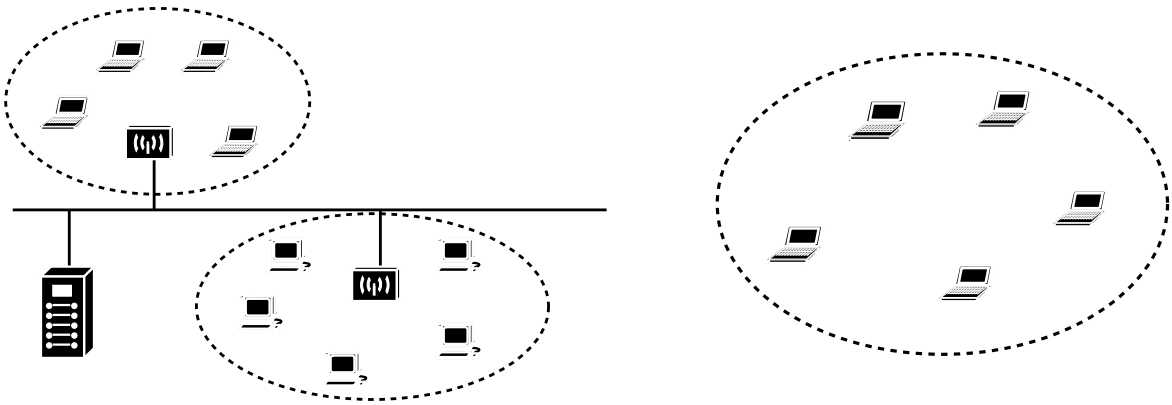
Ein WLAN kann grundsätzlich in zwei Betriebsarten arbeiten.

Infrastructure. Bei dieser Betriebsart kommunizieren die Mobilgeräte über eine Basisstation (access point, portable access unit) miteinander. Die Basisstation ist häufig an ein leitungsgebundenes LAN angeschlossen, so dass das gesamte Netz auch stationäre Geräte enthält.

Ad-Hoc. Bei dieser Betriebsart werden eine gleichberechtigte Menge von Mobilgeräten spontan zu einem LAN zusammengefasst. So können beispielsweise Konferenzen auch ohne

6.2. Wi-Fi ist ein Zertifikat der Wireless Fidelity Alliance, das die Konformität von Geräten mit dem IEEE-Standard und damit die Interoperabilität mit anderen Geräten bescheinigt.

verfügbare Netzinfrastruktur realisiert werden.



a) Infrastruktur b) Ad-Hoc

Abbildung 6.2. Infrstrukturbetrieb und Ad-Hoc-Betrieb

Die Datenraten bei WLAN liegen momentan zwischen 2 Mbit/s und 54 Mbit/s. Höhere Datenraten sind aber schon jetzt absehbar. Der Standard bezieht sich auf die Schichten 1 und 2 des OSI-Referenzmodells. Oberhalb dieser Schichten wird meist die TCP/IP-Protokollhierarchie eingesetzt. Die Zugriffskontrolle CSMA/CA (carrier sense multiple access, collision avoidance) ähnelt der von Ethernet. Wir werden sie später noch genauer betrachten. Ein wichtiger Aspekt des Standards ist das Wechseln eines Mobilgerätes zu einer anderen Basisstation (roaming, hand over). Die Notwendigkeit für einen solchen Wechsel wird meist durch eine Verschlechterung der Übertragungsqualität festgestellt. Es gibt zwei Möglichkeiten für die Auswahl einer neuen Basisstation:

- Nutzung der von den Basisstationen periodisch ausgesendeten Signale (beacon)
- Aktives Aussenden eines Signals (probe) und Abwarten der Antworten der Basisstationen.

Merkmal	Wert
Frequenzbereich	$[2400, 2483]$ MHz oder $[5150, 5725]$ MHz
Bandbreite pro Teilnehmer	$20\,\text{MHz}/120 \approx 166\,\text{kHz}$
Übertragung	digital
Signalisierung	analog, Bandspreizverfahren
Datenrate	2 MBit/s, 11 MBit/s, 54 MBit/s, ...
maximale Sendeleistung	100 mW oder 1000 mW
Reichweite	ca. 150 m im Freien
Multiplex-Verfahren	TDMA

Tabelle 6.3. Technische Daten von IEEE 802.11

Rahmentypen der MAC-Schicht

Auf der MAC-Schicht werden Daten in Form von Rahmen (frames) ausgetauscht. Der Standard unterscheidet dabei zwischen drei Typen deren Aufbau wir später erklären werden.

Kontrollrahmen. Mit diesen Rahmen wird der konkurrierende Zugriff der Stationen auf das gemeinsame Medium organisiert. Sie werden noch im einzelnen erklärt. Abbildung 6.3 zeigt den Einsatz von Kontrollrahmen bei der Übertragung eines Rahmens. Deren Bedeutung ist in Tabelle 6.4 erläutert.

Datenrahmen. In ihnen werden die Nutzdaten von den Stationen übertragen. Auch die Quittierung empfangener Daten kann in diesen Rahmen erfolgen. Es gibt unterschiedliche

Subtypen.

Managementrahmen. Sie dienen zum Informationsaustausch bei Managementaufgaben zwischen Basisstation und Mobilgeräten. Dazu gehören u.a. das oben erwähnte Roaming, der Wechsel zu einer neuen Basisstation.

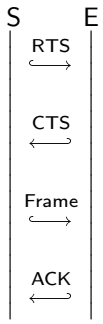


Abbildung 6.3. Zeitlicher Ablauf beim Senden eines Datenrahmens

Abk.	Bedeutung	Erklärung
PS	power save poll	Der Energiesparmodus wurde verlassen
RTS	request to send	Mitteilung an den Empfänger über die Sendeabsicht
CTS	clear to send	Bereitschaft zum Empfang der durch RTS angefragten Sendung
ACK	acknowledgement	Positive Quittung des zuletzt empfangenen Rahmens
CF-END	contention free end	Ende einer konkurrenzlosen Phase
CF-ACK	acknowledgement	Bestätigung für CF-END

Tabelle 6.4. Kontrollrahmen bei IEEE 802.11

Wir betrachten einmal den genauen Aufbau und die Bedeutung der einzelnen Felder der MAC-Frames. Die Rahmen haben einen 240 Bit langen Kopf gefolgt von bis zu 2312 Byte Nutzdaten. Beendet werden die Frames mit einer 32 Bit langen CRC-Prüfsumme, die sich auf den gesamten Rahmen bezieht.

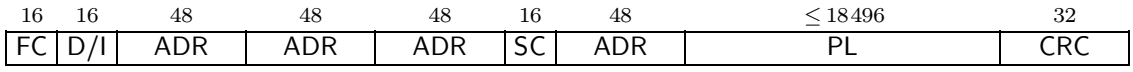


Abbildung 6.4. Aufbau des Rahmens bei IEEE 802.11

Abk.	Bedeutung	Erklärung
FC	frame control	Angabe vom Rahmentyp, Kontrollinformationen
D/I	duration/connection ID	Dauer der Übertragung / virtuelle Verbindung
ADR	address	Adresse von Quelle, Ziel, letztem und nächstem Knoten
SC	sequence control	Sequenz- und Fragmentnummer
PL	payload	Nutzdaten
CRC	cyclic redundancy check	Prüfsumme

Tabelle 6.5. Legende zu Abbildung 6.4

Zugriffskontrolle

Das Grundprinzip der Zugriffskontrolle CSMA/CA erklären wir mit Hilfe des Beispiels in Abbil-

dung 6.5. Bevor eine Station mit einer Sendung beginnt, prüft sie, ob das Medium frei ist. Außerdem gibt es zwei Warteintervalle. Der *interframe space (IFS)* wird vor jeder Sendung gewartet, um Überlappungen mit zuvor gesendeten Rahmen zu vermeiden. Außerdem gibt es (wie bei Ethernet) eine zufällige Wartezeit, die verhindert, dass sendewillige Stationen im selben Moment mit der Sendung beginnen. Da sich diese Wartezeit im Konfliktfall exponentiell verlängert heißt sie *exponential backoff (EB)*. Wir betrachten nun die drei Fälle. Der vollständige Algorithmus ist als Flussdiagramm in Abbildung 6.6 dargestellt.

a) Das Medium ist zum Zeitpunkt t_0 frei. Nachdem die Station den IFS abgewartet hat, prüft sie, ob das Medium noch frei ist. Da das der Fall ist, beginnt sie mit der Sendung.

b) Das Medium ist zum Zeitpunkt t_0 belegt. Da der IFS exakt mit dem Ende der vorherigen Sendung beginnt, verhindert die zufällige Wartezeit EB, dass alle wartenden Stationen gleichzeitig zu senden beginnen. Die betrachtete Station hat entweder einen kurzen EB ermittelt oder wenig Konkurrenz. Das Medium ist nach EB immer noch frei. Die Sendung kann begonnen werden.

c) In diesem Fall ist nach Abwarten von IFS das Medium nicht frei. Die Station wartet diese Sendung ab und verhält sich dann wie in (b).

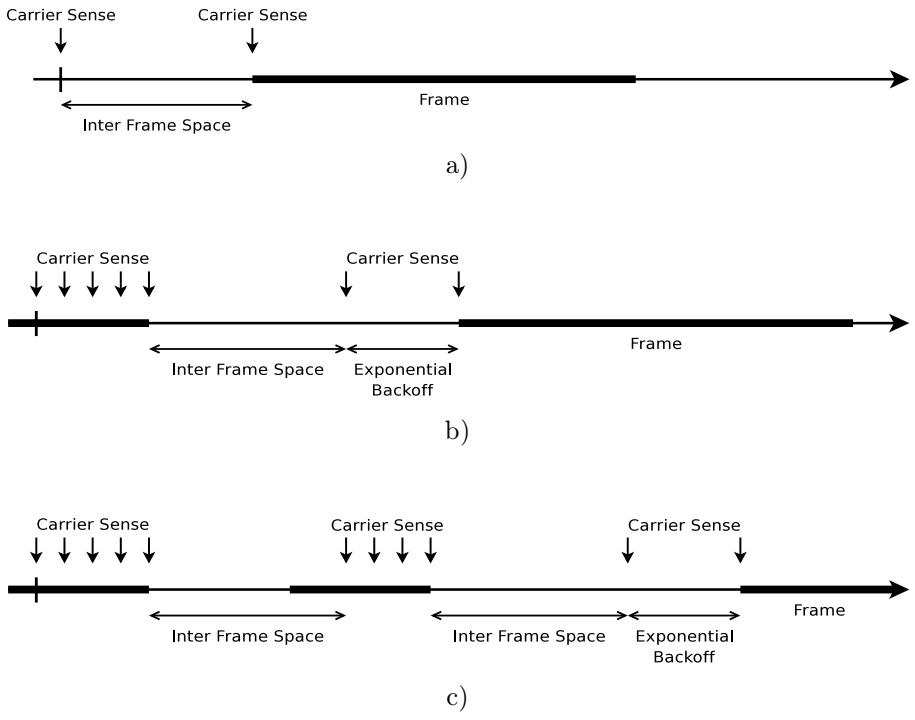


Abbildung 6.5. Beispiele für die zeitlichen Abläufe bei CSMA/CA

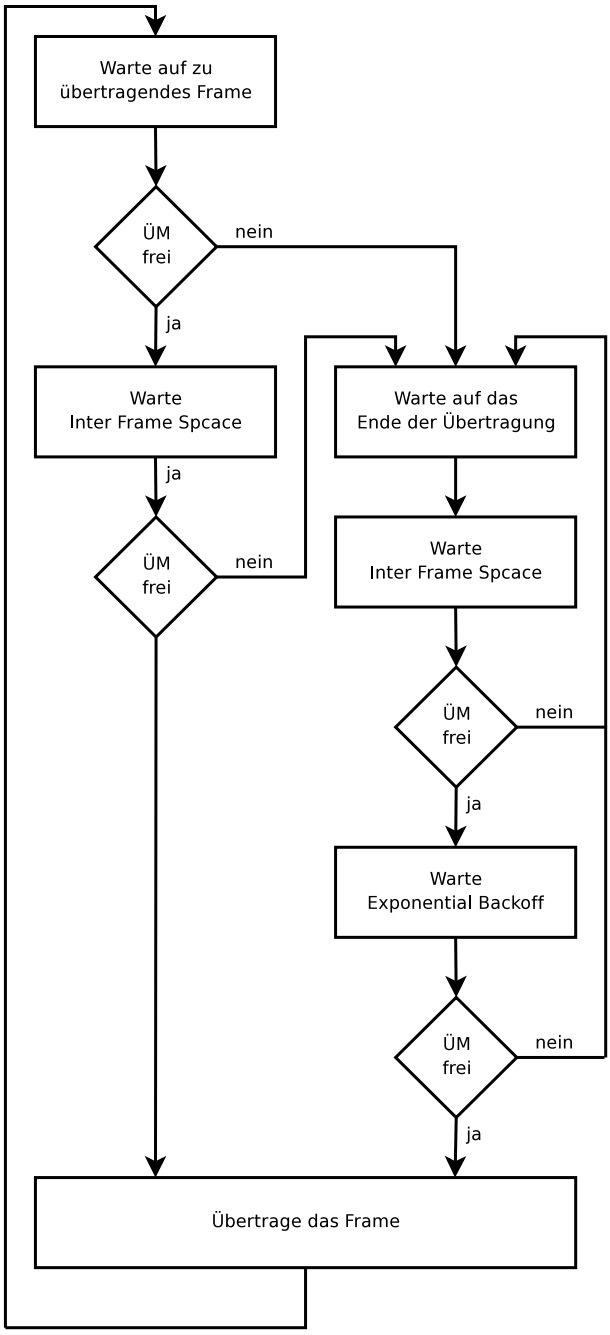


Abbildung 6.6. Flussdiagramm für den CSMA/CA

Die Wartezeit IFS wird außerdem in einfacher Weise zur Priorisierung von Rahmen genutzt, indem höher priorisierte Rahmen einen kürzeren IFS abwarten müssen. Es gibt drei Arten von Wartezeiten.

SIFS. Short Interframe Spacing wird für hoch priorisierte Rahmen wie Kontrollrahmen verwendet. Insbesondere die Quittierung von Sendungen (ACK) wird mit hoher Priorität gesendet.

PIFS. PCF Interframe Spacing hat mittlere Wartezeiten. Es wird insbesondere für Echtzeitkommunikation verwendet. PCF ist ein kollisionsfreies, zentralisiertes Zugriffsverfahren.

DIFS. DCF Interframe Spacing führt zu den längsten Wartezeiten und wird deshalb meist für asynchrone Dienste (keine Echtzeitanforderungen) verwendet. Die Zugriffskontrolle erfolgt

per CSMA/CA.

In der folgenden Tabelle sind die Zeiten für die unterschiedlichen Wartezeiten angegeben. Man erkennt, dass sich die längeren Wartezeiten durch einfache oder doppelte Verlängerung von SIFS um ein Δt ergibt. Die Abkürzungen DSSS und FHSS werden später noch erklärt. In Abbildung 6.7 sind die zeitlichen Abläufe bei der Zugriffsverzögerung durch IFS und EB dargestellt.

	SIFS	PIFS	DIFS	Δt
DSSS	10 μs	30 μs	50 μs	20 μs
FHSS	28 μs	78 μs	128 μs	50 μs

Tabelle 6.6. Unterschiedliche Zeiten für IFS

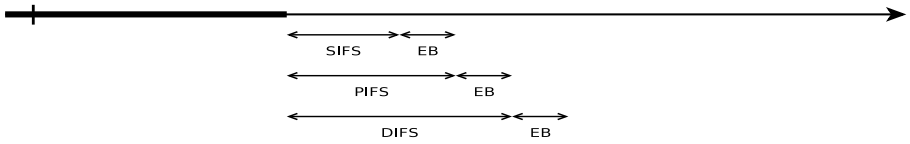


Abbildung 6.7. Wettbewerbsfenster und Zugriffsverzögerung

Verborgene Stationen

Im Gegensatz zu Ethernet heißt der Zugriffskontrollmechanismus bei WLAN "collision avoidance", Kollisionen werden also vermieden und nicht verhindert. Es stellt sich die Frage, warum die Stationen trotz der Fähigkeit, das Medium während der Sendung abzuhören (carrier sense), Kollisionen nicht direkt erkennen können. Der Grund hierfür liegt in der Tatsache, dass die Stationen wegen der begrenzten Reichweite nicht in jedem Fall das gesamte Medium abhören können. So ist es durchaus möglich, dass zwei Stationen A und C gleichzeitig Nachrichten an eine Station B senden, so dass sich diese gegenseitig stören, ohne dass A oder C die Überlagerung erkennen kann. Ein Beispiel hierfür ist in Abbildung 6.8 angegeben.

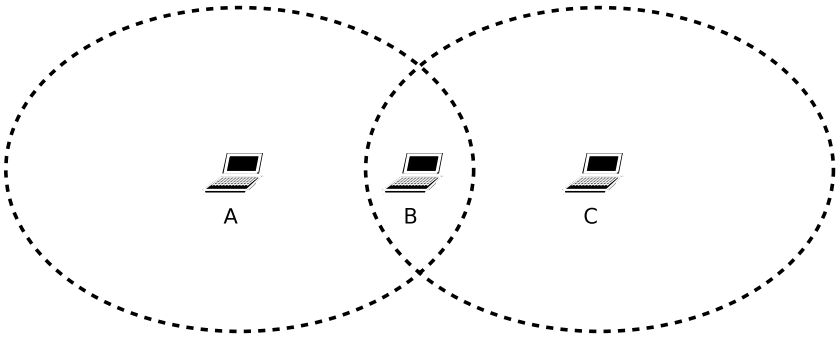


Abbildung 6.8. Beispiel für verborgene Stationen (hidden station)

Bandspreizverfahren

Bei der Übertragung im WLAN werden sogenannte Bandspreizverfahren eingesetzt. Dabei wird ein breiteres Frequenzband verwendet, als für die Übertragung eigentlich nötig wäre. Wir werden sehen, dass sich diese Verfahren positiv auf die Störungsanfälligkeit und Abhörsicherheit auswirken.

Direct Sequence Spread Spectrum (DSSS). Die grundsätzliche Idee besteht darin, jedes Nutzdatenbit durch mehrere Bits darzustellen. Dafür wird eine zufällige Sequenz von Bits (Chipping Sequence) verwendet und per XOR mit den Nutzdaten[6.3] verknüpft. Durch

6.3. Die Nutzdaten werden vorher expandiert: beispielsweise `00010` → `000000000111000` bei einem Spreizfaktor von 3

die so eingefügte Redundanz wird die Störanfälligkeit der Übertragung reduziert. Ist die verwendete Bitsequenz nur dem Sender und dem Empfänger bekannt, wird auch die Abhörsicherheit erhöht. In Abbildung 6.9 wurden die Nutzdaten `01101001` mit der chipping sequence `0110` verknüpft.

Frequency Hopping Spread Spectrum (FHSS). Bei diesem Verfahren wird die Trägerfrequenz der Übertragung regelmäßig geändert. Auch hierfür wird eine zufällige Sequenz von Frequenzen verwendet, die dem Sender und dem Empfänger bekannt ist. In der schnellen Variante werden während der Übertragung eines Bits die Frequenzen gewechselt (beispielsweise dreimal pro Bit), in der langsamen Variante erst nach einigen gesendeten Bits. Auch hier wird eine gewisse Abhörsicherheit erzeugt, wenn die verwendete Sequenz geheim ist. Die Störanfälligkeit wird in zweierlei Hinsicht verbessert. Zum einen treten Störungen häufig nur in einem begrenzten Frequenzband auf. Bei der schnellen Variante wird also nur ein Teil des Bits gestört. Zum anderen werden störende Echos (Reflektionen durch Wände) vom Empfänger weniger wahrgenommen, weil er beim Eintreffen des Echos auf einer anderen Frequenz empfängt.

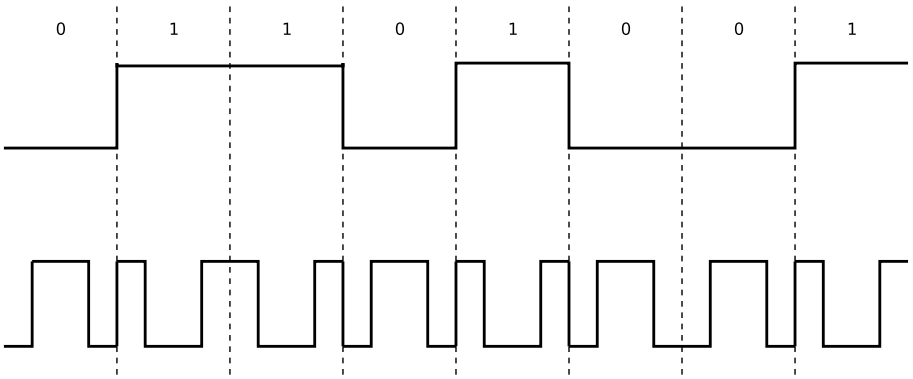


Abbildung 6.9. Beispiel für Direct Sequence Spread Spectrum (Spreizfaktor 4, Chipping Sequence `0110`)

Abbildung 6.10 zeigt Beispiele für FHSS in der schnellen und der langsamen Variante. Wieder werden die Nutzdaten `01101001` codiert. Es werden statt einer Frequenz die drei Frequenzen ν_1, ν_2 und ν_3 verwendet. Zur Codierung wird Frequenzmodulation verwendet. Eine null wird durch eine kleinere Frequenz, eine eins durch eine höhere Frequenz dargestellt.

In der langsamen Variante wird die Sequenz $\nu_3\nu_1\nu_2$... verwendet. Nach jeweils zwei Nutzdatenbits wird die Trägerfrequenz geändert. In der schnellen Variante wird die Sequenz $\nu_3\nu_1\nu_3\nu_2$... verwendet, wobei die Trägerfrequenz während eines Nutzdatenbits viermal gewechselt wird.

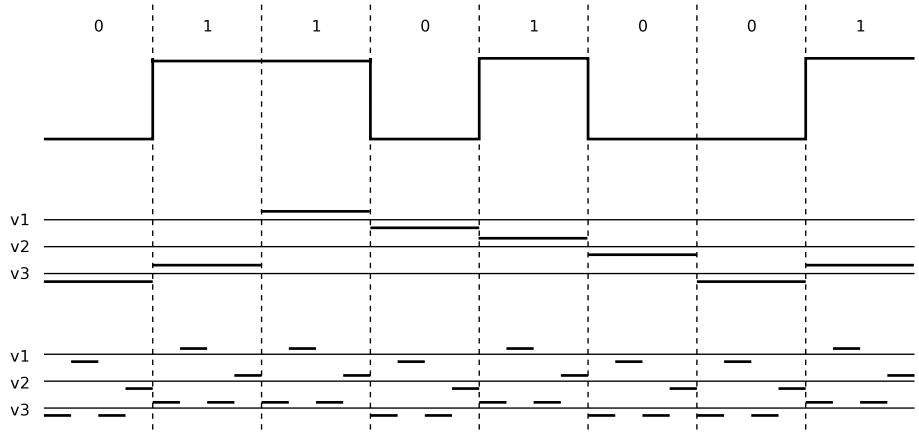


Abbildung 6.10. Beispiele für Frequency Hopping Spread Spectrum

Multi-hop Ad-Hoc Netze

In Ad-Hoc-Netzen kann es vorkommen, dass die Entfernung zwischen zwei Mobilstationen M_1 und M_2 zu groß ist, um eine direkte Datenübertragung durchzuführen. Dennoch ist es denkbar, dass weitere Mobilstationen, die sich zwischen M_1 und M_2 befinden, genutzt werden, um eine indirekte Datenübertragung durchzuführen. Ein Paket würde beispielsweise von M_1 an eine dritte Station, die sich in Reichweite befindet, gesendet und von dieser an M_2 weitergeleitet. Eine indirekte Übertragung über mehrere Zwischenstationen ist ebenfalls denkbar. Diese Zwichenstationen fungieren also nicht nur als Endsysteme sondern auch als Vermittlungsrechner.

Die Topologie eines solchen Netzes ist hochdynamisch. Vermittlungsrechner können plötzlich entstehen oder wegfallen. Zentralisierte Routingverfahren sind daher offensichtlich unbrauchbar. Verwendbar ist ein verteiltes Routing, wie es im Abschnitt 5.3 beschrieben wird. Dabei werden Zustandsinformationen zwischen den Nachbarknoten erster oder zweiter Ordnung ausgetauscht. Diese Informationen können die Netztopologie sowie die Qualität der Funkverbindungen beinhalten. Zu ihrer Bestimmung werden ggf. spezielle Prüfpakete versendet.

6.3 Zellulare Weitverkehrsnetze

Wir haben schon im Abschnitt 2.4.2 über Mobilfunkübertragung angesprochen, dass bei drahtlosen Weitverkehrsnetzen meist eine Aufteilung des geographischen Gebiets in Zellen erfolgt. Wir betrachten nun die Frage nach der idealen Zellengröße. In Abbildung 6.11 wird ein geographischer Bereich durch eine bzw. durch sieben Zellen abgedeckt. Im ersten Fall werden die Frequenzbereiche a, b und c auf dem gesamten Gebiet verwendet. Die Bandbreite für alle Benutzer des Gebiets beträgt $B_a + B_b + B_c$. Im zweiten Fall wird die Bandbreite durch Raummultiplexen für alle Benutzer erhöht. Sie beträgt $3 \cdot B_a + 3 \cdot B_b + B_c$.

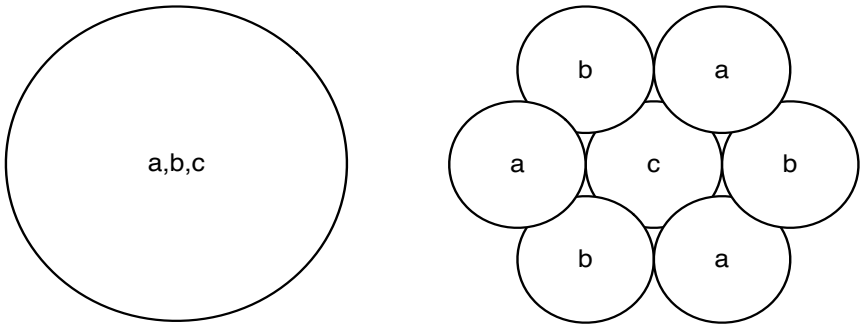


Abbildung 6.11. Große und kleine Zellen bei Mobilfunkübertragung

Je kleiner die Zellen sind, desto mehr Bandbreite kann jedem einzelnen Benutzer zur Verfügung gestellt werden. Allerdings werden auch mehr Basisstationen benötigt und ein Wechsel der Basisstation wird bei (sich bewegenden) Stationen häufiger. Daher steigen bei kleineren Zellen Kosten und Komplexität.

Tabelle 6.7 zeigt einige Merkmale der ersten zellularen Weitverkehrsnetze in Deutschland. Der Betrieb des A-Netzes und des B-Netzes ist mittlerweile eingestellt. Im C-Netz wurde erstmals digitale Signalisierung[6.4] eingesetzt. Im folgenden Abschnitt betrachten wir den wichtigen GSM-Standard.

6.4. Hiermit ist nicht Signalisierung im Sinne unserer Unterscheidung Daten/Signalisierung/Übertragung gemeint, sondern der Teil der Mobilkommunikation, der für den Verbindungsaufbau durchgeführt wird!

	Zeitraum	Frequenzbereich	Datenübertragung	Modulation	Teilnehmer
A-Netz	1958–1977	[156, 174] MHz	analog	FM	ca. 11000
B-Netz	1972–1994	[146, 156] MHz	analog	FM	ca. 26000
C-Netz	seit 1981	[450, 466] MHz	analog/digital	PhM	ca. 770000
GSM	seit 1990	[890, 960] MHz	digital	FM	ca. 160 Mio.*

Tabelle 6.7. Merkmale der ersten zellularen Weitverkehrsnetze in Deutschland *) in 2000, weltweit

6.3.1 GSM

Der Standard GSM (Global System for Mobile Communications) wurde in den späten 1980er Jahren entwickelt, um einen vollständig digitalen Mobilfunkdienst für eine hohe Teilnehmerzahl zu bieten. Der Dienst umfasst sowohl Telefonie als auch unterschiedliche Datendienste (darunter SMS). Sprachqualität und Datenrate sind deutlich höher als im C-Netz. GSM wird in Deutschland als Grundlage für die heutigen D- und E-Netze verwendet.

Systemarchitektur

In Abbildung 6.12 ist die Architektur eines GSM-Netzes schematisch dargestellt. Wir betrachten zunächst die einzelnen Komponenten des Netzes.

Mobile Station (MS). Die Komponenten (Handy, Pager, Notebook-Adapter, etc.), die von den Netzbenutzern zur Kommunikation verwendet werden. Sie kommunizieren per Funk mit einer Basisstation in ihrer Nähe.

Base Station (BS). Basisstationen besitzen einen festen Versorgungsbereich. Sie bedienen eine oder mehrere Funkzellen. Genauer bezeichnet man die Basisstation als Base Station Subsystem (BSS), bestehend aus dem Base Station Controller (BSC) und pro Funkzelle einer Base Transceiver Station (BTS).

Mobile Switching Center (MSC). Eine zentrale Komponente, die die Nachrichtenvermittlung zwischen Mobilstationen und die Erfassung von Abrechnungsdaten realisiert. Dazu verwendet sie unterschiedliche Informationen:

- Im Home Location Register (HLR) sind die Daten der in den entsprechenden Funkzellen "beheimateten" Teilnehmer gespeichert. Das Visitor Location Register (VLR) speichert die Daten der anderen Teilnehmer, die sich in den entsprechenden Zellen befinden.
- Ein Authentication Center (AC oder AUC) und ein Equipment Identification Center (EIC) sind für die Erkennung und Authentifizierung der Mobilstationen zuständig.

Operation and Maintainance Center (OMC). Dies stellt eine zentrale Komponente zur Kontrolle und Wartung des Netzes dar.

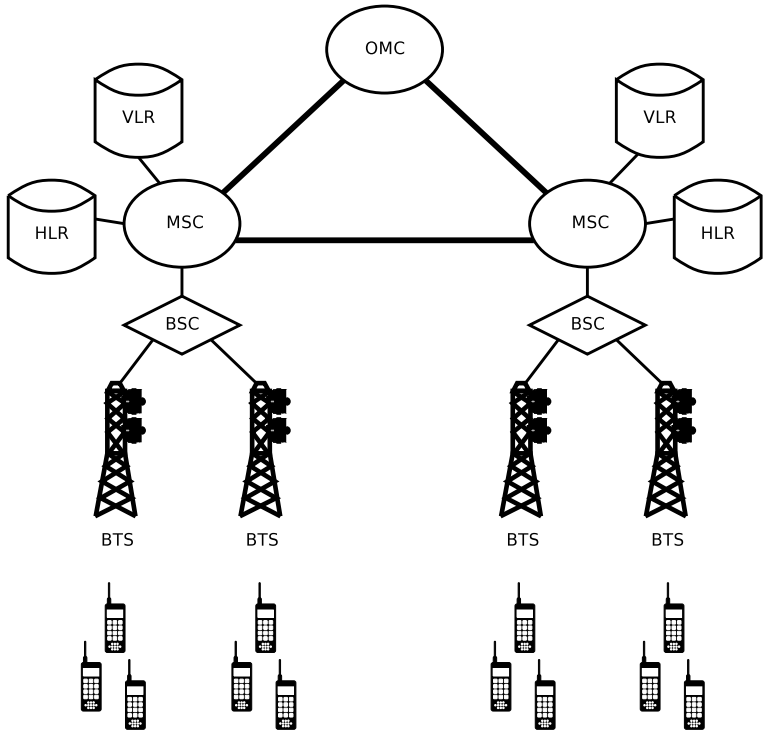


Abbildung 6.12. Schematischer Aufbau eines GSM-Netzes

Neben Raummultiplexen durch die Verwendung von Zellen, wird die verfügbare Bandbreite (siehe Tabelle 6.7) durch Frequenz- und Zeitmultiplexen auf die Benutzer aufgeteilt. Die folgende Abbildung zeigt die hierarchische Struktur der Rahmen bei GSM. Der Frequenzbereich wird in zweimal 124 Kanäle aufgeteilt. Die höheren Frequenzen werden für die Übertragung von Basisstation zur Mobilstation (downlink) verwendet, da sie mehr Energie erfordern. Innerhalb dieser Kanäle werden feste Zeitscheiben verwendet.

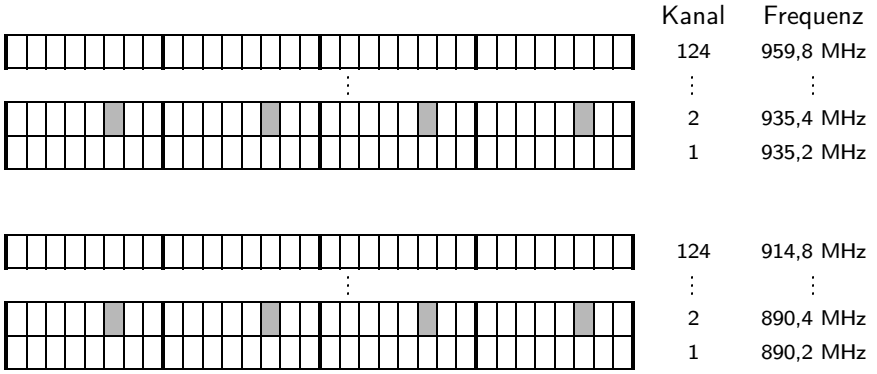


Abbildung 6.13. Einige TDM-Frames auf allen 248 Kanälen. Auf Kanal 2 ist eine Datenübertragung in der jeweils sechsten Zeitscheibe grau dargestellt.

Multiframe. Diese Rahmen haben eine Länge von 32500 Bit und eine Sendedauer von 120ms. Sie werden in 26 TDM-Frames unterteilt. Zwei dieser TDM-Frames sind für Kontrollinformationen bzw. zukünftige Weiterentwicklungen reserviert (siehe Abb. 6.14).

TDM-Frame. Sie bestehen aus 1250 Bit und werden innerhalb von 4, 615 ms übertragen. Sie beinhalten acht Datenrahmen, die jeweils einem Benutzer zugeordnet sind (oder nicht verwendet werden). Zwischen den Datenrahmen liegen Sicherheitszwischenräume (guard times) der Dauer $30\,\mu s$ bzw. $8,25$ Bit. Damit bleiben für jeden Datenrahmen $(1250 - 8 \cdot 8,$ $25)/8$=148 Bit (siehe Abb. 6.15).

Datenframe. Der Aufbau der Datenframes ist in Abbildung 6.16 dargestellt. Während der Dauer seiner Übertragung von $547\,\mu s$ werden 114 Bit Nutzdaten übertragen und 34 Bit Zusatzinformationen übertragen.

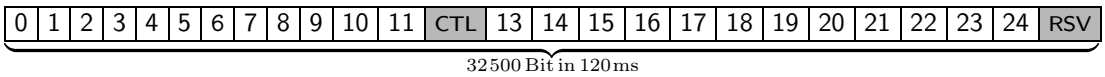


Abbildung 6.14. Multiframe aus 26 TDM-Frames, darunter ein Kontroll-Frame (CTL) und ein für andere Zwecke reservierter Frame (RSV)

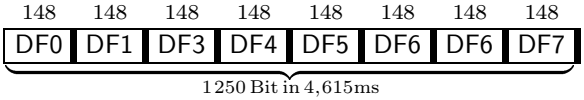


Abbildung 6.15. TDM-Frame mit acht Zeitscheiben und $30\,\mu$s Sicherheitszwischenräumen (guard times)

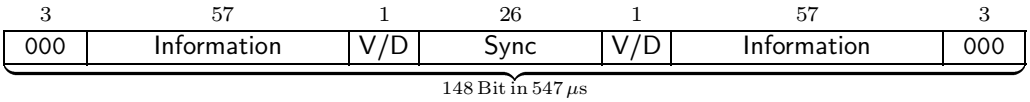


Abbildung 6.16. Datenframe mit Felderm für Nutzdaten, Blocksynchronisation und Sprach-Daten-Flag (V/D)

In der folgenden Tabelle errechnen wir aus den uns bisher bekannten Werten die Netto-Datenrate, die einem Benutzer pro Link (uplink oder downlink) zur Verfügung steht. Dazu teilen wir die Datenrate pro Kanal durch 8 (wegen Zeitmultiplexen), ziehen den Verlust, der durch die guard times und den Overhead des Datenrahmen entsteht, ab. Wir halbieren den Wert, um nur den uplink bzw. downlink zu erhalten. Es bleibt eine Brutto-Datenrate von ca. 23 kBit/s. Der Nettowert für Datenübertragungen liegt deutlich niedriger bei $9,6$ kBit/s.

maximale Datenrate	Rechnung	Wert
pro Kanal	$2 \cdot 1250$ Bit $/4,615$ ms	~ 541712 Bit/s
Kontrollslots	$/\ 24/26$	~ 500040 Bit/s
pro Benutzer	$/8$	~ 62505 Bit/s
abzüglich guard time	$\cdot 547\,\mu s/(547\,\mu s + 30\,\mu s)$	~ 59255 Bit/s
abzüglich Overhead	$\cdot 114$ Bit/148 Bit	~ 45643 Bit/s
pro Link (uplink/downlink)	$/2$	~ 22821 Bit/s
abzüglich weiteren Overheads		$\sim 9,6$ kBit/s

Tabelle 6.8. Berechnung der Netto-Datenrate eines GSM-Kanals

Lokalisierung von Teilnehmern

Wir betrachten den zeitlichen Ablauf beim Anruf eines Mobilfunkteilnehmers aus dem Festnetz. Zur Interkonnektion von Mobilnetz und Festnetz werden spezielle Gateway-MSCs (GMSC) eingesetzt, die mit dem Festnetz verbunden sind.

1. Der Anruf aus dem Festnetz wird an ein nahegelegenes GMSC des entsprechenden Mobilfunknetzes weitergeleitet.
2. Das GMSC erkennt das für die gewählte Nummer zuständige HLR.
3. Das GMSC kontaktiert das HLR und erfährt das aktuelle VLR des Teilnehmers.
4. Das GMSC signalisiert dem VLR über das entsprechende MSC den Verbindungswunsch.
5. Das VLR charakterisiert aus Aufwandsgründen und in Anbetracht seiner eventuell bereits veralteten Zustandsinformation den Aufenthaltsort der Mobilstation nur grob. Deshalb werden durch das MSC alle ihm zugeordneten Basisstationen aufgefordert, die Mobilstation zu suchen (paging).

6. Sofern die Mobilstation das Suchsignal empfängt, meldet es sich beim MSC zurück und veranlasst einen Klingelton.

Während des Gesprächs kann es zu einem Wechsel in eine andere Funkzelle kommen, wenn der Teilnehmer sich bewegt. Ist in dieser Funkzelle noch ein freier Funkkanal verfügbar, wird die Verbindung über ihn weitergeführt. Andernfalls wird die Verbindung abgebrochen. Das Wechseln der Funkzelle, zieht einen Wechsel des zuständigen BTS nach sich. Das kann gleichzeitig einen Wechsel des zuständigen BSS oder gar des zuständigen MSC bedeuten.

Kommunikationsdienste

Der GSM-Standard unterscheidet die folgenden Dienstgruppen.

Trägerdienste. Als Grundlage für höherwertige Dienste dienen die Trägerdienste Sprachübertragung und Datenübertragung. Die Sprachdaten werden hierbei mit einem Codec auf 13 kBit/s komprimiert. Durch Hinzufügen von Redundanz zur Fehlerkorrektur entsteht eine Datenrate von $22,4\,\text{kBit/s}$. Der Datenübertragungsdienst bietet zugunsten geringerer Fehlerwahrscheinlichkeiten eine niedrigere Datenrate von $9,6\,\text{kBit/s}$.

Telekommunikationsdienste. Auf der Basis der Trägerdienste werden Sprachdienste wie Fernsprechen, Notruf, Voice Mail und Nicht-Sprachdienste wie Telefax, SMS und Datentransfer angeboten.

Zusatzdienste. Weitere Dienste, die teilweise auch im ISDN geboten werden, sind die Rufnummernidentifikation, Anrufweiterleitung, automatischer Rückruf, Gebührenanzeige, Konferenzgespräche u.a.

Mehrwertdienste. Auf der Basis der bisher genannten Dienste werden eine Vielzahl von Mehrwertdiensten, wie Flug- und Hotelreservierung, Stauvorhersagen, Börsennachrichten, Pannenhilfe und Klingeltonabonnements angeboten.

Darüber hinaus existiert ein neuer mobiler WWW-Dienst. Mobilgeräte, die das WAP-Protokoll (wireless application protocol) beherrschen und über einen einfachen Browser verfügen, können (ähnlich wie beim normalen WWW-Dienst über HTTP) Informationen in Form von Text und Bildern von speziellen WAP-Servern laden und darstellen.

Sicherheit

Um die Gefahr des Missbrauchs fremder Mobilgeräte zu reduzieren, besitzt jede Mobilstation eine Chipkarte (Subscriber Identity Module, SIM), die zur Benutzung aller Dienste (ausgenommen der Notruffunktion) erforderlich ist. Diese Chipkarte wird mit einem Passwort (Persönliche Identifikationsnummer, PIN) vor fremdem Gebrauch geschützt. Auf der Seite der Basisstation existiert das oben angesprochene Authentification Center (AC). Darüber hinaus werden die Daten verschlüsselt übertragen. Außerdem existiert ein Verfahren, welches das Orten eines Mobilfunkteilnehmers unterbindet.

6.3.2 Bündelfunk

Bündelfunk ist ein Mobilkommunikationssystem zum Austausch meist kurzer Nachrichten im Nahbereich. Er wird in der Regel von geschlossenen Benutzergruppen wie Taxiunternehmen, Rettungs- und Pannendiensten eingesetzt. In der Vergangenheit war es üblich, dass solche Organisationen ein eigenes Betriebsfunksystem benutzten und auf ihnen fest zugewiesenen Kanälen sendeten. Beim Bündelfunk wird das Funksystem meist von einer einzelnen Firma betrieben. Die Kanäle werden von den Nutzern gemietet und werden ihnen dynamisch zugewiesen. So wird in der Regel eine bessere Auslastung und Verfügbarkeit erreicht.

Auch beim Bündelfunk werden die Dienste in Gruppen unterteilt. Es existieren Trägerdienste zur Datenübertragung mit Datenraten zwischen $1,2\,\text{kBit/s}$ und $28,8\,\text{kBit/s}$. Darauf basierend werden Telekommunikationsdienste wie Einzel-, Gruppen-, Direkt- und Notruf, Rufumleitung, Konferenzschaltung, Telefax usw. realisiert.

Wir betonen noch einmal, dass Bündelfunk für betriebliche Anwendung von geschlossenen Benutzergruppen eingesetzt wird. Lediglich unidirektionale Verbindungen zum öffentlichen Telefonnetz sind möglich.

Standards Als wichtige Standards sind MPT 1327 (Ministry of Post, Großbitannien) und TETRA (Terrestrial Trunked Radio, ETSI) zu nennen. Ersterer nutzt analoge Übertragung im $[410, 430]$ MHz Bereich. Die Datenrate für Datenkommunikation ist mit maximal 1,2 kBit/s recht niedrig. Beim TETRA-Standard ist die Übertragung vollduplex, digital und abhörsicher. Die Datenraten können $28,8$ kBit/s erreichen.

Netzarchitektur Abbildung 6.17 stellt die Netzarchitektur beim Bündelfunk schematisch dar. Die Komponenten OMC und MSC sind mit der GSM-Architektur vergleichbar, der TSC (Trunked Site Controller) ähnelt in seiner Funktion dem BSC. Man beachte allerdings, dass die Abkürzung MSC hier für "Master Systems Controller" steht.

Lizenzen Der Betrieb eines Bündelfunksystems erfordert, je nach geographischer Ausdehnung, unterschiedliche Lizenzen. Diese Klassen sind in der folgenden Tabelle aufgeführt.

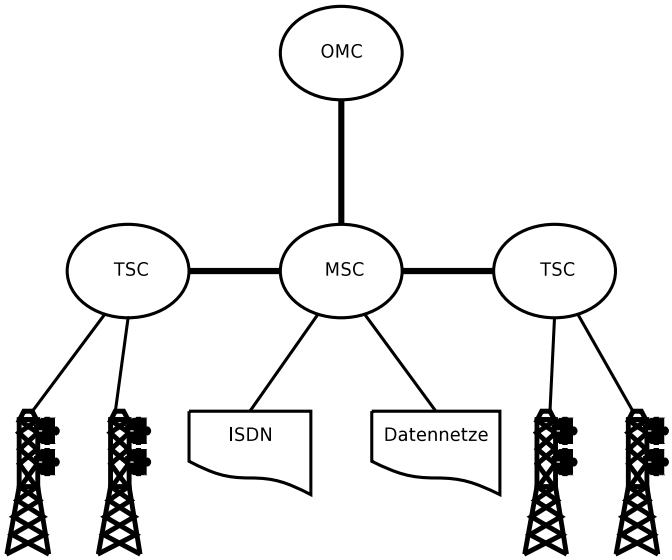


Abbildung 6.17. Architektur von Bündelfunknetzen

Klasse	Einsatzgebiete
A	Regionen mit hoher Nachfrage (z.B. Großstädte)
B	für kleinere Gebiete (evtl. mit Überlappungen)
C	für private Grundstücke
D	für das gesamte Bundesgebiet

Tabelle 6.9. Lizenzklassen beim Bündelfunk in Deutschland

6.3.3 Paging

Eine sehr einfache Form der Mobilkommunikation stellt Paging[6.5] (Funkruf) dar. Es handelt sich hierbei um eine unidirektionale Übertragung kurzer Nachrichten an die Mobilgeräte (Pager, Meldeempfänger). Die Nachrichten werden dabei üblicherweise aus dem öffentlichen Telefonnetz gesendet. Der aktuelle Aufenthaltsort des Teilnehmers ist dem Kommunikationssystem nicht bekannt. Diese Technik besitzt gegenüber der mobilen Sprachkommunikation zwei Vorteile: Zum einen sind die Kosten für die Endgeräte, die über keinen Sender verfügen, gering. Zum anderen

6.5. von engl. to page: jemanden ausrufen

besteht auch Erreichbarkeit an Orten, an denen aktive Endgeräte verboten sind. Man unterscheidet zwischen unterschiedlichen Rufklassen.

Ton. Es wird ausschließlich ein Ton vom Endgerät ausgegeben, welcher den Benutzer über einen eingegangenen Ruf informiert. Diese Klasse eignet sich, wenn nur ein Absender in Frage kommt, der daraufhin telefonisch kontaktiert werden kann. Typische Beispiele hierfür sind Rettungsdienste mit Mitarbeitern, die Bereitschaftsdienst leisten.

Numerik. Die Nachrichten bestehen ausschließlich aus Dezimalziffern. Sie beinhalten in der Regel die Telefonnummer des Absenders, damit der Empfänger sich bei diesem melden kann.

Alphanumerik. Die Nachrichten bestehen aus Zahlen und Buchstaben. So ist die Übertragung kurzer Texte (ähnlich wie beim SMS-Dienst von GSM) möglich.

Da der Aufenthaltsort des Teilnehmers unbekannt ist, muss die Nachricht – anders als bei zellularen Netzen – in einem relativ großen Bereich gesendet werden. Da die Nachrichten sehr kurz sind, hält sich der Aufwand hierfür trotzdem in Grenzen. Die Erreichbarkeit erstreckt sich bei den meisten Paging-Diensten aber nicht auf das gesamte Bundesgebiet. Beim Dienst "Scall" besteht Erreichbarkeit innerhalb von 25 km um das PLZ-Gebiet des Kunden, bei Skyper wählt der Teilnehmer eine von 16 Rufzonen in Deutschland.

öffentlich	nicht öffentlich
Cityruf	Polizei
Scall	Notdienste
Skyper	Hilfsdienste
Quix	Feuerwehr
Telmi	Krankenhäuser
Ermes	

Tabelle 6.10. Benutzergruppen und Dienste beim Paging

6.3.4 Exkurs: Zukünftige Techniken

Man kann wohl davon ausgehen, dass sich in Zukunft ein neuer Standard für Mobilfunknetze mit weit höheren Datenraten als GSM weltweit etablieren wird. In Frage kommen hierfür unter anderem UMTS (Universal Mobile Telecommunications Standard) als existierender europäischer Vorschlag sowie IMT-2000 als zukünftiger ITU-Standard.

Im Gegensatz zu GSM, der das $[890, 960]$ MHz-Band verwendet, nutzt UMTS den Frequenzbereich [1900, 2200] MHz. Die Bandbreite ist damit weit höher und den Benutzern können Datenraten zwischen 2 MBit/s in Gebäuden und 384 kBit/s flächendeckend geboten werden. Sie genügen also für hochwertige Multimediakommunikation oder schnelle Internetzugänge.

Merkmal	Wert
Frequenzbereich	$[1900, 2200]$ MHz
Übertragung	digital
Datenrate	384 kBit/s bis 2 MBit/s
Verfügbarkeit	seit 2004 partiell verfügbar

Tabelle 6.11. Technische Daten von UMTS

Ein weiterer Standard, der für breitbandige Mobilfunknetze im MAN-Bereich an Bedeutung gewinnen kann, ist WiMAX (vgl. [MaF07]). Dabei handelt es sich genau genommen um ein konkretes Profil der IEEE 802.16-Standardfamilie. Entwurfsziele von WiMAX waren eine maximal erreichbare Datenrate im Bereich von 134 Mb/s (für die Summe sämtlicher an eine Basisstation angeschlossenen Teilnehmer) bei einer Reichweite im Bereich mehrerer Kilometer. Insbesonderee sollen mit WiMAX schnell und kostengünstig temporäre Netzinstallationen (etwa bei Sportgroß-

veranstaltungen, Messen, Konzerten, etc.) möglich sein. Innerhalb der Frequenzbereiche [10, 66] GHz (Bereich der Richtfunkfrequenzen) oder [23, 38] GHz (in Deutschland) werden Bandbreiten von 20, 25 oder 28 MHz verwendet.

Der Standard definiert die Schichten 1 und 2 des OSI-Modells. WiMAX kann damit als Basis für IP oder ATM eingesetzt werden. Der Standard sieht desweiteren Dienstgütefelder und unterschiedliche mögliche Topologien vor. Anfang 2006 waren weltweit schon mehr als 150 Pilotnetze im Einsatz.

6.4 Satellitenkommunikation

Auf die Nutzung von Satellitenverbindungen in Kommunikationssystemen sind wir schon in Abschnitt 2.4.1 eingegangen. Wie schon erwähnt, klassifizieren wir Nachrichtensatelliten nach deren Höhe über der Erde.

Geostationary Earth Orbiter (GEO). Sie umkreisen die Erde in ca. 36000 km Höhe. In dieser Distanz beträgt ihre Umlaufzeit exakt einen Tag, so dass sie synchron mit der Erdrotation laufen. Liegt ihre Umlaufbahn über dem Äquator, ändert sich ihre Position am Himmel nicht. Parabolantennen auf der Erde müssen dann nur einmal ausgerichtet werden.

Ein Beispiel hierfür ist Inmarsat (International Maritime Satellite Organisation), das seit 1982 mit neun geostationären Satelliten nahezu die gesamte Erdoberfläche abdeckt und Mobilkommunikaiton für Schiffe, Flugzeuge, Expeditionen u.a. ermöglicht.

Medium Earth Orbiter (MEO). Ihre Höhe liegt zwischen 10000 km und 15000 km. Wichtigster Vertreter dieser Klasse ist momentan das GPS (Global Positioning System), das eine auf wenige Meter genaue Positionsbestimmung erlaubt. Errechnet wird die Position aus den Signallaufzeiten, der von den Satelliten abgesendeten Zeitsignalen. Insgesamt umlaufen 24 Satelliten die Erde, so dass normalerweise die Signale von mindestens drei Satelliten empfangen werden können.

Low Earth Orbiter (LEO). Sie umrunden die Erde in einer Höhe von 700 km bis 1500 km. Unter Verwendung von 66 dieser Satelliten wurde 1998 das Mobilfunknetz IRIDIUM für flächendeckende Mobilkommunikation (im Gegensatz zu GSM) angeboten. Der Betrieb wurde aber schon 2000 wegen geringer Nachfrage vorerst eingestellt.

Bewertungskriterien und Resümee

Obwohl bei Mobilfunknetzen generell ein Frequenzengpass (auch beim Frequenzmultiplexen) besteht, erleben sie momentan ein enormes Wachstum. Die zukünftige Bedeutung der Mobilfunknetze wird vermutlich noch weiter steigen. Zumindest legen das die Prognosen für wichtige Unternehmen wie Nokia und die Kaufpreise der UMTS-Lizenzen nahe. Wir beenden dieses Kapitel mit einer Zusammenstellung der wichtigsten Kriterien zur Bewertung von Mobilfunknetzen.

Kriterium	Aspekte
Kosten (Betreiber)	erstmaliger Aufbau, Reparatur, Wartung
Kosten (Nutzer)	Mobilstation, Kosten für Verbindungen, Grundgebühr
Sicherheit	Fehlerfreiheit, Abhörsicherheit, Authentifikation
Verfügbarkeit	zu verschiedenen Zeitpunkten, an unterschiedlichen Orten
Qualität	Datenrate, Fehlerrate, i.a. besser bei digitaler Übertragung
Sendeleistung	Reichweite, Energiebedarf, Gesundheitsschädlichkeit
Interkonnektivität	Internet, öffentliches Telefonnetz, Mobilfunknetze

Tabelle 6.12. Kriterien zur Bewertung von Mobilfunknetzen

Kapitel 7
Medien- und Echtzeitkommunikation

Durch die steigenden Datenraten im Bereich Datenkommunikation gewinnt (auch bei der Mobilkommunikation) die Übertragung von Sprache, Fest- und Bewegtbildern hoher Qualität an Bedeutung. Dennoch können viele Rechnernetze (insbesondere das globale Internet) die Qualitätsanforderungen der Benutzer oft nicht hinreichend erfüllen. Wir betrachten in diesem Kapitel die Aspekte der Medien- und Echtzeitkommunikation und die Maßnahmen, die zur Verbesserung der Qualität verwendet werden können.

7.1 Motivation und Grundbegriffe

Ähnlich wie im vorangegangenen Kapitel die Begriffe Mobilkommunikation und drahtlose Dateübertragung unterschieden wurden, wollen wir auch hier zunächst zwei eng zusammenhängende Begriffe voneinander abgrenzen.

Definition 7.1. *Unter* Medienkommunikation *und* Multimediakommunikation *verstehen wir den Austausch unterschiedlicher Informationsarten (wie Daten, Festbilder, Sprache, Audio-/Videoströme) mittels eines Kommunikationssystems dergestalt, dass Qualitätsanforderungen der Benutzer berücksichtigt werden. Von Multimedia sprechen wir, wenn mehr als eine Informationsart ausgetauscht wird.*[7.1]

Definition 7.2. *Unter* Echtzeitkommunikation *verstehen wir Kommunikation über ein technisches Kommunikationssystem, bei der Echtzeitbedingungen berücksichtigt werden. Insbesondere besteht das Ziel, die vom Benutzer gestellten Anforderungen an die Übertragungsverzögerung zu erfüllen.*

Echtzeitbedingungen betreffen in erster Linie Durchsatz, Verlustraten bzw. Fehlerwahrscheinlichkeiten (unter Echtzeitbedingungen kommt eine erneute Übertragung meist nicht in Frage), Verzögerungszeit (delay) und Verzögerungsschwankung (delay jitter). Müssen diese Bedingungen quasi immer erfüllt werden, sprechen wir von *harten* Echzeitbedingungen. Genügt eine Erfüllung mit hinreichend großer Wahrscheinlichkeit (beispielsweise paketbasierte Sprachübertragung), von *weichen* Echtzeitbedingungen.

Wenn ein Kommunikationssystem Echtzeitbedingungen (wie z.B. eine maximale Verzögerungszeit) erfüllt, müssen in der Regel auch die Kommunikationspartner bestimmte Bedingungen erfüllen (etwa eine maximale Datenrate bei der Generierung zu übertragender Daten nicht zu überschreiten). Daher ist eine Kontrolle der auftragsgenerierenden Systeme unabdingbar.

Definition 7.3. *Es folgen die Definitionen der ISO (a) und der ITU-T (b) für Dienstgüte (quality of service)*

a) Dienstgüte *stellt eine Menge von Güten dar, die auf das gesamte Verhalten eines oder mehrerer Objekte bezogen sind.*

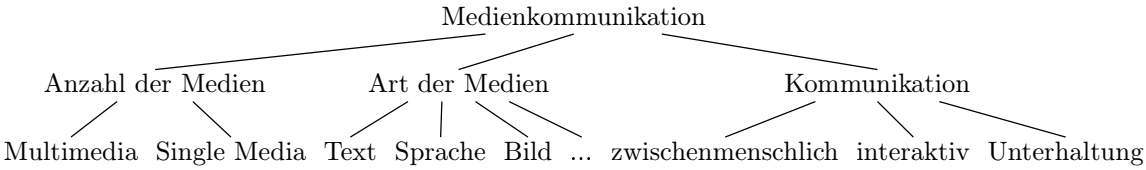
7.1. Insbesondere außerhalb der Informatik sind zahlreiche andere Definitionen möglich und üblich.

b) Dienstgüte *ist definiert als der kollektive Effekt der Dienstleistungen, die den Grad der Zufriedenheit des Benutzers eines Dienstes bestimmen.*

Die Maßnahmen zur Realisierung der Dienstgüte werden oft als *QoS-Management* (Dienstgüteverwaltung) bezeichnet.

7.2 Anwendungen

In der folgenden Tabelle sind einige Anwendungsbeispiele mit Bedarf an Medien- und Echtzeitkommunikation aufgelistet. Zur Klassifikation können die Anzahl und Art der verwendeten Medien, sowie die Kommunikationsart herangezogen werden.



	verwendete Medien	Anwendung
a)	Sprache	Telefonie, Voice-Mail, Telefonkonferenzen
	Festbild	Fax
	Text	E-Mail, Chat
	Text, Festbilder	Computer Supported Cooperative Work (CSCW)
	Sprache, Video	Videotelefonie, Video-Mail, Videokonferenz
	Text, Bilder, Audio, Video	Multimedia E-Mail, Gruppenvideospiele
b)	Audio, Video	Interaktives Fernsehen
	Text	Börsenticker, Newsticker
	Text, Bilder, Audio, Video	Browsen des WWW
c)	Audio, Video	Audio- und Video-on-Demand, Fernsehen
	Audio	Rundfunk

Tabelle 7.1. Anwendungen der Echtzeitkommunikation. a) Kommunikation zwischen Personen b) Interaktive Anwendungen über das Internet c) Unterhaltungsdienste

7.3 Anforderungen bei verteilten Anwendungen

Die Anforderungen, die von Endbenutzern an Medienkommunikationssysteme gestellt werden, sind sehr unterschiedlich. Unterschiedliche *Verkehrsklassen* (Daten, Sprache, Festbild, etc.) besitzen typische Güteanforderungen. In der folgenden Tabelle sind (stark verallgemeinert) einige grobe Werte angegeben. Der *Anwendungstyp* spielt ebenso eine wichtige Rolle: Beispielsweise müssen für medizinische Anwendungen andere Anforderungen erfüllt werden als für Videospiele.

Klasse	Durchsatz	Verzögerung	Verzögerungsschwankung	Zuverlässigkeit
Daten	stark variabel	bis 10 s	wenig relevant	sehr hoch
Sprache	12 kBit/s bis 64 kBit/s	bis 500 ms	gering	eher niedrig
Festbild	kurzfristig hoch	bis 10 s	wenig relevant	eher hoch
A/V	32 kBit/s bis 100 MBit/s	bis 1 s	gering	eher niedrig

Tabelle 7.2. Grobe Größen für Dienstgüteanforderungen in Abhängigkeit von der Verkehrsklasse

Weitere Anforderungen, die häufig an Medienkommunikation gestellt werden, sind:

Synchronisation. Bei Multimediakommunikation ist es häufig notwendig, die einzelnen Medienströme zu synchronisieren. So muss die Synchronisation der Audio- und Videoströme bei Videokonferenzen, Bildtelefonen oder Spielfilmen sehr genau sein. Schon bei einer Zehntelsekunde Unterschied kann der Eindruck der Lippensynchronität verloren gehen.

Multicast-Funktionalität. Bei Gruppenkommunikation ist es wünschenswert, dass der Datenstrom vom Sender nicht mehrfach verschickt werden muss, sondern erst an einem (netztopologisch) sinnvollen Knoten. Diese Funktionalität ist im Internet grundsätzlich nicht gegeben.

Die Definitionen am Anfang dieses Kapitels haben Dienstgüte als subjektiven Effekt definiert. Ein Beispiel hierfür wäre der Wunsch nach einer hochauflösenden lippensynchronen Audio/Video-Übertragung. Es ist nicht leicht, die quantitativen Anforderungen an ein Kommunikationssystem zu spezifizieren, die einen solchen subjektiven Effekt hervorrufen können. Im Abschnitt 7.5 betrachten wir dieses Problem genauer.

7.4 Dienstgüte aus Netzsicht

Der Betreiber eines Netzes betrachtet Dienstgüte meist mit einer eher technischen Perspektive. Er bemüht sich, die Parameter, die für Dienstgüte relevant sind, in den gewünschten Bereichen zu halten. Möglicherweise macht er gewisse Garantien über die Parameter:

- Verzögerungszeit
- Verzögerungsschwankung
- Durchsatz
- Verlustraten

Es stellen sich dabei aber drei wichtige Fragen. Zunächst muss klar sein, ob ein *Maximum, ein Minimum, ein Mittelwert* o.ä. garantiert wird. Darüber hinaus muss die *Bedeutung* des Begriffs "Garantie" festgelegt werden: Bedeutet Garantie eine sehr hohe Wahrscheinlichkeit? Berechtigt sie zu Schadensersatz, wenn sie nicht eingehalten werden kann? Die wahrscheinlich wichtigste Frage ist die nach dem *Beobachtungsintervall*. Dazu betrachten wir das folgende Beispiel.

Beispiel 7.4. Ein Netzbetreiber garantiert eine minimale Datenrate von 10 MBit/s. Konkret garantiert er, in jeder Minute mindestens 600 MBit = 75 MB zu übertragen. Er realisiert diese Garantie, indem er während der ersten sechs Sekunden einer Minute mit 100 MBit/s überträgt. Die restlichen 54 Sekunden werden keine Daten übertragen.

Ein Benutzer, der eine 300 MB große Datei in vier Minuten übertragen möchte würde zufrieden sein, wenn der Betreiber seine Garantie einhält. Ein Benutzer, der eine Videokonferenz durchführen möchte, wäre wahrscheinlich nicht zufrieden. Obwohl die garantierte Datenrate für recht hohe Bild- und Tonqualität ausreicht, müsste der A/V-Strom beim Sender um mindestens 54 Sekunden verzögert werden. Eine Kommunikation über einen solchen Kanal wäre kaum möglich, da die Echtzeitanforderungen an eine Videokonferenz nicht erfüllbar wären.

Es gibt drei allgemeine Ansätze, die QoS-Garantien ermöglichen. Man beachte, dass gegenwärtig viele Netze (z.B. Internet, Ethernet, FDDI) keine QoS-Garantien bieten.

Überdimensionierung. Das Kommunikationssystem wird permanent mit recht niedriger Auslastung betrieben. Obwohl dieser (auch als "dumb fat pipe" bezeichnete) Ansatz auf den ersten Blick etwas unbeholfen wirkt, wird er doch häufig eingesetzt. Die Kosten für eine Überdimensionierung sind oft niedriger als die für den Einsatz komplexerer Vermittlungsrechner mit Priorisierung oder Betriebsmittelverwaltung.

Priorisierung. Dieser zweite Ansatz besteht darin, den Verkehr unterschiedlich zu priorisieren. Echtzeitverkehr könnte in einem Vermittlungsrechner schneller weitergeleitet werden als ein Dateitransfer. Dies wird in der Regel durch mehrere Warteschlangen im Vermittlungsrechner realisiert. Ein Vermittlungsrechner, dessen Speicher zur Neige geht, könnte Dateneinheiten von Computerspielen als erstes löschen. Es ist offensichtlich, dass das Benutzerverhalten kontrolliert werden muss, damit kein zu hohes Verkehrsaufkommen auf den höheren Prioritätsebenen entsteht.

Betriebsmittelreservierung. Hierbei werden Ressourcen des Kommunikationssystems reserviert. Sind die für einen Dienst erforderlichen Ressourcen schon allesamt für andere Verbindungen reserviert, muss ein Benutzer des Dienstes warten. Ein typischer Zeitpunkt für die Reservierung ist der Verbindungsaufbau (häufig bei Mobilfunknetzen). Bei einer gemieteten Standleitung wird die Ressource für einen längeren Zeitraum (z.B. Tage, Monate) reserviert. Die Auslastung der Ressourcen kann bei diesem Ansatz recht niedrig sein.

7.5 QoS-Mapping

Wir haben schon angesprochen, dass eine Abbildung von benutzerseitigen Dienstanforderungen auf netzseitig zu erfüllende Dienstgütemerkmale schwierig, aber notwendig ist. Diese Abbildung wird auch als *QoS-Mapping* bezeichnet. Mit folgendem Beispiel soll gezeigt werden, dass die Abbildung in der Regel auch nicht eindeutig ist.

Aus Benutzersicht wird ein garantierter minimaler Nutzdatendurchsatz von D_g [kbit/s] in jedem Zeitintervall i gefordert. Diese Forderung kann auf unterschiedliche Weisen erbracht werden.

1. Netzseitig wird ein Durchsatz von $D_g + D_k$ gewährleistet, wobei D_k den für die Diensterbringung nötigen Kontrollinformationen entspricht und die Fehler- bzw. Verlustraten vernachlässigbar sind.

2. Netzseitig wird in Durchsatz von $(D_g + D_k) \cdot (1 + x)$ unter Verwendung von Vorwärtsfehlerkorrektur bei einer Paketverlustrate ϵ gewährleistet, wobei $x > 0$. Der Wert $x = f(\varepsilon)$ stellt den für die Vorwärtsfehlerkorrektur benötigten zusätzlichen Durchsatz bei der Verlustrate ε dar.

7.6 Implikationen von Echtzeitanforderungen für kommunizierende Endsysteme

Auch wenn ein Netz die Dienste mit einer adäquaten Qualität erbringt, folgt daraus nicht unbedingt, dass die Qualität aus Sicht des Benutzers angemessen ist. Auch das Endsystem des Benutzers kann in vielerlei Hinsicht zum Engpass werden. Für Echtzeitkommunikation sind daher auch echtzeitfähige Betriebssysteme notwendig. Diese bieten in der Regel folgende Merkmale.

- Priorisierung von Prozessen
- Unterbrechbarkeit von Prozessen
- geeignete Scheduling-Algorithmen

Da diese Algorithmen eher Bestandteil einer Lehrveranstaltung über Betriebssysteme sind, wollen wir nur zwei von ihnen kurz beschreiben.

Rate Monotonic Scheduling

Ein recht einfacher, statischer Algorithmus ist *Rate Monotonic Scheduling (RMS)* für voneinander unabhängige und unterbrechbare Prozesse p_i mit periodisch auftretenden Aufträgen t_i. Die Aufträge sollten spätestens beim Ende ihrer Periode erledigt sein.

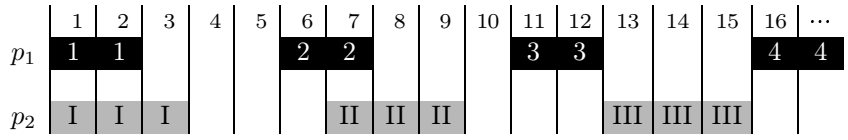


Abbildung 7.1. Beispiel für zwei Prozesse mit unterschiedlichen Perioden

Die Prozesse werden im voraus fest priorisiert und es wird immer der Auftrag mit der höchsten Priorisierung ausgeführt. Entsteht während der Bearbeitung eines Auftrages ein neuer mit höherer Priorität, so wird die Bearbeitung unterbrochen und zunächst der hoch priorisierte Auftrag erledigt. Die folgende Abbildung zeigt das Ergebnis mit unterschiedlichen Priorisierungen. Anders als in diesem Fall kann es unter Umständen von der Priorisierung abhängen, ob jede *Deadline* eingehalten wird.

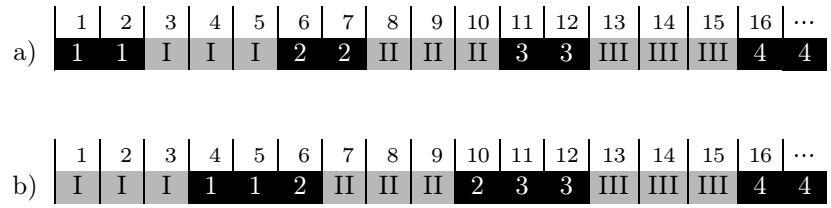


Abbildung 7.2. Zwei Ergebnisse des Scheduling a) p_1 höher priorisiert b) p_2 höher priorisiert

Earliest Deadline First

Bei diesem Algorithmus werden die Aufgaben einzeln priorisiert (anstatt die Prozesse statisch zu priorisieren). Es wird stets die Aufgabe bearbeitet, die die früheste Deadline besitzt. Auch hier werden Aufgaben unterbrochen, wenn eine neue Aufgabe mit früherer Deadline entsteht. Die folgende Abbildung zeigt, wie der Algorithmus die Aufgaben a bis f einteilt. Die Deadline der Aufgaben ist in der ersten Zeile angegeben.

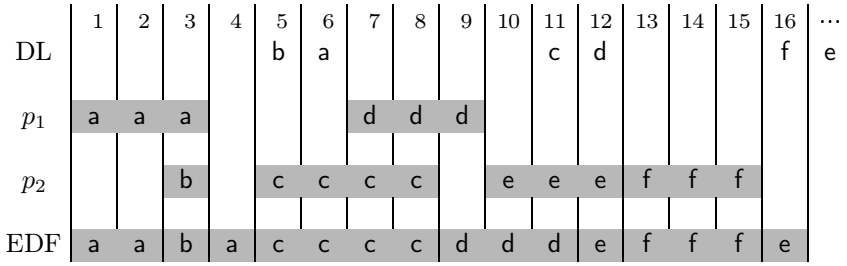


Abbildung 7.3. Entstehende Aufgaben (a-f) mit Deadline DL und Ergebnis des EDF-Scheduling

7.7 Verbesserung der Dienstgüte

Wir betrachten nun, welche unterschiedlichen Wege es gibt, Dienstgüte zu verbessern oder zu gewährleisten. Intuitiv denkt man dabei meist zuerst daran, die Leistungsfähigkeit des Kommunikationssystems zu erhöhen. Oft ist es aber auch möglich, die Anforderungen an das Kommunikationssystem zu reduzieren, ohne den Dienst aus Benutzersicht zu verändern. Beispielsweise kann

die benötigte Datenrate durch bessere Komprimierungsalgorithmen gesenkt werden. Außerdem ist in der Regel eine Benutzerkontrolle erforderlich, um zu prüfen, ob auch er seinen Teil des "Vertrages" einhält.

Verbesserung der Leistungsfähigkeit des Kommunikationssystems

Um die Leistungsfähigkeit eines Kommunikationssystems zu verbessern, können unterschiedliche Komponenten angepasst werden. Zunächst kann *schnellere Hardware* in Form von schnelleren Übertragungsmedien, Vermittlungsrechnern, Netzadaptern und Prozessoren eingesetzt werden. Auch die eingesetzten *Algorithmen* zur Realisierung der Netzdienste können oft beschleunigt werden. Insbesondere können lastadaptive Algorithmen verwendet werden.

Auch die *Protokolle* können um Funktionalitäten erweitert werden, die Dienstgüte gewährleisten können. Hierbei können Priorisierung oder Betriebsmittelreservierung eingeführt werden. Die Reservierung von Betriebsmitteln ist dabei häufig komplexer und somit teurer als die Priorisierung. Für beide Mechanismen gibt es Beispiele im globalen Internet:

- Mit Hilfe des Protokolls RSVP (Resource Reservation Protocol) kann ein Benutzer allen RSVP-fähigen Vermittlungsrechnern auf dem Pfad zum Empfänger die speziellen Anforderungen für eine zukünftige Verbindung mitteilen. Die Vermittlungsrechner versuchen dann, dafür nötige Ressourcen zu reservieren.
- Der Kopf eines IP-Paktes der Version 4 enthält ein Feld (Type of Service, TOS), das zur Priorisierung der einzelnen Dateneinheit verwendet werden kann. Dieses Feld wird aber von den meisten Vermittlungsrechnern nicht beachtet.

Zuletzt kann das Kommunikationssystem intern Redundanzen realisieren. So kann eine *Aufteilung des Verkehrs* auf knoten- und leitungsdisjunkte Pfade erfolgen, um die Konsequenzen des Ausfalls eines Knotens oder einer Leitung gering zu halten. Eine solche Aufteilung des Verkehrs läßt sich gut mit einer *netzinternen Vorwärtsfehlerkorrektur* kombinieren.

Reduktion der Anforderungen

Es gibt eine Vielzahl von Möglichkeiten, die quantitativen Anforderungen, die an ein Kommunikationssystem gestellt werden, zu reduzieren. Wir betrachten einige Arten von Anforderungen und wie diese ohne Beeinträchtigung des Dienstes reduziert werden können.

Durchsatz. Ein Dienst kann mit niedrigeren Anforderungen in Bezug auf Durchsatz arbeiten, wenn (bessere) Kompressionsalgorithmen eingesetzt werden. Für Videokommunikation lassen sich etwa zeitliche Redundanzen (Ähnlichkeiten von Folgebildern) und räumliche Redundanzen (Ähnlichkeiten von Bildbereichen) durch geeignete Algorithmen eliminieren. Zu den wichtigsten Standards für Videokompression gehören MPEG-1, -2 und -4, sowie H.261 und H.263.

Zuverlässigkeit. Bietet das Kommunikationssystem keine ausreichende Zuverlässigkeit bei der Übertragung von Datagrammen, so kann Vorwärtsfehlerkorrektur eingesetzt werden, um Paketverluste ohne eine (meist nicht mögliche) erneute Übertragung auszugleichen. Der Preis dafür ist ein erhöhter Durchsatz sowie zusätzliche Verarbeitungsleistung in den Endsystemen.

Verzögerung. Sollen Audio-/Video-Ströme über ein Kommunikationssystem mit starker Verzögerungsschwankung gesendet werden, wird beim Empfänger normalerweise ein Puffer verwendet, um die Schwankungen auszugleichen. Die Wahrscheinlichkeit dafür, dass die Darstellung aussetzt, wird dadurch reduziert. Der Preis hierfür ist eine gewisse Menge Speicher und eine größere (aber konstante) Gesamtverzögerung.

Sicherheit. Wenn ein Netz die Vertraulichkeit und Integrität der Daten nicht ausreichend gewährleisten kann, werden diese normalerweise netzextern verschlüsselt. Auch die schon angesprochene Aufteilung der Daten auf unterschiedliche Pfade kann zur Verbesserung der Sicherheit genutzt werden.

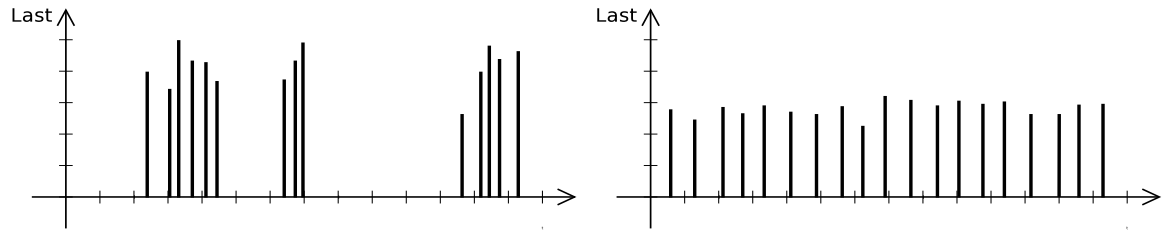
Steuerung des Benutzerverhaltens

Meist wird auch das Benutzerverhalten kontrolliert und gesteuert, um Dienstgüte zu erreichen oder zu verbessern. Wir unterteilen die Maßnahmen in solche, die gezielt das individulle Verhalten eines Benutzers beeinflussen und solche die sich auf die Gesamtmenge der Benutzer beziehen.

- Zwischen einzelnen Benutzern und dem Diensterbringer kann beim Verbindungsaufbau eine Vereinbarung (traffic contract) getroffen werden. Der Benutzer charakterisiert die von ihm zu erwartende Last im für den Betreiber schlechtesten Fall. Während der Verbindung überprüft der Betreiber durch Messungen, ob der Benutzer sich an die Vereinbarung hält. Alternativ kann das Einhalten der Vereinbarung durch eine konkrete Zugangskontrolle realisiert werden. Der Benutzer passt die von ihm tatsächlich erzeugte Last so an, dass sie der Vereinbarung entspricht. Beispielsweise würde eine stark schwankende Last schon beim Benutzer "geglättet" werden, indem die Sendungen mit Hilfe von Puffern gleichmäßig auf bestimmte Zeitintervalle verteilt werden.
- In Bezug auf die Gesamtmenge der Benutzer können neue Verbindungsaufbauwünsche abgelehnt werden, wenn das Kommunikationssystem ausgelastet ist. Um die Dienstgüte laufender Verbindungen beizubehalten ist auch das Abbrechen bestehender Verbindungen denkbar. Wenn Datagramme von den Benutzern mit Prioritäten versehen werden, können diese von den Routern heruntergestuft werden, wenn der Verkehr mit hoher Priorität zu stark wird. Dies kann insgesamt durchaus zu einer besseren Dienstgüte führen.

7.7.1 Das LBAP-Modell

Im vorigen Abschnitt haben wir Vereinbarungen zwischen Benutzer und Betreiber bezüglich der generierten Last angesprochen. Wir betrachten nun ein einfaches Modell, das für konkrete Vereinbarungen eingesetzt werden kann. Die für den Betreiber wahrscheinlich wichtigste Eigenschaft der Last eines Benutzers ist deren Gleichmäßigkeit. Generiert ein Benutzer nur vereinzelt eine sehr hohe Last, so ist das für ein Kommunikationssystem deutlich ungünstiger als ein gleichmäßiges mittleres Verkehrsaufkommen. Die folgende Abbildung zeigt beide Formen der Last.



a) ungleichmäßig ("bursty") b) gleichmäßiger ("smooth")

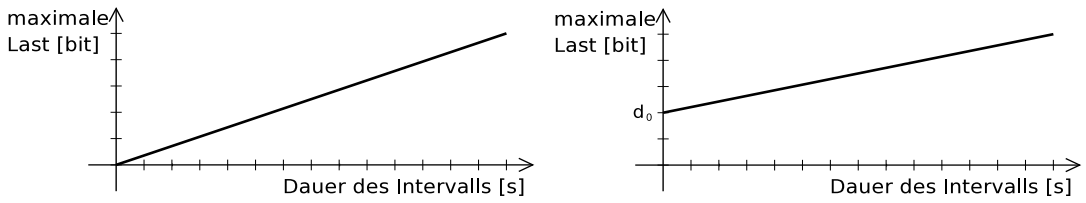
Abbildung 7.4. Von einem Benutzer generierte Last unterschiedlicher Gleichmäßigkeit

Eine primitive Vereinbarung zwischen Benutzer und Betreiber wäre die folgende: Der Benutzer darf eine maximale, konstante Datenrate d_c von 2 MBit/s nicht überschreiten. Überträgt der Benutzer innerhalb eines Intervalls von n Sekunden mehr als $n \cdot 2$ MBit, würde er die Vereinbarung verletzen. Für sehr kleine Zeitintervalle macht diese Art der Vereinbarung und Überprüfung aber oft wenig Sinn. Wenn der Benutzer mit anderen um ein gemeinsames Übertragungsmedium konkurriert oder ihm bei Multiplexverfahren nur bestimmte Zeitscheiben zur Verfügung stehen, kann man von ihm nicht verlangen, eine konstante Last zu generieren. Diese Art der Vereinbarung ist also offenbar zu primitiv:

$$\text{Last}(t) \leq t \cdot d_c$$

Dem trägt das LBAP-Modell (linear bounded arrival process) Rechnung. Es erlaubt im Gegensatz zum vorigen Modell die Übertragung einer bestimmten Datenmenge in beliebig kurzer Zeit. Trotzdem ist die Datenrate für größere Zeitintervalle begrenzt, so dass ein sehr kurzes Intervall mit sehr hoher Datenrate durch ein Intervall mit niedriger Datenrate ausgeglichen werden muss. Für

beide Modelle ist in der folgenden Abbildung die maximale Datenmenge als Funktion der Dauer eines Intervalls angegeben.



a) Primitives Modell b) LBAP-Modell

Abbildung 7.5. Modelle zur Glättung der generierten Last

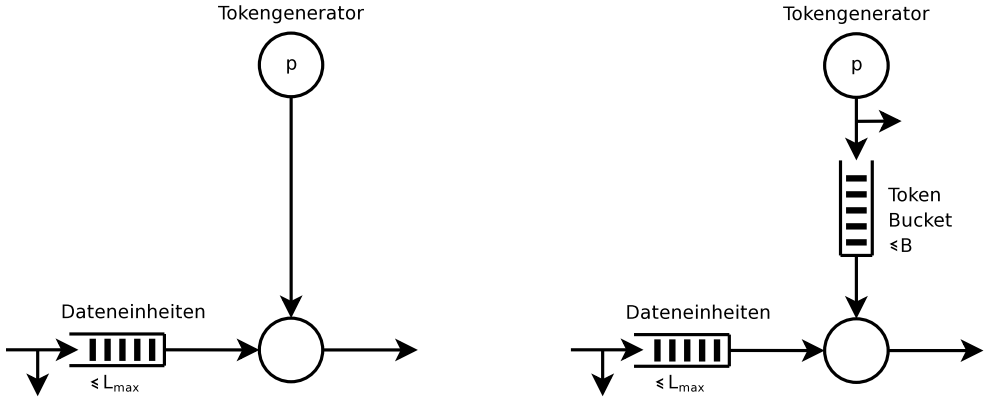
Wir bezeichnen jetzt die maximale Datenmenge, die in beliebig kurzen Intervallen gesendet werden kann als d_0. Die Gleichung für die maximale Datenmenge in einem Intervall der Länge t lautet dann

$$\text{Last}(t) \leq d_0 + t \cdot d_c$$

Es ist offensichtlich nicht praktikabel, das Benutzerverhalten dahingehend zu überprüfen, ob für *jedes* Zeitintervall die entsprechende Ungleichung erfüllt wird. Glücklicherweise gibt es recht einfache Mechanismen, mit denen beide oben beschriebenen Modelle realisiert werden können: "Leaky Bucket" und "Token Bucket". Anschaulich wird das Benutzerverhalten gesteuert, indem das Senden einer Dateneinheit (oder eines Bits, Bytes, Blocks o.ä) ein *Token* erfordert. Diese Token werden mit einer gewissen Rate generiert, so dass die maximal mögliche Last begrenzt wird.

Die folgende Abbildung zeigt schematisch die Funktionsweise der Mechanismen.[7.2] Ein Tokengenerator erzeugt Token mit der Rate ρ. Das Absenden einer Dateneinheit benötigt ein Token. Im Fall des Leaky Bucket werden so die Dateneinheiten mit konstanter Rate abgesendet (genau wie ein undichter Behälter mit konstanter Rate seinen Inhalt verliert). Sind keine Dateneinheiten zu versenden, so verfallen die Token ungenutzt. Die Warteschlange für abzusendende Dateneinheiten kann im übrigen durch eine Größe $L_{\max}$ limitiert sein, so dass es bei einer zu großen angestrebten Senderate zu Datenverlusten kommen kann.

Beim Token Bucket besteht zusätzlich die Möglichkeit, eine gewisse Menge von Token im Token Bucket zu "sammeln". Erst wenn dieser Behälter der Kapazität B gefüllt ist, verfallen weitere Token ungenutzt. Auf diese Weise können bis zu B Dateneinheiten gleichzeitig gesendet werden. Danach muss allerdings eine "sparsame" Phase folgen, um den Behälter wieder zu füllen.



a) Leaky Bucket b) Token Bucket

Abbildung 7.6. Realisierung der Modelle durch Token

Betrachtet man die Gleichungen der beiden Modelle LBAP und Token Bucket so stellt man fest, dass die Datenrate d_c der Generationsrate der Token ρ entspricht. Beim LBAP-Modell korrespondiert außerdem die Größe d_0 mit dem Volumen B des Token Bucket.

7.2. Auch hier wird aus typografischen Gründen der Buchstabe p statt des griechischen ρ verwendet.

7.7.2 "Smart Applications"

Wenn an der Dienstgüte eines Kommunikationssystems (zumindest aus Benutzersicht) nichts zu ändern ist, können sich die kommunizierenden Anwendungen[7.3] so gut wie möglich an die schwankende Qualität der Datenübertragung anpassen. Solche Anwendungen werden als "Smart Applications" bezeichnet. Es gibt eine Fülle von möglichen Reaktionen auf veränderte Netzzustände, von denen wir nur einige nennen wollen:

- Erhöhung der Redundanz bei gesteigerter Verlust- oder Fehlerrate
- Erhöhung der Kompression bei gesunkenem Durchsatz
- Häufigeres Senden von Keyframes bei Videoübertragungen (z.B. I-Frames bei MPEG) bei gestiegener Fehler- oder Verlustrate, um die Fehlerfortpflanzung zeitlich zu begrenzen
- Fragmentierung in kleinere Dateneinheiten (die ein geringeres Verlustrisiko besitzen) bei gestiegener Verlustrate
- Vergrößerung der Puffer bei gestiegener Verzögerungsschwankung

Als Beispiel für eine solche "schlaue" Anwendung, kann eine adaptive Videocodierung betrachtet werden, die je nach Verlusthäufigkeit mehr oder weniger redundant codiert. Beispielsweise wird eine gerade aufgezeichnete Videosequenz von einer Workstation codiert (etwa mit MPEG oder H.261) und über ein paketvermittelndes Netz an eine andere Workstation gesendet. Diese teilt der sendenden Station regelmäßig mit, welche Verlusthäufigkeiten aktuell zu erwarten sind. Dies kann durch eine quantitative Abschätzung, beispielsweise durch Messungen von Leistungskenngrößen der Verbindung oder durch qualitative Abschätzungen durch den Benutzer erfolgen.

7.7.3 Bewertung der Dienstgüte

Um die Dienstgüte zu verbessern oder ein bestimmtes Maß an Qualität zu gewährleisten, muss sie in irgendeiner Form gemessen oder bewertet werden. Solche Bewertungen können periodisch oder ereignisgesteuert durchgeführt werden. Die Bewertung kann entweder durch die Benutzer selbst oder durch Messungen erfolgen:

- der Mean Opinion Score (MOS) ergibt sich aus der Befragung von Benutzern
- der Peak Signal to Noise Ratio (PSNR) lässt sich durch Messungen am Übertragungsmedium bestimmen

Die Beurteilung der Dienstgüte aus Benutzersicht ist letztendlich ausschlaggebend (vgl. Definition 7.3). Allerdings ist eine subjektive Bewertung häufig unpraktikabel, weil sie aufwändig und möglicherweise lästig ist und ihre Ergebnisse auf technische Parameter des Netzzustandes abgebildet werden müssen (QoS-Mapping). Messungen zur Bestimmung der Dienstgüte können entweder netzextern durch Beobachtung der Endsysteme oder netzintern durch Beobachtung der Netzkomponenten durchgeführt werden. Sie beeinträchtigen den Benutzer weniger oder gar nicht. Allerdings können sie sehr komplex sein und das Kommunikationssystem selbst beeinträchtigen.

Modellgestütztes QoS-Management

Eine komplexe Form des Netzmanagements zur Verbesserung und Erhaltung eines Maßes an Dienstgüte stellen modellgestützte QoS-Managementsysteme dar. Diese beobachten den Zustand des Kommunikationssystems, treffen Entscheidungen anhand von Modellen und nehmen Einfluss auf die Komponenten des Kommunikationssystems. Abbildung 7.7 zeigt schematisch den Aufbau solcher Systeme. Das Vorgehen des Managers ist dabei:

1. Periodischer oder ereignisgesteuerter Empfang von Zustandsinformationen
2. Reaktion auf Ereignisse wie QoS-Verletzung, Verbindungsaufbau, Komponentenausfall

7.3. oder die darunterliegende Middleware

3. Modellbasierte Entscheidung für eine Vorgehensweise
4. Steuernder Eingriff in das Kommunikationsnetz

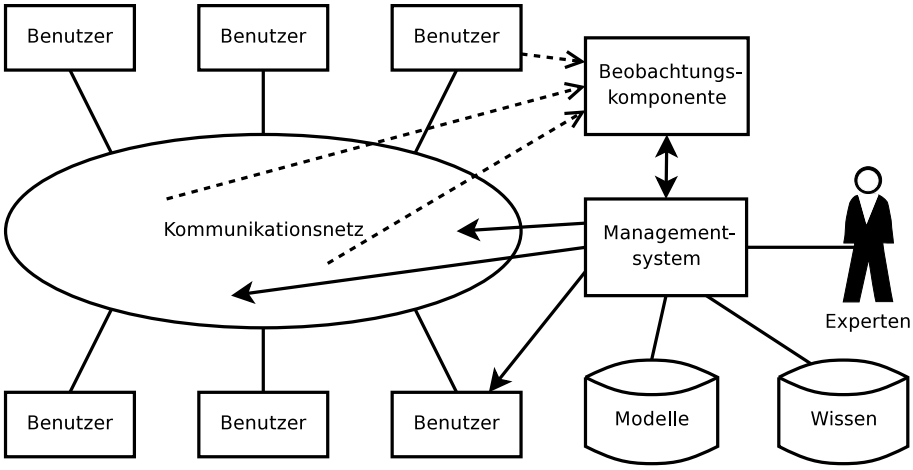


Abbildung 7.7. Modellgestütztes QoS-Managementsystem

7.8 *E-Learning-Werkzeug: MedienExplorativ*

Mit dem Werkzeug MedienExplorativ (mit seinen Anteilen VideoExplorativ und AudioExplorativ) kann dem/der Studierenden die Beeinflussung verteilter Applikationen am Beispiel von Audio- und Videokommunikation in verzögerungs- und verlustbehafteten Netzen (bspw. dem Internet) dargestellt und verdeutlicht werden. Als weiteres wichtiges Lernziel soll der/die Studierende in die Lage versetzt werden, die Grenzen der Qualitätseinbuße durch existierende Echtzeitanforderungen und die Vorteile von adaptiven Applikationen bei der Übertragung über verzögerungs- und verlustbehaftete Netze erkennen zu können.

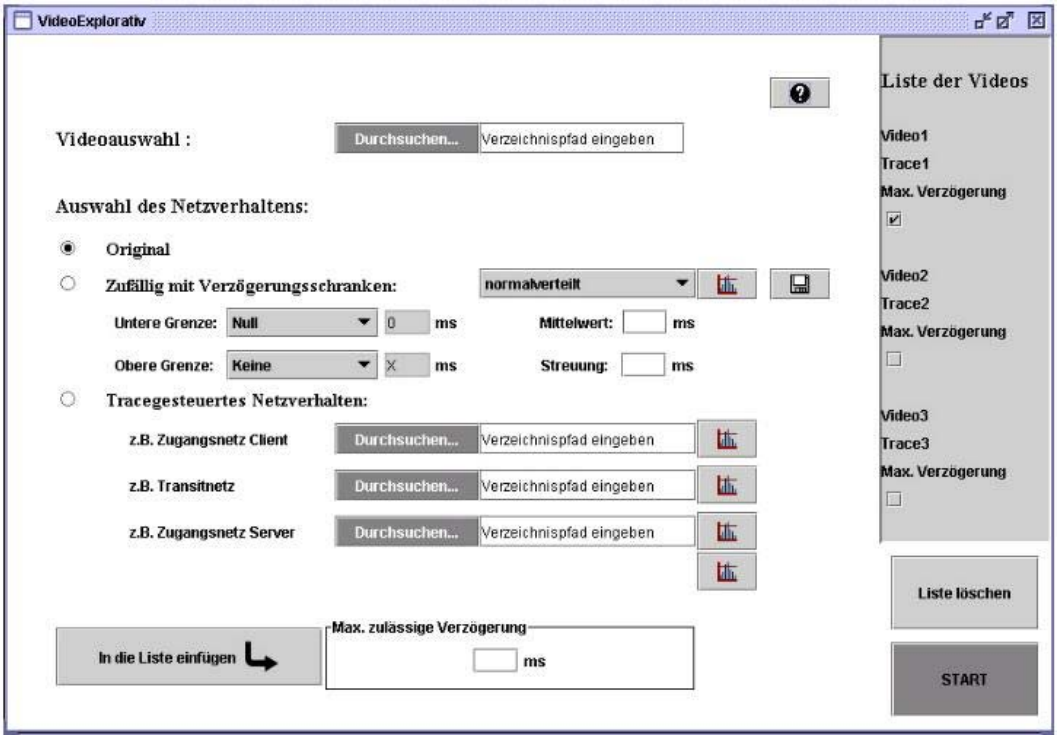


Abbildung 7.8. Screenshot des Programms

Bei dem hier kurz beschriebenen Teilwerkzeug VideoExplorativ können MPEG-Videoströme ausgewählt werden, die dann durch die nutzerseitige Auswahl eines bestimmten Netzverhaltens beeinflusst werden [Hal 01]. Das Netzverhalten kann durch real gemessene, manuell kreierte oder per Verteilung erstellte Traces eingestellt werden [SWS 03]. Die Traces enthalten dazu Verzögerungswerte und Verlustdarstellungen (Verzögerungswert: –1). In einem weiteren Schritt kann durch die Eingabe eines Schwellwertes ab wann die Verzögerung so groß ist, dass sie nicht mehr für die Applikation nutzbar ist (indirekter Verlust), der Trace transformiert und das Netzverhalten an die gewünschte Situation adaptiert werden. Damit können auch unterschiedliche Echtzeitanforderungen bei verschiedenen Szenarien repräsentiert werden. Die Traces (originale wie bereits transformierte) können durch eine integrierte Funktion grafisch dargestellt werden, so dass der/die Studierende durch Einsicht der Grafik bereits vorab erste Vermutungen über den Verlauf der Videoqualität anstellen kann (siehe Abbildung 7.9).

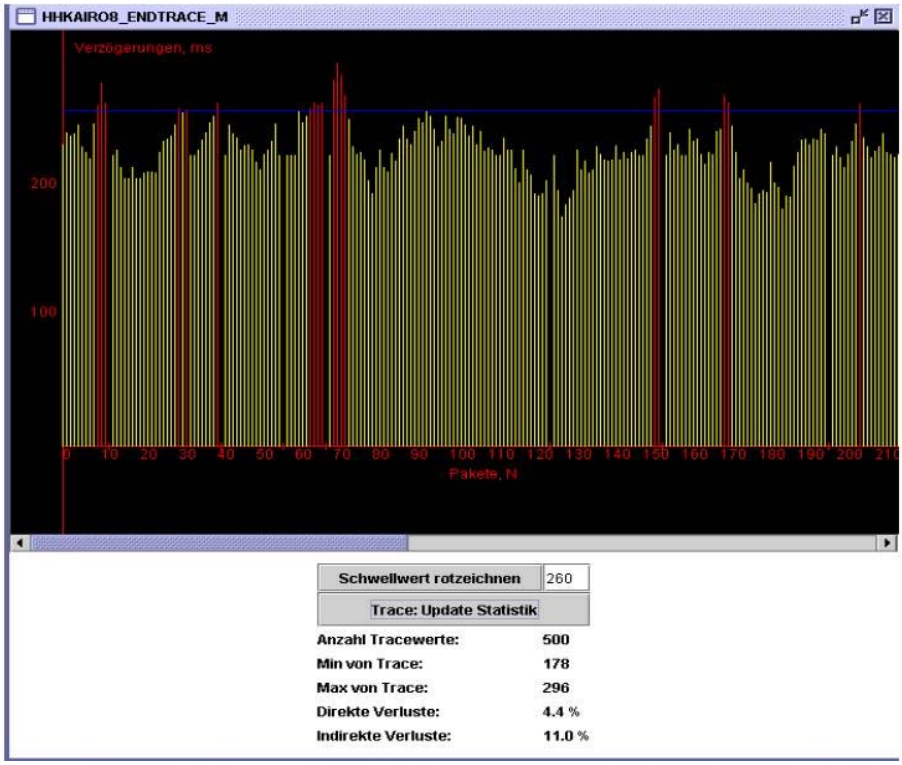


Abbildung 7.9. Grafische Darstellung des Traces

Der transformierte Trace wird in der nächsten Phase der Programmabwicklung mit dem paketisierten Videostrom so verknüpft, dass jeder Tracewert mit einem Paket assoziiert wird. Dieser verzögerungs- und verlustbehaftete Videostrom kann nun mittels eines handelsüblichen "Players" abgespielt werden (für weitere Details auch zur Architektur siehe [FHW 04]).

Abbildung 7.10. Frames aus einem manipulierten MPEG-Videostrom

Hintergrund so einer verzerrten Videodarstellung ist die Tatsache, dass in einem typischen IP-Netz keine Garantie für die zeitgerechte, fehlerfreie und reihenfolgengetreue Übertragung von IP-Paketen gegeben werden kann, das IP-Netz arbeitet nach dem sogenannten "best effort"-Prinzip ("es versucht sein Bestes").

Das E-Learning-Werkzeug MedienExplorativ[7.4] wird in den vorlesungsbegleitenden Übungen benutzt. Zu Details sei auf die entsprechenden Übungsblätter und die separat bereitgestellte Dokumentation zu MedienExplorativ sowie einen zusätzlichen Lernmodul zur Videokommunikation verwiesen.

7.4. realisiert im Rahmen des durch das E-Learning Consortium Hamburg (ELCH) geförderten TeleMuM-Projektes, Arbeitsgruppe TKRN, 5/03–06/05.

Kapitel 8
Protocol Engineering

In Anlehnung an den Begriff "Software Engineering", der den ingenieursmäßigen Charakter dieser Tätigkeit betont, nennen wir eine technisch-wissenschaftliche Beschäftigung mit Kommunikationsprotokollen Protocol Engineering. Die typischen Phasen des Lebenszyklus von Software sind auch für Protokolle gültig. In Anlehung an die allgemeine Software-Entwicklung, geben wir im folgenden entsprechende Begriffsdefinitionen

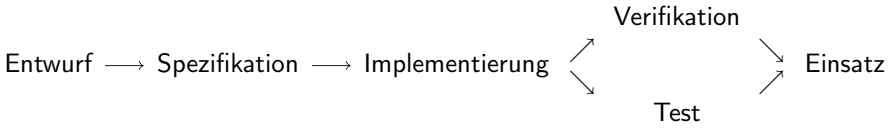


Abbildung 8.1. Typischer Lebenszyklus von Software

Definition 8.1. *Unter einer* Protokollspezifikation *verstehen wir die formale Beschreibung eines Protokolls, insbesondere dessen wesentliche Aspekte:*

- *die Syntax und Semantik der ausgestauschten Protokolldateneinheiten*
- *die zeitlichen Abläufe (timing) während der Kommunikation*

Definition 8.2. *Eine* Protokollverifikation *ist ein formaler Beweis der Konformität einer existierenden Protokollimplementierung mit der entsprechenden, zugehörigen Protokollspezifikation. In einer alternativen Sicht verstehen wir sie als Nachweis der Konformität einer gegebenen Protokollspezifikation mit einer zugehörigen Dienstspezifikation.*

Definition 8.3. *Ein* Protokolltest *ist ein Verfahren zur Überprüfung von Eigenschaften einer Protokollimplementierung, insbesondere um den Grad der Übereinstimmung dieser Implementierung mit der zugehörigen Protokollspezifikaiton beurteilen zu können.*

Im Gegensatz zur Verifikation versucht man in der Testphase, Fehler in der Implementierung zu entdecken. Tests können aber nicht zum Nachweis der Fehlerfreiheit einer Implementierung dienen.

Definition 8.4. *Das Bewerten der Leistungsfähigkeit oder Zuverlässigkeit eines Protokolls, anhand von Messungen, Modellen o.ä., nennen wir* Protokollanalyse.

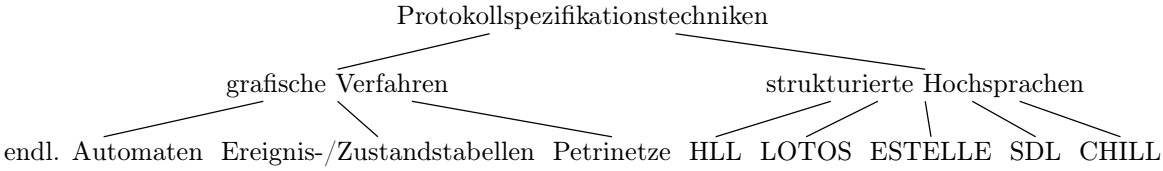
8.1 Protokollspezifikation

Das angestrebte Resultat einer Protokollspezifikation ist eine präzise, vollständige, widerspruchsfreie, formale Beschreibung des entsprechenden Problems. Dabei werden meist die folgenden Ziele

verfolgt.

- Ein hinreichend großer Formalisierungsgrad soll erreicht werden. Allerdings existiert hier häufig ein "Trade-Off" mit der Verständlichkeit für Menschen.
- Die Implementierung des Protokolls soll durch die Spezifikation unterstützt werden. Im Idealfall ist eine automatisierte Umsetzung möglich. Eine effiziente Implementierung führt allerdings oft zu einer starken Abhängigkeit vom Betriebssystem.
- Die Spezifikation soll eine spätere Protokollverifikation unterstützen.
- Formale Eigenschaften wie Fairness, Lebendigkeit und Verklemmungsfreiheit sollen erfüllt werden. Diese betrachten wir am Ende dieses Abschnitts.

Die folgende Klassifikation gibt einen Überblick über die Fülle an Spezifikationstechniken. Es folgen Beispiele für zwei dieser Techniken: endliche Automaten und SDL.



Sequentielle Automaten

Als erstes Beispiel wollen wir mit Hilfe eines sequentiellen Automaten eine Spezifikation für das BSC-Protokoll erstellen. Dieses Protokoll der Datensicherungsschicht regelt die Kommunikation zwischen einer Leitstation und einer Trabantenstation. Nachrichten werden dabei durch Fragmentierung in Blöcke zerlegt, jeder Block wird einzeln quittiert. Es erfolgt eine Flusskontrolle mit einfachem Kredit.

Definition 8.5. *Ein sequentieller Automat ist ein Tupel* (Z, X, Y, μ, ν)*, wobei* Z *eine Menge von internen Zuständen,* X *ein Eingabealphabet und* Y *ein Ausgabealphabet ist.* $\mu\colon Z \times X \to Z$ *ist die Zustandsübergangsfunktion,* $\nu\colon Z \times X \to Y$ *die Ausgabefunktion.*

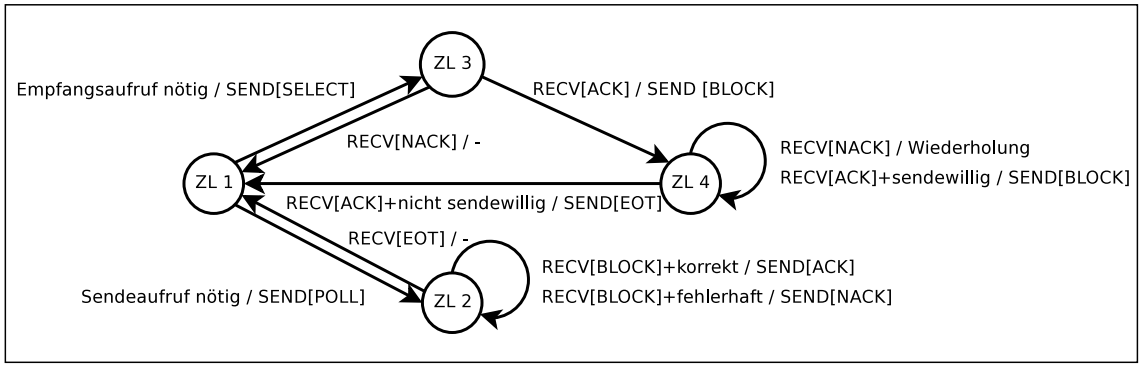
In der grafischen Notation werden die Zustände durch Kreise dargestellt. Existieren Zustände z_1 *und* z_2 *mit* $\mu(z_1, e) = z_2$ *und* $\nu(z_1, e) = a$*, so verbindet ein Pfeil die beiden Zustände:*

$$z_1 \xrightarrow{e/a} z_2$$

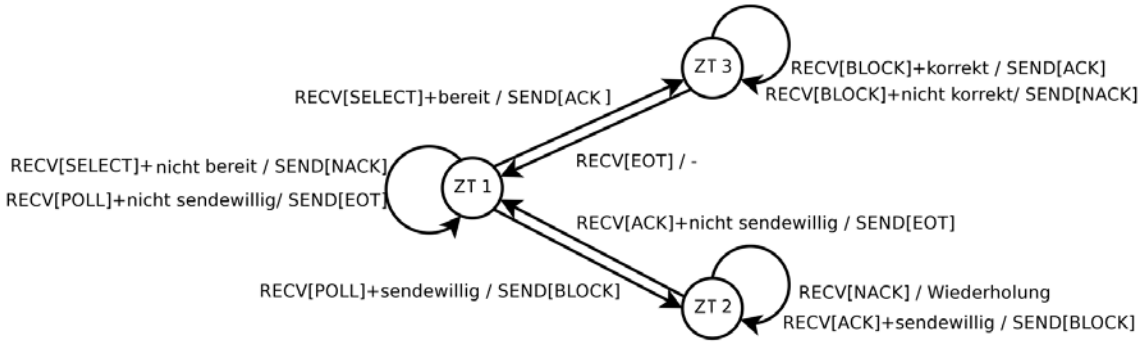
Für die Leitstation und die Trabantenstation legen wir in Tabelle 8.1 zunächst die inneren Zustände fest. Danach stellen wir die möglichen Übergänge als grafische Notationen zweier Automaten dar.

Zustand	Beschreibung
ZL_1	Grundzustand
ZL_2	Warten auf Datenblock oder Übertragungsende
ZL_3	Warten auf Sendeerlaubnis
ZL_4	Warten auf Quittierung
ZT_1	Warten auf Sende- oder Empfangsaufruf
ZT_2	Warten auf Quittierung
ZT_3	Warten auf Datenblock oder Übertragungsende

Tabelle 8.1. Innere Zustände von Leitstation (ZL_i) und Trabantenstation (ZT_i)



a) Leitstation



b) Trabantenstation

Abbildung 8.2. Automaten für Leitstation und Trabantenstation (das Symbol + entspricht dem logischen und)

Die so gegebene Spezifikation ist in vieler Hinsicht sehr stark vereinfacht. So wurde auf eine komplizierte Flusskontrolle und eine Behandlung der Zeitüberwachung (z.B. Timeouts) verzichtet. Auch ein Verbindungsaufbau und -abbau ist nicht dargestellt. Außerdem ist die Fehlerbehandlung, die beispielsweise den Erhalt von ungültigen Dateneinheiten betrifft, weggelassen.[8.1]

Specification and Description Language (SDL)

Die Sprache SDL wurde 1989 von der CCITT (inzwischen ITU) als Standard empfohlen. Sowohl grafische als auch textuelle Repräsentationen sind möglich. SDL-Spezifikationen basieren auf unterschiedlichen Abstraktionsebenen:

- System
- Block
- Prozess
- Prozedur

Die Prozesse werden als erweitere endliche Automaten – unter anderem mit den in der folgenden Abbildung dargestellten Symbolen – modelliert.

8.1. Man beachte, dass dieser Teil oft einen Großteil der Spezifikation ausmacht.

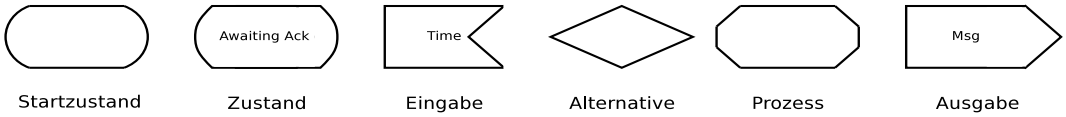


Abbildung 8.3. Symbole des grafischen Formalismus von SDL

Beispiel 8.6. Ein sendender Prozess sendet eine Dateneinheit Data über einen verlustbehafteten Kanal an einen Empfänger. Dieser sendet eine Bestätigung ACK an den Sender zurück, sobald er die Daten korrekt empfangen hat. Erhält der Sender nach einer festgelegten Zeit keine positive Bestätigung, so sendet er die Daten erneut. Nach einer erfolgreichen Übertragung und Bestätigung versuchen Sender und Empfänger weitere Dateneinheiten zu übertragen.

Man beachte, dass auch diese Ausgangssituation stark vereinfacht ist. Unter anderem wird nicht spezifiziert, wie der Sender die Bestätigungen zu den Dateneinheiten zuordnet. Abbildung 8.4 zeigt die zu spezifizierende Ausgangsituation.

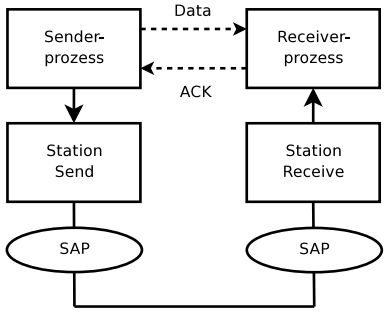


Abbildung 8.4. Ausgangssituation für die Spezifikation mit SDL

Eine Spezifikation mit Hilfe des grafischen Formalismus von SDL ist in der folgenden Abbildung 8.5 angegeben. Hierbei sind Sender und Empfänger als Prozesse modelliert. Der Sender-Prozess wartet auf zu übertragende Daten und erhält diese über die Eingabe Data VIA ARQSend. Er setzt daraufhin einen Timer, sendet die Daten mit der Ausgabe Data VIA SAP an der Empfänger und wechselt in den Zustand AwaitAck. In diesem Zustand sind zwei Eingaben möglich: Das Eintreffen einer Bestätigung und das Ablaufen des Timers. Das Ereignis, das zuerst eintritt, bestimmt den weiteren Ablauf. Wenn der Timer abläuft, wird die Dateneinheit erneut gesendet und der Zustand AwaitAck eingenommen. Trifft eine Bestätigung ein, so wird der Timer zurückgesetzt und es wird in den Zustand AwaitData gewechselt.

Die Spezifikation des Empfängers sollte selbsterklärend sein. Der Kontext der beiden Prozesse ist auf einer höheren Abstaktionsebene im unteren Teil der Abbildung als System spezifiziert.

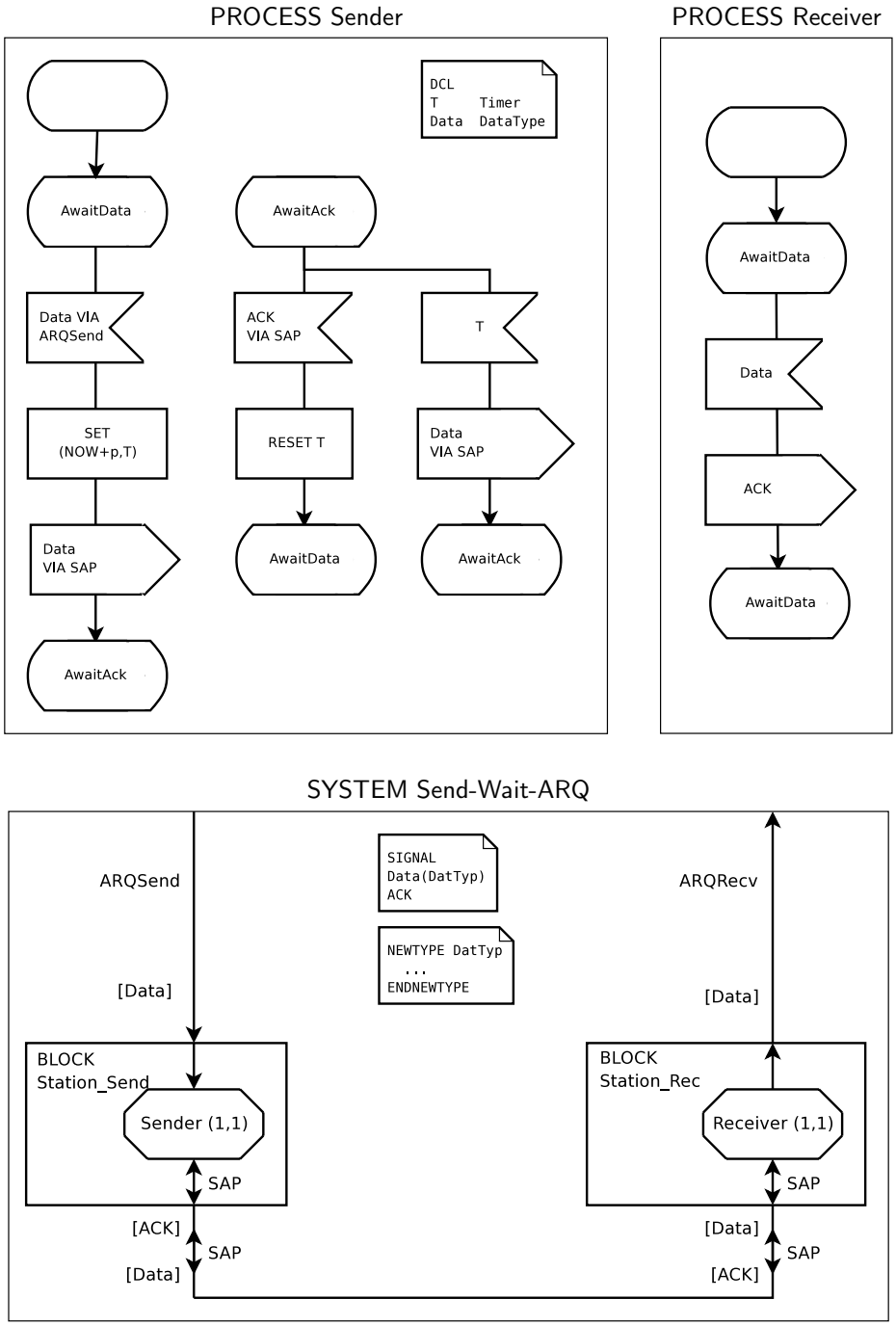


Abbildung 8.5. Beispiel für eine Spezifikation mit SDL

E-Learning-Werkzeug: Protokollautomat

Weitere Erfahrungen mit automatenbasierter Protokollspezifikation kann in den DKR-Übungen mit dem bei TKRN entwickelten Werkzeug "Protokollautomat" gesammelt werden (vgl. auch hier die Fußnote 7.4). Im Gegensatz zum ersten Beispiels dieses Kapitels sind damit auch die genauen zeitlichen Abläufe einfach modellierbar.

8.2 Protokollverifikation

Die Frage, ob eine vorliegende Implementation eines Protokolls sämtliche der durch die Spezifika-

tion geforderten Eigenschaften besitzt, wird im Rahmen einer Protokollverifikation beantwortet. Im Gegensatz zu Tests werden die Eigenschaften bei der Verifikation formal bewiesen. Typische zu beweisende Eigenschaften sind:

Verklemmungsfreiheit (Deadlock-Freiheit). Sie muss eventuell auch bei unzulässigem Benutzerverhalten gewährleistet sein, sofern dieses nicht auszuschließen ist.

Lebendigkeit. Die Spezifikation sollte auch frei von partiellen Verklemmungen sein. Bei diesen Verklemmungen können bestimmte Teile des Protokollgraphen nicht mehr erreicht werden.

Vollständigkeit. Insbesondere soll für jedes mögliche Ereignis – auch in Fehlersituationen – eine Reaktion vorgesehen sein.

Fairnesseigenschaften. Es muss gewährleistet werden, dass die Aufträge eines Benutzers nicht dauerhaft durch andere Aufträge verdrängt werden können.

Spezielle Leistungseigenschaften. Auch quantitative Eigenschaften, wie z.B. die Anzahl der ausgetauschten Protokolldateneinheiten, können verifiziert werden.

Weitere Details der Verifikation – insbesondere unter Nutzung Petrinetz-orientierter Spezifikationstechniken – werden in Lehrveranstaltungen des Arbeitsbereichs TGI vermittelt. Es gibt eine Reihe von Vorgehensweise, die die Verifikation eines Protokolls vereinfachen können:

- die Verwendung sehr einfacher Protokolle (häufig jedoch nicht praxisrelevant)
- die automatisierte Umsetzung der Spezifikation
- die Nutzung höherer Spezifikations- und Programmiersprachen.

8.3 Protokolltest und -analyse

Bei einem Protokolltest wird die Implementierung eines Protokolls kontrolliert mit speziell ausgewählten Eingabedaten ausgeführt. Auf diese Weise ist möglicherweise ein Nachweis für die Anwesenheit von Fehlern möglich. Ein Nachweis für die Abwesenheit von Fehlern ist dagegen unmöglich.

Nachgewiesene Fehler werden in der Regel durch *Debugging* ausfindig gemacht. Man unterscheidet statisches und dynamisches Debugging. Bei ersterem erfolgt eine Suche des Fehlers in den Programmdokumenten, ohne dass das Programm ausgeführt wird. Beim dynamischen Debugging wird das Programm schrittweise ausgeführt. Dabei wird unter anderem der Verlauf der Variablen und der Kontrollfluss überprüft.

Neue zu testende Komponenten (IUT, instance under test) werden in der Regel zusammen mit bereits getesteten Implementierungen getestet. Die folgende Abbildung zeigt wie die IUT in einer getesteten Umgebung eingebettet ist. Für einen Test wird hier das Verhalten der Benutzer B_1 und B_2 der nächsthöheren Protokollschicht in gezielter Weise variiert, um so eine möglichst große Menge unterschiedlicher Kommunikationssequenzen gezielt bezüglich ihrer Korrektheit überprüfen zu können.

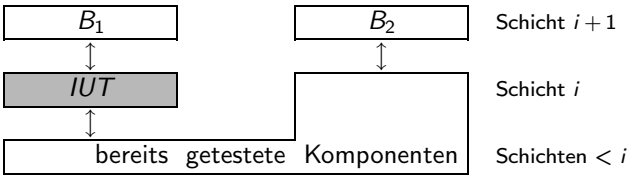


Abbildung 8.6. Testen einer Protokollimplementation

Kapitel 9

Traffic Engineering

Eine technisch-wissenschaftliche Beschäftigung mit Verkehr und Kommunikations- und Rechnernetzen wird als "Traffic Engineering" bezeichnet. Abbildung 9.1 zeigt die unterschiedlichen Teilgebiete und ihren Einfluss aufeinander.

Lastmessungen und *Verkehrsanalysen* können das Benutzerverhalten dokumentieren und als Basis für Vorhersagen über das zukünftige Verhalten dienen. Sie stellen den Ausgangspunkt einer realistischen Lastmodellierung dar. Künstliche Lastgeneratoren, die auf umfangreichen und statistisch ausgewerteten Messdaten beruhen, können evtl. synthetische Lasten auf eine hinreichend valide Art und Weise erzeugen. *Lastmodelle und -spezifikationen* werden für ausreichend realistische analytische und simulative Kommunikations- und Rechnernetzmodelle benötigt. Die auf solchen Modellen bzw. Spezifikationen beruhenden Lastgeneratoren können auch zur (direkten) Analyse existierender Rechnernetze verwendet werden.

Letztendlich kann in realen Rechnernetzen eine *Verkehrsbeeinflussung bzw. -kontrolle* auf der Basis der Erkenntnisse der obigen Methoden erfolgen, um benutzerseitig eine möglichst "sympathische" Last (z.B. durch Verkehrsglättung, ratenbasierte Zugangskontrolle o.ä.) zu erzeugen und so eine Verbesserung der Dienstgüte zu erreichen.

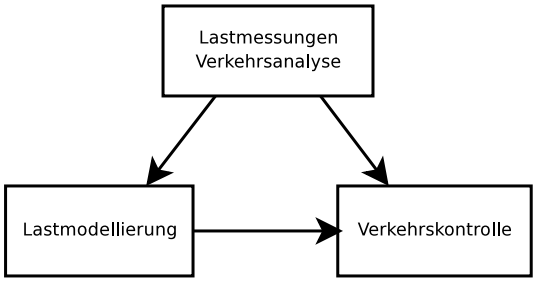


Abbildung 9.1. Teilgebiete des Traffic Engineering

9.1 Teilbereiche

Im folgenden sind einige Bereiche genannt, in denen die Techniken des Traffic Engineering Anwendung finden.

Leistungsmanagement. Es können Engpässe ermittelt und behoben werden. Durch Feineinstellung bestimmter Parameter (tuning) kann die Gesamtleistung erhöht werden. Häufig werden Vorhersagen über das Netzverhalten bei wachsender Last benötigt.

QoS-Management. Das Benutzerverhalten kann ermittelt werden und zur Prognose zukünf-

tiger Betriebsmittelbedarfe dienen.

Abrechnungsmanagement. Um den Betriebsmittelverbrauch einzelner Benutzer zu Abrechnungszwecken zu ermitteln oder abzuschätzen, können individuelle Verkehrsströme erfasst und analysiert werden.

Konfigurationsmanagement. Künstliche Verkehrsgeneratoren erlauben realitätsnahe Netzmodellierungen, wenn sie eine realistische Lastcharakterisierung besitzen.

Sicherheitsmanagement. Umfangreiche Verkehrsanalysen können als Basis für die Erkennung von Angriffsversuchen dienen. Allerdings stellt eine solche Verkehrsanalyse möglicherweise selbst ein Sicherheitsproblem dar.

Definition 9.1. *Eine* (angebotene) Last[9.1] $L = L(U, S, IF, T)$ *sei definiert als vollständige Sequenz von Aufträgen, die*

- *durch eine Umgebung* U
- *an ein Bediensystem* S
- *über eine wohldefinierte Schnittstelle* IF
- *während eines Beobachtungsintervalls* T

übergeben werden. Wir sagen auch, L *ist die seitens* U *für* S *während* T *generierte Last.*

Besonders wichtig ist bei dieser Definition, dass ein Gesamtsystem so in ein Bediensystem und eine Umgebung zerlegt wird, dass eine wohldefinierte und beobachtbare Schnittstelle resultiert. Beispiele für solche Schnittstellen sind S_0 bei ISDN, virtuelle Verbindungen bei X.25- und ATM-Netzen, Dienstschnittstellen in Protokollhierarchien und Sockets bei TCP.

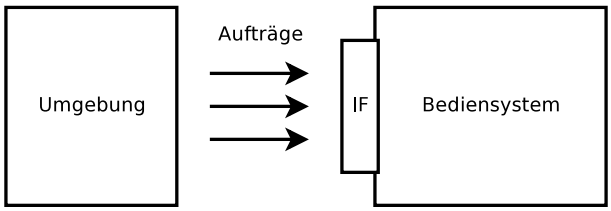
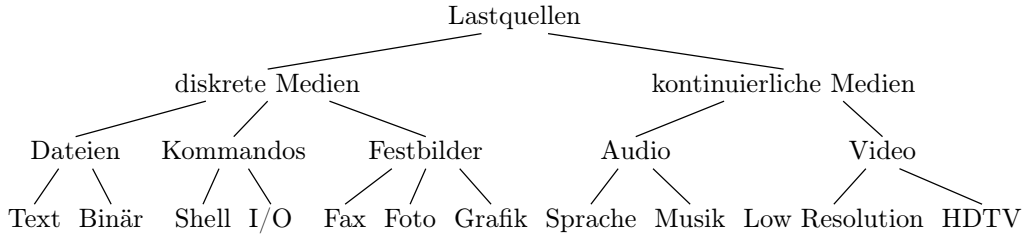


Abbildung 9.2. Veranschaulichung von Definition 9.1

Es folgt eine Übersicht über einige typische Lastquellen bei der heutigen Kommunikation in Rechnernetzen. Wir teilen diese Quellen in diskrete Medien, die in der Regel sehr unregelmäßigen Verkehr verursachen und kontinuierliche[9.2] (auch isochrone) Medien, die über längere Zeiträume zu einem eher einheitlichen Verkehr führen.



9.1. engl.: (offered) load

9.2. Das Wort "kontinuierlich" ist hier nicht im mathematischen Sinne gemeint.

Durch das Traffic Engineering kann einerseits eine Beinflussung des Verkehrs erreicht dahingehend erreicht werden, dass dieser durch das verwendete Netz effizienter verarbeitet werden kann. Auch kann eine Unterstützung der Netzsicherheit durch eine Filterung von Aufträgen erfolgen (vgl. Kapitel 8, Echtzeitkommunikation).

Zum anderen kann auf der Grundlage von Lastmessungen eine Protokollierung des Benutzerverhaltens erfolgen. Diese kann verwendet werden, um das zukünftige Benutzerverhalten zu prognostizieren, um realistische Lastmodelle zu bilden und um Lastgeneratoren zu Erzeugen oder zu parametrisieren.

9.2 *E-Learning-Werkzeug: LastExplorativ*

Die Entwicklung des E-Learning-Werkzeugs "LastExplorativ" (vgl. auch hier die Fußnote 7.4) bezog ihre Motivation aus dem Wunsch, die Studierenden in das wichtige Gebiet des Traffic Engineerings für Rechnernetze einzuführen und ihnen dabei interessante Experimentiermöglichkeiten an die Hand zu geben. Dabei sollte LastExplorativ u.a. das Erreichen folgender Lernziele unterstützen. Die Studierenden sollten durch dieses Werkzeug in die Lage versetzt werden,

- Lastmodelle unter Anwendung eines wissenschaftlich fundierten Procederes zu entwickeln und sie sodann formal zu spezifizieren – und dies für sehr unterschiedliche Schnittstellen eines Rechnernetzes,
- spezifizierte Lastmodelle auszuwerten und dadurch insbesondere ein tieferes Verständnis für die resultierenden Auftragssequenzen zu erhalten,
- die Transformationsprozesse zu verstehen, die in Netzen auf die Lasten einwirken, z.B. auf Lasten, die an anwendungsnahen Schnittstellen erzeugt werden und die dann netzintern umgeformt ("transformiert") werden in Lasten an z.B. übertragungsnahen Schnittstellen.

Da LastExplorativ u.a. die Spezifikation von Lastmodellen unterstützen soll, wollen wir nunmehr die – auf erweiterten endlichen Automaten basierende – Spezifikationstechnik präsentieren, auf die sich LastExplorativ bezieht. Aus Platzgründen kann die Spezifikationstechnik hier nur grob skizziert werden, zu Details sei auf [Wol 99], [WZH 02] verwiesen. Die Spezifikation bezieht sich dabei insbesondere auf die präzise Charakterisierung

- der einzelnen Aufträge selbst, die an der betrachteten Schnittstelle *IF* übergeben werden. Für *IF* wird eine Menge von *Auftragstypen AT* identifiziert, die die Aufträge modellieren, die an dieser Schnittstelle übergeben werden. Jedem Typ werden dabei in eindeutiger Weise *Auftragsattribute* zugeordnet. Jedes Auftragsattribut hat dabei einen a priori fest vorgegebenen Wertebereich, insbesondere `integer`, `real`, `Datenlänge`, `IP_Adresse`, `Portnummer`, `Bitmuster`. Bei der Auftragserzeugung werden jedem Auftrag konkrete Werte für die Auftragsattribute zugeordnet.
- des Ankunftsprozesses der Folge von, seitens eines Benutzers B, erzeugten Aufträgen. Der Ankunftsprozess von Aufträgen wird unter Verwendung sog. *Benutzerverhaltensautomaten (BVA)* spezifiziert, die das Verhalten jeweils eines auftragserzeugenden Benutzers beschreiben.

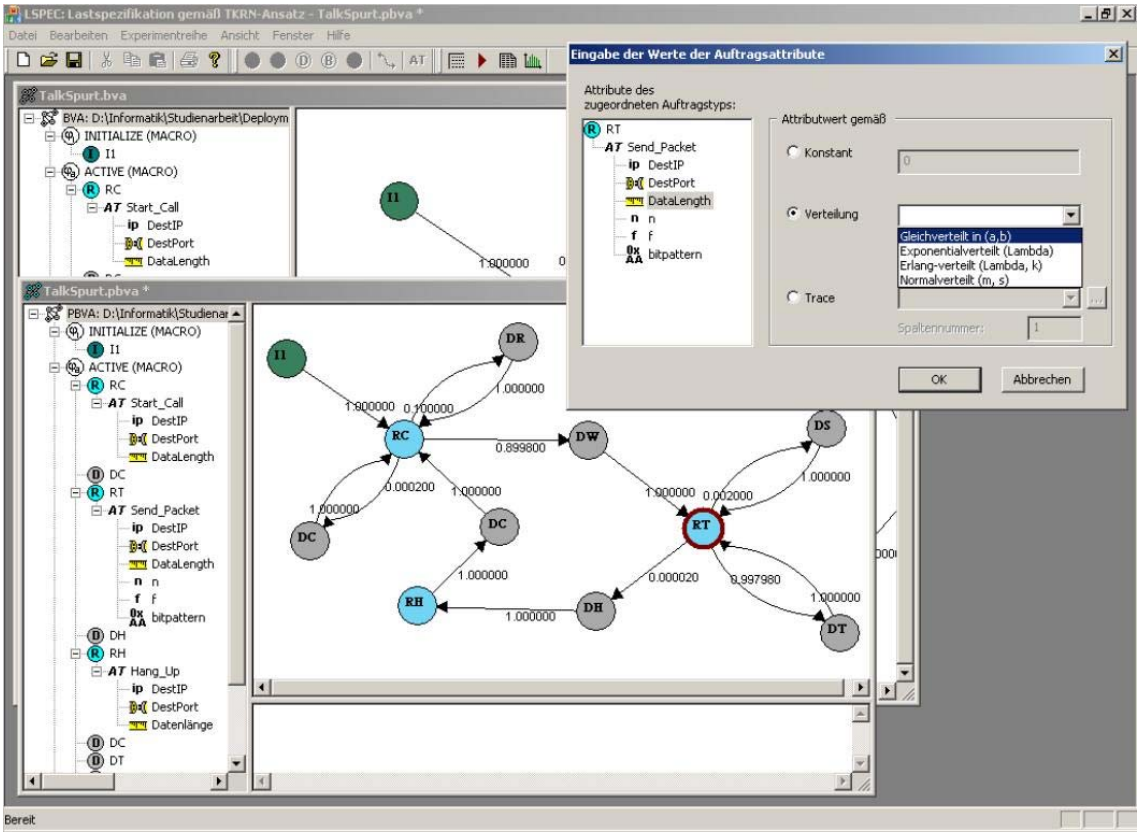


Abbildung 9.3. Lastspezifikation mittels des Werkzeugs LastExplorativ: Exemplarische pBVA-Spezifikation

Definition 9.2. *Ein BVA ist dabei definiert als ein erweiterter, endlicher Automat* $B = (\Phi, T_\varphi)$ *basierend auf einer Menge von Makrozuständen* $\Phi = \{\varphi_i, \varphi_a, \varphi_b, \varphi_t\}$ *mit folgender Bedeutung:*

- φ_i *(idle) modelliert den Ausgangszustand/Ruhezustand des auftragsgenerierenden Benutzers.*
- φ_a *(active) modelliert den Makrozustand, in dem Auftragsgenerierung erfolgt.*
- φ_b *(blocked) modelliert den Makrozustand, in dem der Benutzer für eine Generierung von Aufträgen blockiert ist, dadurch dass er zuvor eine Reaktion des Bediensystems abzuwarten hat.*
- φ_t *(terminated) modelliert den Terminierungszustand des auftragserzeugenden Benutzers.*

T_φ *bezeichnet die Menge der Transitionen zwischen den Makrozuständen.*

Die Makrozustände eines BVA gestatten uns bereits eine klare Separierung zwischen Phasen, in denen Aufträge generiert werden können und Phasen, in denen auf Systemreaktionen gewartet werden muss.

Wir haben das BVA-Modell noch verfeinert, um die als zeitlos betrachteten Ereignisse wie Auftragsgenerierung (d.h. Auftragsübergabe an *IF*) und Eintreten einer Systemreaktion (wiederum angezeigt für *U* an *IF*) von zeitbehafteten Vorgängen klar zu separieren. Dabei werden die Makrozustände φ_a und φ_b durch Mengen von sog. *D*-, *R*- und *S*-Zuständen ersetzt, die wir auch als Elementarzustände bezeichnen, mit Transitionen zwischen diesen Elementarzuständen. Hierbei modellieren *D*-Zustände u.a. die (benutzerspezifischen) Verzögerungen zwischen der Erzeugung zweier aufeinanderfolgender Aufträge. Die *R*-Zustände R_i modellieren Zustände, bei deren Erreichen ein Auftrag (genau) eines speziellen Typs $T(i)$ erzeugt wird. Dabei sind bei Auftragserzeugung

auch die konkreten Werte für die Auftragsattribute zu bestimmen. *S*-Zustände modellieren eine Blockierungssituation für einen Benutzer, bei der Ereignisse (insbesondere Systemreaktionen) abzuwarten sind (und zwar Ereignisse aus einer a priori vorgegebenen Menge von Ereignistypen).

Ein BVA gibt nur die Verhaltensstruktur eines auftragserzeugenden Benutzers vor, die sich mittels Parametrisierung auf ein konkretes Verhalten abbilden lässt. Wir sprechen nach erfolgter Parametrisierung eines BVA auch von einem *parametrisierten BVA (pBVA)*. Die Parametrisierung bezieht sich insbesondere auf

- Zustandsübergänge zwischen Elementarzuständen, z.B. durch Zuordnung von Übergangswahrscheinlichkeiten,
- Werte für Auftragsattribute, z.B. konstant, Verteilung od. Nutzung von Traces,
- Zeiten (u.a. für *D*-Zustände), z.B. konstant, Verteilung od. Nutzung von Traces.

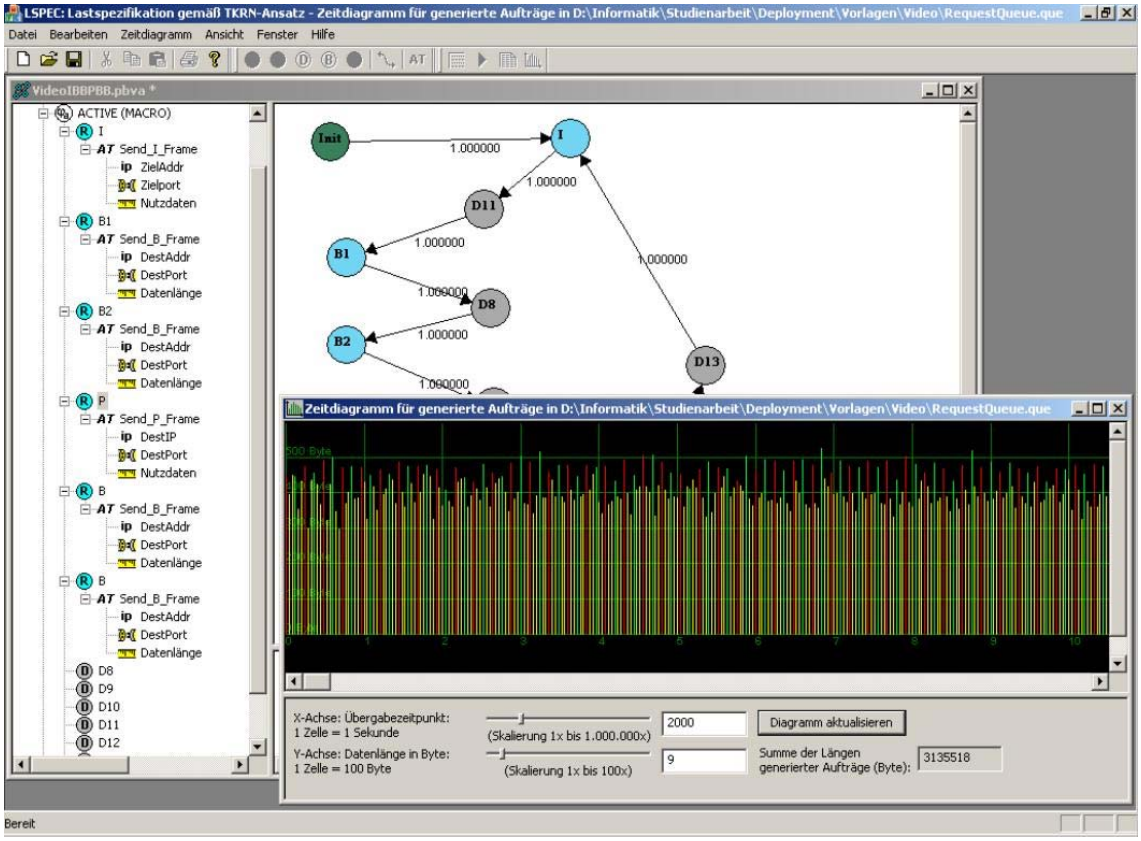


Abbildung 9.4. Grafische Illustration einer resultierenden Auftragssequenz mittels LastExplorativ

Lastcharakterisierung

Es stellt sich die wichtige Frage, in welcher Weise der Automat realitätsnah parametrisiert werden kann. Wir unterscheiden dafür einerseits *Reallasten*, die aus "echten" Ausgangsdaten wie z.B. Videoseqenzen entstehen oder deren Übertragung tatsächlich stattgefunden hat, von *künstlichen Lasten*, die durch Abstraktion auf der Basis des vorhandenen Wissens über reale Lasten gebildet werden. Eine weitere Unterscheidung führt zu vier verschiedenen Lastcharakterisierungen:

- Reallasten, die durch ein reales Netz verarbeitbar sind. Dabei ist es notwendig, dass die Schnittstelle *IF* bei konkreter Erzeugung der Last mit der Schnittstelle des realen Dienstes übereinstimmt.
- Reallasten, die als Ausgangspunkt für Abstraktionsvorgänge dienen, allerdings nicht von einem realen Netz verarbeitbar sind.

- Künstliche Lasten, deren Ausgangsdaten zwar durch Abstraktion/Modellierung von Reallasten entstanden sind, die jedoch weiterhin für ein reales Netz verarbeitbar und interpretierbar sind.
- Künstliche Lasten, die nur für ein Simulationsmodell verarbeitbar sind. Hierbei kann die modellierte Systemschnittstelle in der Regel stark vereinfacht werden.

Grundsätzlich ergeben sich zwei Möglichkeiten, die Parameterwerte für den BVA zu gewinnen:

Traces. Diese entstehen, indem beobachtete bzw. gemessene Wertesequenzen von Reallasten aufgezeichnet werden. Typische Werte sind dabei Datagrammgröße, Auftragsgenerierungszeitpunkte etc.

Wahrscheinlichkeitsverteilungen. Hierbei wird für die konkreten Lastparameterwerte ein Zufallszahlengenerator verwendet. Dieser verwendet eine bestimmte Wahrscheinlichkeitsverteilung, die als angemessene Abstraktion der realen Vorgänge angesehen wird. Eine Parametrisierung mit Hilfe von Wahrscheinlichkeitsverteilungen erfordert meist weniger Aufwand als das Aufzeichnen eines Traces. Eine typische Wahrscheinlichkeitsverteilung ist die Normalverteilung.

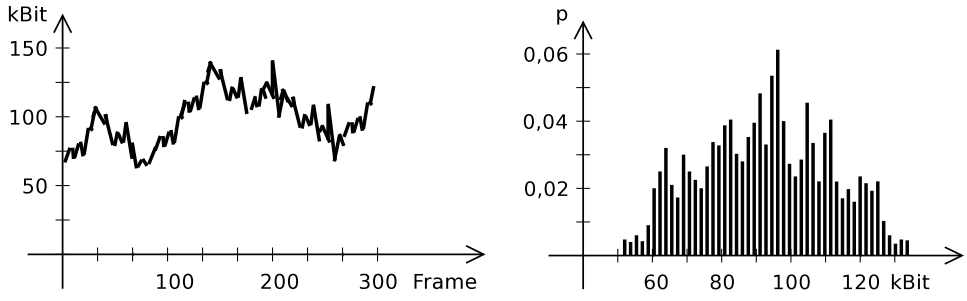
Obwohl die Verwendung von Traces im Gegensatz zu Wahrscheinlichkeitsverteilungen zunächst valider erscheinen mag, besitzen auch Traces ein Problem: Sie stellen zwar einen Vorgang dar, der real stattgefunden hat – ob dieser Vorgang allerdings repräsentativ ist, ist häufig schwer zu beurteilen. Die folgende Fallstudie zeigt, wie künstliche Lasten durch Abstraktionsvorgänge zustande kommen können.

Eine kleine Fallstudie

Bei einer MPEG-komprimierten Videosequenz besitzen die einzelnen Frames im allgemeinen unterschiedliche Größen (in Byte). Das liegt zum einen daran, dass die in einem einzelnen Bild enthaltene Redundanz unterschiedlich ist, und zum anderen daran, dass bei MPEG unterschiedliche Typen von Frames übertragen werden: Sogenannte I-Frames (intraframes) stellen vollständige Bilder dar, die einzeln dekomprimiert werden können.

Auf ein I-Frame folgen meist einige P-Frames (predictive frame). Diese Frames speichern die in Bezug auf ihren Vorgänger auftretenden Unterschiede. Da sich zwei aufeinanderfolgende Frames meist nur wenig unterscheiden, kann so sehr effizient komprimiert werden. Geht ein P-Frame bei der Übertragung verloren, können die folgenden P-Frames nicht korrekt dekomprimiert werden. Erst wenn das nächste I-Frame eintrifft, pflanzt sich der Fehler nicht mehr weiter fort.

Für eine bestimmte Videosequenz (z.B. ein Eishockey-Spiel) sollten nun die Verteilung der Framelängen (in Byte) ermittelt werden. Diese Sequenz besitzt das Komprimierungsmuster `IPPPIPPPIPPP`... Die folgende Abbildung 9.5 zeigt die Länge der einzelnen Frames über die Zeit (a) sowie ein Histogramm, dass die relative Häufigkeit der Framelängen zueinander darstellt (b). Soll eine ähnliche Videosequenz mit Hilfe von Datagrammen fester Länge übertragen werden, kann eine solche Messung Hinweise für eine optimale Datagrammlänge geben.



a) zeitlicher Verlauf b) relative Häufigkeit

Abbildung 9.5. Framelängen bei einer MPEG-komprimierten Viedeosequenz

Außerdem zeigt sich, dass bei hinreichend langen Videosequenzen eine typische relative Häufigkeit feststellbar ist. In Abbildung 9.5b ist schon eine grobe Ähnlichkeit mit der Normalverteilung erkennbar. Nimmt eine solche Ähnlichkeit bei längeren Sequenzen zu, so erscheint die Verwendung einer Normalverteilung an Stelle von umfangreichen Traces als eine gerechtfertigte Vereinfachung. Dennoch ist zu beachten, dass bei solchen Abstraktionen immer gewisse Eigenschaften verloren gehen können. Unter anderem wären folgende Beispiele denkbar:

- Der Verlauf der Framelängen besitzt möglicherweise eine eigene Charakteristik. I-Frames sind relativ groß, da sie die volle Information des Bildes in sich tragen. Der nächste P-Frame ist deutlich kleiner, da nur der Unterschied zum vorangegangenen Bild zu codieren ist. Die Längen der weiteren P-Frames bis zum nächsten I-Frame hängen nun sehr stark von der Bewegungsintensität in der Videosequenz ab, so dass sich eine starke Korrelation zwischen den Längen benachbarter P-Frames (abhängig von der momentan vorhandenen Bewegungsintensität) ergibt.

- Eine echte Normalverteilung besitzt keine Nullstellen. Daher sind mit sehr geringer Wahrscheinlichkeit auch unrealistisch große Frames möglich. Es kann allerdings angenommen werden, dass in einer realen Videosequenz eine bestimmte Framelänge *nie* überschritten wird. Gleichfalls werden generierte Videoframes eine gewisse Minimallänge niemals unterschreiten,[9.3] wenngleich jede Normalverteilung auch negative Werte zulassen würde. Diese Eigenschaft könnte für die realitätsnahe Modellierung wichtig sein. Die Normalverteilung wäre also an den Rändern durch Längen $l_{\min}$ und $l_{\max}$ zu "beschneiden" (truncated distribution), die dann nicht mehr unter- bzw. überschritten werden können.

9.3 Lastmessungen in Rechnernetzen

In Rechnernetzen mit einer Protokollhierarchie kann man die Darstellung aus Abbildung 9.2 wie folgt verfeinern. Beispielsweise werden von einer Umgebung (einem Programm) Aufträge an einen TCP-Socket übergeben. Diese Auftragsgenerierung auf TCP-Ebene impliziert daraufhin Aufträge in Form von IP-Paketen. Je nachdem, welche Schnittstelle man betrachtet, stellt man einen unterschiedlichen Charakter der Last fest. Daran zeigt sich wieder, dass es wichtig ist, die betrachtete Schnittstelle explizit zu benennen.

Möchte man auch niedrigere Schnittstellen und die an ihnen angebotene Last betrachten, so spricht man häufig von *Primärlast*, *Sekundärlast*, *Tertiärlast* usw. In Abbildung 9.6 bilden die A-Aufträge die Primälast. Sie führen zu den B-Aufträgen, die die Sekundärlast bilden.

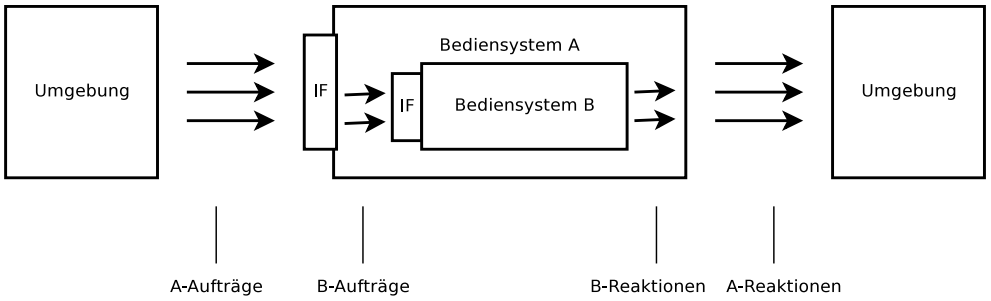


Abbildung 9.6. Primär- und Sekundärlast an einem System A und dessen Subsystem B

Einige grundsätzliche Vorgehensweisen und Fragestellungen bei der Lastmessung sind:

1. Was ist die genaue Motivation und Zielsetzung der Messung?

9.3. z.B. trivialerweise den Wert 0

2. Welche Randbedingungen[9.4] sind festgelegt: Welche Schnittstelle wird betrachtet? Welche Beobachtungsintervalle?

3. Welche Messgrößen (z.B. Pakete, Bytes, Blöcke) werden erfasst?

4. Wie werden die Rohdaten erfasst? Welche zeitliche Auflösung kann die verwendete Uhr bieten? Beeinflusst die Messung den betrachteten Vorgang?

5. Wie werden die Rohdaten ausgewertet? Wie werden sie grafisch aufbereitet?

6. Welche Interpretationen der Messdaten sind möglich? Wie können sie weiterverarbeitet werden?

9.4 Lastmodellierung für Benutzer

Einige Vorschläge für eine formale Modellierung der durch Benutzer genierten Last haben wir im Abschnitt 9.2 gemacht. In diesem Abschnitt wollen wir die einzelnen Schritte einer solchen Modellierung betrachten. Ein mögliches Procedere bei der Lastmodellierung ist das folgende:

1. Dekomposition von System und Umgebung: Zerlege das Gesamtsystem in Bediensystem S und Umgebung U mit einer wohldefinierten Schnittstelle IF zwischen ihnen. Entscheide, welche Aufträge zu berücksichtigen sind. Zerlege die Umgebung U in eine Menge von *lastgenerierenden Benutzern* $B_1, ..., B_n$.

2. Wahl des Detaillierungsgrades: Bilde eine Menge von *Auftragstypen*. Lege für jeden Auftragstyp eine Menge von *Auftragsattributen* fest.

3. Beschreibung der möglichen Interaktion: Spezifiziere für die gegebene Schnittstelle IF die möglichen Sequenzen der Interaktion zwischen S und U (vgl. Dienstspezifikation).

4. Beschreibung der tatsächlichen Interaktion: Bilde jeden Benutzer auf einen *individuellen Lastgenerator* ab, der einen Auftragstrom $S_R = ((t_1, R_1), (t_2, R_2), ..., (t_m, R_m))$ generiert. Dabei bezeichnet t_i den Ankunftszeitpunkt des Auftrags R_i an der Schnittstelle IF. Spezifiziere die Zeitpunkte t_i deterministisch durch einen vorhandenen Trace oder stochastisch durch Wahrscheinlichkeitsverteilungen.

9.4. vgl. Defintion 9.1

Kapitel 10

Netzmanagement, -optimierung und Netzanalyse

Damit Rechnernetze und verteilte Systeme den Anforderungen genügen, die an sie gestellt werden, ist das *Management* eine wichtige Aufgabe beim Betrieb von Rechnernetzen. Zu dieser Aufgabe gehört u.a. das Systemmanagement (die Verwaltung der Endgeräte) und das Netzmanagement[10.1]. Letztere Aufgabe bildet den Inhalt von Abschnitt 10.1.

Die angesprochenen Anforderungen lassen sich in Anforderungen der Benutzer und der Betreiber aufteilen. Sie können durchaus gegensätzlich sein.

- Benutzer stellen Anforderungen in Bezug auf Funktion, Leistung, Zuverlässigkeit, Sicherheit und Kosten.
- Betreiber wünschen einen wirtschaftlichen Betrieb, eine effiziente Nutzung der Ressourcen und (normalerweise) eine hinreichende Zufriedenheit der Benutzer.

Abschnitt 10.2 skizziert und klassifiziert einige Optimierungsaufgaben im Kontext von Rechnernetzen und verteilten Systemen. Netzanalysen basierend auf Mess- und Modellierungsmethoden sind Thema der Abschnitte 10.3 bis 10.5.

10.1 Aspekte des Netzmanagement

Wir beginnen mit einer recht breiten Definition des Begriffs Netzmanagement. Es folgt eine Tabelle mit wichtigen Lösungen für das Netzmanagement. Das im Internet gebräuchliche Protokoll SNMP werden wir genauer betrachten.

Definition 10.1. *Unter* Netzmanagement *verstehen wir die Summe aller Verfahren und Produkte zur Planung, Konfiguration, Steuerung, Überwachung, Fehlerbehebung sowie zur Verwaltung von Rechnernetzen und verteilten Systemen.*

Kontext	Lösung
Internet	SNMP (Simple Network Management Protocol)
ISO/OSI	OSI Management Framework
CCITT	CMIS/CMIP (Common Management Information Service/Protocol)

Tabelle 10.1. Einige wichtige Lösungen für das Management

10.1. auch: Netzwerkmanagement

Konfigurationsmanagement

Ziele des Konfigurationsmanagements sind die Registrierung des Bestandes und des Status aller Objekte eines Rechnernetzes sowie die Registrierung der strukturbestimmenden Zusammenhänge zwischen diesen Objekten. Typische Aufgaben sind das häufig automatisierte Fortschreiben der Konfigurationsdatenbasis, die Änderung der Netzkomponenten bei Fehlern oder bei Erweiterungen des Netzes. Desweiteren gehören die Versionsverwaltung und Auftragsverfolgung zu den Teilaufgaben des Konfigurationsmanagements. In Abbildung 10.1 sind die Objekte und Zusammenhänge nach [Gar 91] dargestellt.

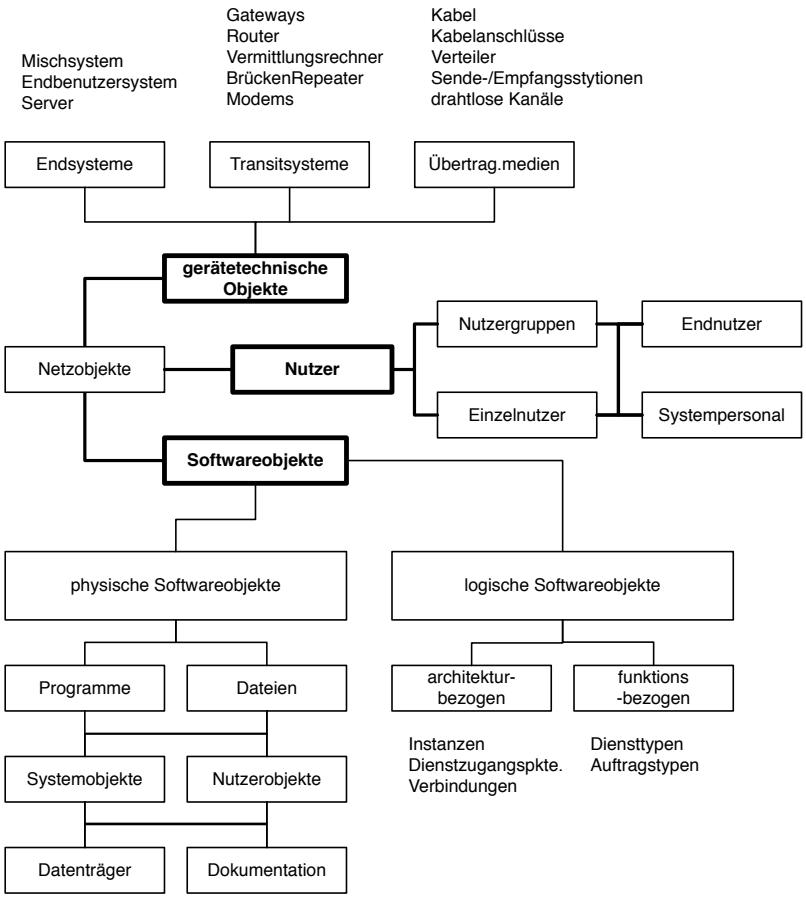


Abbildung 10.1. Objekte des Konfigurationsmanagements

Fehlermanagement

Die Behandlung von anomalen Situationen im Rechnernetz, insbesondere die Realisierung von Überwachungs-, Analyse- und Steuerfunktionen zur Sicherstellung der Verfügbarkeit der Netzfunktionen bilden die Ziele des Fehlermanagements. Typische Teilaufgaben sind die Überwachung, die Verarbeitung von Alarmen, die Fehlerdiagnose sowie die Behebung von Fehlern und das Verhindern einer Fortpflanzung von Fehlern. Häufig werden als "Ticket Systeme" bezeichnete Informationssysteme verwendet, um gemeldete Fehler und den Status ihrer Bearbeitung zu verwalten.

Leistungsmanagement

Das Leistungsmanagement umfasst die Überwachung und Beeinflussung leistungsrelevanter Parameter, insbesondere die Beiseitigung von Überlastungssituationen, die Gewährleistung von geforderten Dienstgüten, die Feineinstellung (tuning) von Netzparametern und die Prognose des zukünftigen Leistungsverhaltens. Im Rahmen des Leistungsmanagements werden Messpunkte und

Messverfahren spezifiziert, mit denen Leistungs- und Lastmessungen durchgeführt werden. Durchgeführte Messungen werden statistisch und grafisch aufbereitet und dienen zur Ableitung und Durchführung von leistungsverbessernden Maßnahmen. In den folgenden Abbildungen sind die Aktivitäten beim Fehler- (siehe Abb. 10.2) und Leistungsmanagement (siehe Abb. 10.3) nach [Gar 91] dargestellt.

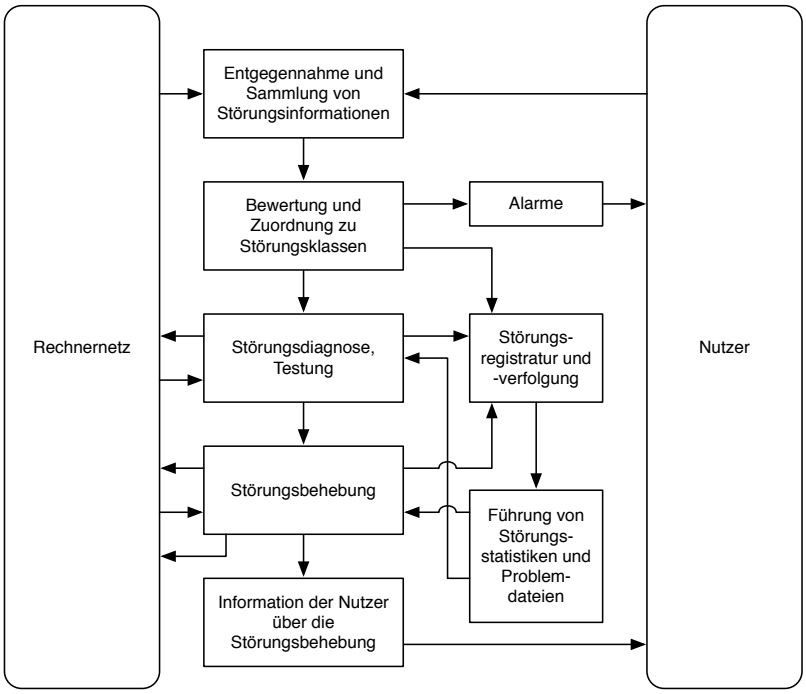


Abbildung 10.2. Aktivitäten beim Fehlermanagement

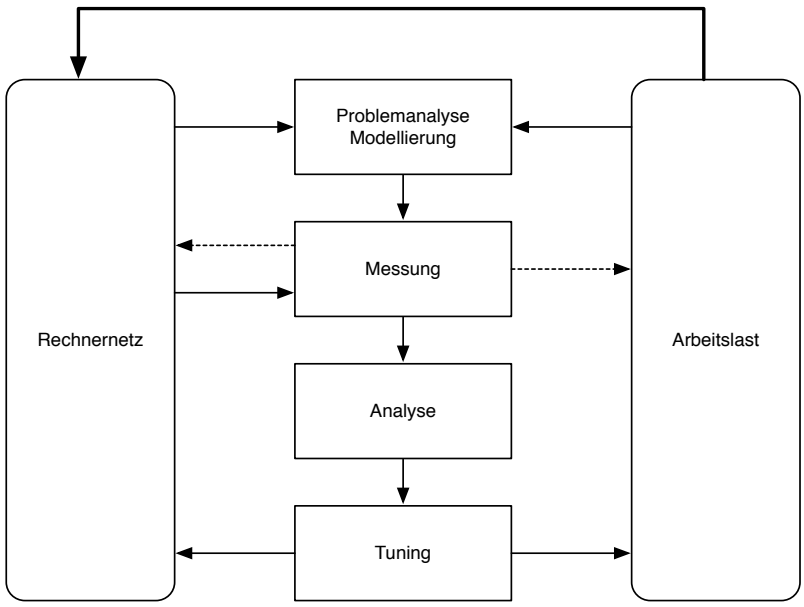


Abbildung 10.3. Aktivitäten des Leistungsmanagements

Abrechnungsmanagement

Eine benutzerbezogene Erfassung der seitens des Netzes bereitgestellten Dienste sowie der beanspruchten Betriebsmittel sowie die Schaffung einer verursachungsgerechten Kostenbelastung für die Nutzer bildet das Abrechnungsmanagement. In diesem Rahmen werden Verbrauchsdaten erfasst, Abrechnungskonten geführt und Zuordnungen der Kosten zu den Konten gebildet. Auch die Verteilung und Überwachung von Kontingenten sowie das Führen von Verbrauchsstatistiken gehört zum Abrechnungsmanagement.

Sicherheitsmanagement

Ziel des Sicherheitsmanagement ist der Schutz gegen gewollte (im Gegensatz zu Störungen), unzulässige Einwirkungen auf das System und dessen Aufträge, z.B. zur Verhinderung von Abhörvorgängen, zur absichtlichen Überlastung, zur Fälschung von Daten, zur Verschleierung von Identitäten und zur unbefugten Inanspruchnahme von Netzdiensten. Die in diesem Rahmen ergriffenen Maßnahmen gliedern sich in bau- und versorgungstechnische, organisatorische, technologische sowie in programm- und gerätetechnische Aktivitäten.

Die Aktivitäten beim Abrechnungsmanagement und die Einflussfaktoren für die Datensicherheit sind nach [Gar 91] in den folgenden Abbildungen 10.4 und 10.5 dargestellt.

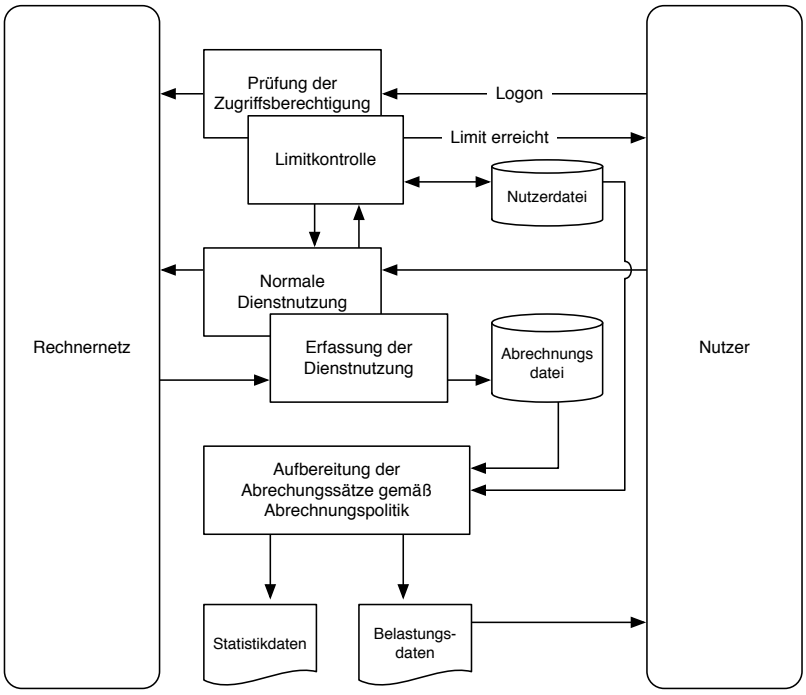


Abbildung 10.4. Aktivitäten des Abrechnungsmanagements

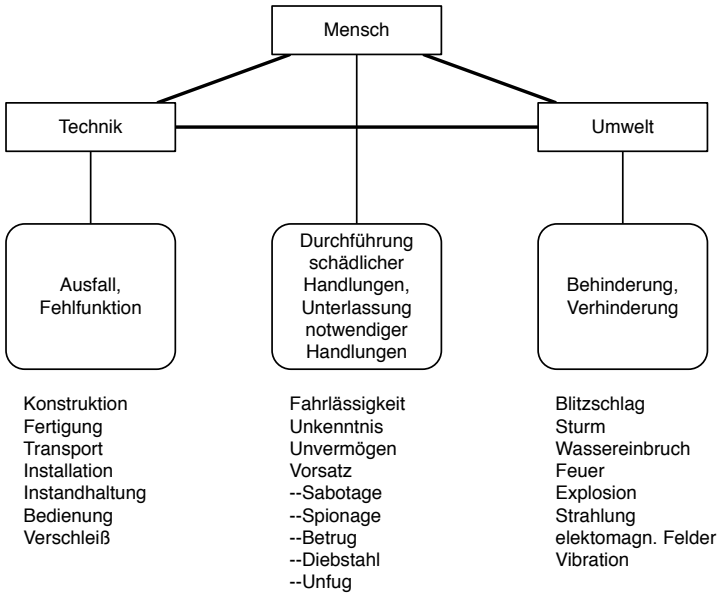


Abbildung 10.5. Einflussfaktoren für die Datensicherheit

Simple Network Management Protocol

Das Simple Network Management Protocol (SNMP) ist ein im Internet verbreitetes Protokoll der Anwendungsschicht für die Verwaltung und Überwachung von Netzkomponenten wie Router, Server, Switches. Es wird auf der Grundlage von UDP/IP realisiert. Abbildung 10.6 zeigt die grobe Architektur für SNMP. Auf einer *Management-Station* läuft ein Manager-Prozess und koordiniert die Komponenten. Dieser Prozess erhält von den *Agenten* Meldungen über den Zustand der *Managed Objects (MO)*, wie Router, Server, Endsysteme. Diese Zustandsinformationen werden in einer verteilten Datenbank, der *Management Information Base (MIB)*, gehalten.

Über den SNMP-Dienst kann der Manager neue "Managed Objects" erstellen oder vorhandene löschen. Er kann die den Managed Objects zugeordneten Attribute lesen und schreiben und von ihnen bereitgestellte Aktionen initiieren. Die Agenten können die Management-Station mit Hilfe von Meldungen über Ausnahmesituationen informieren.

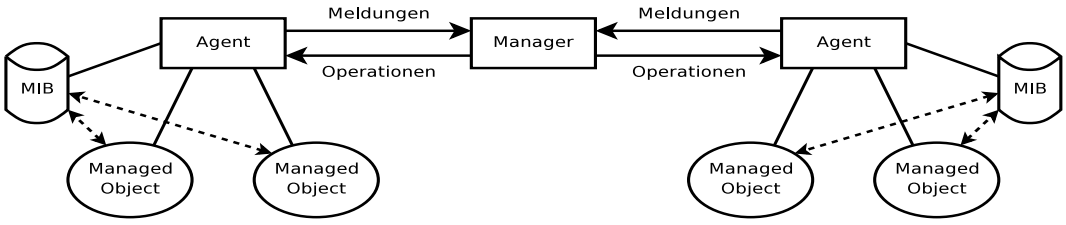


Abbildung 10.6. Grobarchitektur bei SNMP

Die folgende Tabelle gibt eine Übersicht über die verwendeten Protokolldateneinheiten und deren Semantik. Die Syntax der Protokolldateneinheiten ist nach [KrR 02] in Abbildung 10.7 dargestellt.

	PDU-Typ	Bedeutung
1	GetRequest	Abfrage des Werts eines Attributs eines oder mehrerer MO
2	GetNextRequest	Abfrage des lexikographisch nächsten Attributs oder MO
3	SetRequest	Setzen des Wertes eines neuen oder vorhandenen Attributs
4	GetResponse	Antwort auf sämtliche PDU-Typen des Managers
5	Trap	Meldung einer Ausnahmesituation

Tabelle 10.2. PDU-Typen bei SNMP vom Manager zum Agenten (1-3) und vom Agenten zum Manager (4, 5)

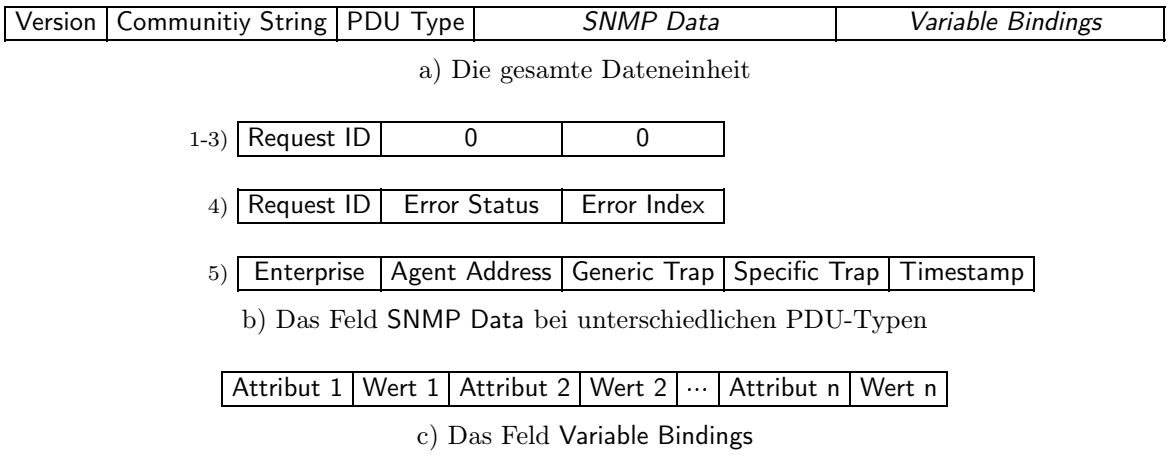


Abbildung 10.7. Syntax der Protokolldateneinheiten bei SNMP

Einige der Datenfelder wollen wir hier erläutern. Das Feld Version gibt die Versionsnummer[10.2] des Protokolls an. Community String ist ein geheimer Bezeichner für einen Agenten. Allerdings wird dieser im Klartext übertragen, was ein Grund für die Entwicklung von weiteren Versionen ist. Typische Ausnahmesituationen werden durch das Feld Generic Trap dargestellt. Die unterschiedlichen Situationen sind in Tabelle 10.3 dargestellt.

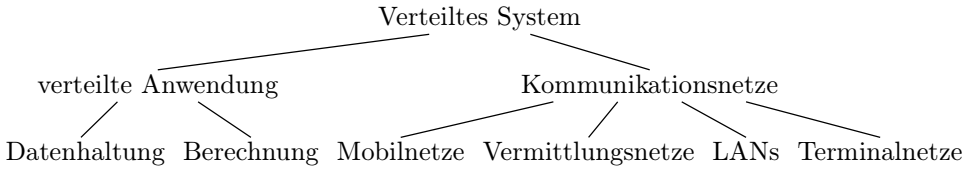
Wert	Bezeichner	Bedeutung
0	`coldStart`	Neustart einer Komponente nach Konfigurationsänderung
1	`warmStart`	Neuinitialisierung einer Komponente
2	`linkDown`	Ausgehende Verbindung nicht mehr verfügbar
3	`linkUp`	Ausgehende Verbindung wieder verfügbar
4	`authFailure`	Fehlerhafter CommunityString
5	`egbNeighborloss`	Fehlende Konnektivität
6	`enterpriseSpecific`	Herstellerspezifischer Eintrag im Feld Specific Trap

Tabelle 10.3. Ausnahmesituationen im Feld Generic Trap

10.2 Optimierung im Bereich Kommunikationsnetze und Verteilte Anwendungen

Während der Planung und Umkonfigurierung von Systemen mit verteilter Verarbeitung oder verteilter Datenhaltung erfolgt in der Regel eine Suche einer optimalen oder suboptimalen Konfiguration, bei der entweder eine maximale Leistungsfähigkeit bei limitierten Kosten oder eben minimale Kosten bei der Gewährleistung einer Mindestleistung angestrebt werden. Dabei sind in der Regel eine sehr große Zahl von Konfigurationsvarianten möglich. Um eine effiziente Bewertung durchzuführen, müssen entweder bestimmte Varianten möglichst schnell ausgeschlossen werden oder die Bewertungen auf der Grundlage von sehr groben Modellen erfolgen. Die zu berücksichtigenden Komponenten lassen sich wie folgt klassifizieren.

10.2. Zur Zeit exitstieren drei Versionen.



Die Abbildung 10.8 zeigt eine mögliche Notation für eine Konfigurierung eines Rechnernetzes. Sie zeigt Arbeitsrechner (AR), Vermittlungs- und Gatewayrechner (VR, GW), Terminals und Terminalkonzentratoren (T, TK), die Module einer verteilten Anwendung (M) und die von ihnen verwendeten Datenobjekte (F). Durch gestrichelte Kästen sind vier Teilbereiche markiert, für die eine optimale Konfigurierung gefunden werden soll.

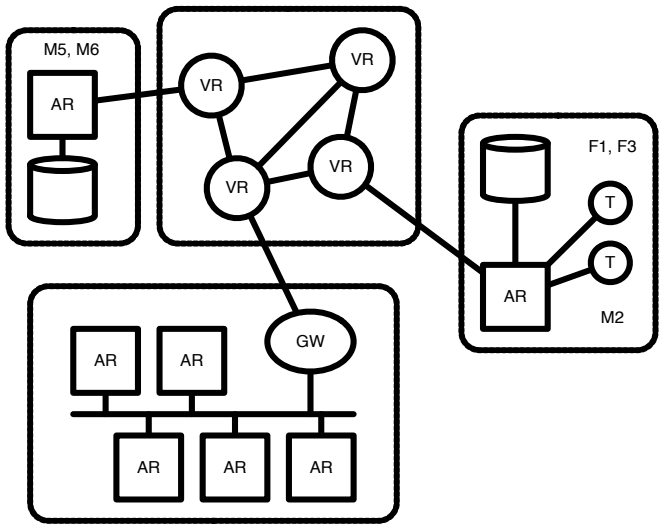


Abbildung 10.8. Exemplarisches Konfigurationsmodell

10.3 Ziele und Methoden der Netzanalyse

Sowohl die Entwickler von Netzkomponenten (in der Forschung und der Industrie) als auch die Betreiber von Rechnernetzen (Telefongesellschaften, Internet Service Provider etc.) haben ein Interesse an Methoden zur Analyse und Bewertung neuer Produkte, Algorithmen, Protokolle und Technologien. Entwickler benötigen sowohl in der Entwurfsphase als auch während der Realisierung und Evaluierung Informationen – u.a. bezüglich der zu erwartenden Leistungsfähigkeit und Zuverlässigkeit eines zu entwickelnden Systems. Betreiber großer Netze benötigen solche Informationen bei der Konfigurierung ihrer Komponenten und bei ihrer Einkaufsstrategie. Es folgt eine Liste mit typischen Bewertungskriterien bei der Netzanalyse:

- Leistungsfähigkeit (performance)
- Verlässlichkeit (dependability)
- Zuverlässigkeit (reliability)
- Sicherheit (safety/security[10.3])
- Verfügbarkeit (availability)

10.3. "Safety" steht für den Schutz vor Datenverlust, "Security" für den Schutz vor unbefugtem Zugriff.

- Wartbarkeit (maintainability)
- Wirtschaftlichkeit (economy)

Im folgenden liegt der Fokus auf der Leistungsfähigkeit. Diese ist unter anderem durch Zeitgrößen wie *Bedienzeiten*, *Verzögerungszeiten*, *Verweilzeiten* und *Antwortzeiten* für übertragene Dateneinheiten und ausgeführte Prozesse sowie Zählgrößen wie *Durchsatz*, *Auslastung*, *Ankunftsrate* und *Bedienrate* quantifizierbar. Analysen der Leistungsfähigkeit können auf unterschiedliche Weise erfolgen:

a) durch direkte Messungen am Realsystem
   - mit Hilfe spezieller Hardware (Hardware-Monitore)
   - mit Hilfe von Messprogrammen (Software-Monitore)
   - durch Kombinationen aus Hardware und Software (Hybrid-Monitor)

b) durch indirekte Messungen an Ersatzsystemen (insbesondere an Modellen)
   - mit Hilfe von analytischen Methoden
   - mit Hilfe von Simulationen
   - durch Kombinationen analytischer und simulativer Methoden

c) durch Kombinationen von Modellen und Realsystemen

Die folgende Abbildung zeigt zwei typische Beispiele für Kombinationen von Modellen und Realsystemen bei der Netzanalyse.

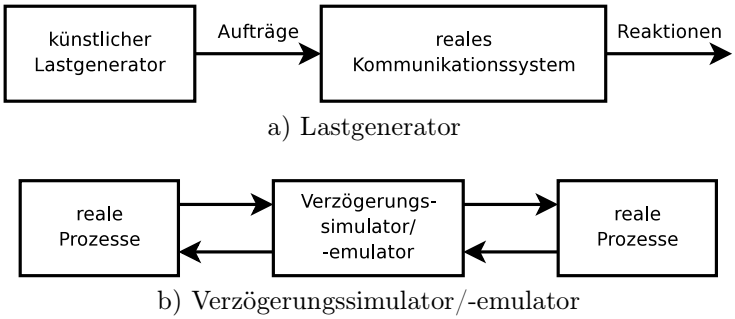


Abbildung 10.9. Kombinationen von Modellen und Realsystemen

10.4 System- und Leistungsmessungen

Wir unterteilen nun Messungen bei der Netzanalyse in drei Klassen, von denen die ersten beiden den Inhalt dieses Abschnittes bilden.

Systemmessungen. Sie beziehen sich auf solche Kenngrößen des betrachteten Netzes, die lastunabhängig ermittelt werden können. Beispiele sind Auftragsbedienzeiten und die grundsätzliche Leistungsfähigkeit der Hardware.

Leistungsmessungen. Hierbei beziehen sich die Messungen auf das Netzverhalten unter einer speziellen Last. Diese kann real sein oder künstlich generiert werden.

Lastmessungen. Sie beziehen sich auf die an der beobachteten Schnittstelle angebotene Last, also der benutzergenerierten Sequenz von Aufträgen. Diese Klasse haben wir im vorigen Kapitel besprochen.

Wie im vorigen Kapitel folgt eine Grobbeschreibung der Vorgehensweise bei solchen Messungen. Dabei sind wieder einige wichtige Fragestellungen angegeben.

1. Festlegung der Ziele: Welche Arten von Erkenntnissen sind angestrebt?
2. Festlegung der Randbedingungen: Welche Netzinfrastruktur wird betrachtet? Welche Limitationen besitzen verwendete Modelle?
3. Realisierung der Werkzeuge: Welche Mess-Monitore (Hardware- oder Softwaremonitore) werden eingesetzt?
4. Experimentvorbereitung: Installation der Messwerkzeuge, Erzeugung künstlicher Last.
5. Durchführung: Erfassung, Speicherung und ggf. Filterung der Messdaten (Rohdaten).
6. Aufbereitung: grafische Aufbereitung, statistische Auswertung, Histogramme etc.
7. Interpretation: Deutung der Ergebnisse. Sind weitere Messungen nötig?

Allgemeine Probleme bei derartigen Messungen sind der oft hohe Aufwand – insbesondere dann, wenn für eine statistische Auswertung große Mengen an Rohdaten gesammelt werden müssen – und die eventuell auftretenden Störungen der Abläufe durch deren Messung. Abgesehen davon sind Messungen in der Entwurfsphase oder bei fehlenden Informationen über das Benutzerverhalten oft unmöglich.

Spezielle Probleme bei der Messung in Kommunikationsnetzen sind der "Black Box"-Charakter vieler Netze. Bei ihnen sind viele Attribute der internen Komponenten nicht messbar. Außerdem sind Randbedingungen (wie die Gesamtlast) schwer oder gar nicht erkennbar. Darüber hinaus ist eine Synchronisation der verwendeten Uhren nötig, um Zeitgrößen zu messen. Zuletzt können Messungen in Rechnernetzen eine Beeinträchtigung der Sicherheit bedeuten, wenn z.B. unverschlüsselter Verkehr "abgehört" und aufgezeichnet wird.

10.5 Modellierung von Kommunikationsnetzen

Netzmodelle stellen eine wichtige Möglichkeit zur Analyse von Rechnernetzen dar. Sie werden aus unterschiedlichen Gründen erstellt, von denen wir einige schon genannt haben:

- Das System befindet sich in der Entwurfsphase, Messungen sind unmöglich.
- Das System ist für die gewünschten Messungen unzugänglich.
- Reale Messungen sind zu aufwändig, langwierig oder kostspielig.
- Es sollen Randbedingungen variiert werden, die nicht zugänglich sind.

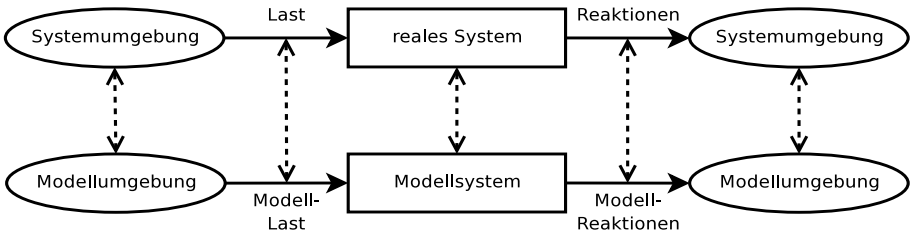


Abbildung 10.10. Korrespondierende Bereiche bei Modell und realem System

Die zentralen Fragen bei der Modellierung, insbesondere beim Modellentwurf, sind: Welcher Ausschnitt der Realwelt (bestehend aus System und Umgebung) soll modelliert werden? Mit welchem Detaillierungsgrad soll modelliert werden?

Mögliche Phasen eines Modellierungsprozesses sind in der folgenden Abbildung angegeben. Dabei ist zu beachten, dass der Modellierungsprozess kein strikt sequentieller Prozess sein muss. Häufig ist ein iteratives Vorgehen, also ein wiederholtes Durchlaufen von bestimmten Schritten sehr sinnvoll.

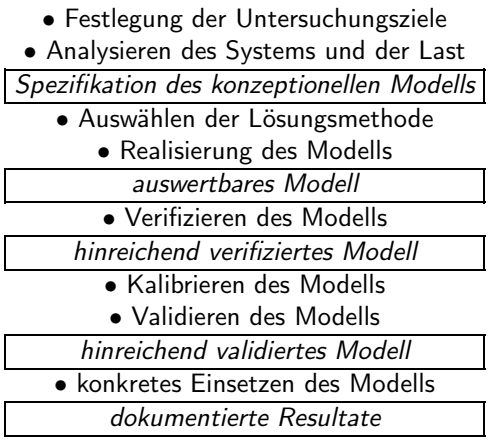


Abbildung 10.11. Ein Phasenmodell der Modellierung

10.5.1 Wartesysteme

Nachdem wir in Abschnitt 10.3 bereits einige elementare Leistungskenngrößen zur Leistungsbewertung eingeführt haben, wollen wir uns der Frage zuwenden, welche Art von Modellen sich für die Leistungsbewertung von Kommunikations- und Rechnernetzen besonders gut eignen.

Die bei weitem meisten der existierenden Leistungsmodelle für Rechensysteme und Rechnernetze basieren auf *Wartesystemen* und *Wartenetzen*, auf die wir uns in der Folge daher konzentrieren werden.

Ein Wartesystem (queueing system) W besteht aus einer Bedienstation (BS) mit zugeordneter Warteschlange Q. Die Bedienstation selbst umfasst m Bediener $B_1, B_2, ..., B_m$ mit $m \geq 1$. Im Falle $m = 1$ sprechen wir auch von einem *elementaren Wartesystem*.

Ein Wartesystem wird charakterisiert durch eine Beschreibung:

- des Ankunftsstroms von Aufträgen, d.h. die Menge der Aufträge, die über der Zeit in die Warteschlange Q eingefügt werden und die dort auf ihre Bedienung/Bearbeitung durch einen Bediener warten,
- der geforderten Auftragsbedienzeiten,
- der Anzahl m vorhandener Bediener,
- der Bedienstrategie, d.h. der Art und Weise, wie Aufträge aus Q für eine Bedienung ausgewählt werden (u.a. Festlegung der Bedienreihenfolge),
- der maximalen Warteschlangenlänge $N \in \mathbb{N}$.

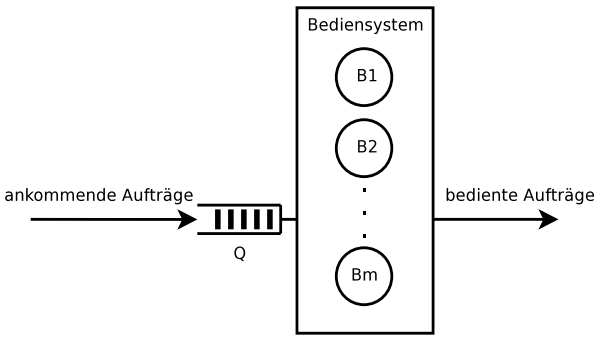


Abbildung 10.12. Grafische Darstellung eines Wartesystems mit der Warteschlage Q und m Bedienern

Um Wartesysteme in sehr kompakter Weise zu charakterisieren, hat sich die sogenannte *Kendall-Notation* etabliert.[10.4] In Kendall-Notation wird ein Wartesystem W in folgender Weise ausgedrückt.

$$\boldsymbol{W = A/B/m/N - s}$$

- $\boldsymbol{A}$: Verteilung der Auftrags-Zwischenankunftszeiten. Beispiele für Verteilungen sind

  M : Exponentialverteilung (M steht für Markov),
  E_k : Erlangverteilung mit k Phasen,
  D : Deterministische Verteilung,
  G : Beliebige Verteilung (G für general).

- $\boldsymbol{B}$: Verteilung der Auftrags-Bedienzeiten. Dieselben Verteilungen wie bei $\boldsymbol{A}$ sind möglich.
- $\boldsymbol{m}$: Anzahl der Bediener
- $\boldsymbol{N}$: maximale Anzahl von Aufträgen im Wartesystem (Systemkapazität c: maximale Warteschlangenlänge plus Anzahl Bediener, d.h. $c = N + m$)
- $\boldsymbol{s}$: Bedienstrategie. Beispiele sind unter anderem:

  FIFO, *FCFS* (First-In-First-Out, First-Come-First-Served) : Die Aufträge werden in der Reihenfolge ihrer Ankunft bedient.
  LIFO, *LCFS* (Last-In-First-Out, Last-Come-First-Served) : Der zuletzt ankommende Auftrag wird als nächster bedient.
  SIRO : (Service-In-Random-Order)
  RR (Round Robin) Reigen : vorgegebene Zeitscheiben für die Bedienung.

Beispiel 10.2. Ein Beispiel für die Notation eines Wartesystems mit zwei Bedienern, einer Warteschlange für zehn Aufträge, die nach der Strategie "First-Come-First-Served" mit beliebiger Verteilung der Bedienzeiten abgearbeitet werden und deren Eintreffen mit der Exponentialverteilung dargestellt wird, lautet dann:

$$\boldsymbol{W = M/G/2/10 - FCFS}$$

10.5.2 Wartenetze

Da Aufträge in komplexen Systemen (wie Rechnernetze) nicht nur durch eine Bedienstation bedient bzw. bearbeitet werden, bietet es sich an, Wartesysteme zu sog. Wartenetzen zu verallgemeinern, um so auch die Modellierung komplexerer Bedienvorgänge zu unterstützen. Dabei ist ein *Wartenetz* (queueing network) eine Menge untereinander verbundener Wartesysteme.

Für Wartesysteme und Wartenetze lassen sich typische Leistungskenngrößen berechnen wie Durchsatz (z.B. betrachtet an Übergabepunkten zwischen direkt benachbarten Wartesystemen eines Wartenetzes), Auslastung (z.B. von Bedienern eines Wartesystems) oder Auftragsverzögerung (z.B. in einem Wartesystem oder in einem offenen Wartenetz zwischen Zu- und Abgangszeitpunkt von Aufträgen).

Zur Berechnung der Leistungskenngrößen eines Wartesystems oder Wartenetzes können zum einen mathematisch-analytische Berechnungsmethoden herangezogen werden, z.B. wenn ein formelmäßiger Zusammenhang zwischen den Kenngrößen eines speziellen betrachteten Wartesystems/Wartenetzes und den gesuchten Leistungskenngrößen bekannt ist. Man spricht hier von *analytischer Modellauswertung*. Alternativ kann ein Programm (Simulationsprogramm) die Auftragsbedienung in einem Wartenetz z.B. durch rechnergestützte Simulation nachvollziehen und auf diese Weise gesuchte Leistungskenngrößen ermitteln. Man spricht hier von *simulativer Modellauswertung*.

10.4. zurückgehend auf D.G. Kendall

Eine kleine Fallstudie

Im folgenden soll die Modellierung eines ATM-Vermittlungsrechners als Fallbeispiel dienen. Üblicherweise hat die angestrebte Auswertetechnik (analytisch oder simulativ) einen starken Einfluss auf die möglichen Modelle. Bei analytischen Methoden sind allzu komplexe System-Modelle und Last-Modelle in der Regel nicht akzeptabel. In Abbildung 10.13 ist zunächst der zu modellierende Vermittlungsrechner (bereits als Modell) dargestellt. Er besitzt Eingangsprozessoren (EP) und Ausgangsprozessoren (AP) für die Eingangs- bzw. Ausgangsleitungen. Diese sind über einen systeminternen Bus miteinander verbunden.

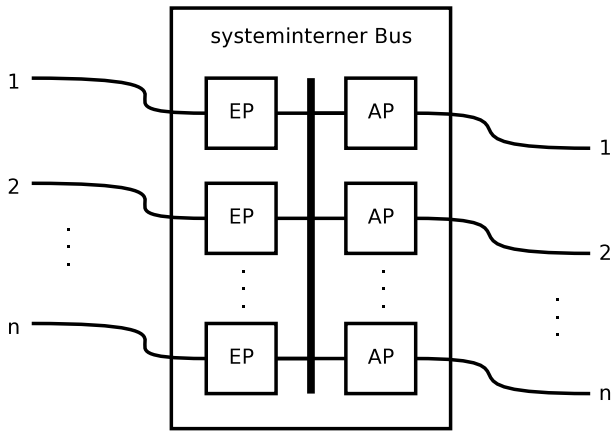


Abbildung 10.13. Vereinfachter ATM-Vermittlungsrechner

Die folgenden Abbildungen zeigen nun Modelle unterschiedlicher Granularität in Form von grafisch notierten Wartenetzmodellen. In Abbildung 10.14 ist der Vermittlungsrechner in sehr einfacher Weise mit einem Bediener und einer Warteschlange modelliert. In der Kendall-Notation könnte das Modell z.B. als $\boldsymbol{W_1 = M/G/1}$ oder $\boldsymbol{W_1 = D/G/1}$ interpretiert werden.

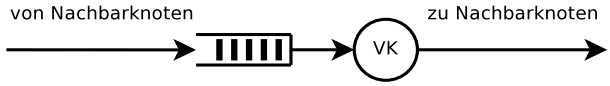


Abbildung 10.14. Elementares Vermittlungsrecher-Modell

Ein Modell für einen sehr leistungsfähigen Vermittlungsrechner, der zwar für jeden Auftrag eine konstante Bedienzeit $t_0 > 0$ benötigt, jedoch jeden Auftrag sofort bedient, ist in Abbildung 10.15 dargestellt. Dieses Modell entspricht der Spezifikation $\boldsymbol{W_2 = G/D/\infty}$.

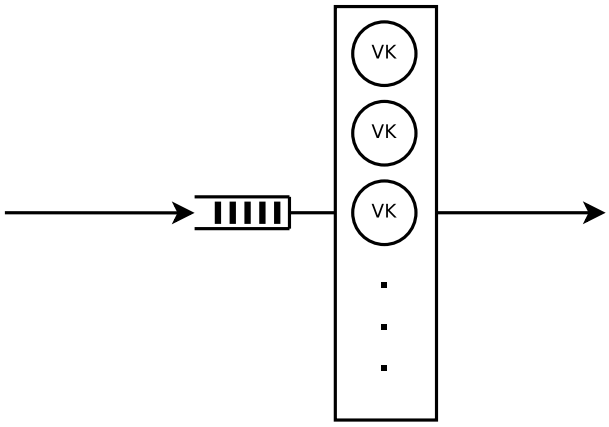


Abbildung 10.15. Sehr leistungsfähiger Vermittlungsrechner (infinite server)

Unser letzter Modellierungsversuch bildet den Vermittlungsrechner deutlich detaillierter ab. Hier wird der Tatsache Rechnung getragen, dass Aufträge unterschiedliche Komponenten des Systems in Anspruch nehmen und somit mehrfach Bedienzeiten hinnehmen müssen.

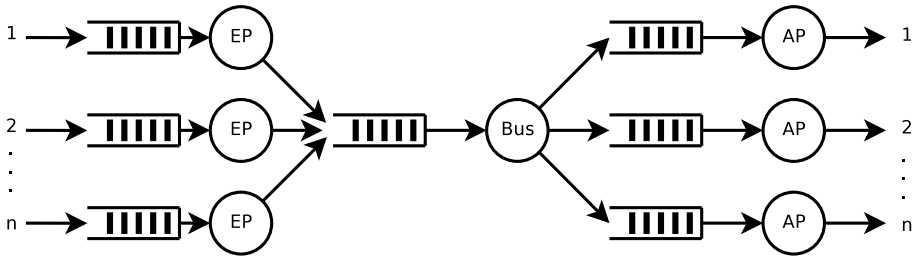


Abbildung 10.16. Berücksichtigung der Eingabe- und Ausgabeprozessoren

In diesem Fall liegt ein Wartenetz vor, das durch mehrere Wartesysteme in Kendall-Notation beschrieben werden kann. Für dieses Wartenetz sind sowohl analytische als auch simulative Modellauswertungen möglich, um beispielsweise günstige Parametrisierungen der einzelnen Wartesysteme zu finden.

Kapitel 11
Netzsicherheit

Das Thema Sicherheit ist für Rechnernetze sehr wichtig und wird durch die wachsende Nutzung des globalen Internet für kritische Bereiche wie Zahlungsverkehr immer bedeutender. In Anbetracht der zeitlichen Restriktionen dieser Vorlesung wird das Thema allerdings nur rudimentär behandelt. Weitere Details sind den Lehrbüchern und einschlägigen Veranstaltungen des Arbeitsbereichs SVS zu entnehmen.

11.1 Datenschutz und Datensicherung

Im englische Sprachraum werden die Begriffe “Security“ für Geheimhaltung oder Zugangskotrolle und “Safety“ für Gefahrlosigkeit oder Schutz vor Verlust oder Verfälschung verwendet. Im Deutschen muss in beiden Fällen meist das Wort “Sicherheit“ herhalten. Um die recht nützliche Unterscheidung der englischen Sprache nicht aufzugeben, werden oft folgende Begriffe benutzt:

Datenschutz. Identifizierbare Daten über persönliche oder sachliche Verhältnisse von Personen oder Organisationen werden vor unbefugtem Zugriff geschützt.

Datensicherung. Ein Schutz vor Zerstörung, Verfälschung oder Wertverlust von Daten durch technische Fehlfunktionen und unbeabsichtigte oder böswillige Handlungen von Personen. Im engeren Sinne bezieht sich Datensicherung auf den Schutz der Daten vor Verfälschungen beim Transport zwischen zwei direkt miteinander verbundenen Kommunikationspartnern. Wir verwenden den Begriff aber allgemeiner.

Netzsicherheit umfasst dabei Datenschutz und Datensicherung. Dabei ist es das Ziel Angriffe bei der Übertragung und während der Zwischenspeicherung durch geeignete Maßnahmen zu vermeinden. Solche Angriffe können durch legitime Benutzer mit Fehlverhalten oder durch Personen ohne jegliche Befugnis geschehen und bestehen in der Verletzung von Sicherheitsanforderungen. Diese Anforderungen, die an Netzsicherheit gestellt werden, sind häufig wie folgt klassifiziert:

Vertraulichkeit. Objekte wie Daten und Programme können nur von den dazu berechtigten Personen gelesen, vervielfältigt oder verwendet werden. Häufig wird auch von Geheimhaltung gesprochen.

Integrität. Objekte befinden sich in einem korrekten Zustand. Sie wurden nicht unbefugt oder ungewollt verändert.

Verfügbarkeit. Objekte können von Personen (im Rahmen ihrer Berechtigung) ohne Einschränkung verwendet oder verändert werden.

Nichtabstreitbarkeit. Aktionen, Nachrichten, Dokumente u.a. können eindeutig einer Person als Erzeuger zugeordnet werden. Diese Person kann ihre Rolle nicht abstreiten. Sender und Empfänger können die Identität des jeweils anderen im Nachhinein beweisen.

Authentizität. Die Identität eines Benutzers oder Systems ist sichergestellt. Sie kann nicht vorgetäuscht werden.

Während der Übertragung von Daten über Rechnernetze kann deren Sicherheit durch Angreifer beeinträchtigt werden. Angriffsmöglichkeiten bieten sich sowohl bei der Übertragung (besonders bei Bus-, Ring- und Broadcast-Topologie) als auch bei der Zwischenspeicherung der Daten in Vermittlungsrechnern oder Endgeräten. Gründe für derartige Angriffe gibt es viele. Zu ihnen zählen:

- der einfache und (relativ) anonyme Zugriff auf entfernte Daten und Betriebsmittel,
- die zunehmende Menge an wertvollen Informationen, die zwischen kommunizierenden Rechnern ausgetauscht wird,
- Netztechnologien, die bestimmte Angriffe erleichtern (Offene Systeme, Sicherheitslücken)
- Übertragungsmedien, die von sich aus wenig abhörsicher sind (Funkstrecken, Kupferdrähte)
- die große Zahl möglicher Angreifer in gekoppelten Rechnernetzen

Wir unterscheiden Angriffe auf die Netzsicherheit in *passive Angriffe* und *aktive Angriffe*. Bei ersteren wird die Kommunikationsbeziehung nicht direkt beeinflusst. Beispielsweise werden Nachrichteninhalte durch Abhören der Verbindung ermittelt oder der Verkehr zwischen Rechnernetzknoten analysiert. Bei aktiven Angriffen findet eine Veränderung, Löschung, Erzeugung, Umleitung, Wiederholung o.ä. der Daten statt. Beispielsweise wird durch Veränderung einiger Dateninhalte eine falsche Identität vorgetäuscht.

Spezielle Angriffstechniken

Es folgt eine Auflistung von Angriffsmethoden in Rechnernetzen mit einigen typischen Beispielen.

Sniffing. Dabei handelt es sich um ein passives Abhören des Datenverkehrs. Ein solches Abhören ist in Ring-, Bus- und Broadcast-Topologien besonders einfach. Es können aber auch Rechnernetzknoten auf dem Pfad zwischen den Kommunikationspartnern "belauscht" werden.

Man in the Middle. Hierbei wird der Datenverkehr aktiv abgehört. In einem Ethernet können beispielsweise zwei Kommunikationspartner recht leicht (über das Protokoll ARP) davon überzeugt werden, dass es sich bei dem Angreifer um den jeweils anderen Kommunikationspartner handelt. Diese schicken unwissend alle Dateneinheiten an den Angreifer, der diese (nachdem er sie durchsucht, speichert oder verändert) an das tatsächliche Ziel weiterleitet. Sniffing und Man in the Middle werden auch als **passives bzw. aktives Wiretapping** bezeichnet.

Spoofing. Hierunter versteht man das Fälschen der Absenderadresse einer Dateneinheit. Eine fremde Identität wird vorgetäuscht, die eigene verschleiert. Soll allerdings auch der rückwärtige Datenverkehr an den Angreifer gerichtet werden, müssen andere Techniken eingesetzt werden.

Hijacking. Ein Endpunkt einer aktiven Verbindung wird durch den Angreifer übernommen. Meist wird danach die Rolle des ursprünglichen Kommunikationspartners unbemerkt übernommen. Eine solche feindliche "Übernahme" erfordert meist eine recht genaue Analyse des bisherigen Datenverkehrs auf allen darüber liegenden Schichten.

Denial of Service (DoS). Bei solchen Angriffen werden die Betriebsmittel eines Rechners (insbesondere dessen CPU, Haupt- und Sekundärspeicher) gezielt so stark verbraucht, dass der Rechner seine Aufgaben nicht mehr durchführen kann. Häufig wird der "three way handshake" beim Aufbau einer TCP-Verbindung dafür missbraucht. Um sehr leistungsfähige Rechner außer Betrieb zu setzen, werden oft Angriffe von zahlreichen Angreifern[11.1] in Form einer **Distributed Denial of Service (DDoS)** Attacke kombiniert.

Um solchen und anderen Angriffen vorzubeugen, können unterschiedliche Maßnahmen getroffen werden. Der folgende Abschnitt behandelt die erste dieser Maßnahmen.

1. Verschlüsselung der Daten

11.1. In der Regel gehören die Endrechner dabei ahnungslosen Benutzern, deren System kompromittiert wurde.

2. legislative Maßnahmen in Form von Gesetzen
3. organisatorische Maßnahmen wie physische Zugriffskontrolle
4. Identitätskontrolle in Form von Passwörtern, Biometriekontrollen u.ä.
5. Entdeckung von Angriffen (intrusion detection)

11.2 Chiffrierverfahren

Verfahren zur Verschlüsselung von Daten verfolgen zwei grundsätzlich unterschiedliche Vorgehensweisen: *symmetrische Verschlüsselung* und *asymmetrische Verschlüsselung*.

- Bei symmetrischen Verfahren besitzen die Kommunikationspartner A und B einen gemeinsamen Schlüssel S, der zur Verschlüsselung und Entschlüsselung verwendet wird. Bei der Verschlüsselung wird eine Nachricht M in $V_S(M)$ umgewandelt. Wird die Entschlüsselungsfunktion auf die verschlüsselte Nachricht angewendet, gewinnt man die ursprüngliche Nachricht zurück. Es gilt also stets:

  $$E_S(V_S(M)) = M$$

  Der *Data Encryption Standard (DES)* verwendet ein symmetrisches Verfahren. Die wichtigste Maßnahme zur Gewährleistung der Vertraulichkeit ist das Geheimhalten des Schlüssels S. Wenn zwei Kommunikationspartner über ein unsicheres Medium kommunizieren möchten und noch nicht über ein geheimes, gemeinsamen Wissen in Form eines Schlüssels verfügen, müssen sie diesen über einen sicheren Kanal (wie z.B. ein persönliches Treffen) austauschen. Eine Alternative stellen die asymmetrischen Verfahren dar.

- Bei asymmetrischen Verfahren besitzt jeder Kommunikationspartner ein Paar von zwei Schlüsseln $(S, \bar{S})$. Der öffentliche Schlüssel S wird zum Verschlüsseln der Nachricht verwendet, der private Schlüssel $\bar{S}$ zum Entschlüsseln. Es gilt

  $$E_{\bar{S}}(V_S(M)) = M$$

  Der öffentliche Schlüssel wird an alle Kommunikationspartner gegeben, die verschlüsselte Nachrichten senden möchten, der private Schlüssel wird geheimgehalten. Solche Verfahren müssen gewährleisten, dass der private Schlüssel nicht[11.2] aus dem öffentlichen gewonnen werden kann. Ein wichtiger Vertreter dieser Klasse ist das *RSA-Verfahren*.[11.3]

Data Encryption Standard (DES)

Der symmetrische Verschlüsselungsstandard DES ist ein Blockalgorithmus, der Blöcke von 64 Bit Länge mit Hilfe eines 64 Bit langen Schlüssels chiffriert. Acht Bit des Schlüssels sind Paritätsbits, so dass die effektive Schlüssellänge 56 Bit beträgt. Somit gibt es $2^{56} \approx 10^{17}$ verschiedene mögliche Schlüssel. Die Schlüssellänge wird mittlerweile als zu kurz erachtet.

Der Algorithmus erzeugt nach einem festen Schema 16 Unterschlüssel der Länge 48 Bit. Mit Hilfe dieser Unterschlüssel werden in 16 Runden jeweils eine *Transposition* und eine *Substitution* durchgeführt. Diese Begriffe sind in der Kryptologie elementar und sollen daher formal definiert werden.

Definition 11.1. *Sei ein n Zeichen langer Block durch die Funktion b: $\{1...n\} \to A$ dargestellt, wobei A ein Alphabet ist. Eine Bijektion s: $A \to A$, die jedes Zeichen des Alphabets auf ein (nicht notwendigerweise anderes) Zeichen abbildet, heißt* Substitution. *Eine Bijektion t: $\{1...n\} \to \{1...n\}$, die jedem Zeichen eine (nicht notwendigerweise andere) neue Position zuweist, heißt* Transposition.

11.2. oder nur mit gewaltigem Aufwand

11.3. benannt nach seinen Entwicklern Rivest, Shamir und Adleman

Beispiel 11.2. Wir betrachten einen Block der Länge $n = 8$ mit Zeichen aus dem Alphabet $A = \{\texttt{a},\texttt{b},\texttt{c},\texttt{d}\}$ und die folgenden Funktionen

$$s(a) = c, \quad s(b) = b, \quad s(c) = d, \quad s(d) = a$$

$$t(1) = 1, \quad t(2) = 4, \quad t(3) = 3, \quad t(4) = 7, \quad t(5) = 6, \quad t(6) = 2, \quad t(7) = 5, \quad t(8) = 8$$

Wendet man die Substitution auf den Block `abbdacba` an, so erhält man `cbbacdbc`. Die Transposition des (substituierten Blocks) führt zu `cdbbbcac`.

Der genaue und recht komplizierte Ablauf bei DES ist in den folgenden Abbildungen dargestellt. Grob betrachtet, wird die Eingangsbitfolge in zwei Hälften geteilt. Die zweite Hälfte des Eingangsblocks wird zur ersten Hälfte des Ausgangsblocks. Die zweite Hälfte des Ausgangsblocks ergibt sich aus dem Eingangsblock und den Unterschlüsseln.

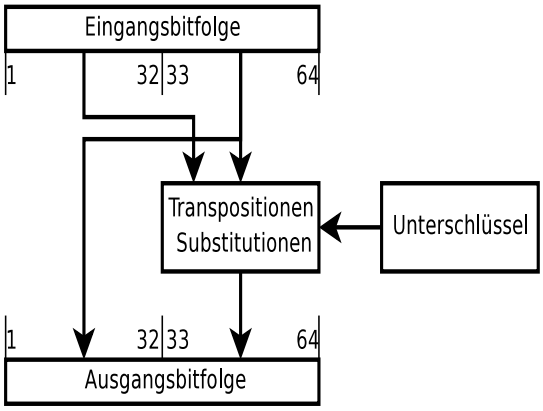


Abbildung 11.1. Ablauf einer einzelnen Runde bei DES

Eine etwas genauere Darstellung liefert die folgende Abbildung. Allerdings sind auch hierbei Anteile des Algorithmus noch weggelassen.

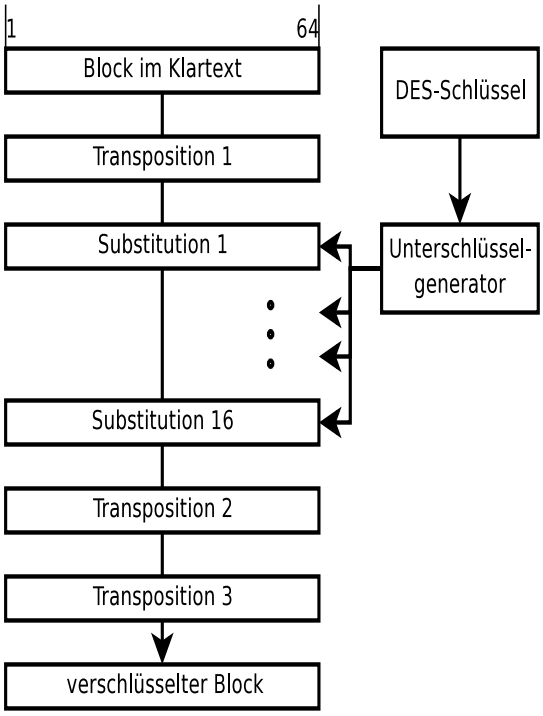


Abbildung 11.2. Substitutionen und Transpositionen bei DES

Blockalgorithmen haben eine grundsätzliche Schwäche: Sie erlauben es Angreifern, einige ungewünschte Handlungen durchzuführen. Dazu gehören:

1. Das Einfügen eines korrekt verschlüsselten Blocks. So kann ein früher mitgelesener Block mit bekanntem Inhalt von einem Angreifer unbemerkt in den Datenstrom eingefügt werden.
2. Das Löschen eines Blockes kann unbemerkt bleiben.
3. Als Kombination aus 1 und 2 kann ein Block durch einen früher mitgelesenen Block ersetzt werden. Hat der Angreifer X beispielsweise den korrekt verschlüsselten Block "Betrag: 100 Euro, Empfänger X" bei einer elektronischen Überweisung mitgelesen und gespeichert, so kann er bei zukünftigen Überweisungen an andere Empfänger den entsprechenden Block durch den gespeicherten ersetzen.

Der klassische Ausweg für dieses Problem ist das *Chaining*. Dabei wird jeder fertig verschlüsselte Block mit seinem Vorgänger über die Operation XOR verknüpft. Der erste Block wird mit einer beiderseits bekannten Zufallszahl[11.4] verknüpft. DES nimmt eine solche Verknüpfung im CBC-Modus[11.5] vor.

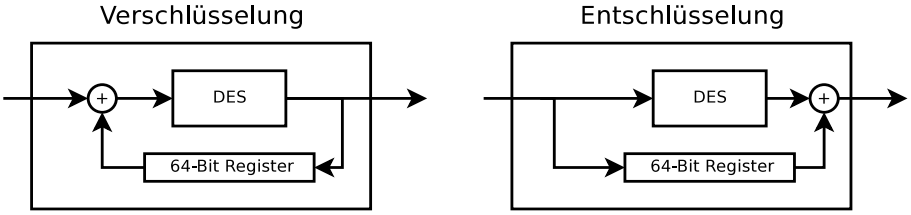


Abbildung 11.3. Vorgehensweise beim Chaining

Sitzungsschlüssel

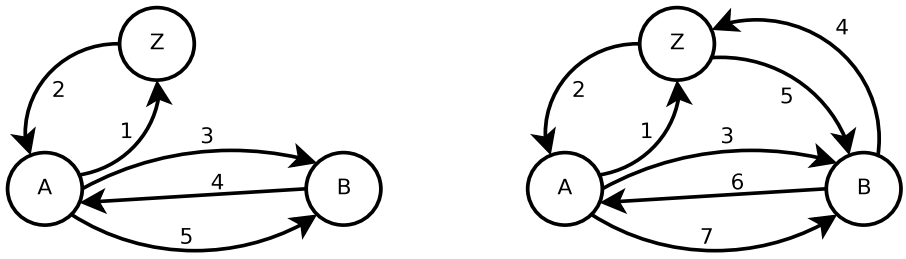
Nicht immer besitzen zwei Kommunikationspartner einen gemeinsamen Schlüssel. Es folgen zwei Verfahren, mit denen diese Kommunikationspartner einen solchen Schlüssel mit Hilfe einer Zentrale gewinnen können – ein symmetrisches und ein asymmetrisches (die zeitlichen Abläufe dieser Verfahren sind in Abbildung 11.4 zusammengefasst).

- Die Kommunikationspartner A und B besitzen jeweils einen gemeinsamen geheimen Schlüssel S_A bzw. S_B mit einer Zentrale Z. Sie gehen nun folgendermaßen vor:
  1. A sendet $V_{S_A}(A, B, \mathrm{id}_A)$ an Z. Z entschlüsselt: $E_{S_A}(V_{S_A}(A, B, \mathrm{id}_A)) = (A, B, \mathrm{id}_A)$ und erkennt, dass A einen Sitzungsschlüssel für die Kommunikation mit B benötigt. Die mitgesendete Identifikation id_A kann beispielsweise eine Netzwerkaddresse sein.
  2. Z sendet $V_{S_A}(\mathrm{id}_A, B, S_S, V_{S_B}(\mathrm{id}_A, A, S_S))$ an A. A kann diese Nachricht entschlüsseln und erhält den Sitzungsschlüssel S_S. Außerdem erhält A eine verschlüsselte Nachricht, die nur B entschlüsseln kann.
  3. A sendet aus der vorigen Nachricht $V_{S_B}(\mathrm{id}_A, A, S_S)$ an B. B erhält so ebenfalls den Sitzungsschlüssel S_S.
  4. B sendet $V_{S_S}(\mathrm{id}_A, \mathrm{id}_B)$ an A. A entschlüsselt die Nachricht und bestätigt damit, dass er den Sitzungsschlüssel erhalten hat. Außerdem erhält A auf diese Weise die Identifikation von B.
  5. A sendet $V_{S_S}(\mathrm{id}_B)$ an B zurück und quittiert somit die Bestätigung aus Schritt 4.
- Die Kommunikationspartner A und B sowie die Zentrale Z besitzen jeweils ein Paar aus privatem und öffentlichem Schlüssel: $(S_A, \bar{S}_A)$, $(S_B, \bar{S}_B)$ und $(S_Z, \bar{S}_Z)$. Die Zentrale kennt die öffentlichen Schlüssel von A und B. Der *private* Schlüssel der Zentrale ist sowohl A als auch B bekannt.
  1. A sendet (A, B, id_A) an die Zentrale Z.

11.4. Dieser Wert wird häufig als Initialisierungsvektor bezeichnet.

11.5. Chain Block Cipher Modus

2. Z sendet $V_{S_Z}(\mathrm{id}_A, S_B, B)$ – also mit dem *öffentlichen* Schlüssel von Z verschlüsselt – an A, und teilt so den öffentlichen Schlüssel von B mit. Da A den privaten Schlüssel der Zentrale kennt, kann er die Nachricht entschlüsseln.
3. A kontaktiert B und erklärt seinen Kommunikationswunsch mit $V_{S_B}(\mathrm{id}_A, A)$.
4. B kennt den öffentlichen Schlüssel von A bisher nicht und wendet sich deshalb mit (B, A, id_B) an die Zentrale.
5. Die Zentrale sendet $V_{S_Z}(\mathrm{id}_B,\ S_A,\ A)$ zurück, so dass B den öffentlichen Schlüssel erhält.
6. B benutzt den öffentlichen Schlüssel von A und sendet die Nachricht $V_{S_A}(\mathrm{id}_A, \mathrm{id}_B)$ an A, so dass A erfährt, dass B zur Kommunikation bereit ist.
7. Zuletzt quittiert A die Nachricht aus Schritt 6, indem er $V_{S_B}(\mathrm{id}_B)$ an B sendet. B kann nun sicher sein, dass A erfahren hat, dass B bereit ist.

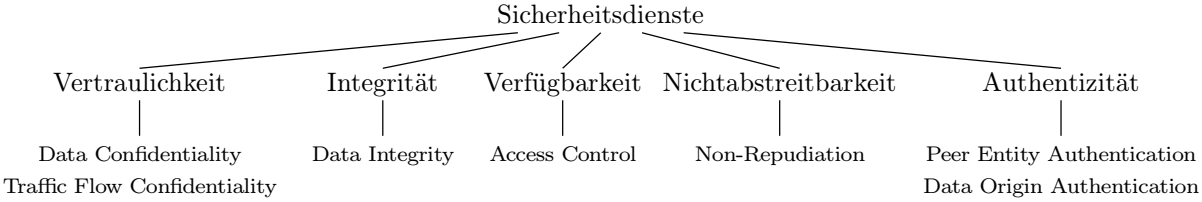


a) symmetrisch b) asymmetrisch

Abbildung 11.4. Zeitliche Abläufe bei der Sitzungsverschlüsselung

11.3 Sicherheitsmechanismen und -dienste

Verschlüsselungsverfahren sind nur eine Gruppe von Mechanismen, die zur Gewährleistung der Netzsicherheit eingesetzt werden. In diesem Abschnitt wollen wir kurz weitere Mechanismen betrachten. In Abschnitt 11.1 haben wir die Anforderungen an Netzsicherheit klassifiziert. Diesen Klassen lassen sich nun konkretere *Sicherheitsdienste* zuordnen:[11.6]



Vertraulichkeit kann sich sowohl auf den Inhalt der Kommunikation (data confidentiality) als auch auf die Kommunikationsbeziehungen (traffic flow confidentiality) beziehen. Selbst wenn der Inhalt bei der Kommunikation durch Verschlüsselung vor Unbefugten geschützt ist, müssen manchmal bestimmte Mechanismen eingesetzt werden, um zu verhindern, dass Unbefugte erkennen können, *dass* kommuniziert wird, *wer* kommuniziert, *wann* und *in welchen Umfang* diese Kommunikation stattfindet.

Authentizität kann ebenfalls differenziert werden. Zum einen ist es für viele Anwendungen erforderlich, dass ein Kommunikationspartner seine Identität gegenüber einem anderen unter Beweis stellt. Dies gewährleistet allerdings noch nicht, dass die Herkunft der Daten (z.B. in Bezug auf eine Adresse) authentisch ist.

11.6. vgl. u.a. Trusted Computer System Evaluation Criteria (TCSEC) des National Computer Security Center (NCSC) und das OSI Security Protocol Reference Model

Betrachten wir nun die *Sicherheitsmechanismen*, die zur Gewährleistung der Netzsicherheit genutzt werden. Tabelle 11.1 fasst zusammen, welche Mechanismen zur Realisierung der Sicherheitsdienste eingesetzt werden können.

Encryption (Verschlüsselung). Daten werden mit speziellen Algorithmen für Unbefugte unlesbar gemacht. Es existieren symmetrische und asymmetrische Algorithmen. Sie können eingesetzt werden, um Abschnitte des Übertragungsmediums zu schützen (link encryption) oder auf Anwendungsebene zwischen Benutzern eines Kommunikationssystems genutzt werden (end-to-end encryption). Dieser Mechanismus wurde in Abschnitt 11.2 besprochen. Da symmetrische Algorithmen in der Regel schneller arbeiten, werden asymmetrische Algorithmen oft nur verwendet, um einen Sitzungsschlüssel auszutauschen.

Digital Signature (digitale Unterschrift). Dieser Mechanismus stellt die Authentizität und Integrität von Daten sicher und bietet eine Möglichkeit, den Urheber einer Nachricht unzweifelhaft zu identifizieren. Asymmetrische Verschlüsselungsalgorithmen können verwendet werden, um solche Signaturen zu erzeugen. Beispielsweise kann ein Benutzer A eine Nachricht mit seinem eigenen privaten Schlüssel "entschlüsseln" und versenden. Der Empfänger B kann nun die Nachricht mit dem ihm bekannten öffentlichen Schlüssel "verschlüsseln".[11.7] Erhält er eine korrekte Nachricht, so kann diese nur von A stammen.

$$E_{\bar{S}_A}(M) = M_{\text{sig}} \qquad V_{S_A}(M_{\text{sig}}) = M$$

Dieses Verfahren wird u.a. von dem populären Sicherheitssystem PGP (pretty good privacy) verwendet. Wie auch bei der Verschlüsselung stellt sich hier die Frage, wie ein Benutzer den öffentlichen Schlüssel eines anderen erfährt und zweifelsfrei annehmen kann, dass er authentisch ist. Zur Lösung dieses Problems kann der Mechanismus selbst angewendet werden – entweder in einer dezentralen oder zentralen Weise:

- Die Benutzer signieren gegenseitig ihre öffentlichen Schlüssel. Ein Kommunikationspartner A kann einem anderen B seinen öffentlichen Schlüssel senden – signiert von einem Dritten C. Sofern B den öffentlichen Schlüssel von C kennt und diesem vertraut, kann er nun auch auf die Echtheit des öffentlichen Schlüssels von B vertrauen. Dieser Mechanismus wird oft als "Web of Trust" bezeichnet.
- Stattdessen kann eine vertrauenswürdige Zentrale die Zugehörigkeit eines öffentlichen Schlüssels zu einer Person oder Institution bescheinigen. Auch hier signiert die Zentrale den Schlüssel. Solche Bescheinigungen werden oft *Zertifikate* genannt.

Es sollte noch erwähnt werden, dass aus Effizienzgründen häufig nicht die gesamte Nachricht signiert wird, sondern nur ein Hash-Wert der Nachricht. Abbildung 11.5 zeigt eine mit PGP signierte Nachricht.[11.8]

```
---BEGIN PGP SIGNED MESSAGE---
Hash: SHA1

Diese Nachricht wurde zweifelsfrei von Merlin Senger geschrieben.
---BEGIN PGP SIGNATURE---
Version: GnuPG v1.4.0 (GNU/Linux)

iD8DBQFDY3lrtw++lTUDVJQRAt/EAJ9/Gf9AIPleYRaFw1NBaiWudSUH7QCgv3+9
0t1NpYOyeZOLm0vurCAd3o4=
=d5x9
---END PGP SIGNATURE---
```

Abbildung 11.5. Eine Nachricht mit digitaler Signatur

11.7. Da die Begriffe "verschlüsseln" und "entschlüsseln" im Rahmen von digitalen Signaturen etwas irreführend sind, spricht man in der Regel von "signieren" und "verifizieren". Dennoch unterscheiden sich die Funktionen nicht von denen bei der Ver- und Entschlüsselung.

11.8. Natürlich wird die Verifikation in der Regel von den Anwendungen vorgenommen, so dass sich der Benutzer nicht mit solchen kryptischen Zeichenketten beschäftigen muss.

Access Control (Zugangskontrolle). Hierunter verstehet man die Zuordnung von Zugriffsrechten auf Betriebsmittel zu Entitäten (z.B. Benutzer, Prozesse). Die dafür offensichtlich notwendige Identifikation der Entität wird häufig durch Passwörter gesichert. Die Rechte werden häufig in Form von Listen oder Matrizen repräsentiert. Oft werden auch die Konzepte *Gruppen* oder *Rollen* umgesetzt. Ein einfaches Beispiel hierfür ist das Unix-Dateisystem. Eine Datei ist einem Benutzer und einer Gruppe zugeordnet. Die Zugriffsrechte sind in Form einer 9 Bit langen Sequenz dargestellt:[11.9]

```
rw-rw-r--  merlin  freunde        0  Oct 2 13:04  gästebuch.txt
rwxr-x--x  merlin  programmierer  0  Oct 7 15:47  script.sh
rw------  merlin  merlin         0  Oct 9 19:25  tagebuch.txt
```

Data Integrity (Datenintegrität). Dieser Mechanismus soll gewährleisten, dass Daten sich in einem korrekten Zustand befindet und nicht beispielsweise durch Übertragungsfehler oder Angreifer verändert wurden. Mechanismen zur Erkennung und Behebung von Übertragungsfehlern wurden in Abschnitt 2.12 beschrieben. Mechanismen zur Quittierung sowie zur Wiederherstellung der Sendereihenfolge von Datagrammen wurden ebenfalls schon beschrieben. Der Schutz vor Angreifern ist damit allerdings noch nicht gewährleistet, da diese auch die Prüfsummen selbst manipulieren könnten. Oft werden signierte Prüfsummen an die Nachrichten angehängt (siehe auch Abbildung 11.5).

Authentication (Authentifikation). Hierbei handelt es sich um Mechanismen, die im weitesten Sinne die Identität einer Entität nachweisen. Dazu werden kryptografische Techniken, Passwörter und neuerdings auch biometrische Eigenschaften verwendet.

Traffic Padding. Wir haben schon erwähnt, dass neben den Inhalten einer Kommunikationsbeziehung auch andere Merkmale schützenswert sein können. Soll etwa der Raum, die Häufigkeit, die Dauer oder der Umfang einer Kommunikation verschleiert werden, so können zusätzliche Dateneinheiten (dummy messages) versendet werden. Ein solches Vorgehen wird Traffic Padding genannt.

Routing Control. Hiermit sind Mechanismen gemeint, die für Dateneinheiten einen geeigneten Pfad in Hinblick auf deren Vertraulichkeit wählen. Geeignete Pfade können durch als sicher angesehene Netzwerke oder Leitungen realisiert sein. Auch das Aufteilen des Verkehrs auf unterschiedliche Pfade kann ein gewisses Maß an Vertraulichkeit gewährleisten.

Notarization. Bei diesem Mechanismus werden bestimmte Eigenschaften der gesendeten Daten durch einen vertrauenswürdigen Dritten garantiert. Typische Eigenschaften sind dabei die Integrität der Nachricht, der Zeitpunkt der Sendung sowie die Echtheit von Absender und Empfänger.

	Encryption	Digital Signature	Access Control	Data Integrity	Authentication	Traffic Padding	Routing Control	Notarization
Data Conf.	•						•	
Traffic Flow Conf.	○					○	•	
Data Integrity	○	•		•				
Access Control	○		•					
Non-Repudiation	○	•		○				○
Peer Entity Auth.	○	•			•			
Data Origin Auth.	○	•						

• eignet sich, um den Dienst zu erbringen
○ kann den Dienst in Kombination mit anderen erbringen

Tabelle 11.1. Zusammenhänge zwischen Sicherheitsdiensten und -mechanismen

11.9. Dabei steht `r` für lesen (read), `w` für schreiben (write) und `x` für ausführen (execute). Neun Bits geben so die Rechte für den Besitzer, die Gruppe und alle anderen an.

Kapitel 12
Ausblick und Trends

In dieser Lehrveranstaltung haben wir wichtige Aspekte von Datenkommunikation und Rechnernetzen kennengelernt. Diese wurden mit system- und methodenorientierten Sichtweisen dargestellt. So erfolgte auf der einen Seite eine Beschäftigung mit Architekturen, Algorithmen, Protokollen sowie mit Standards und Aspekten der Realisierung. Dabei haben wir bedeutende Beispiele im breiten Spektrum von der rechnerinternen Kommunikation bis hin zum globalen Internet kennengelernt und die Datenkommunikation von der physikalischen bis zur Anwendungsebene beleuchtet.

Auf der anderen Seite haben wir uns mit Methoden zur Modellierung von Systemen und Lasten, formalen Beschreibungstechniken, Analyseverfahren sowie der wissenschaftlichen Durchführung und Auswertung von Experimenten auseinander gesetzt. Insbesondere in den letzten Kapiteln wurden Methoden aus den Bereichen Protocol Engineering, Traffic Engineering und Netzmanagement dargestellt.

Da der Bereich der Datenkommunikation nach wie vor starken Veränderungen und Weiterentwicklungen unterworfen ist, haben wir uns nicht auf die jüngsten Entwicklungen beschränkt, sondern versucht, einen möglichst breiten Überblick über Systeme und Methoden zu liefern. Die Kenntnis von aktuellen wie auch von historischen Systemen erleichtert das Verständnis von Neuerungen in diesem Bereich.

Entwicklungstendenzen

In der jüngeren Vergangenheit wurden große Fortschritte bei der Erhöhung der Datenraten erreicht. Insbesondere geschah dies bei der optischen Signalübertragung um weitere Größenordnungen und in Funknetzen. Auch bei der sogenannten “letzte Meile“ zum Endteilnehmer konnten die Datenraten mit DSL im Gegensatz zu ISDN um ein Vielfaches gesteigert werden.

Auch bei den Netzkomponenten setzt sich der Trend zu immer schnelleren Endsystemen fort. Hinzu kommt eine stetig kompakter werdende Bauweise, die es ermöglicht, bestehende Technologien in immer kleineren Endgeräten zu nutzen. Eine um Größenordnungen schnellere Vermittlung wird wohl künftig durch rein optische Vermittlungsrechner möglich sein.

Durch die Entwicklung leistungsfähigerer Systeme und Algorithmen werden laufend neue Standards zur Datenkommunikation festgelegt. Dies ist bei leitungsgebundenen Verbindungen wie Ethernet aber auch in Funknetzen wie UMTS und WLAN der Fall.

Nicht zu vergessen sind dabei jedoch die Risiken der immer enger werdenden globalen Vernetzung. Zum einen entsteht durch sie zunehmend eine gesellschaftliche Abhängigkeit von diesen Technologien. Diese betrifft wirtschaftliche Unternehmen, staatliche Behörden und Privatpersonen. Zum anderen werfen die neuen Technologien neue Sicherheitsrisiken auf, die Spionage, Abhörung und Verfälschung von Informationen ermöglichen.

Persönlichkeitsveränderungen, die auf die neuen Technologien zurückgeführt werden, lassen sich bei einer nicht geringen Zahl der Nutzer beobachten. Eine Sucht nach den neuen Medien ist dabei genauso zu beachten wie die Jugendgefährdung, die von den zugänglichen Inhalten ausgeht.

Zuletzt hat die globale Vernetzung auch negative Auswirkungen auf die Staatengemeinschaft. Ein Beispiel hierfür ist eine bedeutender werdende Kluft zwischen denen, die über die Technologien und Infrastrukturen verfügen (“haves“) und jenen, die sie der breiten Bevölkerung nicht bereitstellen können oder wollen (“don’t haves“ bzw. “have nots“). Hinzu kommt der Mangel an weltweiten Gesetzen und Regelungen, deren Ausgestaltung und Umsetzung wohl eine große Herausforderung für die Zukunft darstellen wird.

Aktuelle Forschungsschwerpunkte zur Kommunikation in Rechnernetzen, insbesondere am Arbeitsbereich TKRN sind u.a.

- Verbesserungen der Dienstgüte bei Audio- und Videokommunikation im Bereich der Echtzeitkommunikation,
- Methoden zum Traffic Engineering für dienstintegrierte Netze, wie Lastmessung, -modellierung, -generierung und Netzemulation,
- WLAN-Nutzung und -analyse im Bereich des e-Learning sowie
- Netzmanagement für gekoppelte Netze im Bereich des Internet.

Wirklich spannende Zeiten mit großen Herausforderungen für Informatik-Absolventen und -Nachwuchswissenschaftler.

Index

Literaturverzeichnis

Als Ergänzung zu diesem Skript besonders geeignete Lehrbücher sind fett gedruckt.

[Bad 04] A. Badach: Voice over IP – Die Technik, C. Hanser-Verlag 2004, 347 S.

[BaH 01] A. Badach, E. Hoffmann: Technik der IP-Netze, C. Hanser-Verlag 2001

[BaH 07] A. Badach, E. Hoffmann: Technik der IP-Netze, 2. Aufl., C. Hanser-Verlag 2007

[Bat 99] A. Bateman: Digital Communications, Addison-Wesley 1999, 221 S.

[BeG 99] F. Bergmann, H. J. Gerhardt (Hrsg.): Taschenbuch der Telekommunikation, C. Hanser-Verlag 1999

[BöH 02] W. Böhmer: VPN – Virtual Private Networks, Hanser 2002, 367 S.

[Chi 98] E. Chimi: High-Speed Networking: Konzepte, Technologien, Standards, C. Hanser-Verlag 1998, 596 S.

[Com 04] D. Comer, Computernetzwerke und Internets, 3. Aufl., Prentice-Hall 2004, 688 S.

[Com 04] D. Comer, Computer Networks and Internets with Internet Applications, 7th ed., Prentice-Hall 2004

[CoR 00] R. G. Cole, R. Ramaswamy: Wide-Area Data Network Performance Engineering, Artech House Telecommunications Lib. 2000, 417 S.

[DBR 97] M. Duck, P. Bishop, R. Read: Data Communications for Engineers, Addison-Wesley 1997

[DeP 96] De Prycker: Asynchronous Transfer Mode, Prentice-Hall 1996, 430 S.

[DYN 03] J. Duato, S. Yalamanchili, L. Ni: Interconnection Networks: an Engineering Approach. Kaufmann Publ. 2003

[Far 04] A. Farrel: The Internet and its Protocols – A Comparative Approach, Morgan Kaufmann 2004, 809 S.

[FHW 04] K. Fiolka, K. Heidtmann, B. Wolfinger: Ein eLearning-Werkzeug zur Videokommunikation über simulierte verlustbehaftete Netze, GI-Workshop "Elektronische Unterstützung der Präsenzlehre" im Rahmen der GI-Jahrestagung INFORMATIK 2004, Ulm, Sept. 2004.

[For 06] B. A. Forouzan: TCP/IP Protocol Suite, 3rd ed., Mc. Graw-Hill 2006, 861 S.

[Gli 06] S. G. Glisic: Advanced Wireless Networks: 4G Technologies, Wiley-Verlag 2006

[Göb 99] J. Göbel: Kommunikationstechnik, Grundlagen und Anwendungen, Hüthig-Verlag 1999, 969 S.

[Gre 99] R. Greenlaw: Introduction to the Internet for Engineers, Mc. Graw-Hill 1999

[HaB 04] J. F. Hayes, T. Babu: Modeling and Analysis of Telecommunications Networks, J. Wiley 2004

[Hal 01] F. Halsall: Multimedia Communications, Addison-Wesley 2001

[Hal 05] F. Halsall: Computer Networking and the Internet, 5th ed., Addison-Wesley 2005

[HAN 99] H.-G. Hegering, S. Abeck, B. Neumair: Integriertes Management vernetzter Sysŋteme: Konzepte, Architekturen und deren betrieblicher Einsatz, dpunkt-Verlag 1999, 607 S.

[HeL 00] E. Herter, W. Lörcher: Nachrichtentechnik, 8. Aufl., C. Hanser-Verlag 2000

[HeQ 95] W. Heise, P. Quattrocchi: Informations- und Codierungstheorie, 3. Aufl., Springer-Verlag 1995

[HPS 00] H. Häckelmann, H. J. Petzold, S. Strahringer: Kommunikationssysteme – Technik und Anwendungen, Springer-Verlag 2000

[JuW 98] V. Jung, H.-J. Warnecke: Handbuch für die Telekommunikation, Springer-Verlag 1998

[Kes 97] S. Keshav: An Engineering Approach to Computer Networking, ATM Networks, the Internet, and the Telephone Network, Addison-Wesley 1997

[Klu 00] N. Klussmann: Lexikon der Kommunikations- und Informationstechnik, 2. Aufl., Hüthig-Verlag 2000

[KMK 04] A. Kumar, D. Manjunath, J. Kuri: Communication Networking – An Analytical Approach, Morgan Kaufmann 2004, 929 S.

[KrR 02] G. Krüger, D. Reschke (Hrsg.): Lehr- und Übungsbuch Telematik, 2. Aufl., Fachbuchverlag Leipzig C. Hanser 2002

[KuR 08] J. F. Kurose, K.W. Ross: Computer Networking Ũ A Top-Down Approach, 4th ed., Pearson /Addison-Wesley 2008, 852 S.

[KüW 02] P. J. Kühn, B. E. Wolfinger (Hrsg.): Themenheft "Internet Traffic Engineering", PIK-Zeitschrift, 24. Jg. Heft 2, 2002

[Lin 05] J. Lindner: Informationsübertragung, Springer 2005

[MaF 07] J. Maucher, J. Furrer: WiMAX - Der IEEE-802.16-Standard: Technik, Anwendung, Potenzial, Heise-Verlag, 2007, 407 S.

[MäG 02] R. Mäusel, J. Göbel: Analoge und digitale Modulationsverfahren, Hüthig-Verlag 2002

[MeR 07] D. Medhi, K. Ramasamy: Network Routing Ũ Algorithms, Protocols, and Architectures, Elsevier 2007, 788 S.

[PeD 04] L. L. Peterson, B. S. Davie: Computernetze, 3. Aufl., dpunkt-Verlag 2004

[PeD 07] L. L. Peterson, B. S. Davie: Computer Networks, 4. ed., Elsevier-Verlag, 2007

[Per 01] C. Perkins: Ad Hoc Networking, Addison-Wesley 2001

[Pro 02] W. Proebster: Rechnernetze - Technik, Protokolle, Systeme, Anwendungen, 2. Aufl., Oldenbourg-Verlag 2002

[PrS 04] J. G. Proakis, M. Salehi: Grundlagen der Kommunikationstechnik, 2. Aufl., Pearson Studium 2004

[Ram 01] R. Ramaswami, K. N. Sivarajan: Optical Networks – A Practical Perspective, Morgan Kaufmann 2001

[Sal 05] D. Salomon: Coding for Data and Computer Communications, Springer 2005

[Sch 03] J. Schiller: Mobilkommunikation, 2. Aufl., Addison-Wesley 2003

[Sch 97] B. Schürmann: Rechnerverbindungsstrukturen – Bussysteme und Netzwerke, Vieweg-Verlag 1997, 414 S.

[She 01] T. Sheldon: Encyclopedia of Networking & Telecommunications, Mc. Graw-Hill 2001 (incl. CD)

[Sie 01] G. Siegmund (Hrsg.): Intelligente Netze, 2. Aufl., Hüthig-Verlag 2001

[Sie 02] G. Siegmund: Technik der Netze, 5. Aufl., Hüthig-Verlag 2002

[SMW 06] R. Steinmetz, M. Mühlhäuser, M. Welzl: Rechnernetze, C. Hanser-Verlag 2006

[Sta 00] W. Stallings: Local and Metropolitan Area Networks, 6th ed., Prentice-Hall 2000

[Sta 02a] W. Stallings: Wireless Communications and Networks, Prentice-Hall 2002

[Sta 02b] W. Stallings: High-Speed Networks and Internets – Performance and Quality of Service, 2nd ed., Prentice-Hall 2002

[Sta 07] W. Stallings: Data & Computer Communications, 8th ed., Prentice-Hall 2007

[StN 04] R. Steinmetz, K. Nahrstedt: Multimedia Systems, Springer 2004

[SWS 03] C. Scherpe, B. Wolfinger, I. Salzmann: Model Based Network Emulation to Study the Behavior and Quality of Real-Time Applications, 7th IEEE Internat. Symp. on Distributed Simulation and Real Time Applications (DS-RT 2003), 23.–25. Okt. 2003, Delft, NL

[Tan 03] A. S. Tanenbaum: Computer Networks, 4th ed., Prentice-Hall 2003, 891 S.

[Wal 02] B. Walke: Mobilfunknetze und ihre Protokolle 1 – Grundlagen, GSM, UMTS und andere zellulare Mobilfunknetze, 3. Aufl., Prentice-Hall 2002

[WaV 02] J. Walrand, P. Varaiya: High-Performance Communication Networks, 2nd ed., Morgan Kaufmann 2002, 693 S.

[WoH 07] B.E. Wolfinger, K.-D. Heidtmann (Hrsg.): Proc. 4. GI/ITG-Workshop: Leistungs-, Zuverlässigkeits- und Verlässlichkeitsbewertung von Kommunikationsnetzen und verteilten Systemen, MMBnet 2007, Hamburg, Sept. 2007

[Wol 99] B. Wolfinger: Characterization of Mixed Traffic Load in Service-Integrated Networks, Systems Science Journal, Vol. 25, No. 2, 1999, S. 65–86

[WUH 99] B. Wolfinger, T. Uhl, W. Halang (Hrsg.): Proc. Symposium "Verteilte multimediale Anwendungen und dienstintegrierende Kommunikationsnetze", Flensburg 1999

[WZH 02] B.E. Wolfinger, M. Zaddach, K. Heidtmann et al.: Analytical Modeling of Primary and Secondary Load as Induced by Video Applications Using UDP/IP, Computer Communications Journal, Vol. 25, Issue 11–12, 2002, S. 1094–1102

[Zen 01] A. Zenk: Lokale Netze – Die Technik fürs 21. Jahrhundert, 2. Aufl., Addison-Wesley 2001