



Motivierende Fragen

- Wie kann ein Protokoll eindeutig beschrieben werden?
- Welche Grundmechanismen können in Protokollen identifiziert werden?
- Wie können Nachrichten übermittelt werden und mit welchen Problemen muss man rechnen?
- Welche Schichten gibt es im Kommunikationsmodell?



Übersicht

- | | |
|--|--|
| <ol style="list-style-type: none"> 1. Einführung und Motivation <ul style="list-style-type: none"> ▪ Bedeutung, Beispiele 2. Begriffswelt und Standards <ul style="list-style-type: none"> ▪ Dienst, Protokoll, Standardisierung 3. Direktverbindungsnetze <ul style="list-style-type: none"> ▪ Fehlererkennung, Protokolle ▪ Ethernet 4. Vermittlung <ul style="list-style-type: none"> ▪ Vermittlungsprinzipien ▪ Wegwahlverfahren 5. Internet-Protokolle <ul style="list-style-type: none"> ▪ IP, ARP, DHCP, ICMP ▪ Routing-Protokolle 6. Transportprotokolle <ul style="list-style-type: none"> ▪ UDP, TCP 7. Verkehrssteuerung <ul style="list-style-type: none"> ▪ Kriterien, Mechanismen ▪ Verkehrssteuerung im Internet | <ol style="list-style-type: none"> 8. Anwendungsorientierte Protokolle und Mechanismen <ul style="list-style-type: none"> ▪ Netzmanagement ▪ DNS, SMTP, HTTP 9. Verteilte Systeme <ul style="list-style-type: none"> ▪ Middleware ▪ RPC, RMI ▪ Web Services 10. Netzsicherheit <ul style="list-style-type: none"> ▪ Kryptographische Mechanismen und Dienste ▪ Protokolle mit sicheren Diensten: IPSec etc. ▪ Firewalls, Intrusion Detection 11. Nachrichtentechnik <ul style="list-style-type: none"> ▪ Daten, Signal, Medien, Physik 12. Bitübertragungsschicht <ul style="list-style-type: none"> ▪ Codierung ▪ Modems |
|--|--|



Grundlagen: Rechnernetze und Verteilte Systeme

Kapitel 2: Begriffswelt und Standards

Dienst, Protokoll, Automat, IETF, ITU, IEEE

Prof. Dr.-Ing. Georg Carle
 Lehrstuhl für Netzarchitekturen und Netzdienste
 Technische Universität München
 carle@net.in.tum.de
 http://www.net.in.tum.de



Ziele

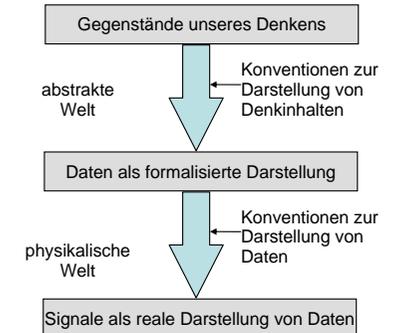
- In diesem Kapitel wollen wir vermitteln
 - Grundlegende Begriffe
 - Kommunikationsprobleme
 - Funktionsweise der Nachrichtenübermittlung
 - Geschichtete Kommunikationsmodelle
 - Formale Protokollspezifikation

Kapitelgliederung

- 2.1. Grundlegende Begriffe
- 2.2. Grundlegende Problemstellungen der Kommunikation
- 2.3. Charakterisierung von Kommunikationsvorgängen/-beziehungen
 - 2.3.1. Menge der beteiligten Kommunikationspartner (KP)
 - 2.3.2. Übertragungsverfahren/Schnittstellen
 - 2.3.3. Nutzungsrichtung
 - 2.3.4. Auslieferungsdisziplin
 - 2.3.5. Qualität
- 2.4. Technischer Hintergrund
- 2.5. Kommunikationsarchitekturen
 - 2.5.1. Netztopologien
 - 2.5.2. Dienste und Protokolle
- 2.6. ISO/OSI-Basisreferenzmodell
 - 2.6.1. OSI-Kommunikationseinheiten
 - 2.6.2. Bezeichnungskonventionen
 - 2.6.3. Charakterisierung der Schichten
- 2.7. Protokollspezifikation mit SDL

Der Begriff „Signal“

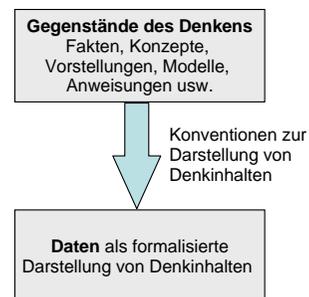
- Signal
 - Ein Signal ist die *physikalische Darstellung (Repräsentation)* von Daten durch charakteristische räumliche und/oder zeitliche Veränderungen der Werte physikalischer Größen.
 - Signale sind somit die *reale physikalische Repräsentation* abstrakter Darstellungen der Daten
 - Sprache, 8 Bit PCM codiert
 - Text als ASCII-Character



2.1. Grundlegende Begriffe - Der Begriff „Daten“

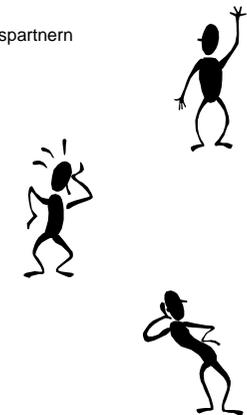
- Daten
 - Was wird dargestellt? Und wie?
 - *Darstellung* von Fakten, Konzepten, Vorstellungen und Anweisungen *in formalisierter Weise*, geeignet für
 - Kommunikation,
 - Interpretation und die
 - Verarbeitungdurch Menschen und/oder technische Mittel.
 - Allgemeine Beispiele für Datendarstellungen:
 - gesprochene Sprache
 - Zeichen-/Gebärdensprache
 - geschriebene Sprache
 - Datenkommunikation: Datenaustausch über immaterielle Träger (Energieflüsse, meist elektrische Ströme, elektromagnetische Wellen) und größere Entfernungen zwischen Menschen und/oder Maschinen

Modell zur Erzeugung von Daten durch den Menschen:



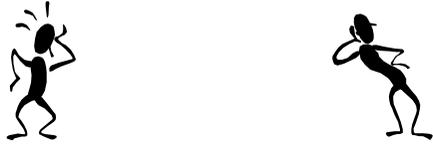
2.2. Grundlegende Problemstellungen der Kommunikation

- Regelung des Kommunikationsablaufs → Protokolle, Protokollschichten
- Ressourcenverteilung bei mehreren Kommunikationspartnern → Vielfachzugriff (Multiple Access)
- Kommunikation über Zwischenknoten → Vermittlung (Switching)
- Abarbeitung paralleler Kommunikationsvorgänge → Scheduling
- Identifikation von Kommunikationspartnern → Namen und Adressen
- Wahl des besten Kommunikationspfades → Routing
- Umgang mit Übertragungsfehlern → Fehlerkontrolle (Error Control)
- Anpassung der Übertragungsgeschwindigkeit → Flusskontrolle (Flow Control)



Protokolle, Protokollschichten

- Definition einer gemeinsamen Sprache und Anwendung vereinbarter Abläufe



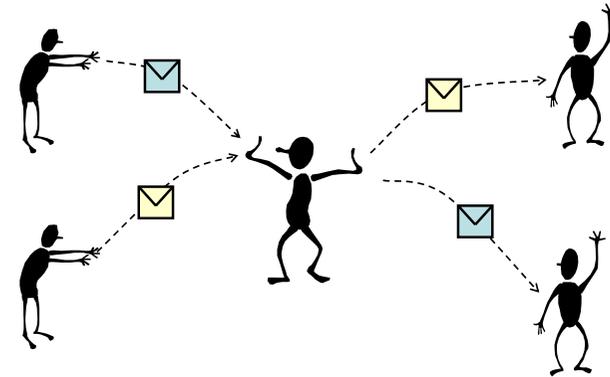
- Protokollschichten ermöglichen Arbeitsteilung



→ mehr dazu noch in diesem Kapitel

Vermittlung (Switching)

- Funktion von Nachrichtenvermittlern/Zwischenknoten



→ mehr dazu in Kapitel 4

Vielfachzugriff (Multiple Access)

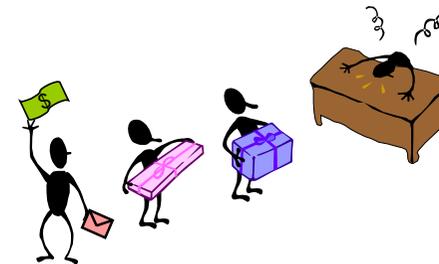
- Regelung des Zugriffs auf gemeinsames Medium zur Vermeidung von Störungen und Kollisionen



→ mehr dazu in Kapitel 3

Scheduling

- Bestimmung der Abarbeitungsreihenfolge für verschiedene Aufgaben

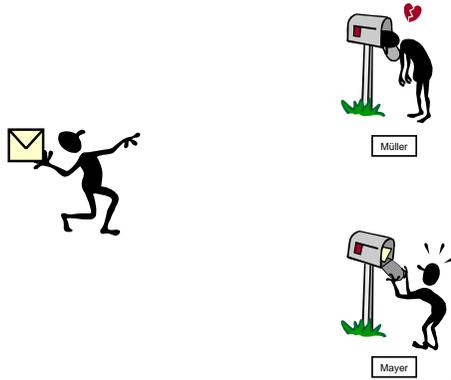


→ mehr dazu in Kapitel 7



Namen und Adressen

- Bestimmung des Empfängers und ggf. auch des Absenders



→ mehr dazu
in Kapitel 4



Fehlerkontrolle

- Erkennen und Behebung von Übertragungsfehlern

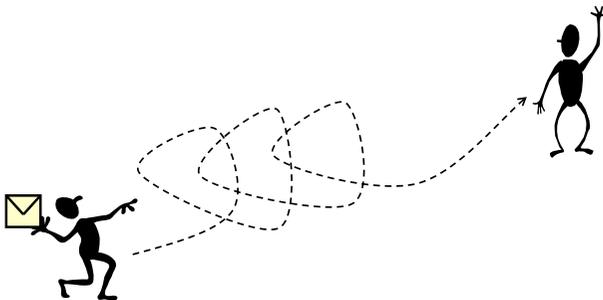


→ mehr dazu
in Kapitel 3,6



Wegwahl (Routing)

- Auffinden des günstigsten Pfades zum Empfänger

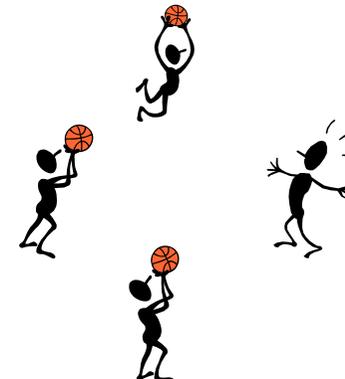


→ mehr dazu
in Kapitel 4,5



Flusskontrolle

- Anpassung der Übertragungsgeschwindigkeit an die Empfangsfähigkeiten des Empfängers



→ mehr dazu
in Kapitel 6,7

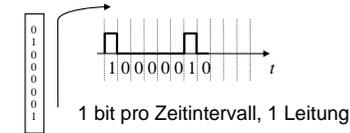
2.3. Charakterisierung von Kommunikationsvorgängen

Ein Kommunikationsvorgang kann aufgrund folgender Kriterien charakterisiert werden:

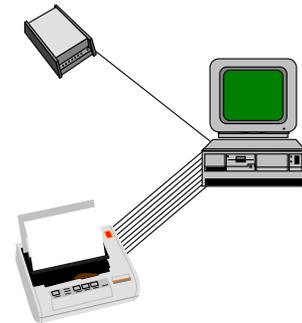
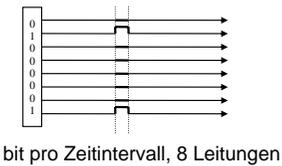
- (1) Beteiligten Kommunikationspartner (KP)
- (2) Übertragungsverfahren/Schnittstellen
- (3) Nutzungsrichtung
- (4) Auslieferungsdisziplin
- (5) Qualität

(2) Übertragungsverfahren/Schnittstellen

▪ Serielle Übertragung

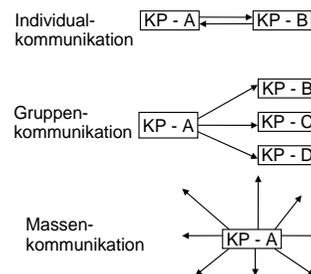


▪ Parallele Übertragung

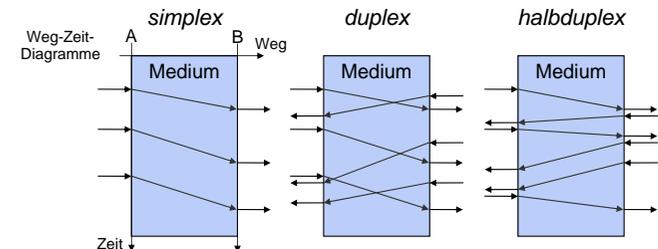


(1) Beteiligte Kommunikationspartner (KP)

- Akteure
 - Mensch-Mensch
 - Mensch-Maschine
 - Maschine-Maschine
- Menge der Kommunikationspartner
 - Dialog (*Unicast*): Zwei Partner tauschen über eine Punkt-zu-Punkt-Kommunikationsstrecke Daten aus.
 - Gruppenruf (*Multicast*): Ein Kommunikationspartner spricht gleichzeitig mehrere empfangende Kommunikationspartner an.
 - Rundruf (*Broadcast*): Es werden von einem Kommunikationspartner sehr viele (in der Regel unbekannte) Empfänger angesprochen, potentiell alle (Rundfunk).
 - *Anycast*: Ein beliebiger Kommunikationspartner einer Gruppe wird angesprochen.
 - *Concast*: viele Kommunikationsknoten senden an einen Einzelnen.



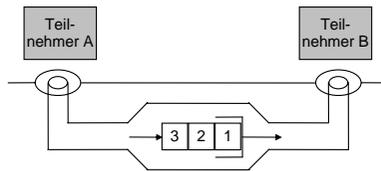
(3) Verbindungseigenschaften: Nutzungsrichtung



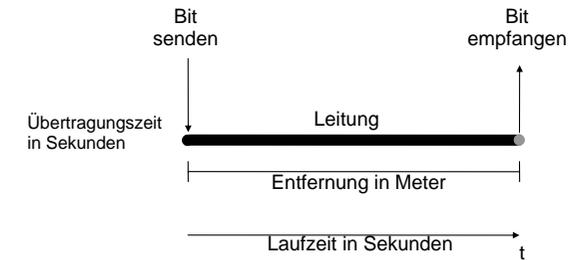
- Feuermelder
- Sensoren
- Pager
- Telefon
- Wechselsprechen
- Datenkommunikation mit geteilten Medien

(4) Auslieferungsdisziplin

- Die Auslieferungsdisziplin beschreibt die Reihenfolge der beim Empfänger ankommenden Daten in Bezug auf die Reihenfolge, wie sie abgeschickt wurden:
 - treu zur Einlieferungsreihenfolge (FIFO)
 - FIFO + priorisiert
 - keine Reihenfolgentreue garantiert



2.4. Technischer Hintergrund - Technische Leistung



Durchsatz (auch: Bandbreite)
= Anzahl der pro Sekunde übertragenen Bits
[Einheit bit/s]

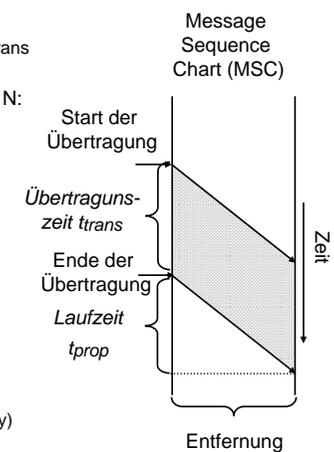
Bandbreiten-Verzögerungs-Produkt
= Speicherkapazität einer Leitung

(5) Qualität

- Bezüglich Qualität sind folgende Eigenschaften von Kommunikationsdiensten zu betrachten:
 - Technische Leistung
 - Antwortzeit, Durchsatz, Sende-/Empfangsrate, ...
 - Kosten
 - Investitionskosten, Betriebskosten, ...
 - Zuverlässigkeit
 - Fehlertoleranz, Ausfallsicherheit, Störunanfälligkeit, Verfügbarkeit, ...
 - Schutz
 - Abhörsicherheit, Manipulationssicherheit, Authentifizierung, Autorisierung, Maßnahmen gegen Dienstverweigerung, ...

Signalausbreitung im Medium, Datenspeicherung

- Senden einer Nachricht benötigt Übertragungszeit (transmission delay) t_{trans}
 - Übertragungszeit abhängig von Datenrate r and Länge der Nachricht N :
 $t_{trans} = N / r$
- Signale erreichen nach Laufzeit (propagation delay) t_{prop} ihr Ziel
 - Abhängig von Entfernung und Ausbreitungsgeschwindigkeit im Übertragungsmedium
- Über die Laufzeit t_{prop} werden $r \cdot t_{prop}$ bit generiert
 - Gespeichert im Medium
- Gesamtverzögerung:
 $t = t_{trans} + t_{prop} (+ t_{proc} + t_{queue})$
 - t_{proc} : Verarbeitungszeit (processing delay)
 - t_{queue} : Wartezeit (queuing delay)

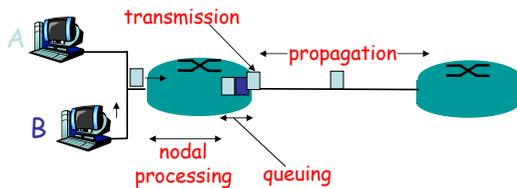




Verzögerungen in paketvermittelten Netzen

Vier unterschiedliche Verzögerungen an jedem Knoten

- 1) Verarbeitungszeit (processing delay)
- 2) Wartezeit (queuing delay)
- 3) Übertragungszeit (transmission delay)
- 4) Laufzeit (propagation delay)



2.5. Kommunikationsarchitekturen

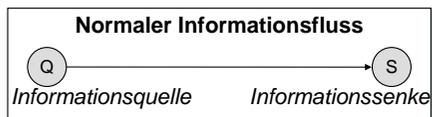
- Zur Realisierung von Kommunikationsvorgängen wird eine Kommunikationsarchitektur benötigt für:
 - physikalische Konnektivität
Verbindung über Kupferkabel, Lichtwellenleiter, Luftschnittstelle, ...
 - Kommunikationsfunktionalität
 - Steuerung des Ablaufs
 - Adressierung der Kommunikationspartner
 - Garantie einer geforderten Qualität
 - Anpassung unterschiedlicher Formate
 - ...
 - Schnittstelle zu den Anwendungen
- Aufgrund der *unterschiedlichen Aufgaben*:
 - Kommunikationsarchitektur mit geschichtetem Aufbau üblich
 - eine Schicht nutzt die Funktionalität der darunter liegenden Schicht, um ihre eigenen Funktionen zu realisieren



Sicherheitsgefahren und Schutzmaßnahmen

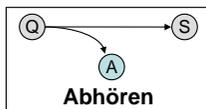
- Schutzmaßnahmen

- Verschlüsselung (kryptographische Codes)
- Schaffung vertrauenswürdiger Systeme (Authentisierung, Autorisierung)

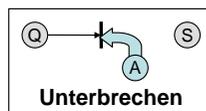
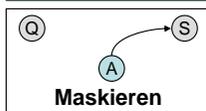
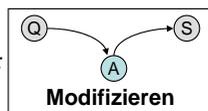


- Angriffe

Passiv:



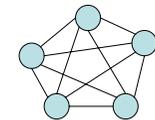
Aktiv:



2.5.1. Netztopologien

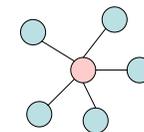
- vermaschtes Netz

- voll vermascht:
 - N Knoten
 - $N(N-1)/2$ Kanten/Verbindungen
 - stets direkte Verbindung zwischen zwei Knoten, zusätzlich N-2 alternative Pfade mit 2 Hops
 - unwirtschaftlich für große N



- Sternnetz

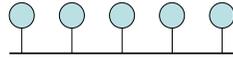
- Kanten mit unterschiedlichen Rollen:
 - Zentraler Vermittlungsknoten
 - Endknoten
- Grundkonzept eines hierarchischen Netzes
- N Endknoten \rightarrow N Kanten/Verbindungen
- 2 Hops zwischen zwei beliebigen Endknoten
- keine alternativen Pfade
- wirtschaftlich für große N



Netztopologien

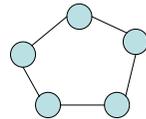
□ Busnetz

- gemeinsamer Bus als Broadcast-Medium
- passive Kopplung der Knoten an den Bus
- Vielfachfachzugriffssteuerung notwendig

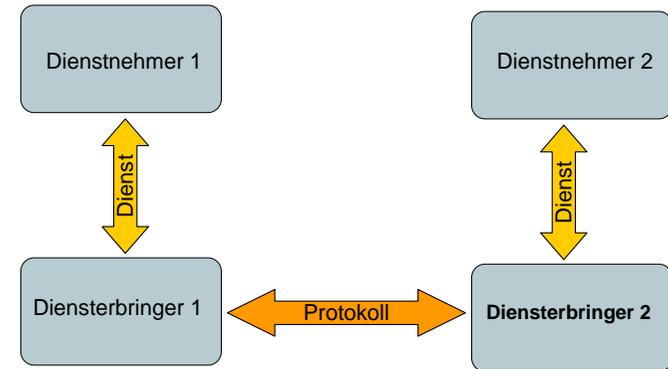


□ Ringnetz

- gemeinsamer Ringbus
- aktive Kopplung der Knoten an den Bus
- Kanten/Verbindungen unidirektional (simplex) oder bidirektional (duplex)
- bidirektionale Verbindungen
⇒ zwei unabhängige Pfade zwischen zwei Knoten
- Vielfachzugriffsteuerung durch reservierte Zeitschlitze (TDM) oder Token

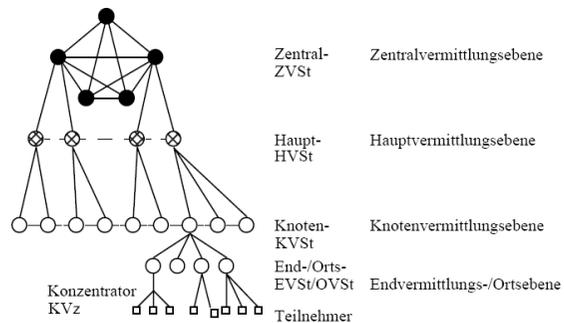


2.5.2. Dienst und Protokoll - Übersicht

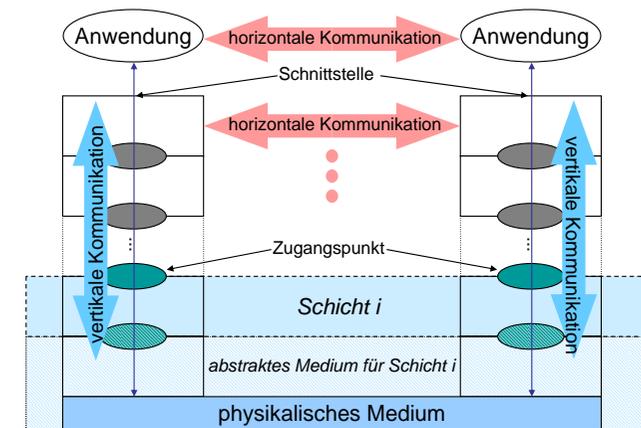


Hierarchische Netztopologien

□ Beispiel: klassisches Telefonnetz



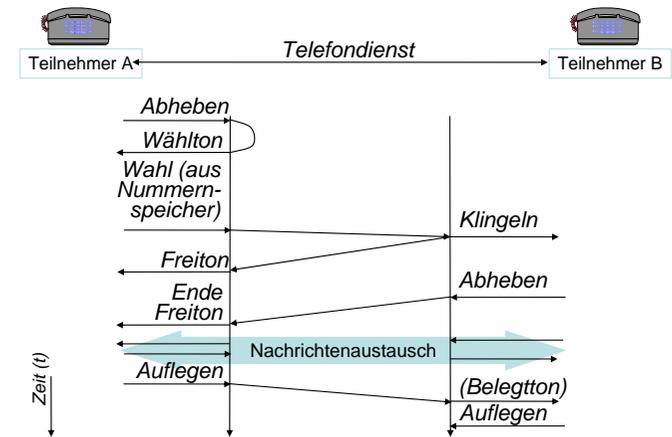
Geschichtetes Kommunikationssystem



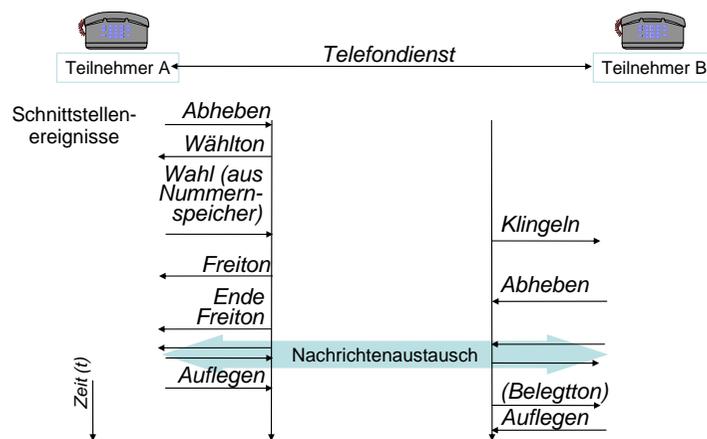
Dienst und Protokoll

- Partner einer Schicht
 - benutzen einen Dienst (außer unterste Schicht)
 - bieten einen Dienst (außer oberste Schicht)
 - brauchen nichts zu sehen / kennen außer direkt unterliegendem Dienst (Konzept der „virtuellen Maschine“)
 - „unterhalten sich“ gemäß Regeln (Protokollen)
 - z.B. „Telefon“-Schicht: wählen/klingeln/besetzt
 - Bei Menschen viel kontextsensitiv / implizit:
 - z.B. „Melden am Telefon“
 - Übersetzer: „Übersetz-Modus“, „Rückfragen-Modus“, „Selbst-Vorstellen“, „Chef-Vorstellen“, ...
- Kommunikationsarchitekturen basieren auf
 - „Dienst“ = (Kommunikations-) Dienst [(Communication) Service]
 - „Regeln“ = (Kommunikations-) Protokoll [(Communication) Protocol]

Beispiel Telefon – Dienstnehmersicht (2)

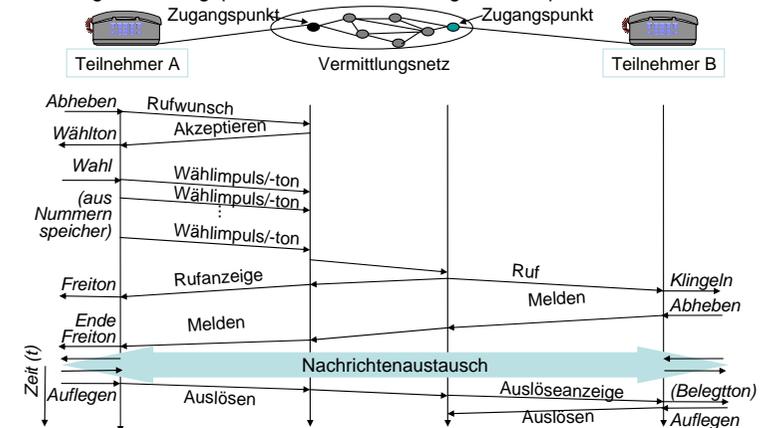


Beispiel Telefon – Dienstnehmersicht (1)



Beispiel Telefon - Dienst und Protokoll

- Signalisierungsprotokoll im alten analogen Fernsprechnetz:



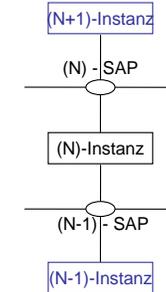
Begriffswelt „Dienst“

- Funktionalität einer Schicht wird als Menge von **Diensten** zur Verfügung gestellt.
- Die Dienste einer Schicht werden durch den Datenaustausch zwischen (Partner-) **Instanzen** erbracht. Dieser Datenaustausch erfolgt gemäß festgelegten Regeln und Formaten, die man **Protokoll** nennt.
- Ein Dienst wird an der **Dienstschnittstelle** einem Dienstbenutzer von einem Dienstbringer angeboten.
- Die **Dienstdefinition** spezifiziert verfügbare Dienste und Regeln für ihre Benutzung (in der darüber liegenden Schicht).
- Ein **Dienstprimitiv** (Schnittstelleneignis) dient zur Anforderung oder Anzeige eines Dienstes beim Dienstbenutzer, Grundtypen sind:
 - Anforderung (Req , Request)
 - Anzeige (Ind , Indication)
 - Antwort (Rsp , Response)
 - Bestätigung (Cnf , Confirmation)

(N) - Dienstzugangspunkt / (N) - SAP

- Innerhalb eines geschichteten Kommunikationssystems kommunizieren (N+1)-Instanzen und (N)-Instanzen über einen **(N)-Dienstzugangspunkt** [(N)-SAP, (N)-Service Access Point] miteinander.

Beziehungen zwischen (N-1)-SAP, (N)-Instanz und (N)-SAP



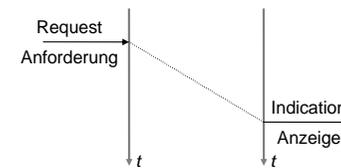
- Die (N)-Instanz bietet die von ihr erbrachten (N)-Dienste der (N+1)-Instanz am (N)-SAP an.
- Die (N)-Instanz benutzt die Dienste, die ihr am (N-1)-SAP angeboten werden.

Dienst der Schicht N

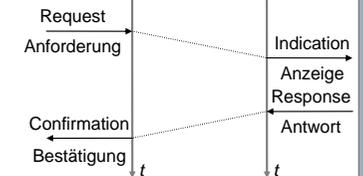
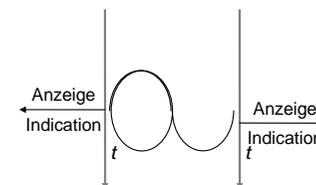
- **(N) - Dienst / (N) - Service**
 - Menge von Funktionen, welche die (N)-Schicht den (N+1)-Instanzen an der Schnittstelle zwischen der (N)- und (N+1)-Schicht anbietet (vertikale Kommunikation).
 - Die (N)-Instanzen erbringen die Dienste der (N)-Schicht mit Hilfe von Nachrichtenaustausch (horizontale Kommunikation). Dazu verwenden sie die Dienste der (N-1)-Schicht.
 - Wie die Dienste der (N) - Schicht erbracht werden, bleibt der (N+1) - Schicht verborgen.

Diensttypen

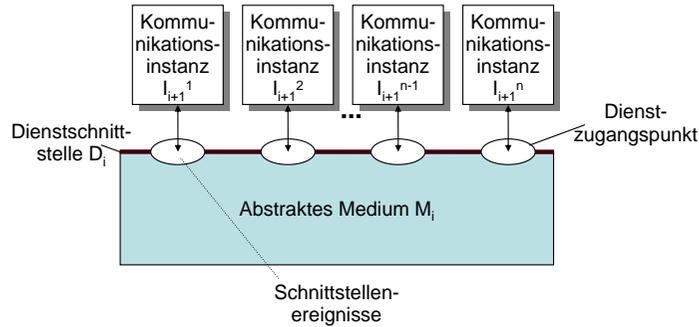
- **Unbestätigter Dienst**
 - Beispiel: Briefübermittlung
 - Vom Dienstnehmer initiiert:
- **Bestätigter Dienst**
 - Beispiel: Buchung
 - Vom Dienstnehmer initiiert:



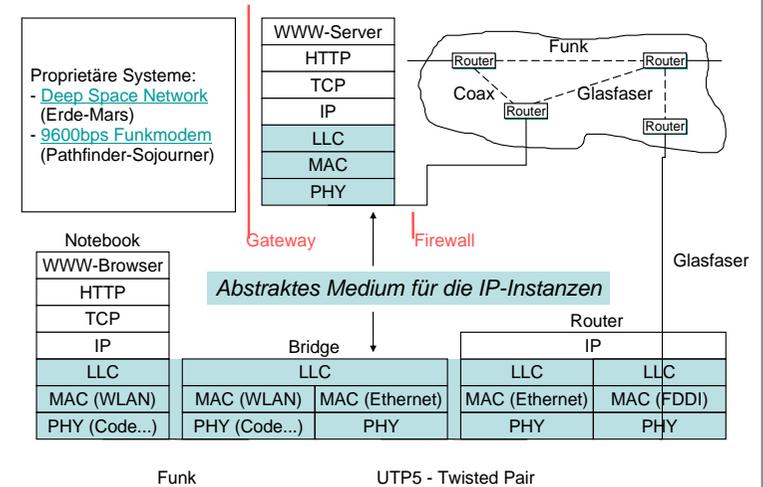
- Vom Dienstbringer initiiert:



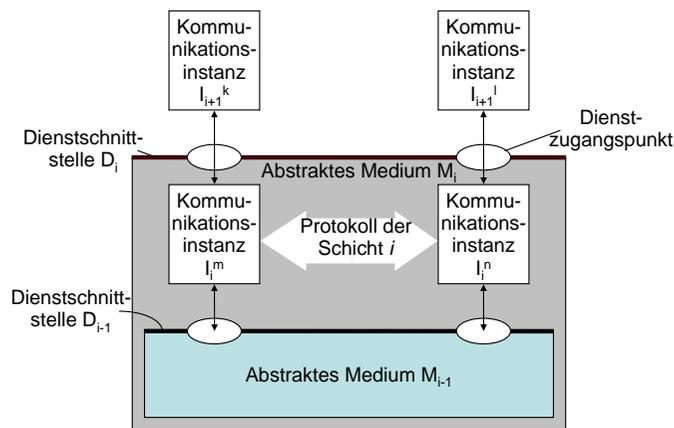
Der Dienstbegriff



Abstraktes Medium im Beispiel



Diensterbringung: Protokollablauf



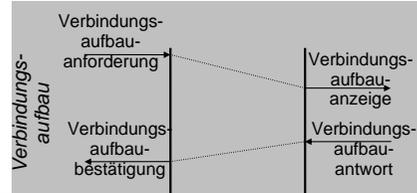
Verbindungsorientierte vs. verbindungslose Kommunikation

- Verbindungsorientierte Dienste
 - Vor dem Datenaustausch zwischen Dienstnehmern auf Schicht n wird eine Verbindung durch die beteiligten Instanzen der Schicht $n-1$ aufgebaut
 - Anforderung erfolgt mithilfe entsprechender Dienstprimitive der Schicht $n-1$
 - Protokollabhängige Aushandlung von Übertragungsparametern
 - z.B. Teilnehmer (immer), Dienstqualität, Übertragungsweg
 - Datenaustausch innerhalb dieser Verbindung erfolgt unter Berücksichtigung des aktuellen Verbindungszustandes
 - ⇒ Der Kontext einer jeden Datenübertragung wird somit berücksichtigt.
- Verbindungslose Dienste
 - Jeder Datenaustausch wird gesondert betrachtet, ohne Betrachtung vorhergegangener Kommunikationsvorgänge (gedächtnislos)
 - ⇒ Der Kontext einer Datenübertragung wird somit nicht berücksichtigt.

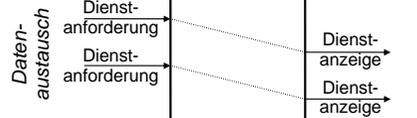
Verbindungsorientierte Dienste

3-Phasen-Prinzip

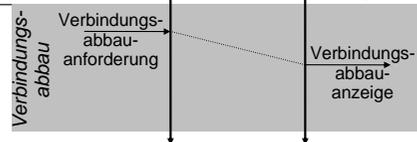
1. Verbindungsaufbau
Kontexterzeugung
 - Endsysteme
 - Netz



2. Datenaustausch
(hier: simplex)
weniger laufende Kontext-
informationen erforderlich



3. Verbindungsabbau
Kontextfreigabe
Ressourcenfreigabe



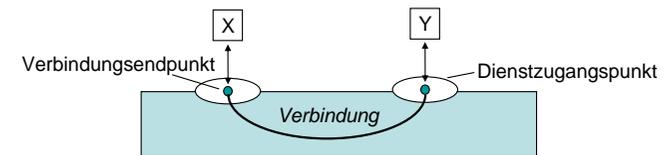
Dienstnehmer-Adressierung

□ Datagramm

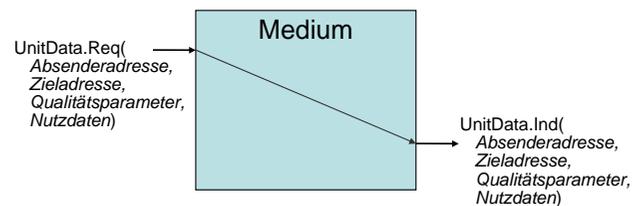
- Anforderung: Mit Adresse des Beantworters
- Anzeige: Ggf. mit Adresse des Initiators

□ Verbindungen

- Kontext, etabliert durch Verbindungsaufbau, beinhaltet Adressierungsinformation
- Bei mehreren Verbindungen vom selben Dienstzugangspunkt: Verbindungsidentifikation



Datagramm-Dienste



- Vom Datagramm-Dienst wird *kein Zusammenhang* zwischen verschiedenen Übertragungsleistungen unterstützt.
- Der Datagramm-Dienst unterstützt *keine Auslieferungsdisziplin*, z.B. keine Garantie für Reihenfolgetreue.
- Der Datagramm-Dienst realisiert eine *unbestätigte Dienstleistung* (keine Aushandlung zwischen Kommunikationspartnern).

2.6. Das ISO/OSI-Basisreferenzmodell

□ Ziel:

- Internationale Standardisierung ([ISO](#) = International Organization for Standardization) von Diensten und Protokollen zur Realisierung sogenannter "Offener Systeme" (OSI = Open System Interconnection)
- Grundlage zur Kommunikation von Systemen unterschiedlicher Hersteller
- Wichtig: Das Basisreferenzmodell dient als Denkmodell, anhand dessen sich Kommunikationssysteme erklären und klassifizieren lassen.
- Implementierung des Modells vor allem in öffentlichen Netzen in Europa (weitgehende Verdrängung durch Internet-Protokolle)

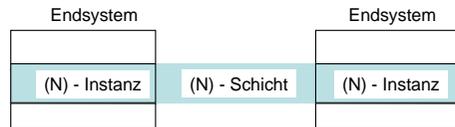
□ Standard:

- [ISO/IEC IS 7498](#): Information Processing Systems - Open Systems Interconnection - Basic Reference Model, Internationaler Standard, 15. Oktober 1994.
- Übernommen von der CCITT bzw. [ITU-T](#) in der Norm X.200

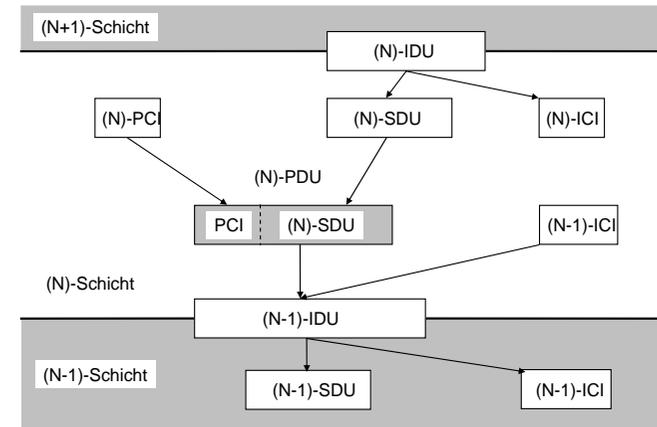


Prinzipien des ISO/OSI-Basisreferenzmodells

- OSI - Endsystem
 - Rechnersysteme, die sich bei der Kommunikation an OSI-Standards halten
- (N) - Schicht (Layer)
 - Sämtliche Einheiten einer (N) - Hierarchiestufe in allen Endsystemen
- (N) - Instanz (Entity)
 - Implementierung eines (N) - Dienstes in einem Endsystem.
 - Es kann verschiedene Typen von (N) - Instanzen geben ((N) - Instanz - Typen), z.B. IP im Router/Endsystem, oder die z.B. verschiedene Protokolle für eine Schicht implementieren. Eine Kopie einer (N) - Instanz wird Vorkommnis der (N) - Instanz genannt.
- Partnerinstanzen (Peer-Entities)
 - Instanzen einer Schicht.
 - Partnerinstanzen erfüllen Funktionen eines Dienstes durch Datenaustausch.

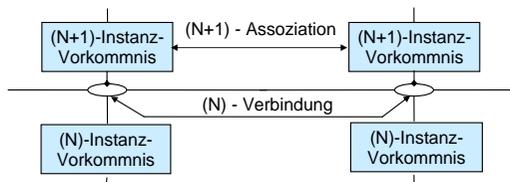


Generische OSI-Kommunikationseinheiten



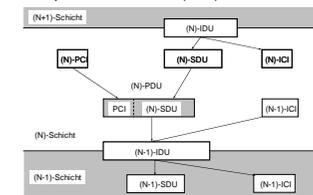
Verbindung und Assoziation

- (N) - Assoziation (Association)
 - Kooperative Beziehungen zwischen zwei (N)-Instanz-Vorkommnissen. Dazu gehört Verwaltung von Zustandsinformationen.
 - (N)-Assoziation wird durch (N-1)-Verbindungen (oder (N-1)-verbindungslosen Dienst) unterstützt. Sie kann zeitlich nacheinander verschiedene (N-1)-Verbindungen verwenden.
- (N) - Verbindung (Connection)
 - Beziehung zwischen zwei (oder mehr) (N+1)-Instanz-Vorkommnissen auf Ebene der (N)-Schicht. Diese Beziehung wird mit Hilfe des (N)-Protokolls unterstützt.

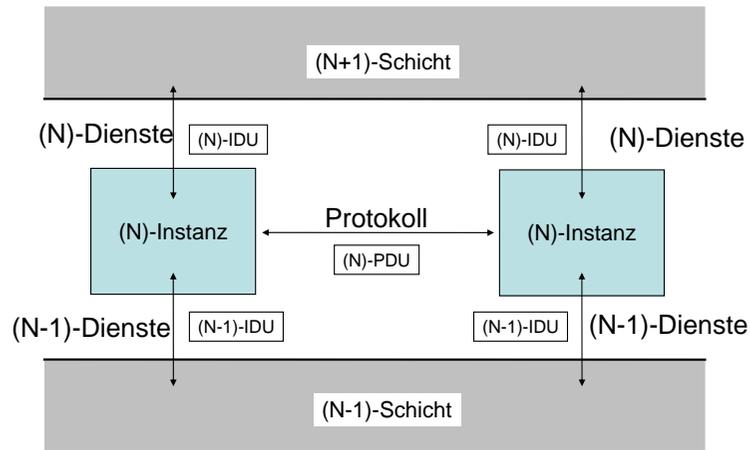


2.6.1. OSI-Kommunikationseinheiten, Beschreibung

- (N)-Schnittstellendateneinheiten
 - Interface Data Unit, IDU
 - Zwischen (N+1)- und (N)-Instanzen über einen (N)-SAP ausgetauschte Dateneinheit.
 - Setzt sich zusammen aus (N)-ICI und (N)-SDU.
- (N)-Schnittstellenkontrollinformation
 - Interface Control Information, ICI
 - Zwischen (N)-Schicht und (N+1)-Schicht ausgetauschte Parameter zur Steuerung von Dienstfunktionen (z.B. Adressen).
- (N)-Dienstdateneinheiten
 - Service Data Unit, SDU
 - Daten, die transparent zwischen (N)-SAPs übertragen werden.
- (N)-Protokollkontrolldaten
 - Protocol Control Information, PCI
 - Daten, die zwischen (N)-Instanzen ausgetauscht werden, um die Ausführung von Operationen zu steuern (z.B. Folgenummern o.ä.).
- (N)-Protokolldateneinheit
 - Protocol Data Unit, PDU
 - Dateneinheit, die zwischen (N)-Instanzen unter Benutzung eines Dienstes der (N-1)-Schicht ausgetauscht wird.
 - Zusammengesetzt aus (N)-PCI und (N)-SDU.
 - Entspricht somit der (N-1)-SDU.



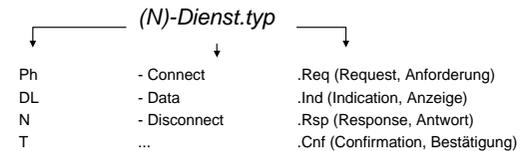
Kommunikationsmodell - OSI-Systeme



2.6.2. Bezeichnungskonventionen

- (N)-Schicht
 - A -Schicht: Anwendungsschicht (Application Layer)
 - P -Schicht: Darstellungsschicht (Presentation Layer)
 - S -Schicht: Kommunikationssteuerungsschicht (Session Layer)
 - T -Schicht: Transportschicht (Transport Layer)
 - N -Schicht: Vermittlungsschicht (Network Layer)
 - DL -Schicht: Sicherungsschicht (Data Link Layer)
 - Ph -Schicht: Bitübertragungsschicht (Physical Layer)

- (N)-Dienstprimitive



- Dienstprimitive in der A-Schicht werden gemäß ihres Application Service Element (ASE) benannt.

Die OSI-Schichten im Überblick

| | | |
|--------------------------------------|-----------------------------|--------------------|
| Anwendungsschicht | Schicht 7 (A - Schicht) | Application Layer |
| Darstellungsschicht | Schicht 6 (P - Schicht) | Presentation Layer |
| Kommunikations- steuerungsschicht | Schicht 5 (S - Schicht) | Session Layer |
| Transportschicht | Schicht 4 (T - Schicht) | Transport Layer |
| Vermittlungsschicht | Schicht 3 (N - Schicht) | Network Layer |
| Sicherungsschicht | Schicht 2 (DL - Schicht) | Data Link Layer |
| Bitübertragungsschicht | Schicht 1 (Ph - Schicht) | Physical Layer |

Dienstprimitive

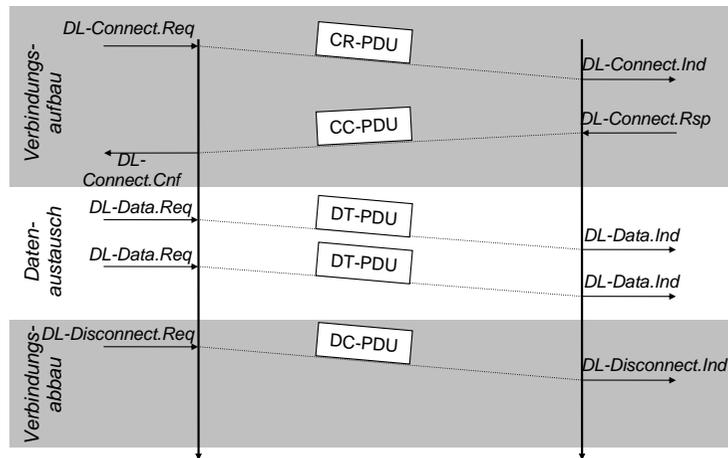
- Die Benennung eines Dienstprimativs besteht aus folgenden Komponenten:

| Name der Schicht/Anwendung | Dienstleistung | Ereignistyp | Parameter |
|----------------------------|-----------------------|--------------------|------------|
| Physical (Ph) | Connect (Con) | Request (Req) | (beliebig) |
| Data Link (DL) | Data (Dat) | Indication (Ind) | |
| Network (N) | Release (Rel) | Response (Rsp) | |
| Transport (T) | Abort (Abo) | Confirmation (Cnf) | |
| HTTP | Provider Abort (PAbo) | | |
| FTP | Disconnect (Dis) | | |
| ... | ... | | |

- Beispiel:
 - T-Con.Req(Adressen) = Verbindungsaufbauanforderung an der Schnittstelle zum Transportdienst
 - HTTP-Get.[Req](URL) = Anforderung der HTML-Seite, die durch URL identifiziert wird



Bezeichnungskonventionen am Beispiel



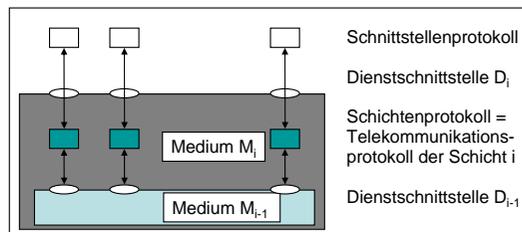
Protokollmechanismen

- Ein Protokollmechanismus ist ein Verfahren, welches abgeschlossene Teilfunktion innerhalb des Protokollablaufs beschreibt: generischer Charakter (ähnlich 'Systemfunktion').
- In verschiedenen Kommunikationsarchitekturen verwendet.
- Oft in mehreren Protokollen/Schichten einer Kommunikationsarchitektur anzutreffen.
 - Multiplexen / Demultiplexen
 - Teilung / Vereinigung
 - Segmentieren / Reassemblieren
 - Blocken / Entblocken
 - Verkettung / Trennung
 - (Mehrfach-)Kapselung
 - Fehlerbehandlung
 - Sicherung (ggf. fehlererkennend)
 - Sequenzüberwachung
 - Quittierung (Acknowledgement)
 - Zeitüberwachung (Timeout)
 - Wiederholen; Rücksetzen
 - Flusskontrolle (Sliding window)
 - Routing (Wegewahl, Weiterleiten)
 - Medienzuteilung für geteilte Medien
 - Synchronisation
 - Adressierung
 - Verbindungsverwaltung
 - Datentransfer



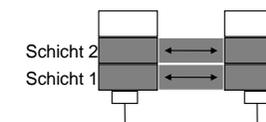
Protokoll: Modelle

- Überbrückung funktionaler und qualitativer Unterschiede zwischen D_{i-1} und D_i
- Art und Weise der Erbringung der Dienste D_i durch Instanzen I_i auf Basis der Dienste D_{i-1}
- Nebenläufiger Algorithmus
- Verteilter Algorithmus, wobei Dienste D_{i-1} das Zusammenwirken der I_i -Instanzen ermöglichen
- Berücksichtigung der Auswirkungen von Störungen in D_{i-1}
- Beschreibung: i.allg. nur 2 Instanzen, Automatenmodell, Weg-Zeit-Diagramm



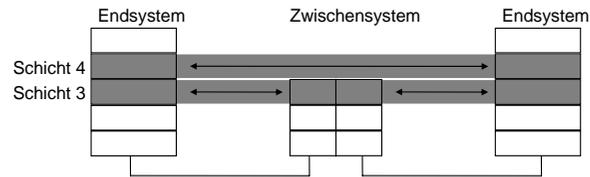
2.6.3. Charakterisierung der Schichten Bitübertragungsschicht und Sicherungsschicht

- Bitübertragungsschicht (Schicht 1)
 - ungesicherte Verbindung zwischen Systemen
 - Übertragung unstrukturierter Bitfolgen über physikalisches Medium
 - umfasst u.a. physikalischen Anschluss, Umsetzung Daten \leftrightarrow Signale
 - Normung vor allem der physikalischen Schnittstelle Rechner/Medien
- Sicherungsschicht (Schicht 2)
 - gesicherter Datentransfer
 - Zerlegung des Bitstroms (Schicht 1) in Rahmen (Frames)
 - Fehlererkennung und -behandlung
 - Protokollmechanismen: Quittierung, Zeit-/Sequenzüberwachung, Wiederholen/Rücksetzen

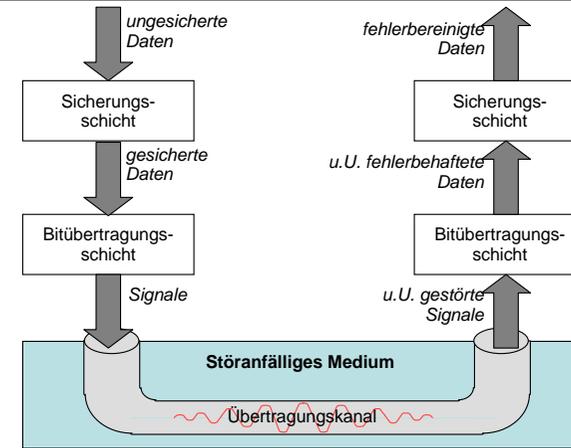


Vermittlungsschicht und Transportschicht

- Vermittlungsschicht (Schicht 3, auch 'Netzwerkschicht')
 - verknüpft Teilstreckenverbindung zu Endsystemverbindungen
 - Wegewahl (Routing) bei Vermittlung, Staukontrolle
 - evtl. aufgeteilt in 'Internetzwerk-/Subnetz-/Routing-'Subschichten
 - verbindungslos oder -orientiert
- Transportschicht (Schicht 4)
 - Adressierung von Transportdienstbenutzern
 - Datentransfer zwischen Benutzern in Endsystemen
 - bietet Transparenz bzgl. Übertragungs- und Vermittlungstechnik, Subnetzen
 - verbindungsorientiert, ggf. -los



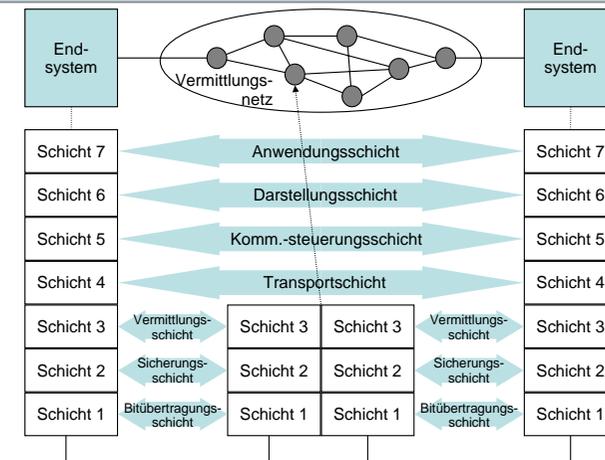
Daten und Signale



Anwendungsorientierte Schichten

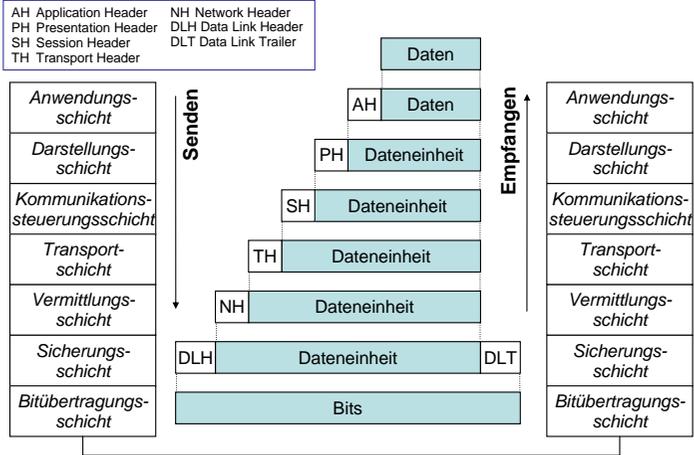
- Kommunikationssteuerungsschicht (Schicht 5)
 - Ablaufsteuerung und -koordinierung (Synchronisation im weitesten Sinne)
 - Sitzung (Session)
 - ergibt erst Sinn bei Verwendung durch den Benutzer
- Darstellungsschicht (Schicht 6)
 - behandelt die Darstellung von Informationen (Syntax) für den Datentransfer
 - Marshalling
 - Prozess des Packens von Daten in einen Puffer, bevor dieser über die Leitung übertragen wird. Dabei werden nicht nur Daten verschiedenen Typs gesammelt, sondern diese werden auch in eine Standard-Repräsentation umgewandelt, die auch der Empfänger versteht.
- Anwendungsschicht (Schicht 7)
 - macht dem OSI-Benutzer Dienste verfügbar
 - stellt verschiedene Dienste zur Verfügung, je nach Anwendung, z.B.
 - Dateitransfer
 - zuverlässiger Nachrichtenaustausch
 - entfernter Prozeduraufruf

OSI: Die 7 Schichten





Einkapselung von Daten

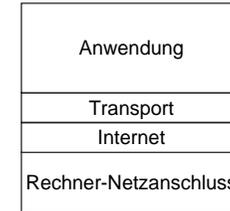


OSI und Internet

OSI-Referenzmodell



Internet-Referenzmodell

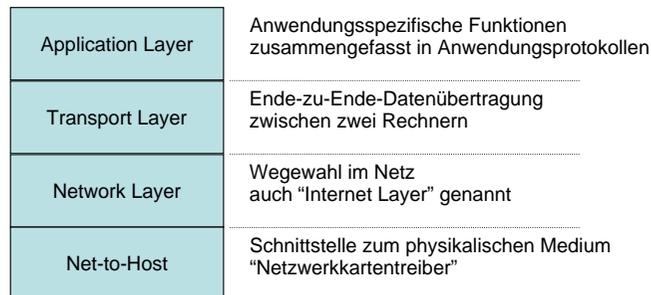


Unterschiede:

- Aufgaben der OSI-Schichten 5 und 6 werden beim Internet-Referenzmodell als Teil der Anwendung betrachtet.
- Die OSI-Schichten 1 und 2 sind zu einer den Anschluss des Rechners an das Kommunikationsnetz beschreibenden Schicht zusammengefasst.



Internet-Referenzmodell



Gegenüber ISO/OSI sind die drei anwendungsorientierten Schichten zu einer einzigen Schicht zusammengefasst.



2.7. Protokollspezifikation mit SDL



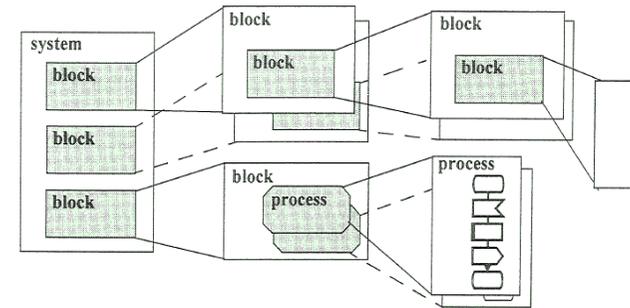


Specification and Description Language (SDL)

- Formale Sprache zur Beschreibung und Spezifizierung von Kommunikationssystemen
- Standard der ITU (früher: CCITT) (1984, 1988, 1992)
 - ITU = International Telecommunications Union
 - CCITT = Comité Consultatif International Téléphonique et Télégraphique
- Ziele:
 - Beschreibung des Verhaltens bestehender Systeme
 - Spezifizierung des Verhaltens neuer Systemkonzepte
- Verwendung u.a. bei der Spezifikation digitaler, leitungsvermittelter Systeme:
 - ISDN (Integrated Services Digital Network)
 - SS7 (Signaling System No 7)



Hierarchische Strukturierung in SDL



Aus König: SDL, Kap. 8

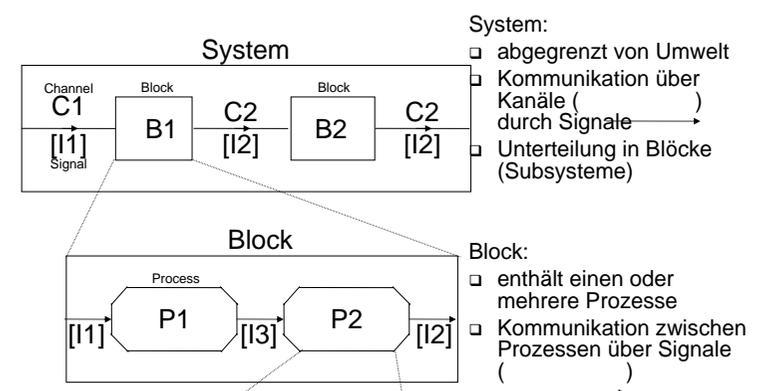


Eigenschaften von SDL

- Prozess als Grundelement
 - erweiterter endlicher Automat (Extended Finite State Machine - EFSM)
 - kommuniziert mit anderen Prozessen durch den Austausch von Nachrichten (Signalen) über Verbindungswege (Kanäle)
 - mehrere Prozesse arbeiten parallel und existieren gleichberechtigt nebeneinander
- Vordefinierte und benutzerdefinierte Datentypen
- Zwei äquivalente Darstellungsformen:
 - SDL/GR (Graphical Representation)
 - SDL/PR (Phrase Representation)
- Vorteile einer formalen Sprache
 - Exakte Spezifizierung
 - Möglichkeit von Werkzeugen - Editoren, Simulatoren, Prototyp-Generatoren, Testfall-Generatoren, Werkzeuge zur formalen Verifikation
 - Generatoren (Compiler) zur direkten Übersetzung von SDL in ausführbare Programme oder Programmgerüste

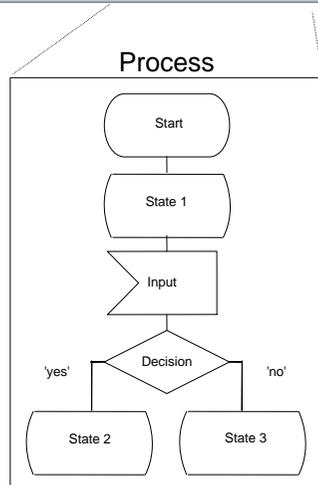


Hierarchische Strukturierung in SDL





Hierarchische Strukturierung in SDL



Prozess:

- kommunizierende *Extended Finite State Machine* (EFSM)
- Zustände, Übergänge, Aufgaben
- weitere Unterteilung in Prozeduren und Macros



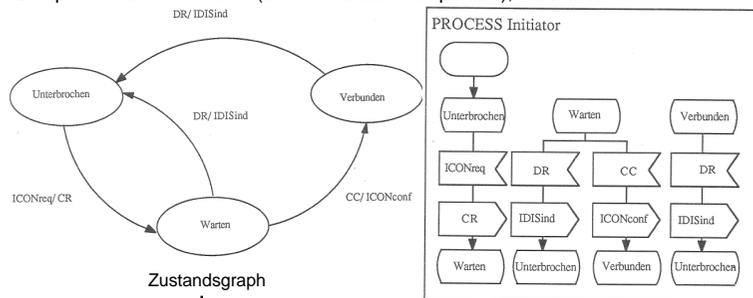
Prozesse in SDL

- Prozesse auf Basis erweiterter endlicher Automaten (EFSM):
 - endliche Zustandsanzahl und vorgegebene Zustandsübergänge
 - Eingangssignale lösen Zustandsübergänge aus
 - Aufgaben werden während eines Zustandsübergangs ausgeführt, z.B. auch Aussendung von Ausgangssignalen an andere Prozesse
 - eine Eingabewarteschlange puffert eingehende Nachrichten zwischen, falls Prozess sich gerade in einem Zustandsübergang befindet
 - es kann mehrere Instanzen eines Prozesses geben
- Erzeugung von Prozessen
 - bei Systemstart
 - zur Laufzeit durch andere Prozesse (CREATE)
- Beendigung von Prozessen
 - bei Erreichen eines STOP-Knotens



Übersetzbarkeit von Automaten in SDL-Graphen

Beispiel ⇒ InRes-Protokoll (InRes= Initiator-Responder), c.f. Folie 81



Zustandsgraph

Prozess in SDL/GR

Signale von/zu Dienstnehmer

- ICONreq: InRes-Connection-Request
 - ICONconf: InRes-Connection-Confirm
 - IDISreq: InRes-Disconnection-Request
 - IDISind: InRes-Disconnection-Indication
- Signale von/zu entfernter Instanz
- CC, DR, ...

aus: Hogrefe, „ESTELLE, LOTOS und SDL“, Springer Compass, 1989, S.121ff



Symbole im Prozessgraphen

| Symbol | Bedeutung des Knotens |
|--------|--|
| | Start Node (Startknoten): • kennzeichnet Beginn eines Prozesses • enthält Name des Prozesses |
| | State Node (Zustandsknoten): • für einen oder mehrere Zustände • enthält den/die Zustandsnamen |
| | Task Node (Aufgabenknoten): • zwischen zwei Zuständen • führt Befehle aus • enthält Namen und optional die Befehlsabfolge oder informellen Text |



Symbole im Prozessgraphen

| Symbol | Bedeutung des Knotens |
|--------|---|
| | Create Request Node: <ul style="list-style-type: none"> • erstellt und startet neue Prozessinstanz innerhalb eines Übergangs • wohldefiniert, enthält Name des Prozesses und seine Parameter |
| | Stop Node: <ul style="list-style-type: none"> • beendet die Prozessinstanz |
| | Decision Node: <ul style="list-style-type: none"> • ermöglicht Auswahl zwischen alternativen Pfaden innerhalb eines Übergangs • enthält eine Bedingung oder Abfrage • Antworten kennzeichnen Pfade/Alternativen |



Symbole im Prozessgraphen

| Symbol | Bedeutung des Knotens |
|--------|---|
| | Flow Line: <ul style="list-style-type: none"> • Pfad (Kante), um zwei Symbole (Knoten) miteinander zu verbinden |
| | Input Node (In-Connector): <ul style="list-style-type: none"> • markiert die Stelle, an der der Pfad von gleichnamigem Out-Connector weitergeht |
| | Output Node (Out-Connector): <ul style="list-style-type: none"> • markiert die Stelle, an der der Pfad unterbrochen wird, um an In-Connector weiterzulaufen |
| | Comment: <ul style="list-style-type: none"> • zusätzlicher informeller Text |



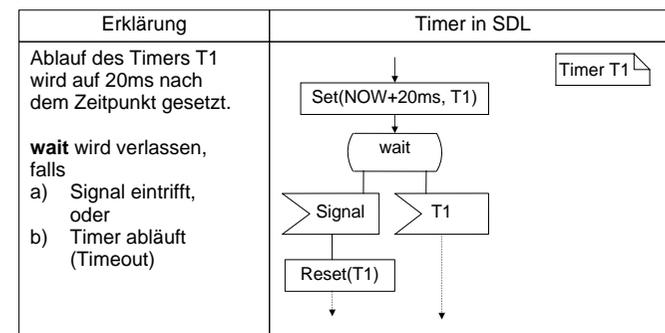
Symbole im Prozessgraphen

| Symbol | Bedeutung des Knotens |
|--------|---|
| | Save Node (SYNCHRONISATION): <ul style="list-style-type: none"> • verzögert ein Signal innerhalb eines Übergangs (ohne dass dazu ein Zustand existieren muss) • enthält gespeicherte Signale |
| | Input Node: <ul style="list-style-type: none"> • wartet auf den Erhalt eines oder mehrerer Signale innerhalb eines Übergangs • enthält den/die Signalnamen |
| | Output Node: <ul style="list-style-type: none"> • sendet ein oder mehrere Signale innerhalb eines Übergangs • enthält den/die Signalnamen und optional Zielprozessname/Kommunikationspfad |



Zeitverhalten

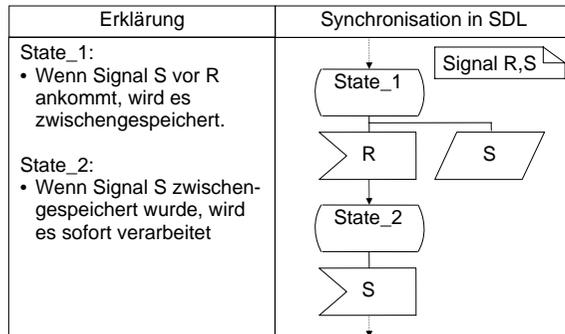
- Zeitverhalten spielt eine große Rolle in der Telekommunikation
- Einführen von Timer-Prozessen:
 - gibt vor, wie lange ein Zustand maximal gehalten wird, bis eines der erwarteten Eingangssignale eintrifft
- Beispiel zur Verwendung eines Timers:





Signalverzögerung (Implicit Delays)

- Normalerweise Abarbeitung der Eingangssignale nach dem FIFO-Prinzip
- Bei gleichzeitigem Eintreffen zweier Eingangssignale zufällige Auswahl
- Reihenfolge für die Verarbeitung von Eingangssignalen kann durch SAVE-Knoten geändert werden
- Beispiel zur Verarbeitung zweier Signale R und S mit der Reihenfolge R,S:



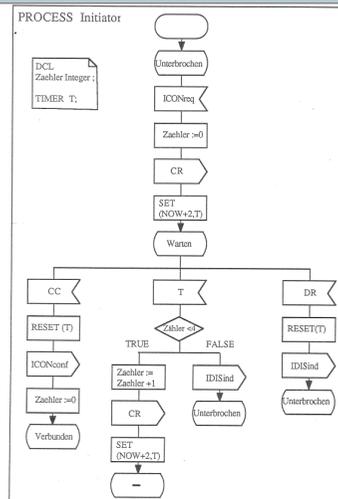
Standardisierung: Überblick

- Die Erfordernisse einer internationalen Telekommunikation erzwingen die Festlegung international gültiger Standards.
 - Standardisierung des Fernmeldewesens
 - Gremienarbeit mit gut strukturierten Lösungen, aber lange „Time To Market“
 - Weltweit einheitlich über Fernmelde-Betriebsgesellschaften (Telekommunikations-Dienstleister)
 - Beispiele: [ITU-T](#), [ETSI](#) (European Telecommunication Standards Institute)
 - Internet
 - Diskussionen direkt Betroffener und [IETF](#) (Internet Engineering Task Force) führen zu Standards
 - Beispielimplementierungen stehen im Vordergrund, daher sehr schnelle „Time To Market“
 - Herstellervereinigungen
 - Ebenfalls realisierungsorientiert mit relativ schneller „Time To Market“
 - Beispiele: [The Open Group](#) (ehemals OSF und X/Open), [ECMA](#) (European Computer Manufacturers Association), [ATM-Forum](#)



Beispiel

- InRes-Protokoll: Einfaches Protokoll zum Verbindungsaufbau zwischen zwei Protokollinstanzen
- Signale:
 - ICONreq: Verbindungsanforderung durch Benutzer
 - ICONconf: Verbindungsbestätigung an Benutzer
 - IDISind: Meldung eines Verbindungsabbruchs an den Benutzer
 - CR: Connection-Request-Nachricht an Gegenstelle
 - CC: Connection-Confirm-Nachricht von Gegenstelle
 - DR: Disconnect-Request-Nachricht von Gegenstelle

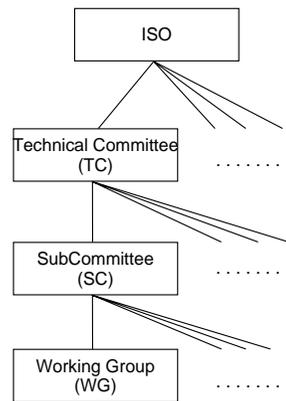


Standardisierung: Traditionelle Organisationen

- ITU** International Telecommunication Union (ehemals CCITT und CCIR)
Internationaler beratender Ausschuss für Telekommunikation
- CCITT** Consultative Committee on International Telegraphy and Telephony
Comité Consultatif International Télégraphique et Téléphonique
ehem. Internationaler beratender Ausschuss für Telefon und Telegrafie
(neue Bezeichnung: ITU-T)
- CCIR** Consultative Committee on International Radio
ehem. Internationaler beratender Ausschuss für den Funkdienst
(neue Bezeichnung: ITU-R)
- ISO** International Organization for Standardization
(ISO griech. „gleich“)
Internationale Organisation für Standardisierung
ISO koordiniert die internationale Normungsarbeit außerhalb des Telekommunikations-Bereichs.
- DIN** (Deutsches Institut für Normung) ist deutscher Partner der ISO.



Standardisierung: Beispiel ISO



WG-Meetings:

Alle 6-9 Monate, damit die nationalen Organisationen Einverständnis mit den Konzepten erreichen. Dann startet der Standardisierungsprozess:

Standardisierungsprozess:

DP: Draft Proposal

DIS: Draft International Standard

IS: International Standard

Das Fortschreiten auf eine höhere Stufe erfolgt durch eine internationale Abstimmung und die Einarbeitung der Kritik der „Nein“-Stimmen.

→ sehr langer Prozess!



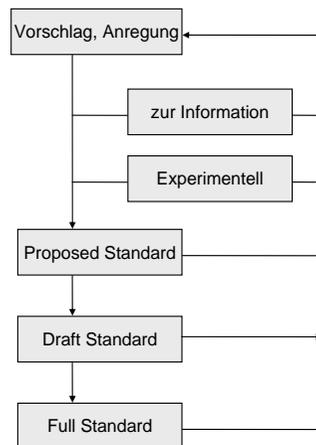
Standardisierung: RFC - Beispiele

- ❑ RFC 768 User Datagram Protocol (UDP), August 1980
 - ❑ RFC 791 Internet Protocol (IP), Sept. 1981
 - ❑ RFC 792 Internet Control Message Protocol (ICMP) Sept. 1981
 - ❑ RFC 793 Transmission Control Protocol (TCP), Sept. 1981
 - ❑ RFC 959 File Transfer Protocol (FTP), Oktober 1985
 - ❑ RFC 997 Internet Numbers, März 1987
 - ❑ RFC 3261 SIP: Session Initiation Protocol, Juni 2002
 - ❑ RFC 4509 Use of SHA-256 in DNSSEC Delegation Signer (DS) Resource Records (RRs), Mai 2006
- ❑ Weiter Informationen unter www.ietf.org!



Standardisierung: Beispiel Internet

- ❑ Der Standardisierungsweg geht über die Internet Engineering Task Force (IETF).
- ❑ Die Internet Engineering Steering Group (IESG) steuert die Diskussionen.
- ❑ Allgemein akzeptierte Arbeitsdokumente (Internet Drafts) erhalten permanenten Status (Request for Comments, RFC)
- ❑ Mögliche Ergebnisse:
 - Standard Track RFC (Proposed/Draft/Full Standard)
 - Experimenteller RFC
 - RFC zur Information
- ❑ Bereits ab dem Status Draft Standard müssen mindestens zwei interoperable, unabhängig voneinander entwickelte Implementierungen vorhanden sein.



Standardisierung: RFC – Beispiele (2)

- ❑ RFC 1149—Standard for the transmission of IP datagrams on Avian Carriers. D. Waitzman. 1 April 1990. Updated by RFC 2549; see below. A deadpan skewering of standards-document legalese, describing protocols for transmitting Internet data packets by homing pigeon.
- ❑ RFC 2322—Management of IP numbers by peg-dhcp. K. van den Hout et al. 1 April 1998.
- ❑ RFC 2324—Hyper Text Coffee Pot Control Protocol (HTCPCP/1.0). L. Masinter. 1 April 1998.
- ❑ RFC 2549—IP over Avian Carriers with Quality of Service. D. Waitzman. 1 April 1999. Updates RFC 1149, listed above.
- ❑ RFC 3251—Electricity over IP. B. Rajagopalan. 1 April 2002.
- ❑ RFC 3514—The Security Flag in the IPv4 Header (Evil Bit). S. Bellovin. 1 April 2003.
- ❑ RFC 4824—The Transmission of IP Datagrams over the Semaphore Flag Signaling System (SFSS). Jogi Hofmueller, Aaron Bachmann, IOhannes zmoelnig. 1 April 2007.