Grundlagen der Informatik

- Logische und mathematische Grundlagen
- Digitale Daten
- Computerprogramme als Binärdaten
- Betriebssysteme
- Rechnernetzwerke
  - Protokolle und Schichtenmodell
  - Local Ara Networks (LAN)
  - Internet Protocol (IP)
  - Transmission Control Protocol (TCP)
  - Adressierung im Internet (DNS und URLs)

Rechnernetzwerke

- Aus Sicht eines Computers (bzw. seines Betriebssystems) ist ein Kommunikationsnetzwerk nichts anderes als eine andere Art von Ein-/Ausgabegerät.

  (Am ehesten vergleichbar einem Modem, also einer Ein-/ Ausgabeeinheit für Telefonübertragungen).

- Allerdings ist die Kommunikation wesentlich komplexer.

  → *Kommunikationsprotokolle* zur unmissverständlichen Verständigung zwischen Computern sind notwendig.

- Die Abgrenzung, welche beteiligten Dienste noch zum Betriebssystem gehören und welche Dienste eigenständig sind, ist willkürlich.

  → Wir machen hier der Einfachheit halber keinen Unterschied, d.h. zählen alles mehr oder weniger zum Thema Betriebssysteme im weitesten Sinne.

Netzwerke und Kommunikationsprotokolle

- Jeder Computer (*Netzwerk–Knoten*) in einem Netzwerk hat eine *Netzwerkadresse*, die im gesamten Netzwerk nur einmal vergeben ist.
- Eine Botschaft (z.B. E-Mail), die von einem Computer zu einem anderen durch ein Netzwerk hindurch verschickt werden soll, wird üblicherweise in kleine *Pakete* (*Datagramme*) zerlegt, die einzeln verschickt werden.
  - An jedes Paket wird vorne und/oder hinten weitere Zusatzinformation angehängt.

    → Eine Art ”Briefumschlag“ für die eigentlich zu übertragenden Daten.
  - Minimal notwendige Zusatzinformation:

    ◊ Netzwerkadresse des Zielknotens,

    ◊ Identifikation des Prozesses, der das Datenpaket auf dem Zielknoten in Empfang nehmen und weiterbehandeln soll.
- Ein Regelwerk für die Formatierung von Datenpaketen nebst Zusatzinformationen bezeichnet man als *Kommunikationsprotokoll* (oder kurz *Protokoll*).

Schichten-Modell

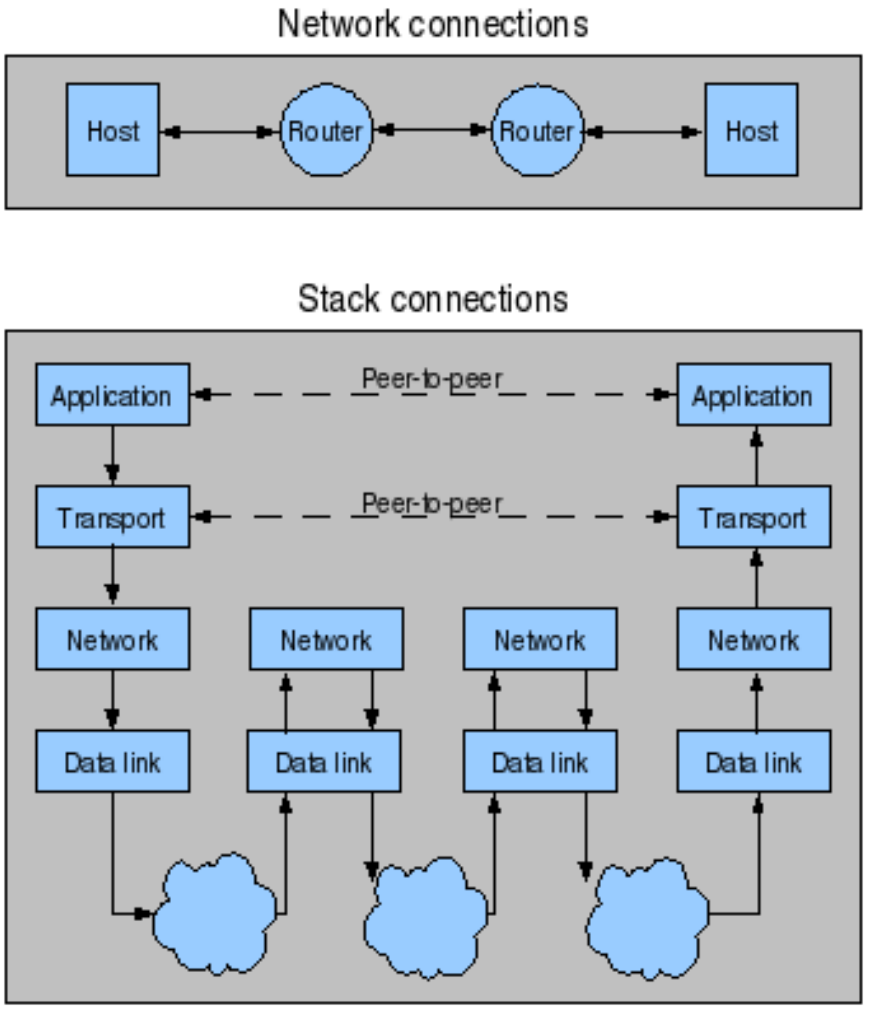
- TCP/IP/Ethernet ist ein Beispiel für eine generelle Strategie zur Entwicklung von Dienstprogrammen:
  - Die Gesamtaufgabe wird gedanklich in verschiedene Abstraktionsebenen (Layer, Schichten) zerlegt.
  - Jede Abstraktionsebene wird durch einen möglichst autonom agierenden Teildienst erledigt
  - die tieferliegenden Schichten werden von höherliegenden Schichten genutzt.
- Vorteile:
  - Die Implementation einer Anzahl separater Teildienste ist meist wesentlich einfacher als die Implementation eines monolithischen Dienstes.
  - Für die einzelnen Teildienste kann es mehrere Alternativen geben, die ohne Effekt auf die anderen Teildienste ausgetauscht werden können.
- Nachteil:
  - Erhöhte Laufzeit durch erhöhten Kommunikationsaufwand.

  → Kann die Antwortzeiten spürbar erhöhen.

TCP/IP 5-Schichtenmodell

1. Physical Layer
   - physikalische Verbindung zwischen den Rechnern
   - ermöglicht die Übertragung von Bits zwischen den Rechnern
2. Data Link Layer
   - ermöglicht die Übertragung von Datenpaketen zwischen benachbarten Netzwerkknoten
   - z.B. Ethernet
3. Network Layer
   - ermöglicht die Übertragung von Datenpaketen zwischen beliebigen Netzwerkknoten
   - z.B. IP
4. Transport Layer
   - organisiert die Datenübertragung in Pakete
   - z.B. TCP, UDP (no error recovery)
5. Application Layer
   - Protokolle, die auf einzelne Anwendungen maßgeschneidert sind.
   - z.B. HTTP, FTP (Web), SMTP, POP3 (E-mail), TELNET, SSH, etc.

TCP/IP 5-Schichtenmodell



- die Netzwerk-Verbindung zwischen den beiden Rechnern wird (hier) über zwei Zwischenknoten (Router) hergestellt
- über Application und Transport Layer kommunizieren die beiden Rechner (logisch gesehen) direkt miteinander
- die Verbindung über Network und Data Link Layer findet jedoch über mehrere Stationen statt.

Datenübertragung

- Auf jedem Computer in einem Netzwerk läuft ein Server, der nach diesem Protokoll Datenpakete von anderen Netzwerk–Knoten in Empfang nimmt bzw. an andere Netzwerk–Knoten weiterreicht.

  → Im folgenden <*Protokoll*>–*Server* genannt (Ethernet–Server, IP–Server, TCP–Server...).

- Ein Prozess kann diesem Server ein Datenpaket zur Weiterleitung durch das Netzwerk schicken.

  → Das Datenpaket wird vom Server gemäß Kommunikations-protokoll in den ”Briefumschlag“ gesteckt und losgeschickt.

- Wird ein Datenpaket von diesem Server aus dem Netzwerk in Empfang genommen, dann wird

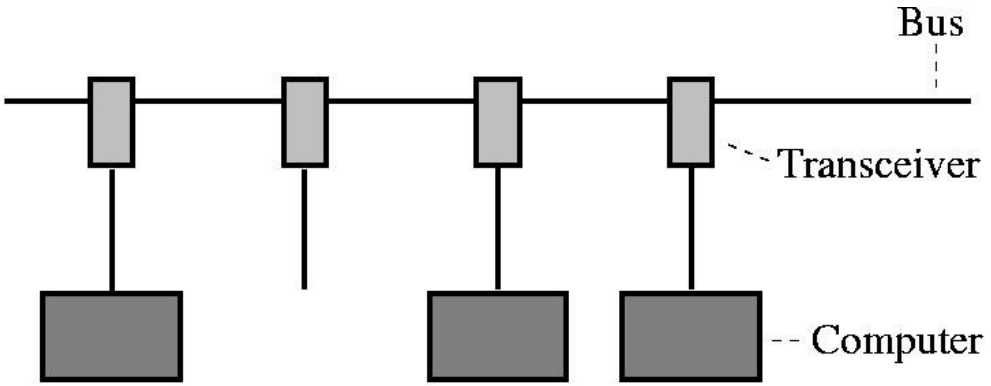
  ◊ der ”Briefumschlag“ des Kommunikationsprotokolls entfernt
  ◊ der eigentliche Inhalt an den spezifizierten Prozess weitergereicht.

Intranet / Local Area Network (LAN)

LANs sind oft so realisiert, dass mehrere oder sogar alle darin untereinander vernetzten Computer am selben ”Draht“ hängen.

Weit verbreitete Lösung als Beispiel: Ethernet

- Alle Computer hängen über *Transceiver* an einem einzigen Draht, dem *Bus*.



- Die Datenübertragung über den Bus läuft durch die Transceiver ungestört hindurch.

→ Computer können in Transceiver ein- und ausgeklinkt werden (bzw. sogar ausfallen), ohne dass der Datenfluss zwischen anderen Computern dadurch irgendwie berührt wird.

Carrier Sense Multiple Access (CSMA)

- Auf jedem Computer hört der Ethernet–Server den gemeinsamen Draht nach Datenpaketen ab und verarbeitet die Pakete, die an seine eigene Netzwerkadresse (*Ethernet–Adresse*) adressiert sind.
- Wenn der Ethernet–Server ein Datenpaket verschicken soll, horcht er den Draht ab und wartet bis die momentan laufende Datenübertragung beendet ist.
- In diesem Moment beginnt er sofort selbst mit der Datenübertragung.

CSMA with Collision Detection (CSMA-CD)

Problem: Was passiert, wenn zwei Ethernet–Server

- gleichzeitig darauf warten, dass der Draht wieder frei wird und
- mehr oder weniger gleichzeitig mit der Datenübertragung beginnen?

Antwort:

- Falls zwei oder mehr Ethernet–Server mehr oder weniger im selben Moment zu senden beginnen, erkennt jeder von ihnen das sofort, weil er natürlich weiterhin den Draht abhört.
- Jeder der betroffenen Ethernet–Server
  - ◊ bricht daraufhin seinen Übertragungsversuch sofort ab und
  - ◊ wartet eine zufällig bestimmte Zeitspanne, bis er von vorne beginnt, d.h. wieder auf die Beendigung der momentanen Datenübertragung wartet, um daraufhin selbst loszusenden.

Effekt

- Durch den (bewusst eingesetzten) Faktor Zufall kommt es nur in extrem ungünstigen Fällen zu einer erneuten Kollision derselben Datenpakete.
- Falls die Übertragung eines Datenpakets dennoch zu häufig wegen Kollisionen mit Übertragungsversuchen anderer Ethernet–Server abgebrochen werden musste,
  - ◊ bricht der Ethernet–Server den Übertragungsversuch ganz ab und
  - ◊ schickt eine Fehlermeldung an den sendenden Prozess (z.B. Mail–Server) zurück.
- Dieser Prozess wird dann typischerweise
  - ◊ das Paket sofort noch einmal versuchen zu verschicken oder
  - ◊ zu einem späteren Zeitpunkt noch einmal oder
  - ◊ dem sendenden Endbenutzer Bescheid geben oder
  - ◊ den Versuch der Verschickung ganz aufgeben.

Internet (vereinfacht)

- Das Internet besteht konzeptionell aus Sendekanälen jeweils von einem einzelnen Knoten zu einem anderen einzelnen Knoten.
- Ein Datenpaket wird von Computer zu Computer weitergereicht.
- Jeder Internet–Knoten *X* hält dazu eine Tabelle (*Routing–Tabelle*) mit Einträgen der Art

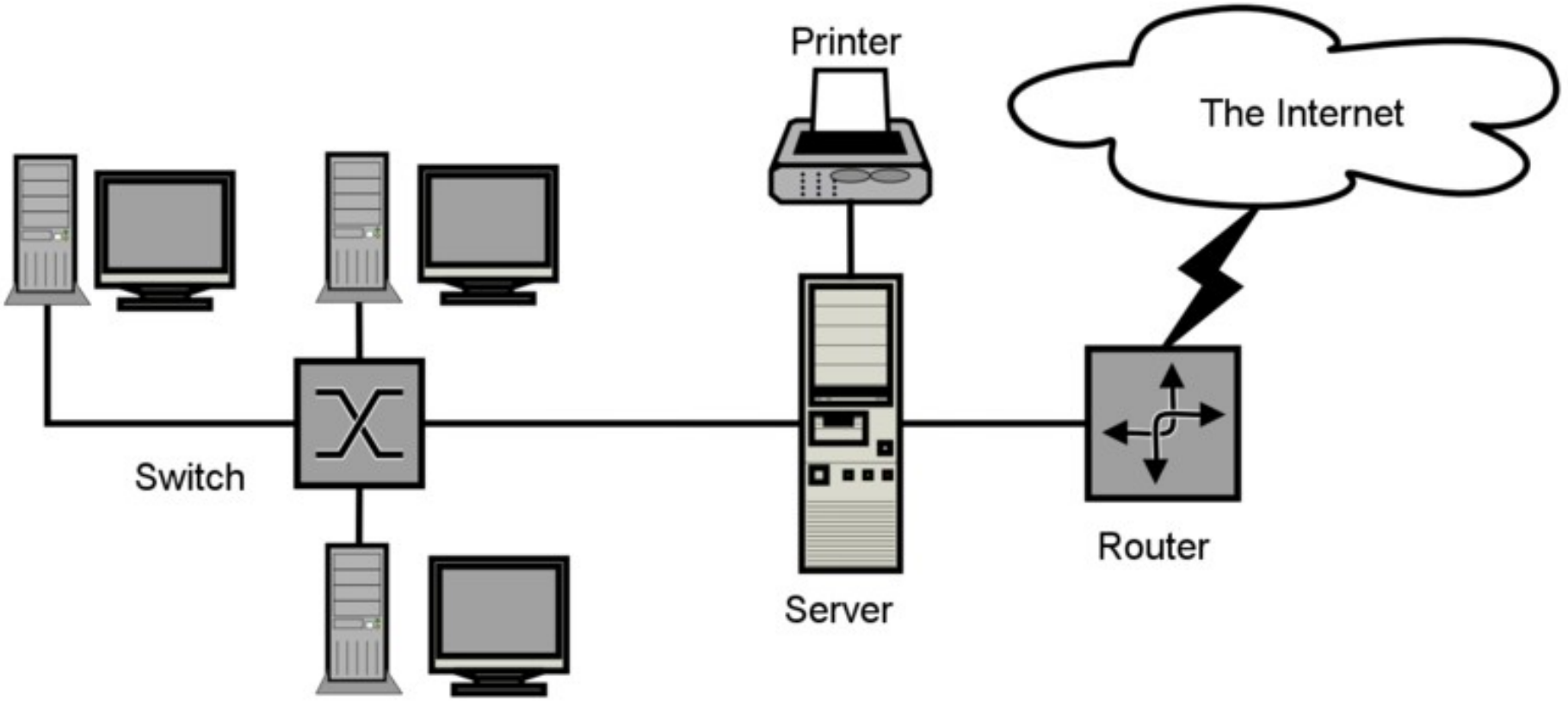
Intervall von Internet–Adressen

↔

Nächster zuständiger Internet–Knoten

  - In der Tabelle kommen nur Internet–Knoten vor, zu denen es einen direkten Sendekanal von *X* aus gibt.

Einfaches Rechnernetz mit Anbindung ans Internet



Auswertung der Routing-Tabelle

Wenn ein Internet–Knoten ein Datenpaket verschicken soll,

- entweder weil ein Prozess auf diesem Computer selbst das Datenpaket verschicken will
- oder weil das Datenpaket von einem anderen Internet–Knoten an diesen weitergereicht wurde,

dann

- wird die erste Zeile der Routing–Tabelle, bei dem die Internet–Adresse des Empfängers im Intervall enthalten ist, herausgesucht, und
- das Datenpaket wird an den in dieser Zeile eingetragenen Internet–Knoten auf dem direkten Sendekanal weitergeschickt.

Eigenschaften der Routing-Tabelle

- *Vollständigkeit*: Damit jedes Datenpaket behandelt werden kann, müssen alle Internet–Adressen durch die Vereinigung dieser Intervalle überdeckt sein.

- *Redundanz*: In der Regel ist jede Internet–Adresse in mehr als einem Intervall in der Routing–Tabelle enthalten.

  → Wenn die Verbindung über den ersten zuständigen Knoten nicht zustande kommt, kann es mit der zweiten Möglichkeit versucht werden, dann mit der dritten usw.

Organisation des Internets

- Die Wege der Datenübertragung vom Sender zum Empfänger
  - ◊ werden nicht durch zentrale Instanzen festgelegt,
  - ◊ sondern durch das oben beschriebene lokale ”Durchwursteln auf gut Glück“ von Knoten zu Knoten.
- *Konsequenz*: Totalausfälle sind recht unwahrscheinlich, da keine (verwundbaren) Zentralinstanzen nötig sind.
- *Allgemeine Erfahrung*:
  - ◊ Dieses ”systematische Chaos“ funktioniert auch im Normalbetrieb erstaunlich gut,
  - ◊ d.h. vermeidet ”Verkehrsstaus“ recht gut bzw. löst sie schnell wieder auf.

Angeblicher Hintergrund:

Das Internet ist ursprünglich auf das (militärisch motivierte) Ziel hin entworfen worden, dass möglichst viele Botschaften auch beim Zusammenbruch sehr weiter Netzwerkteile noch ihr Ziel erreichen (”atomkriegsicher“).

IP (Internet Protocol)

Basisprotokoll des Internets:

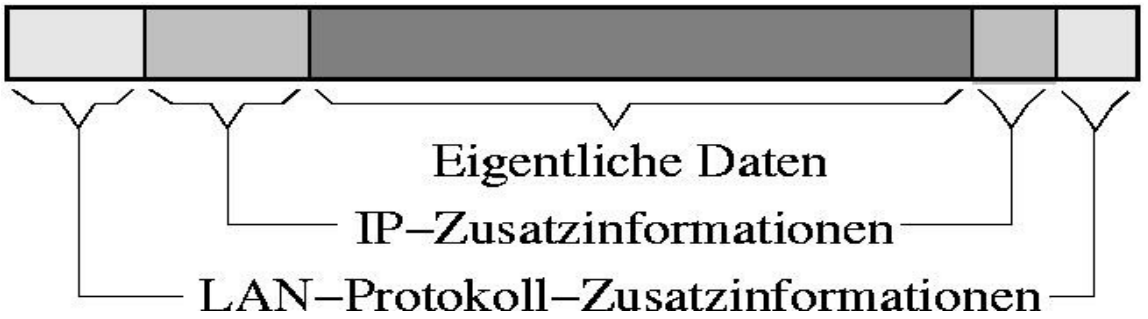
- Die einzelnen direkten Sendekanäle von Internet–Knoten zu Internet–Knoten sind typischerweise auf einem LAN–Protokoll (z.B. Ethernet) basiert.
- Die verschiedenen direkten Kanäle, zu denen ein Computer als Direktsender oder Direktempfänger gehört, sind in der Regel völlig separate LANs.

Problem:

- Das LAN–Protokoll kann nur die LAN–Adresse des allernächsten Zwischenknotens innerhalb des LANs als Zusatzinformation aufnehmen.
- Die Internet–Adresse des letztendlichen Ziels des Datenpakets muss auch irgendwie gespeichert werden.

Realisierung

- In den ”Briefumschlag“ des LAN–Protokolls wird ein zweiter ”Briefumschlag“ gesteckt, der (unter anderem) die Internet-Zieladresse als Zusatzinformation enthält.



- Standard für das Internet: das sogenannte *IP* (*Internet Protocol*)

  → Meist entgegen der Sprachlogik *IP–Protokoll* genannt.

  - Die Zusatzinformation im IP–Protokoll enthält u.a. die Internet–Adressen von Sender und Empfänger.

Realisierung

- Datenpakete, die über das Internet geschickt werden sollen, werden zunächst einmal vom IP–Server behandelt.

Verschicken:

- der IP–Server reicht das Datenpaket an den Server des LAN–Protokolls weiter
- mit der Identifikation des IP–Servers als empfangender Prozess (auf dem Empfänger im LAN).
- dieser IP-Server behandelt dann das Paket oder reicht es seinerseits weiter

Bemerkung:
Internet–Adressen werden meist *IP–Adressen* genannt.

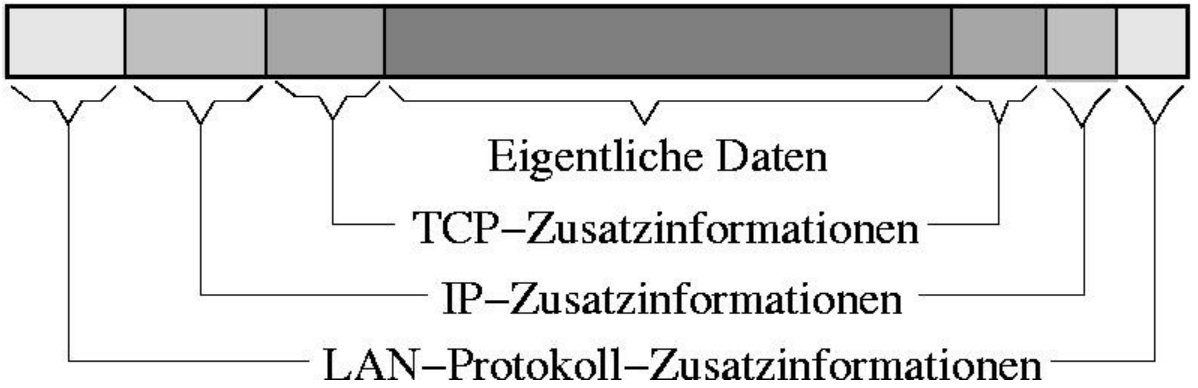
- eine IP-Adresse besteht aus 4 Bytes (32 bit), die zur beseren Lesbarkeit durch . von einander getrennt sind
- **Beispiel:** **`130.83.47.128`** (Web-Server der TU Darmstadt)

Probleme mit IP

- Ein Datenpaket wird einfach vom IP–Server (via LAN–Protokoll–Server) losgeschickt — und dann vergessen!
- Ob alle Datenpakete einer Botschaft planmäßig angekommen sind oder ob irgendwelche Datenpakete irgendwo hängengeblieben sind, lässt sich nicht mehr nachvollziehen.
- Die Datenpakete können auch auf unterschiedlichen Wegen durch das Internet gelaufen sein und in der falschen Reihenfolge ankommen.
- Datenpakete können durch Übertragungsfehler auch verfälscht ankommen.

TCP (Transmission Control Protocol)

- Das Datenpaket wird zusätzlich in einen dritten Briefumschlag im Innern der ersten beiden Briefumschläge gesteckt.
  → Standard für die meisten Internetdienste: *TCP*



TCP Zusatzinformationen:

- Eine laufende Nummer des Datenpakets innerhalb der Gesamtbotschaft, die bei der Zerlegung der Gesamtbotschaft in einzelne Datenpakete vergeben wird.
- *Port–Nummern* des Ausgangs- und Zielprozesses.
- *Checksumme* zum Erkennen von Übertragungsfehlern.

TCP Port Nummern

- *Erinnerung*: Der Prozess auf dem Empfänger, der die Botschaft eigentlich empfangen soll, muss natürlich spezifiziert sein.
  - Dafür ist im Internet das TCP–Protokoll zuständig.

- *Genauer*: Jeder solche Prozess muss eine numerische Kennung haben, die *Port–Nummer*.
  - Überall verbreitete Standarddienste wie E-Mail, Telnet, Ftp, Http (=WWW–Server) haben standardisierte Port-Nummern.
  - *Konkretes Beispiel*: Http hat Port–Nummer 80.

Checksummen

- Die Berechnung von Checksummen ist eine Klasse von mathematischen Verfahren zur Aufdeckung von Übertragungsfehlern.
- Solche Verfahren basieren auf der Interpretation des Datenpakets als eine Sequenz von Binärzahlen fester Länge.
- Mit überwältigender Wahrscheinlichkeit hat ein durch Übertragungsfehler verfälschtes Datenpaket eine andere Checksumme.

  → Wenn der Empfänger die Checksumme des empfangenen Datenpakets berechnet und mit der mitgelieferten Checksumme vergleicht, werden Übertragungsfehler mit sehr großer Sicherheit durch Nichtübereinstimmung der beiden Checksummen entdeckt.

- Zur Erläuterung betrachten wir beispielhaft das TCP–Verfahren.

TCP Checksummen

- Das Datenpaket wird als eine Sequenz

$$b_1, b_2 ..., b_n$$

von n 16–Bit–Binärzahlen interpretiert.

→ Ggf. wird das Datenpaket mit bis zu 15 Nullen aufgefüllt, wenn die Anzahl Bits im Datenpaket kein Vielfaches von 16 ist.

- *Notation*: Für $i = 1, 2, ..., 16$ sei $b_j[i]$ die i–te Stelle von b_j.

- Die Checksumme ist ebenfalls eine 16–Bit–Binärzahl.

- *Genauer*: Die *i*–te Stelle der Checksumme ist

  = 0 falls $b_1[i] + b_2[i] + \cdots + b_n[i]$ eine gerade Zahl ist,

  = 1 falls $b_1[i] + b_2[i] + \cdots + b_n[i]$ eine ungerade Zahl ist.

(Un-)Sicherheit des Verfahrens

- Offensichtlich kann es immer noch fehlerhaft übertragene Bits geben, die der Empfänger durch Vergleich von empfangener und selbst berechneter Checksumme nicht findet.

- Da sich je zwei fehlerhafte Bits an einer der 16 Stellen bei der Checksumme gegenseitig wegheben, passiert das genau dann, wenn
  an jeder der sechzehn Stellen eine gerade Anzahl von Fehlern im Datenpaket (einschließlich Checksumme) auftreten.

  → Insbesondere mindestens zwei Fehler.

- Dass genau zwei Fehler auftreten — und das auch noch an derselben der 16 Stellen — ist schon extrem unwahrscheinlich: Wenn zwei fehlerhafte Bits so speziell verteilt auftreten, ist die Wahrscheinlichkeit sehr hoch, dass noch weitere Fehler im Datenpaket sind.

(Un-)Sicherheit des Verfahrens

- Bei mehr als zwei fehlerhaften Bits ist die Wahrscheinlichkeit aber sehr gering, dass sie so ungewöhnlich verteilt sind, dass der Vergleich der Checksummen keinen Fehler findet.
- Daher kann man die verbleibende Restunsicherheit von Checksummen in der Praxis unbeschadet ignorieren.

Allerdings:

- Diese Überlegungen setzen voraus, dass fehlerhafte Bits durch zufälliges ”Hintergrundrauschen“ entstehen.
- Zum Beispiel bei periodisch wirkenden Störquellen gelten diese Überlegungen nicht mehr so ganz.

  → Obwohl es auch in diesem Fall schon mit dem Teufel zugehen müsste...
- Und bei gezielter, absichtlicher Störung der Übertragung kann man sowieso nichts mehr aussagen.

Botschaften verschicken mit TCP/IP

- Vor dem Versenden der Datenpakete ”sprechen“ sich der sendende und der empfangende TCP–Server mit Hilfe dreier Datenpakete ab:
  - ◊ Der Sender schickt ein TCP–Datenpaket zur Eröffnung einer Verbindung (notfalls wiederholt).
  - ◊ Falls der Empfänger bereit ist, schickt er eine positive Antwort zurück.
  - ◊ Der Sender schickt eine Empfangsbestätigung für die Antwort an den Zielknoten.
- Sobald der Sender seine Datenpakete übertragen hat, leitet er den Verbindungsabbau analog zum -aufbau durch den Austausch von drei Datenpaketen ein.

Handshake Protokolle

- Protokolle, die wie TCP (im Gegensatz zu Ethernet und IP) vorsehen, dass der Empfänger auch Nachrichten an den Sender schickt, heißen *Handshake Protokolle*.

- Der obige Austausch dreier Datenpakete ist das Kennzeichen von *Three–Way Handshake Protokollen*.

Zuverlässigkeit der Datenübertragung

- Für jedes korrekt empfangene Datenpaket (d.h. auch die Check-summe muss stimmen) schickt der Empfänger eine Empfangs-bestätigung mit der laufenden Nummer des Datenpakets an den Sender zurück.

- Falls der Sender diese Bestätigung nach einer gewissen Zeit immer noch nicht erhalten hat, schickt er das Datenpaket einfach noch einmal.

- Das tut er eine gewisse Anzahl von Malen, gibt dann schließlich auf und sendet statt dessen eine Fehlermeldung an den Prozess, von dem die zu versendenden Daten stammen.

- Wenn die Übertragung geklappt hat, leitet der empfangende TCP–Server die Datenpakete in der Reihenfolge ihrer laufenden Nummern an den eigentlichen Adressaten weiter.

Zuverlässigkeit (2)

- Wenn die Datenpakete nicht in dieser Reihenfolge beim empfangenden TCP–Server eintreffen, werden verfrühte Datenpakete von diesem TCP–Server zurückgehalten, bis alle Datenpakete mit kleinerer laufender Nummer eingetroffen sind.

- Bleiben die Daten nach einer gewissen Wartezeit unvollständig, ist die Übertragung gescheitert, und die bisher gesendeten Datenpakete werden vom Empfänger wieder ”vergessen“.

Anmerkung:

- Solche Checksummen werden sehr häufig angewandt.

- *Beispiel*: Die letzte Ziffer der ISBN–Nummer eines Buches ist eine solche Checksumme.

Domain Name System (DNS)

- Numerische IP–Adressen sind natürlich alles andere als benutzerfreundlich.

- Zur Verbesserung der Benutzerfreundlichkeit ist das *Domain Name System* (*DNS*) entwickelt worden.

- Spezielle Internet–Knoten (*DNS–Server*) speichern Tabellen, mit denen DNS–Adressen in IP–Adressen übersetzt werden können.

  → Die anderen Internet–Knoten müssen nur die Information halten, welchen DNS–Server sie für welche Adresse fragen können.

- Das DNS ist hierarchisch in *Top–Level–Domains*, *Subdomains*, *Subsubdomains* etc. gegliedert.

Beispiel

`ultra10.rbg.informatik.tu-darmstadt.de`

Erläuterungen:

- DNS–Top–Level–Domain ”de“ für Deutschland,
- DNS–Subdomain ”`tu-darmstadt`“,
- DNS–Subsubdomain ”`informatik`“ für den Fachbereich Informatik,
- DNS–Subsubsubdomain ”`rbg`“ für die Rechnerbetriebsgruppe am Fachbereich Informatik,
- Computer mit DNS–Namen ”`ultra10`“ in dieser Subsubsubdomain.

Beispiele für Top-Level Domains

- *Länder*:
  - `de`: Bundesrepublik Deutschland
  - `at`: Österreich (Austria)
  - `au`: Australien
  - `ch`: Schweiz (Confoederatio Helvetica)
  - `fr`: Frankreich
  - `uk`: Großbritannien (United Kingdom)
- *Sparten* (in erster Linie in den USA):
  - `com`: Firmen (commercial)
  - `edu`: Schulen und Hochschulen (education)
  - `gov`: US–Behörden (government)
  - `net`: Zentrale Netzadressen
  - `mil`: US–Militär
  - `org`: Nichtstaatliche Organisationen (z.B. wissenschaftliche)

Uniform Resource Locators

- Über das Internet lässt sich jede Informationsressource jedweder Art durch Adressen in einem allgemeinen Format ansprechen.
- Stichwort in der Literatur: *Uniform Resource Locator* (*URL*).
- *Beispiel*: Die Übersichtsseite zu Neuigkeiten am Fachbereich Informatik der TU Darmstadt hat die URL

```
http://www.informatik.tu-darmstadt.de:80/web/index.shtml
```

Aufbau von URLs

- *Allgemeine Grobstruktur von URLs*:
  - Zuerst das Protokoll (”`http`“),
  - nach ”`://`“ dann der Rechnername (oder auch seine IP-Adresse) („`www.informatik.tu-darmstadt.de`“)
  - optional dann eine Portnummer, mit Doppelpunktabgetrennt (”`:80`“)
  - nach einem “`/`“ schließlich die Adresse der Ressource auf diesem Rechner (”`web/index.shtml`“).
- Speziell ”`http`“ ist das gängige Standardprotokoll für WWW-Ressourcen (z.B. HTML–Seiten).

Beispiele für URLs

`http://www.allgemeineinformatik.de/index.php`
Dies bezeichnet die Homepage dieser Lehrveranstaltung. Der Port wurde an dieser Stelle weggelassen, das Hypertext Transfer Protocol (HTTP) ist standardmässig auf Port 80 aktiv.

`ftp://ftp.tu-darmstadt.de/pub/liesmich.txt`
Dies verweist auf die Datei `liesmich.txt` auf dem FTP-Server der TU. FTP steht für File Transfer Protocol. Der Port kann auch hier weggelassen werden, da der TU-Server unter dem Standard-Port für FTP, 21, zu erreichen ist.

`ipp://printers.tu-darmstadt.de:631/printers/drucker1`
Diese URL beschreibt einen Drucker, der über das Internet Printing Protocol (ipp) angesprochen wird. An dieser Stelle wurde der an sich optionale Port angegeben.