Rechnernetze & Netzwerkprogrammierung - VO-Vorbereitung

T. Auer

Latest Update: 16. Februar 2018

Inhaltsverzeichnis

1 ISO/OSI-Referenzmodell vs TCP-IP **3**
1.1 OSI-Modell . . . 3
1.1.1 Application Layer . . . 3
1.1.2 Presentation Layer . . . 3
1.1.3 Session Layer . . . 3
1.1.4 Transport Layer . . . 3
1.1.5 Network Layer . . . 4
1.1.6 Data Link Layer . . . 4
1.1.7 Physical Layer . . . 5
1.2 TCP-IP Modell . . . 5
1.2.1 Application Layer . . . 5
1.2.2 Transportlayer . . . 5
1.2.3 Vermittlungsschicht (IP) . . . 5
1.2.4 Network Layer . . . 6
1.3 Networklayer – Data Plane and Control Plane . . . 6

2 Wichtige Protokolle **6**
2.1 DHCP . . . 6
2.2 HTTP . . . 6
2.3 FTP . . . 7
2.4 SMTP . . . 8
2.5 POP3 . . . 9
2.6 IMAP . . . 9
2.7 SNMP . . . 10
2.8 DNS . . . 10
2.9 TCP . . . 11
2.10 UDP . . . 13
2.11 BGP . . . 14

2.12 OSPF . . . 15
2.13 SDN . . . 16
2.14 TDMA, FDMA, CDMA . . . 16
2.15 SALOHA . . . 17
2.16 CSMA/CD . . . 18
2.17 CSMA/CA . . . 18
2.18 ARP . . . 19
2.19 IEEE 802.11 Wireless LAN . . . 19

3 Networking in Java 20
3.1 Processes . . . 20
3.2 Threads . . . 20
3.3 Readers and Writers . . . 20
3.4 Sockets and Exceptions . . . 20

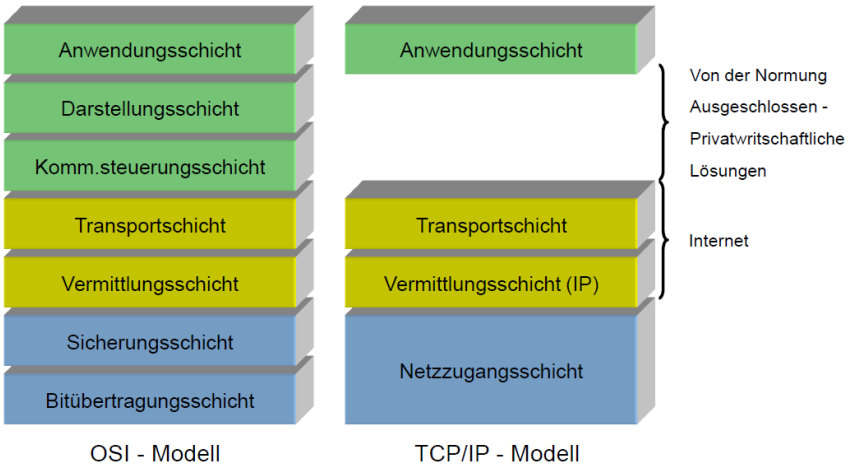
4 Begriffe und Definitionen 20
4.1 Weitere Übertragungstypen . . . 23
4.2 Berechnungen . . . 24
4.2.1 Graph Abstractions of the Network . . . 24
4.2.2 IP-Addressberechnungen . . . 25
4.2.3 Link Layer - Parity Checking . . . 26
4.2.4 Cyclic Redundancy Check . . . 26
4.2.5 CDMA Encoding/Decoding . . . 27

5 Typische Klausurfragen & Fallbeispiele 27
5.1 Probeklausur WS2016/2017, 3. Termin . . . 27
5.2 Frei erfundene Bsp. . . . 27
5.2.1 Theorie . . . 27
5.2.2 Berechnungen . . . 28
5.2.3 Wahr oder Falsch . . . 29

6 Typische Klausurfragen & Fallbeispiele – Lösungen 30
6.1 1. Klausur WS2017, 1. Termin . . . 30
6.2 Probeklausur WS2016/2017, 3. Termin . . . 31
6.3 Frei erfundene Bsp. . . . 35
6.3.1 Theorie . . . 35
6.3.2 Berechnungen . . . 36
6.3.3 Wahr oder Falsch . . . 36

7 Sources 42

1 ISO/OSI-Referenzmodell vs TCP-IP



1.1 OSI-Modell

1.1.1 Application Layer

Im OSI-Modell dient der Application Layer dem Userinterface. Es handled I/O, bietet Userfunktionen und initiiert Verbindungen zu unterliegenden Schichten.
Bsp.: Webbrowser, E-Mail Dienste, Instant Messaging.
Protokolle: HTTP, DNS, BGP, SMTP, POP3, IMAP

1.1.2 Presentation Layer

Wandelt systemabhängige Daten (ASCII, EBCDIC) in systemunabhängige Datenformate (ASN.1) um oder behandelt Datenkompression und Verschlüsselung.
Protokolle: Telnet, Tox, Network Data Representation, NetWare Core Protocol.

1.1.3 Session Layer

Steuert Verbindungen in Form von Prozesskommunikation zweier Systeme, zum Datenaustausch. Es handelt sich hierbei um einen semi-permanenten Dialog bestehend aus Anfragen und Antworten.
Hauptfunktionalitäten: Authentifizierung, Session Restoration.
Protokolle: Rmote Procedure Call Protocol, Point-to-Point Tunelling Protocol, Real-Time Transport Control Protocol, Session Control Protocol.

1.1.4 Transport Layer

In TCP/IP als auch im OSI-Modell: Realisiert Verbindung zwischen Quell und Zielhost mittels TCP und UDP. Es dient dem Zuordnen von Datenpaketen zu einer spezifischen Anwendung.

TCP	UDP
Congestion Control	Best Effort
Flow Control	Fast
'Handshake'	Low Overhead
Full-Duplex Data	Multicast
Connection-oriented	Connectionless
Point-to-Point	Point-to-X
Reliable stream	Unreliable
Pipelined	

Weitere Protokolle:

1. SCTP – Stream Control Transmission Protocol
2. TLS – Transport Layer Security: TCP+Verschlüsselung
3. DTLS – Datagram Transport Layer Security: TLS+UDP

1.1.5 Network Layer

Implementiert Internet Protocol, IP. Das Zustellen der Daten zum richtigen Empfänger ist dessen Aufgabe, e.g. das Routing, in sowohl Leitungsorientierten als auch Paketorientierten Verbindugnen. Zudem das Bereitstellen netzwerkübergreifender Adressen, Aktualisierung von Routingtabellen und Fragmentierung von Datenpaketen.

1. IP – Internet Protocol: Übertragung
2. IPsec – Internet Protocol Security: Sichere Datenverbindung
3. ICMP – Internet Control Message Protocol: Kontrollnachrichten, Teil jeder IP.
4. OSPF – Open Shortest Path First: Informationsaustausch zwischen Routern
5. BGP – Border Gateway Protocol: Informationsaustausch zwischen autonomen Systemen
6. IGMP – Internet Group Management: Definiert Multicast-Gruppen.

1.1.6 Data Link Layer

Segmentiert Pakete in Frames und checkt Prüfsummen. Man prüft die fehlerfreie Übertragung als auch Zugriff auf Übertragungsmedium, Bridge und/oder Switch. Nach IEEE ist sie in Logical Link Control und MAC aufgeteilt. (In der VO: Data Plane, Control Plane)

1. ARP - IPv4 Adressierung in Ethernet-Netzwerken (IPv6 – NDP)
2. Shortest Path Bridging

3. IEEE 802.11 - WLAN
4. IEEE 802.4 - Token Bus, 802.5 - Token Ring

1.1.7 Physical Layer

Wandelt Bits in ein angemessenes Signal zur Übertragung um und handled physikalische Übertragungsmöglichkeiten. Behandelt die unterliegenste Hardware: Repeater, Hubs, Leitungen, Stecker u.s.w.

1.2 TCP-IP Modell

1.2.1 Application Layer

Umfasst alle Protokolle die mit Anwendungsprogrammen zusammenarbeiten und Netzwerke zum Datenaustausch nutzen.

1. NFS – Network File System: Rechnerverbindungen, virtuelle Verbindung zwischen Festplatten.
2. DNS – Domain Name System: Umsetzung zwischen Domainnamen und IP-Addressen.
3. SMTP – Simple Mail Transfer Protocol: E-Mail Versand
4. FTP – File Transfer Protocol: Dateien externer Rechner übertragen, löschen, ändern oder umbenennen. Ports 20, 21.
5. Telnet: Remote login via Terminal, TCP-Port 23
6. IMAP – Internet Message Access Protocol: Zugriff auf E-Mail
7. POP3 – Post Office Protocol V3: E-Mail Abruf
8. Sowie: HTTPS, NTP, RTP, SNMP, SSH

1.2.2 Transportlayer

Behandelt End-to-End-Kommunikation. Nutzt TCP als auch UDP zum Datenaustausch.

1.2.3 Vermittlungsschicht (IP)

Umfasst die Weitervermittlungen von Paketen sowie Routing durch das Web, ermöglicht durch IP. Dual-Stacks erkennen Ipv4 oder Ipv6.

1.2.4 Network Layer

Festlegung von Datentransfer: Der Host muss einem Netzwerk zugehören. Es gilt allumfassend als Schicht zur Punkt-zu-Punkt-Datenübertragung.

1. Ethernet mit CSMA/CD: IEEE 802.3!
2. Token Bus - IEEE 802.4
3. Token Ring - IEEE 802.5
4. WLAN: IEEE 802.11
5. ARP – Adress Resolution Protocol: Adressumsetzung zwischen IP und MAC
6. RARP – Reverse Adress Resolution Protocol, deprecated.
7. PPP: Point-to-Point Protocol

1.3 Networklayer – Data Plane and Control Plane

Man differenziert den Network Layer grundlegend in Data Plane und Control Plane. Der **Dataplane** beschreibt alles rund um den lokalen Router, während der **Control Plane** das gesamte Netzwerk betrachtet. Der Dataplane implementiert das Forwarding und Datenfragmentierung, während die Control Plane das Routing beschreibt. Letzteres kann via Routing Algorithmen oder SDNs definiert werden.

2 Wichtige Protokolle

2.1 DHCP

Das Dynamic Host Configurations Protocol dient einem jeden Gerät sich an das Internet zu verbinden. Es ist teil der Anwendungsschicht. DHCP wird primär verwendet, wenn ein Gerät zu einem neuen System angeschlossen wird.

Ein DHCP-Request wird an den Router gesendet, welcher dem Gerät eine IP-Addresse zuweist. Zudem weiß der Klient nun die Adresse des DNS Servers als auch die IP Adresse des FIRST-HOP Routers.

→ DNS (IPs) & ARP (MACs) resolve following adresses.

2.2 HTTP

Das Hyper Text Transfer Protokol ist ein zustandsloses Protokoll zur Datenübertragung auf der Anwendungsschicht. Es gilt dabei Dokumente in einen Webbrowser zu laden.

1. HTTP/1.0
   Bei HTTP/1.0 wird vor jeder Anfrage eine neue TCP-Verbindung aufgebaut und nach der Datenübertragung wieder geschlossen. Hier müssen per geladenem Element 1 Verbindung aufgebaut werden.

2. HTTP/1.1
   In HTTP/1.1 kann der Klient durch den KeepAlive Headereintrag die Verbindung aufrecht erhalten. Mithilfe von HTTP-Pipelining kann somit mittels einer einzigen TCP-Verbindung alle Elemente eines Dokumentes anfordern. Dies fürt zu Performanzsteigerungen.

   Hinzu kommt der PUT-Befehl, womit man dem Server Daten senden kann, der DELETE-Befehl um diese Daten wieder zu löschen und eine TRACE-Methode zum Tracking von Packeten zum Server.

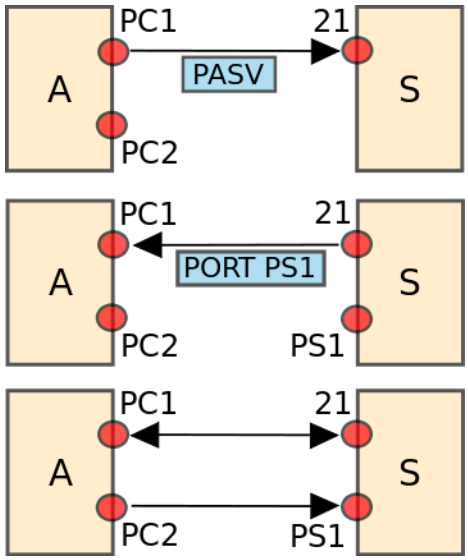
3. HTTP/2.0
   HTTP/2.0 bietet das Zusammenfassen mehrerer Anfragen (Multiplexing), Datenkompressionsmöglichkeiten (HPACK), Inhaltsübertragung in Binärform und Push-Verfahren, Serverseitig initiierte Datenübertragung.

Befehle: GET, POST, HEAD, PUT, PATCH, DELETE, TRACE, OPTIONS, CONNECT.

2.3 FTP

Das File Transfer Protokol ist ein zustandsbehaftetes Protokol zum Datentransfer über IP-Netzwerke. FTP ist Teil der Anwendungsschicht. Man soll mithilfe von FTP den Datenaustausch vom Server zum Client, vom Client zum Server oder zwischen zwei Servern ermöglichen.

FTP nutzt zur Steuerung und zum Übertragen von Daten zwei separate Verbindungen, wobei üblicherweise auf Port 21 der Controller aufgebaut wird und auf einem separaten Port die Datenübertragung stattfindet.



A: Client – S: Server

Man unterscheidet zwischen Aktivem FTP und Passivem FTP:
Aktives FTP wird vom Klienten mithilfe von PORT und EPRT initiiert und man unterscheidet wie oben beschrieben den Befehlskanal und den Datenkanal. **Passives FTP** wird via PASV und EPSV Befehlen gestartet. Dieser Modus wird genutzt, wenn keine direkte Verbindung zum Klienten aufgebaut werden kann, etwas aufgrund einer Firewall oder Adressumschreibungen via NAT.

2.4 SMTP

Das Simple Mail Transfer Protocol dient dem Austausch von Emails, primär jedoch zum Senden und Weiterleiten – Zum Empfangen dieser dient üblicherweise POP3 oder IMAP. SMTP-Server liegen üblicherweise auf Port 25 oder 587 für authentifizierte Mails.

Normalerweise wird SMTP vom Mail User Agent ausgeführt, welcher sich zum SMTP-Server verbindet und dann dem Nutzer die Mails weiterleitet:

Client	Server	
telnet x.y.z 25		Der User Agent ruft auf
	220 Ready	Server Meldet sich
EHLO Oi		Klient identifiziert sich
	250 OK	Server bestätigt
MAIL FROM: <@.org>		Klient definiert Absenderadresse
	250 OK	Server bestätigt
RCPT TO: <@.org>		Klient definiert Empfängeradresse
	250 OK	Server bestätigt
DATA		Klient initiiert die Email selbst
	354 Start Input	
(MailInput)		Die Email wird geschrieben. Ein '.' beendet die Email.
	250 OK	Server bestätigt die EMail.
QUIT		Klient initiiert Verbindungsabbau
	221 Closing	Verbindung ist terminiert.

2.5 POP3

Das Post Office Protokol 3 dient dem Empfangen von Daten auf dem lokalen Gerät. Es ist das minimalste Protokol, da es lediglich Auflisten, Abholen und Löschen implementiert. Im Gegensatz zu IMAP werden bei einem Pull-Request die Daten direkt vom Server gezogen, wo diese auch gelöscht werden. Zudem ist keine permanente Verbindung zum Server notwendig. Allerdings gibt es zwischen unterschiedlichen Mailclients keine einheitliche Synchronisierung – Wird eine Mail gelesen, so können andere Klienten diese Information nicht erhalten.

Um bei POP3 Datensicherheit zu gewährleisten wird APOP implementiert und/oder mittels SSL/TLS verschlüsselt.

Befehle: USER, PASS, STAT, LIST, RETR, DELE, NOOP, RSET, QUIT.

2.6 IMAP

Das Internet Message Access Protocol dient dem Aufruf von Emails. Es erweitert POP3 um Userdefinierte Konfigurationen des Mailservers. Bei einem Serverzugriff werden Kopien der Mails dem Server entnommen, um Datenverlust vorzubeugen. Der Datentransfer verläuft wie folgt:

```
-- Der Server begrüßt den Klienten
-- Der Klient authentifiziert sich
-- Der Server bestätigt
-- Der Klient gibt einen Befehl ein (z.B. 'select Inbox')
-- Der Server gibt Daten über vorhandene und gelesene Mails wieder
-- Der Klient pullt die Mails oder Informationen
-- Der Server beantwortet je nach Befehl mit angefordertem Datensatz
-- ... Der Klient meldet sich ab
-- Server terminiert die Verbindung.
```

2.7 SNMP

Das Simple Network Management Protocol dient der zentralen Überwachung und Steuerung von Geräten innerhalb eines Netzwerkes, als auch Fehlererkennung und -benachrichtigung. Es gilt daher als Protokol der Anwendungsschicht.

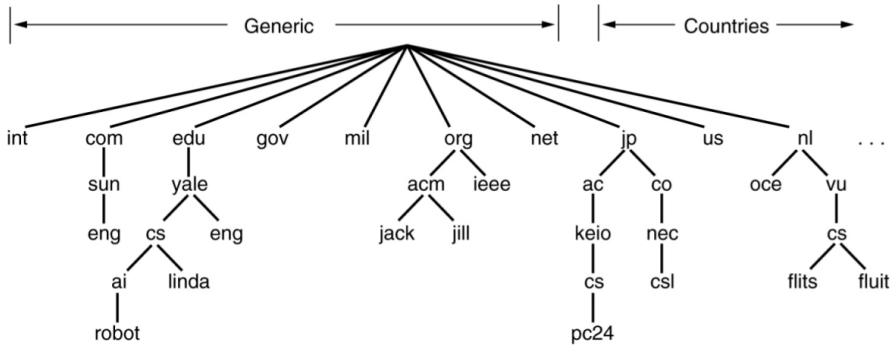
SNMP Agents empfangen an den jeweiligen Geräten die Befehle: GET-REQUEST, GETNEXT-REQUEST, GETBULK, SET-REQUEST, GET-RESPONSE, INFORM-REQUEST und Trap. Die 3 Get-Befehlspakete können vom Manager zu einem Agenten gesendet werden, um Daten anzufordern. Die Antwort erfolgt mittels Response-Paket. Mittels SET können Agenten konfiguriert werden.

Sicherheitsprobleme: SNMP unterstützt keine Anmeldung mit Kennwort und Usernamen – Es werden lediglich Communities zum Management verwendet, etwa PUBLIC (read-only) und PRIVATE(read-write).

2.8 DNS

Der Domain Name Service bietet in Netzwerken die Namensauflösung der IPs und ist Teil des Application Layers – Nicht zu verwechseln mit ARP/RARP im Link Layer. Ähnlich einem Telefonbuch wird bei einer Anfrage ermittelt wo die angeforderte IP liegt und übermittelt.

Während DNS dezentral aufgebaut ist, gelten hierarchische Strukturierungen, in Form eines Baumes:



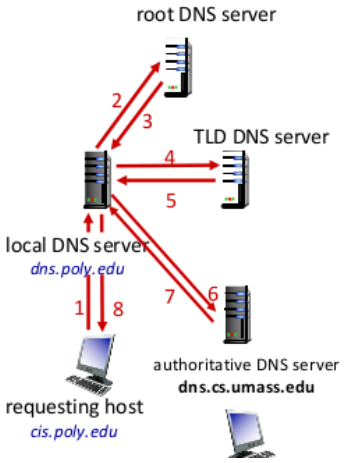
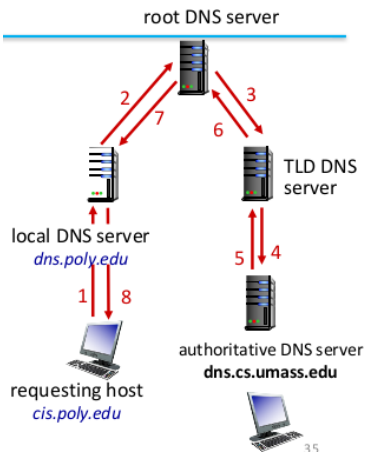
Man iteriert von oben nach unten

Eine komplette Domain besteht aus der Konkatinierung aller Labels eines spezifischen Pfades. Die Adresse 'de.sharelatex.com' wäre beispielsweise

```
.
|
com
   \
    \
 sharelatex
     |
    de
```

Jedes einzelne Label sind jeweils mindestes 1 Byte und maximal 63 Byte lang und enden mit einerm '.'.

Die Adressauflösung geschieht anhand der Zusammenarbeit mehrerer DNS-Server.

Iterative	Recursive
	
Der Kontaktierte darüberliegende Server beantwortet mit der gesuchten IP oder mit einem weiteren DNSServer, welcher womöglich die Adresse kennt.	Der Server fragt alle darunterliegenden Stationen nach der IP. Gibt es keine Nachricht, so wird die Anfrage dem darüberliegenden DNSServer weitergeleitet.

Für gewöhnlich werden Einträge von IPs in den jeweiligen Zonen unterscheidet.

2.9 TCP

Das Transmission Control Protokol arbeitet auf der Transportschicht und dient der Datenübertragung. TCP implementiert den sog. 'Handshake', wobei ein sicherer Datenübertragungkanal zwischen Server und Klient aufgebaut wird. Es gilt als Stop-and-wait Protokol.

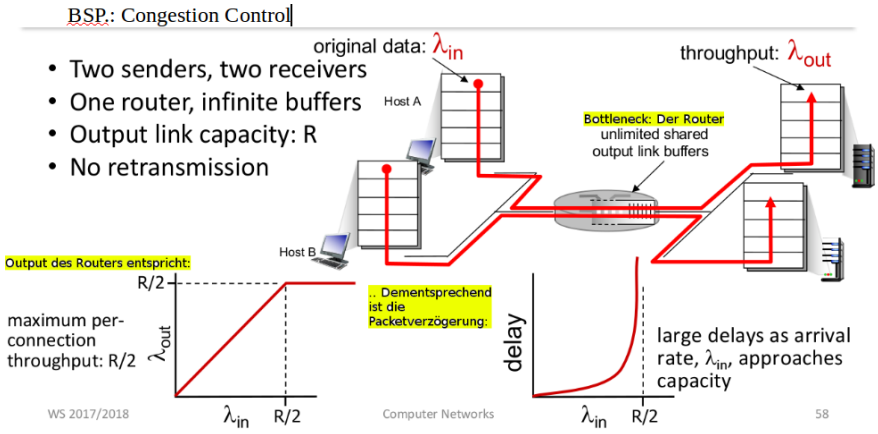
Struktur des TCP-Segmentes

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Source Port          |       Destination Port        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sequence Number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

```
|                    Acknowledgment Number                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Data |           |U|A|P|R|S|F|                               |
| Offset| Reserved  |R|C|S|S|Y|I|            Window             |
|       |           |G|K|H|T|N|N|                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Checksum            |         Urgent Pointer        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             data                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Merkmale:

1. Full Duplex Data – Bidirectional Dataflows
2. Connection-oriented – Initializes Channel before sending and terminates channel once everything's been sent.
3. Flow Control: Receiver controls sending rate of the Sender – Empfänger übermittelt Überlauf des Empfangsbuffers.
4. Congestion Control: Bei zuvielen Daten von zuvielen Quellen: Verwurf von Daten auf der Netzwerkschicht durch Bufferoverflow



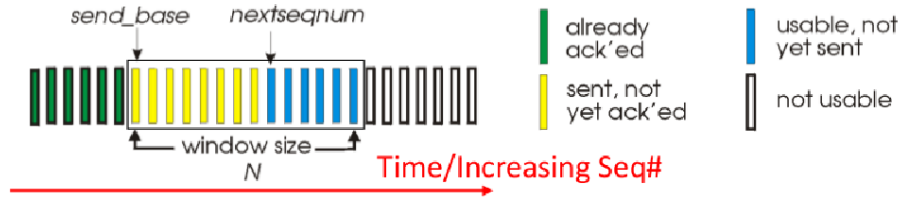
Wichtige Flags bei TCP:

1. SYN-Flag: Initiiere Channel
2. FIN-Flag: Terminiere Channel

3. ACK-Flag: Authentifiziert die ACK.

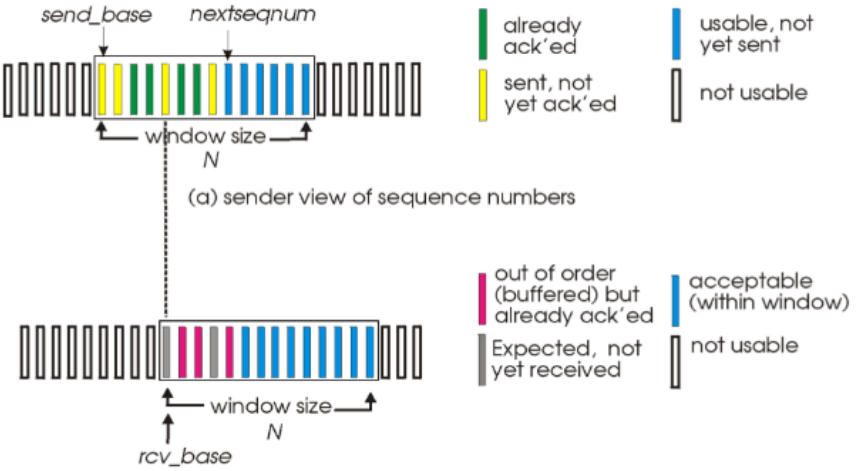
TCP im großem Ausmaß wird üblicherweise via Go-Back-N oder Selective Repeat implementiert.

→ Go-Back-N:
Sender kann N Packete durchschicken und Sender schickt kumulative ACKs.
- Bei Datenverlust, so erfolgt keine Bestätigung. Lediglich diese, bis zum Loch werden bestätigt.
Sender:



* Bei Timeout werden die Gelben packete wieder gesendet.

→ Selective Repeat:
Sender sendet N Packete, wobei jedes Packet einzeln bestätigt wird.
Jedes Packet bekommt Timer – Bei Auslauf wird das Packet neu übertragen.



Sobald das letzte Packet mit kleinster ACK bestätigt wird, so wird das Window nach vorne verschoben.

2.10 UDP

UDP ist im Grunde das Gegenstück zu TCP: Verbindungsfreie Datenübertragung ohne Sicherheiten, Zustandslose, jedoch weitaus schneller als TCP, aufgrund des fehlenden Overheads durch den Handshake und mit kleinerem Header.

UDP-Struktur:

```
0       7 8     15 16    23 24    31
+--------+--------+--------+--------+
|     Source      |   Destination   |
|      Port       |      Port       |
+--------+--------+--------+--------+
|                 |                 |
|     Length      |    Checksum     |
+--------+--------+--------+--------+
|
|              Payload
|           e.g. AppData
+---------------- ...
```

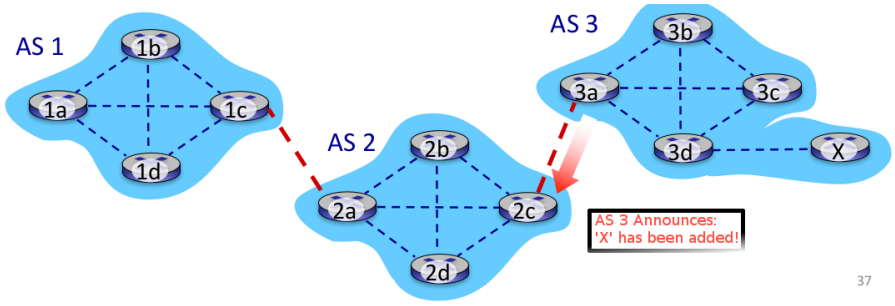
→ Daraus resultiert: Das kleinste UDP-Segment ist 8 Byte lang - Mit leerem Payload und den Headern zu je 2 Byte.

2.11 BGP

Das Border Gateway Protocol ist das Routingprotokoll das Autonome Systeme miteinander verbindet. Es ist Teil der Anwendungsschicht und basiert auf TCP.

Im Rahmen von BGP existieren **eBGP**, external BGP und **iBGP**, internal BGP. Internal BGP steht für das Routing innerhalb eines Subsystemes und External BGP steht für das Routing von Paketen in andere BGP-Systeme.

Das iBGP trackt hierbei stets, welche Elemente im jeweiligen System sind. Wird eine Komponente hinzugefügt, so wird dies über den eBGP-Knoten weitervermittelt.



AS2 informs AS1 aswell that AS3 has added 'X'

Das Routing innerhalb von BGP basiert auf mehreren Möglichkeiten:

1. Lokale Preferenz
2. Shortest AS-Path
3. Closest NEXT-HOP – Also known as Hot Potatoe Routing
4. Other, specifiable Criteria or identifiers.

2.12 OSPF

Das Open Shortest Path First ist ein Link State-Protocol und basiert auf dem Dijkstra-Algorithmus. Es ist im Gegensatz zu BGP in der Vermittlungsschicht angesiedelt.

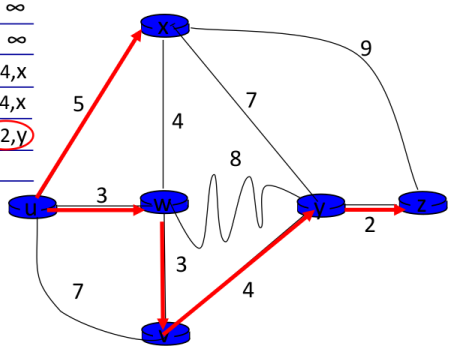
Der Dijkstra-Algorithmus:

Step	N'	D(v) p(v)	D(w) p(w)	D(x) p(x)	D(y) p(y)	D(z) p(z)
0	u	7,u	3,u	5,u	∞	∞
1	uw	6,w		5,u	11,w	∞
2	uwx	6,w			11,w	14,x
3	uwxv				10,v	14,x
4	uwxvy					12,y
5	uwxvyz					

Notes:

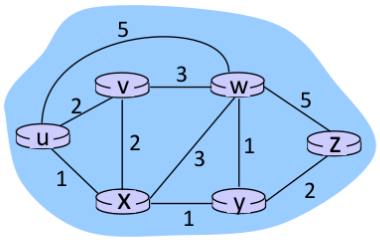
Construct shortest path tree by tracing predecessor nodes

Ties can exist (can be broken arbitrarily)



Wähle günstigste Kante für jeden Fall. Beachte noch offene Knoten!

Alternativ ist auch der Distanz-Vektor Algorithmus anwendbar:



Clearly, $d_v(z) = 5$, $d_x(z) = 3$, $d_w(z) = 3$

Bellman-Ford equation says:

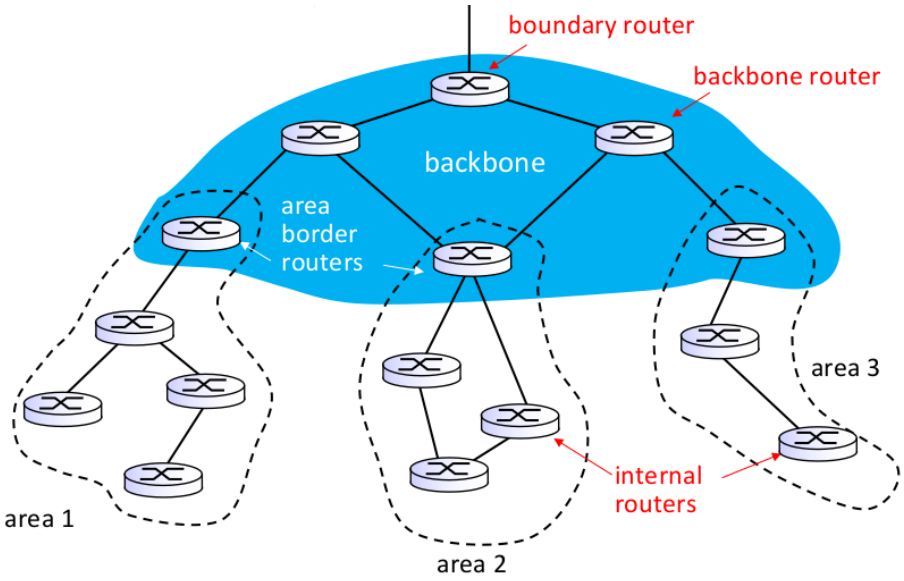
$$d_u(z) = \min \{ c(u,v) + d_v(z), \; c(u,x) + d_x(z), \; c(u,w) + d_w(z) \}$$
$$= \min \{2 + 5, \; 1 + 3, \; 5 + 3\} = 4$$

Node achieving minimum is next hop in shortest path, used in forwarding table

Wähle Minimum aller Pfade von Knoten a zu b, wobei d_u die Distanz zum Knoten u von Knoten $d(z)$ darstellt.

Es garantiert schleifenfreies Routing, es überwacht Nachbarn (Area-Konzept) und ist damit leichter zu warten, es unterstützt Klassenlose Internetadressierungen und kann im Fehlerfall des Routings das Bidirectional Forwarding Detection-Protokoll nutzen.

OSPF hält sich beim Routing immer an eine spezifische Hierarchie:

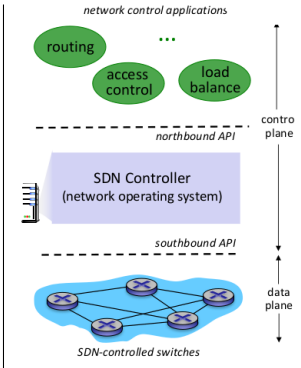


2.13 SDN

Das Software-Defined Networking ist ein, umgangssprachlich genannter, Clusterfuck mehrerer Protokolle, die aufeinander aufbauen und via Hardware gemanaged werden. Es implementiert Flow-Based forwarding, es trennt Dataplane und Control Plane mittels Switches und verfügt über ein programmierbares Netzwerk.

Prinzipiel gibt es die SDN Control Applications, den SDN Controller und die SDN-Controlled switches.

1. Die Kontrolapplikationen dienen dem konkreten Steuern des SDN.
2. Der SDN Controller erhält die Konsistenz des Netzwerkes. Als Verteiltes System hat es Zugriff auf sowohl Applikationen als auch Switches.
3. Die SDN Switches dienen dem Forwarding.



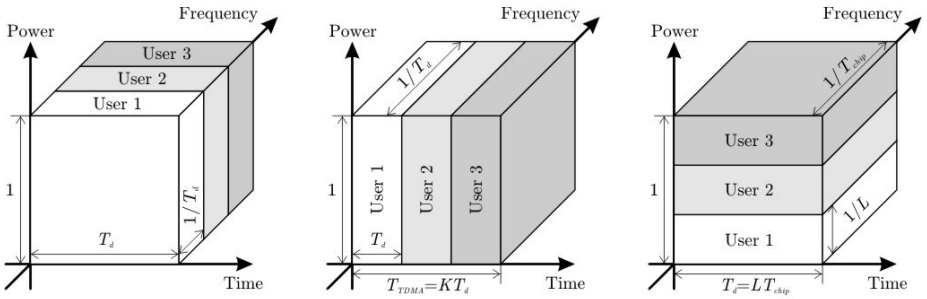
2.14 TDMA, FDMA, CDMA

TDMA, FDMA und CDMA dienen beispielgebend dazu, wie man Verbindungen effektiv einteilen kann.

1. TDMA
   Im Time Division Multiple Access-Verfahren bekommt jeder User mittels zeitlichem Multiplexing einen Zeitrahmen zugesprochen, in welchem er Senden darf. Hierfür bekommt dieser User den gesamten Frequenzbereich zugesprochen.

2. FDMA
   In Frequency Division Multiple Access bekommt jeder User eine fixierte Frequenz auf welcher er senden darf.

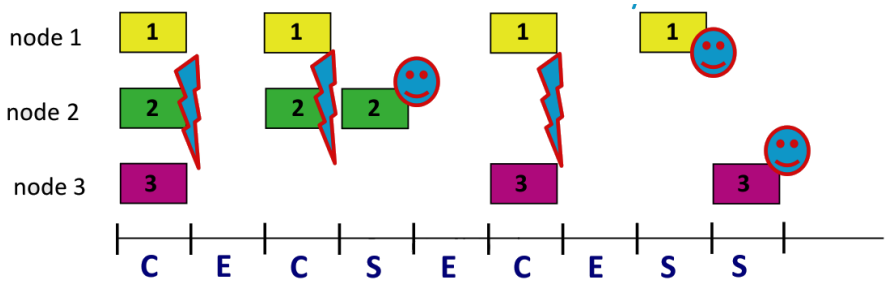
3. CDMA
   Code Division Multiple Access gilt dem Wireless Bereich. Man kann simultan senden, wobei encodieren der Daten den Sender und Empfänger spezifizieren.



FDMA **TDMA** **CDMA**

2.15 SALOHA

SALOHA, oder Slotted Additive Links On-Line Hawaii Area bedient sich dem TDMA, wobei jeder Slot Zeitslot miteinander synchronisiert wird. Sobald ein Frame zur Übertragung kommt, so wird dies unmittelbar für den nächsten Zeitslot eingetragen. Bei Kollissionen wird dem jeweiligen Paket ein Zeitstempel verpasst und diesem entsprechend in einen neuen Slot eingetragen.

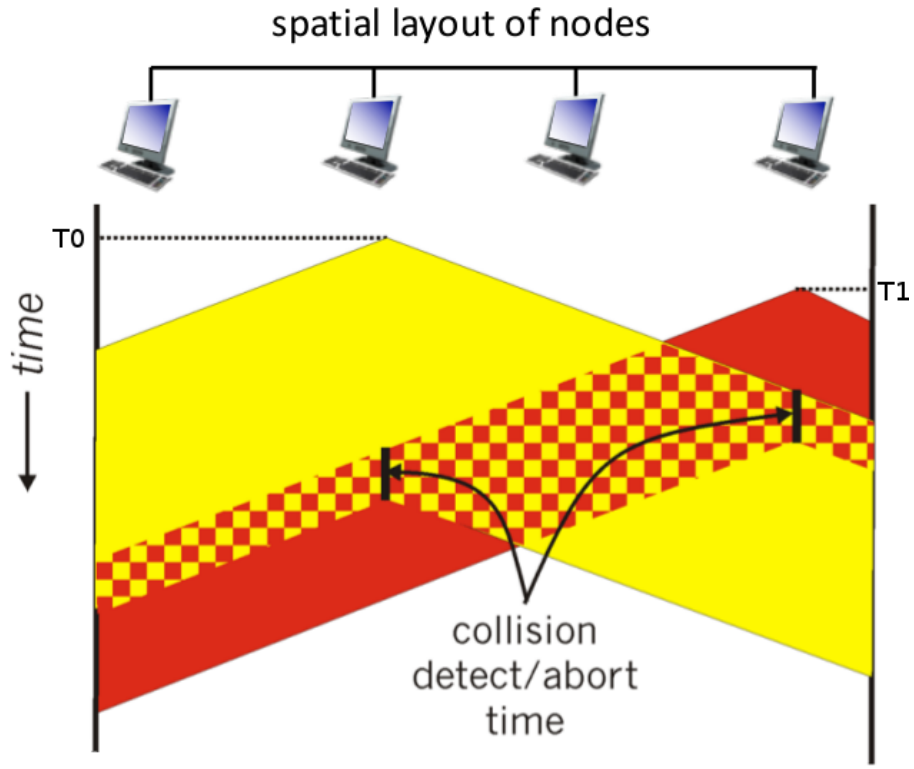


Eine weitere Version, das PURE oder unslotted ALOHA ist simpler, ohne Synchronisation. Hierbei werden eintreffende Frames sofortig gesendet. Bei einer Kollission werden die Übetragungen abgebrochen und man erwartet eine ACK des Empfängers – Siehe CSMA/CD.

2.16 CSMA/CD

Das Carrier Sending Multiple Access/Collision Detection ist eine umfassende Übertragungsstruktur, aufbauend auf TDMA. In CSMA wird aktiv mitgehört wann eine Station etwas sendet. Wenn der Übertragungskanal unbenutzt ist, so startet unmittelbar die Übertragung. Ist der Kanal besetzt, so wird die Übertragung verlegt. Generell dient es im Ethernet.

Einzelne Übertragungsstationen agieren in einem spezifischen Frequenzbereich:



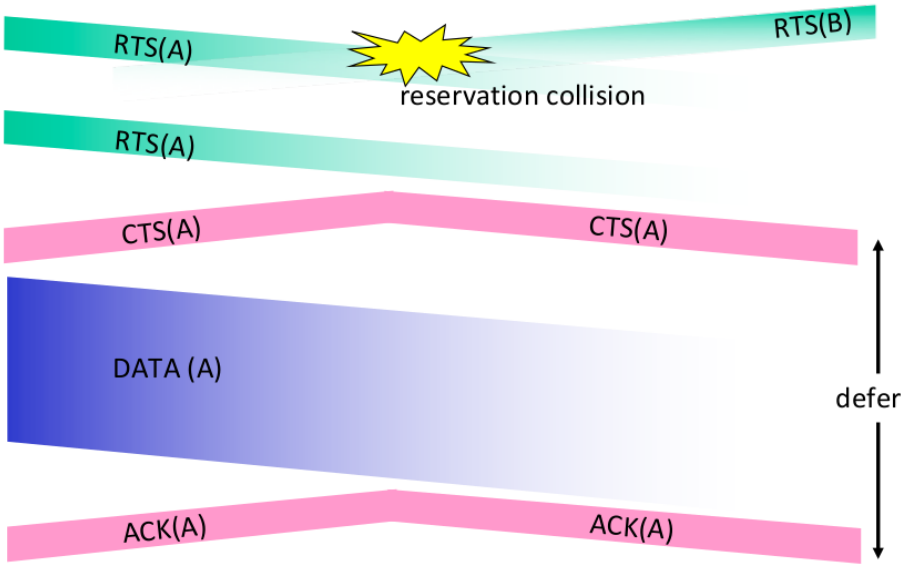
Kollision entdeckt – erwarte ACK

Wird keine Kollission entdeckt, so wird die Datei normal übertragen. Sollte eine Kollision entdeckt werden, so wird die Übertragung abgebrochen. Und die Übertragung wird später ausgeführt, wobei man per Kollission exponentiell lange wartet.

2.17 CSMA/CA

Carrier Sensing Multiple Access/Collision Avoidance unterscheidet sich weitgehend von dessen Konterpart, CD. Es kombiniert mehrere Token-Topologien zu einem Protokoll und wird üblicherweise für 802.11, also WLAN, genutzt.

Innerhalb eines Netzwerkes bekommt der Router Anfragen, 'Requests to send', von den jeweiligen Geräten. Der Server sendet anschließend an alle Beteiligten im Netzwerk, dass der spezifisch gewählte Knoten senden wird und damit der Kanal besetzt ist. Nach der Datenübertragung sendet der Server eine ACK an alle, um den freien Kanal anzukündigen.



2.18 ARP

Das Address Resolution Protocol ist Teil der Netzzugangsschicht und dient der Zuordnung einer IPv4 Adressierung in Ethernet-Netzen zu einer MAC-Adresse. (IPv6 – Neighbor Discovery Protocol) Die Adresse einer jeder Schnittstelle ist dabei, theoretisch gesehen, weltweit eindeutig. Wird die MAC-Adresse des Zielrechners nicht spezifiziert, so ist ein IP-Paket unzustellbar. Jedoch kann man mithilfe von ARP die MAC-Adresse des Zielrechners.

2.19 IEEE 802.11 Wireless LAN

Das Wireless Local Area Network ist eine Verbindung via Funk. Man spezifiziert hier wiederum zwischen verschiedenen Frequenzbereichen, wodurch die Kompatibilität zu anderen Versionen nicht gegeben sein muss.

3 Networking in Java

3.1 Processes

3.2 Threads

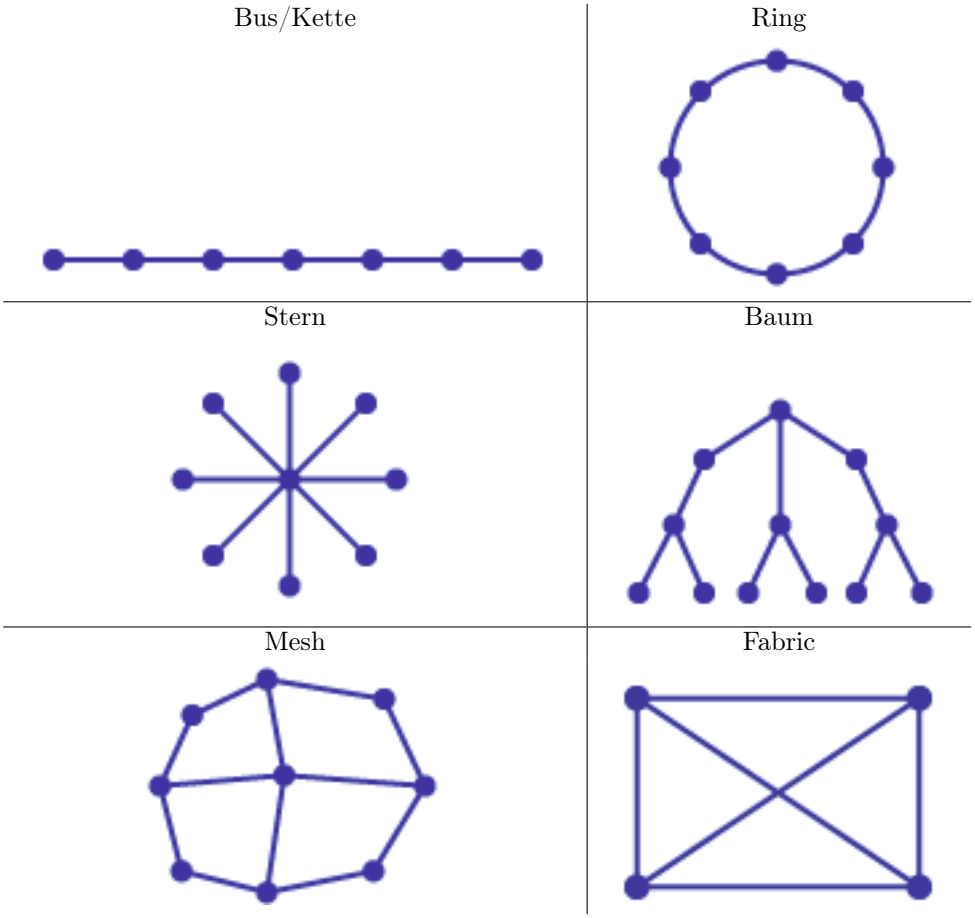
3.3 Readers and Writers

3.4 Sockets and Exceptions

4 Begriffe und Definitionen

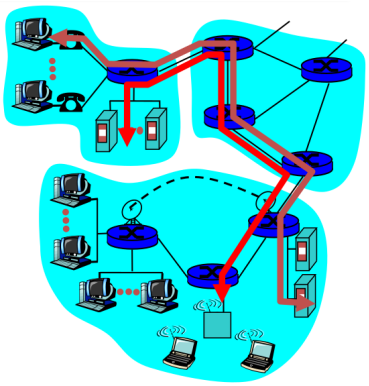
- **Das Internet**: Das Internet ist ein weltweiter Verbund von Rechnernetzwerken und autonomen Systemen (AS). Datenaustausch zweier Systeme wird spezifiziert in Protokollen, beschrieben in RFCs der Internet Engineering Task Force (IETF).
  Vorläufer: ARPANet – Advanced Research Project Agency, US-Militär gefolgt von TCP/IP, DNS, Usenet und darauf Kommerz-WWW: 1991 weltweit öffentlich verfügbar.

- **Netzwerktopologie** Eine Netzwerk-Topologie ist die physikalische Anordnung von Netzwerk-Stationen, die miteinander verbunden sind. Sie ist unterteilbar in dessen *Logik – Wo befindet sich ein Gerät im Netzwerk & Physik – Ports und Kabel* . Typische Topologien sind:

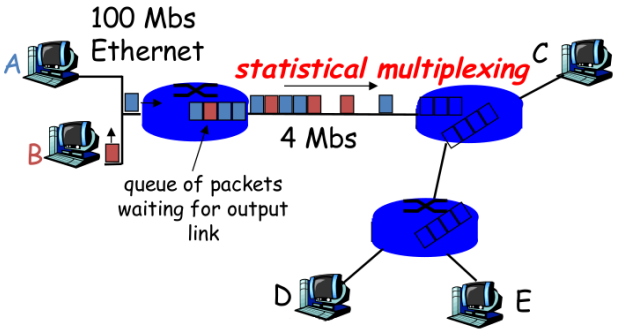


- **Servicetypen**:
  • Connection oriented Services, auch 'Verbindungsorientierte': Ähnlich TCP. Vergleichbar mit einem Telefonanruf, beispielsweise Reliable Data Streams (Sequence of Pages or Remote Logins).
  • Connectionless Services, auch 'Verbindungslose': Ähnlich UDP. Vergleichbar mit Briefpost, oft in Form von Emails und Database querys.

- **Circuit Switching und Packet Switching**:
  • Bei Circuit Switching werden Leitungen gänzlich reserviert. Solange 2 Medien miteinder kommunizieren, können andere u.U. nichts versenden da die Leitung blockiert ist (no Sharing). Implementiert via FDMA (4 User auf 4) oder TDMA (4 User auf 1 Kanal).
  • Packet Switching bedient sich dem Aufteilen der Daten in Pakete. Zieladresse gibt an wohin das Paket gesendet werden muss, wobei die Routen je nach Forwarding-Funktion variieren.



Circuit Switching within a network.



Statistical Packet Switching.

Store And Forward Switching.

- Netzwerktypen:
  (a) Digital Subscriber Line
  Nutzt Telefonverbindung zur Verbindung zum Netzwerk, DSLAM.
  (b) Cable Network
  FDMA: Mehrere Nutzer hängen an einem Kabel.
  (c) Home Network
  Typisches Heimnetzwerk mit zum Route verbundene Geräte.
  (d) Ethernet oder Enterprise Access Networks
  Geräte sind zu einem Switch verbunden, welche mit ISP gekoppelt sind.
- Multiplexing/Demultiplexing
  Beschreibt das Hinzufügen/Entfernen von Steuer- und Kontrolinformationen.

- Congestion Control und Flow Control
  • Congestion Control: Aufgrund zuvieler Quellen und Sender – Datenverwurf auf 2ter Schicht. • Flow Control: Empfänger benachrichtig Sender Datenüberlauf im Routerbuffer. Flow Control:

- Reliable Data Transfer
  Zuverlässiger Datentransfer wird üblicherweise mittels verschiedener States gewährleistet. Vgl.: TCP

- Forwarding and Routing
  • Forwarding beschreibt den zu wählenden Pfad des Paketes.
  • Routing beschreibt den Pfad, welches das Paket zurückgelegt hat.

- MAC Addressen
  Media Access Control-Adressen konkretisieren das Gerät innerhalb eines Netzes – Teil der 2. Schicht. Es gibt mehrere Arten der Syntax: 00-80-41-ae-fd-7e, wobei die Zeichentrennung via ',' '.' gesetz wird. Selten gibt es auch keine Zeichentrennung und es ist eine Zeichenkette.

- IPv4 und IPv6

- Routing Algorithm Classifications Es gibt mehrere Arten von Routing Algorithmen.

- Autonomous Systems in Scalable Routing

- Hot Potato Routing

- IP-Anycast

- Multiple Access Protocols

- Channel Partitioning

- The 5G Atom

4.1 Weitere Übertragungstypen

1. Adaptive Streaming
2. Client-Server Communication
3. Request-Response Protokol
4. Stop-and-Wait
5. Go-Back-N
6. Selective Repeat
7. Classless Interdomain Routing & Network Address Translation

8. Multicast
9. OpenFlow
10. Network Management
11. Ethernet, Switches & Routers
12. Flooding
13. The Wireless Network
14. IEEE
    (a) 802.2 Logical Link Protocol - Medienzuteilung bei MAC
    (b) 802.4 Token Bus
    (c) 802.5 Token Ring
    (d) 802.11 WLAN

4.2 Berechnungen

4.2.1 Graph Abstractions of the Network

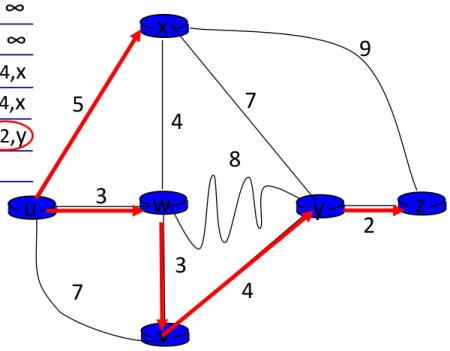
1. Dijstra

Step	N'	D(v) p(v)	D(w) p(w)	D(x) p(x)	D(y) p(y)	D(z) p(z)
0	u	7,u	3,u	5,u	∞	∞
1	uw	6,w		5,u	11,w	∞
2	uwx	6,w			11,w	14,x
3	uwxv				10,v	14,x
4	uwxvy					12,y
5	uwxvyz					

Notes:

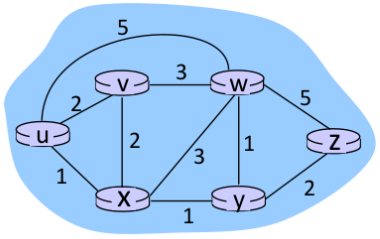
Construct shortest path tree by tracing predecessor nodes

Ties can exist (can be broken arbitrarily)



Wähle günstigste Kante für jeden Fall. Beachte noch offene Knoten!

2. Distance-Vector



Clearly, $d_v(z) = 5$, $d_x(z) = 3$, $d_w(z) = 3$

Bellman-Ford equation says:

$$d_u(z) = \min \{ c(u,v) + d_v(z), \; c(u,x) + d_x(z), \; c(u,w) + d_w(z) \}$$
$$= \min \{2 + 5, \; 1 + 3, \; 5 + 3\} = 4$$

Node achieving minimum is next hop in shortest path, used in forwarding table

Wähle Minimum aller Pfade von Knoten a zu b, wobei d_u die Distanz zum Knoten u von Knoten $d(z)$ darstellt.

4.2.2 IP-Addressberechnungen

IP-Adressberechnungen basieren auf deren Klassifizierung bzw. auf deren Netzanteil. IPv4 – Adressberechnung Bei einem Subnetz von 11111111.11111111.11111111.00000000 (255.255.255.0) wird der Netzteil auf 192.168.0 festgelegt ↔ E.g. wir besitzen 8 Bits für unsere Hostadressen.
→ Der Netzname ist predefiniert mit .0
→ Der Broadcsat liegt auf .255
→ Wir können damit noch $2^8 - 2$ Adressen belegen, dementsprechend hat dieses Netz noch 254 Adressen zur Verfügung.

Dies funktioniert analog mit dem Klassenlosen IPv4 Verteilungsprinzip, bei welchem man die höchstwertigen Bits (von Links) dem Netzteil zuspricht. Der Rest resolviert die Anzahl an zu vergebene Adressen.
Für einen Netzteil von 192.168.120 ↔ 11000000.10101000.01111 existiert ein möglicher Host-Adressbereich von $32 - 8 - 8 - 5 = 11$ setzbaren Bits, dementsprechend haben wir $2^11 - 2 = 2046$ Adressen.

IPv6 – Adressberechnung
.. verläuft nach demselben Schema wie bei IPv4. Die Adressen werden in ihre Bits transformiert und es werden Netz- und Hostadressbereiche eingeteilt. Der Wert nach einer IPv6 Adresse (.../12) beschreibt die Anzahl der Bits für den Netzanteil. Der Rest der Adresse beschreibt den Hostanteil.

IPv4 to IPv6 Für eine IPv4, z.B. 192.168.99.1 erzeugt man die IPv6 Adresse durch Umschreibung in Hexadezimal. Wähle jedes Oktet und dividiere durch 16, um eine Hexadezimalzahl zu erzeugen.

```
192 -> C0       168 -> A8       99 -> 63        1 -> 01
```

Damit ist die IPv6-Representation von 192.168.99.1 die Konkatenation der Hexadezimalwerte C0A8:6301. Die Umsetzung ist vordefiniert nach dem Schema 2002:<32-Bit IPv4>:<16-Bit Network>::/64 Die vollständige IPv6-Adresse lautet 2002:C0A8:6301::1/64. (64 → '0' als 2ter Parameter von Links)

IPv6 to IPv4 Analog, nur rückwärts. Man extrahiert die 32-Bit aus der IPv6 Adresse und wandelt um. Mit C0A8 6301 haben wir insgesamt 8 Hexawerte zu je 4 Bit, dies entspricht 32 Bit. Wandle *16 in eine Dezimalzahl um und interpretiere das Ergebnis u.U. in Binärdarstellung.

4.2.3 Link Layer - Parity Checking

4.2.4 Cyclic Redundancy Check

Die Zyklische Redundanzberechnung dient der Fehlerkorrektur in Datenpaketen. Zur Datenüberprüfung wird ein Berechnungsverfahren angewandt und wenn dessen Ergebnis 0 ist, so ist der Datenblock fehlerfrei. Es dient jedoch nur zur Erkennung zufälliger Fehler, nicht zur Bestätigung der Paket-Ganzheit.

Es basiert auf der Polynomdivision und die Folge der zu übertragenden bits werden als binäres Polynom betrachtet: 1001 ist beispielsweise: $1 * x^3 + 0 * x^2 + 0 * x^1 + 1 * x^0$. Die Bitfolge wird durch ein festzulegendes Generatorpolynom mod 2 geteilt, wobei ein Rest bleibt. Dieser ist der CRC-Wert. Bei der Datenübertragung wird dieser Wert an den originalen Datenblock angehängt.

Zur Verifikation wird der Datenblock mit angehängtem CRC-Wert als Binärfolge interpretiert und erneut durch das CRC-Polynom Modulo geteilt. Ist das Ergebnis 0, so gibt es keinen Fehler.

Bsp.: CRC-Polynom: 110101 Datenblock: 11011 Der Datenblock mit Anhang ist nun 11011|00000, wobei die Anzahl der 0en der Länge des CRC-Polynoms -1 entspricht. Man erinnere sich: Das Exklusive-Or erzeugt Ergebnisse wie folgt:

```
0 ^ 0 = 0       1 ^ 1 = 0
0 ^ 1 = 1       1 ^ 0 = 1
```

Mittels Exklusive-OR wird nun der Block mit Anhang durchdividiert:

```
1101100000
110101
----------
0000110000
    110101
----------
     00101   -> Rest
```

Der Rest, 00101 wird nun dem originalen Datenblock angehänt und komplett übertragen: 1101100101

Bei Datentransfer wird erneut durch das CRC-Polynom dividiert:

```
1101100101
110101
----------
0000110101
    110101
----------
     00000
```

Hierbei haben wir den Restwert 0 und es ist damit kein Fehler aufgetreten.

4.2.5 CDMA Encoding/Decoding

5 Typische Klausurfragen & Fallbeispiele

5.1 Probeklausur WS2016/2017, 3. Termin

1. **ISO/OSI vs TCP-Referenzmodell**: Worin unterscheidet sich das ISO/OSI- vom TCP/IP Referenzmodell
2. **Transport**: Beschreibe die allgemeinen Eigenschaften von TCP bzw. UDP.
3. **HTTP**: Beschreibe HTTP (a) allgemein, hinsichtlich (b) persistenter Verbindungen, (c) Zustände und (d) Caches.
4. **Transportschicht**: Beschreibe und Skizziere die TCP-Verbindungsaufbau und TCP-Verbindungsabbau.
5. **Netzwerkschicht**: Nenne und beschreibe die drei Hauptfunktionen der Netzwerkschicht und nenne die zwei bekanntesten Routingalgorithmen.
6. **Netzwerkschicht**: Beschreibe IPv4 bzw. IPv6 und gehe auf Unterschiede ein und skizziere mögliche Ansätze zum Übergang von IPv4 und IPv6.
7. **Sicherungsschicht**: Beschreibe CSMA/CD und CSMA/CA. Wo werden die Methoden eingesetzt?
8. **Sicherungsschicht**: Beschreibe ARP.
9. **Rechnernetze Allgemein**: Beschreibe die Funktionsweise und Unterschiede von paketvermittelnden und leitungsgebundenen Netzwerken.

5.2 Frei erfundene Bsp.

5.2.1 Theorie

1. Ordne folgende Protokolle den Schichten des TCP-Modelles zu:

SMNP	•	•	Anwendungsschicht
POP3	•	•	Transportschicht
ARP	•	•	Vermittlungsschicht
DNS	•	•	Netzzugangsschicht
TLS	•	•	

2. Was ist der Unterschied zwischen Flow Control und Congestion Control?
3. Was sind die unterschiedlichen Elemente eines Wireless Networks und wie arbeiten sie miteinander?
4. Was passiert wenn man innerhalb eines WLan-Netzwerkes den Standort wechselt?
5. Was sind die 5 Eckpfeiler des 5G Atoms und welche Teilbereiche gibt es?
6. Was bedeutet "Multiple Access" im Rahmen des IEEE 802.11?
7. Was sind nennenswerte Unterschiede zwischen ARP und DNS?
8. Worin unterscheiden sich Router und Switches?
9. Wie funktioniert der Link Layer und was ermöglicht dieser? Skizziere!
10. Wie unterscheiden sich TDMA, FDMA, CDMA? Skizziere.
11. Wie funktioniert BGP? Gibt es äquivalente Protokolle?
12. Welche Arten der Datenübertragungen gibt es und welche grundlegenden Protokolle werden dafür implementiert? *Tip: Uni-Cast sendet von 1 zu 1 weiterem.*
13. Wie funktioniert SDN?
14. Was sind Grundfunktionalitäten der Applikationsschciht. Welche Protokolle werden hier implementiert?
15. Wie unterscheiden sich Data Plane und Control Plane? Gibt es unterschiedliche Variationen des Control Plane?
16. Wie wird ein IPv6 Paket behandelt, sollte man lediglich IPv4 weitersenden können?
17. Welche Versionen von HTTP existieren und was sind Unterschiede zwischen den einzelnen Versionstypen? Ist HTTP ein sicheres Protokoll?

5.2.2 Berechnungen

1. I seriously fucking hope this wont come.
2. Welche Arten der IP-Adressierungen gibt es? Wie werden Geräte adressiert?

5.2.3 Wahr oder Falsch

1. IEEE 802.11 ist ein Transportprotokoll.
2. Mithilfe der Datenrate-adaption kann man Datenqualität optimieren. Das SNR verstärkt diese Optimierung.
3. DAS SNR ist eine relative Maßzahl von Geräusch und Signal. Je höher das SNR, desto schlechter ist es.
4. Die Verarbeitung der MAC Addresse gehört zur Vermittlungsschicht
5. Das 802.3 ist eine veraltete Version des 802.11.
6. Der Link Layer wird abstrahiert in 2 Unterebenen, Dataplane und Controlplane.
7. P2P-Protokolle, also PPPs verwenden Switches zur Übertragung.
8. TDMA und FDMA sind zeitlich gesehen gleich schnell mit Datenübertragungen.
9. Die Abkürzung 'AS' steht für "automated Services" und steht im Zusammenhang mit Routerregionen.
10. BGP dient dem Forwarding von Paketen.
11. Bei "Multicast" sendet genau 1 Sender an beliebig viele Empfänger.
12. SDN implementiert Congestion Control.
13. Es ist möglich mittels SNMP auf Geräte im gleichen Netzwerk zuzugreifen.
14. Die Begriffe "Openflow" und "Flooding" beschreiben exakt das gleiche.
15. Die Firewall, etwa im OpenFlow, vergleicht IP-Adressen und TCP/UDP Port Nummern. Anhand dessen werden Daten akzeptiert oder gesperrt.
16. Forwarding ist ein äquivalenter Ausdruck für Routing.
17. Das Hot Potatoe Routing nutzt Flooding zur Datenweitergabe.
18. Der Networklayer implementiert die Funktionen des Forwarding und des Routing.
19. Eine IP-Addresse deckt einen Adressbereich von $2^(32)$ Addressen ab.
20. Jeder einzelne Router, welche ein Paket durchläuft, prüft die Checksumme des Paketes auf dessen Richtigkeit.
21. IPv6 und IPv4 unterscheiden sich lediglich durch Header, welche in IPv6 neu implementiert wurden: Priority, Flow Label und Next Header-Field.
22. TCP implementiert Multicast.

23. Existieren Bitfehler in einem Datenpaket, so können diese mithilfe der Checksumme erkannt werden.

24. Bei Selective Repeat existieren Fehlermöglichkeiten, welche den Algorithmus außer Kraft setzen.

25. Für Go-Back-N gilt: Es wird solange das gesamte Window neugesendet, bis die ACKs aller Pakete empfangen wurden.

26. Die Abfrage von IP-Adressen mittels DNS ist sowohl rekursiv als auch iterativ ausführbar.

27. Aufgerufene Adressen werden werden im Schnitt für 2 Tage gecached.

28. HTTP-Fehlercodes beginnend mit 5 beschreiben Klient-Fehler.

6 Typische Klausurfragen & Fallbeispiele – Lösungen

6.1 1. Klausur WS2017, 1. Termin

1. Was ist ein Request/Resposne Protokoll? Gebe ein Beispiel an.

2. Warum ist HTTP zustandslos? Welche Möglichkeiten gibt es Zustände einzuführen?

3. HTTP und Persistente Verbindungen und Pipelining - Wie geht das?

4. Was ist SMTP?

5. Wofür ist die Transportschicht?

6. Was ist BGP und wie unterscheidet es sich zu OSPF?

7. Welche Konflikte können beim Link-State Algorithmus und bei Distance-Vector Algorithmus entstehen?

8. Wofür steht SNMP?

9. Was ist Data Plane und Control plane? Wie wird Control plane implementiert?

10. Was ist ein kritischer Abschnitt? Gebe ein Beispiel.

11. Welche Aufgaben hat ein Router? Was ist Congestion Control und Flow Control? Skizziere letztere.

6.2 Probeklausur WS2016/2017, 3. Termin

1. **ISO/OSI vs TCP-Referenzmodell**: Worin unterscheidet sich das ISO/OSI- vom TCP/IP Referenzmodell?
   Sie unterscheiden sich in der Aufteilung ihrer Schichten.
   Das **ISO/OSI** Model nutzt 7 Schichten zum beschreiben des Netzwerkaustausches, während **TCP-IP** lediglich 4 nutzt.

ISO/OSI	TCP/IP
Anwendungsschicht	Anwendungsschicht
Darstellungsschicht	
Kom-Schicht	
Transportschicht	Transportschicht
Vermittlungsschicht	Vermittlungsschicht (IP)
Sicherungsschicht	Netzzugangsschicht
Bitübertragungsschicht	Netzzugangsschicht

2. **Transport**: Beschreibe die allgemeinen Eigenschaften von TCP bzw. UDP.

TCP	UDP
Flow Control	No Control
Congestion Control	No Control
In-Order Stream	Best-Effort-IP
Reliable Transmission	Unreliable Transmission
High overhead	Low Overhead
Connectionbuildup (Slow)	Connectionless (Fast)
Predefined Ports	Use next best Port (mostly)

3. **HTTP**: Beschreibe HTTP (a) allgemein, hinsichtlich (b) persistenter Verbindungen, (c) Zustände und (d) Caches.
   (a) Das Hyper Text Transfer Protokol beschreibt die Übertragung von Daten auf der Anwendungsschicht.
   Die Anfrage hat üblicherweise die Form von

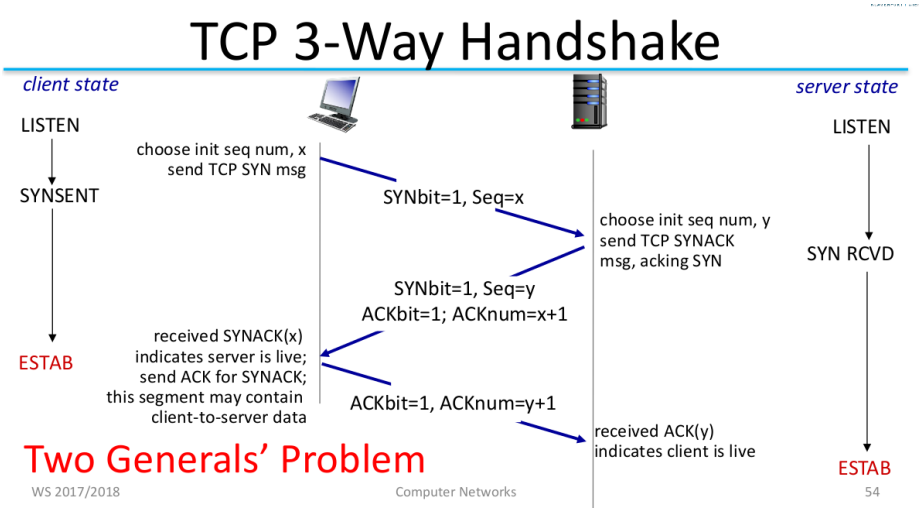
```
GET / http/1.1
Host: www.campus.aau.at
```

   .. wobei GET die Form der Anfrage ist, '/' spezifiziert den Ort, hier root und 'http/1.1' ist die genutzte Protokolversion. (b) Persistente Verbindungen werden erst mit Http/1.1 implementiert. Hier wird dem Server durch 'Keepalive'-Headereintrag signalisiert, dass die Verbindung nicht abgebrochen werden soll. Pipelining ermöglicht das mittels einer Anfrage mehrere Antworten gesendet werden können - 1 Für das HTML und x für die darin enthaltenen Dokumente oder Bilder für lediglich eine Anfrage.

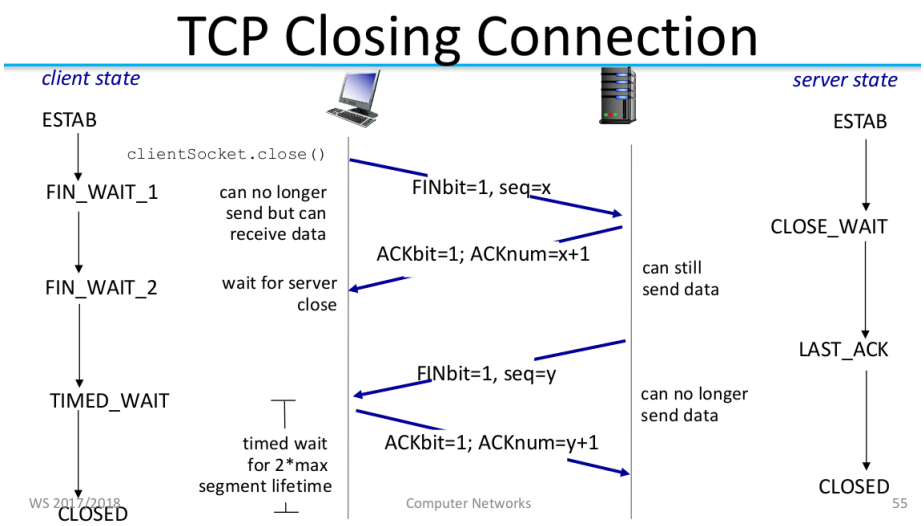
(c) HTTP ist ein Zustandsloses Protokol.
(d) Ein (shared) HTTP Cache ist ein lokaler dient dem Speichern von Anfragen, welche von mehr als einem User genutzt werden. Ziel ist es Antworten zu speichern, zur Performanzsteigerung.

`Siehe auch: https://tools.ietf.org/html/rfc7234}`

4. **Transportschicht**: Beschreibe und Skizziere die TCP-Verbindungsaufbau und TCP-Verbindungsabbau.
   Der TCP-Verbindungsaufbau wird vom Klienten mittels der SYN-Flag initiiert. Erst mit einer Antwort des Server, mit SYN und ACK Flags, gilt die Verbindung als aufgebaut.



Der Abbau geschieht mittels der FIN-Flag. Sobald der Klient die Verbindung schließen möchte, so wird die FIN-Flag gesetzt, wodurch der Klient keiner weiteren Daten schicken darf, jedoch noch empfangen kann. Die Verbindung ist dann abgebaut, wenn ein Klientseitiger Timer ausläuft oder die FIN-Flag seitens Server eintrifft.



5. **Netzwerkschicht**: Nenne und beschreibe die drei Hauptfunktionen der Netzwerkschicht und nenne die zwei bekanntesten Routingalgorithmen. Die Netzwerkschicht, oder Networklayer, dient dem Forwarding, dem Routing und der Fragmentierung der Datenpakete in Frames.

   Zu den bekannten Routingalgorithmen zählen der **Dijkstra-Algorithmus oder Link State** und der **Distance-Vector**

6. **Netzwerkschicht**: Beschreibe IPv4 bzw. IPv6 und gehe auf Unterschiede ein und skizziere mögliche Ansätze zum Übergang von IPv4 und IPv6. IPv4 als auch IPv6 spezifiziert jedes Gerät anhand einer eindeutigen Adresse im jeweiligen Subnets.

   Während IPv4 einen Adressbereich von $2^{(}32)$, oder $256^{(}4)$, bietet, so bietet IPv6 einen Adressbereich von $2^{(}128)$. IPv6 ist üblicherweise codiert in Hexdadezimalzahlen, wobei einzelne Blöcke mittels ':' getrennt werden.

```
https://213.213.220.07/                          -- IPv4
http://[2001:0db8:85a3:08d3::0370:7344]/         -- IPv6
http://[2001:0db8:85a3:08d3::0370:7344]:8080/    -- IPV6 + Port
```

   IPv4 Adressen werden in 4 verschiedene Klassen eingeteilt (Heuertags eigentlich Klassenlos berechnet):

   (a) Klasse A: 8 Bit-Netz & 24 Bit Host-Adresse

   (b) Klasse B: 16 Bit-Netz & 16 Bit Host-Adresse

   (c) Klasse C: 24 Bit-Netz & 8 Bit Host-Adresse

   (d) Klasse D: 28 Bit-Multicastgruppen

Die zu vergebenden Hostadressen berechnet sich mit $2^(AnzahlanBitsfrHostadresse)-2$. Dabei sind spezifische Adressen stets reserviert: 192.168.0.0/8 ist das Netzwerk selbst, während 192.168.0.255/32 ist der Broadcast innerhalb dieses Netzes.

IPv4 – Klasse C-Netzberechnung:
Bei einem Subnetz von 11111111.11111111.11111111.00000000 (255.255.255.0) wird der Netzteil auf 192.168.0 festgelegt $\leftrightarrow$ E.g. wir besitzen 8 Bits für unsere Hostadressen.
$\rightarrow$ Der Netzname ist predefiniert mit .0
$\rightarrow$ Der Broadcsat liegt auf .255
$\rightarrow$ Wir können damit noch $2^8 - 2$ Adressen belegen, dementsprechend hat dieses Netz noch 254 Adressen zur Verfügung.

Dies funktioniert analog mit dem Klassenlosen IPv4 Verteilungsprinzip, bei welchem man die höchstwertigen Bits (von Links) dem Netzteil zuspricht. Der Rest resolviert die Anzahl an zu vergebene Adressen.
Für einen Netzteil von 192.168.120 $\leftrightarrow$ 11000000.10101000.01111 existiert ein möglicher Host-Adressbereich von $32 - 8 - 8 - 5 = 11$ setzbaren Bits, dementsprechend haben wir $2^11 - 2 = 2046$ Adressen.

IPv6 – Adressberechnung
.. verläuft nach demselben Schema wie bei IPv4. Die Adressen werden in ihre Bits transformiert und es werden Netz- und Hostadressbereiche eingeteilt. Der Wert nach einer IPv6 Adresse (.../12) beschreibt die Anzahl der Bits für den Netzanteil. Der Rest der Adresse beschreibt den Hostanteil.

Subnetz Tabelle

```
2001:0db8:0126:0000:0000:0000:0000:0000                    Anzahl der IP-Adressen
     |||| |||| |||| |||| |||| |||| ||||
     |||| |||| |||| |||| |||| |||| |||128 ------------------------------------ 1
     |||| |||| |||| |||| |||| |||| ||124 ------------------------------------ 16
     |||| |||| |||| |||| |||| |||| |120 ------------------------------------ 256
     |||| |||| |||| |||| |||| |||| 116 ------------------------------------ 4 096
     |||| |||| |||| |||| |||| |||112 ------------------------------------ 65 536
     |||| |||| |||| |||| |||| ||108 ------------------------------------ 1 048 576
     |||| |||| |||| |||| |||| |104 ------------------------------------ 16 777 216
     |||| |||| |||| |||| |||| 100 ------------------------------------ 268 435 456
     |||| |||| |||| |||| |||96 ------------------------------------ 4 294 967 296
     |||| |||| |||| |||| ||92 ------------------------------------ 68 719 476 736
     |||| |||| |||| |||| |88 ------------------------------------ 1 099 511 627 776
     |||| |||| |||| |||| 84 ------------------------------------ 17 592 186 044 416
     |||| |||| |||| |||80 ------------------------------------  281 474 976 710 656
     |||| |||| |||| ||76 ------------------------------------ 4 503 599 627 370 500
     |||| |||| |||| |72 ------------------------------------ 72 057 594 037 927 900
     |||| |||| |||| 68 ------------------------------------ 1 152 921 504 606 850 000
     |||| |||| |||64 ------------------------------------ 18 446 744 073 709 600 000
     |||| |||| ||60 ------------------------------------ 295 147 905 179 353 000 000
     |||| |||| |56 ------------------------------------ 4 722 366 482 869 650 000 000
     |||| |||| 52 ------------------------------------ 75 557 863 725 914 300 000 000
     |||| |||48 ------------------------------------ 1 208 925 819 614 630 000 000 000
     |||| ||44 ------------------------------------ 19 342 813 113 834 100 000 000 000
     |||| |40 ------------------------------------ 309 485 009 821 345 000 000 000 000
     |||| 36 ------------------------------------ 4 951 760 157 141 520 000 000 000 000
     |||32 ------------------------------------ 79 228 162 514 264 300 000 000 000 000
     ||28 ------------------------------------ 1 267 650 600 228 230 000 000 000 000 000
     |24 ------------------------------------ 20 282 409 603 651 700 000 000 000 000 000
     20 ------------------------------------ 324 518 553 658 427 000 000 000 000 000 000
```

7. **Sicherungsschicht**: Beschreibe CSMA/CD und CSMA/CA. Wo werden die Methoden eingesetzt?

8. **Sicherungsschicht**: Beschreibe ARP.

9. **Rechnernetze Allgemein**: Beschreibe die Funktionsweise und Unterschiede von paketvermittelnden und leitungsgebundenen Netzwerken?

6.3 Frei erfundene Bsp.

6.3.1 Theorie

1. Ordne folgende Protokolle den Schichten des TCP-Modelles zu:

SMNP	•	•	Anwendungsschicht
POP3	•	•	Transportschicht
ARP	•	•	Vermittlungsschicht
DNS	•	•	Netzzugangsschicht
TLS	•	•	

2. Was ist der Unterschied zwischen Flow Control und Congestion Control?

3. Was sind die unterschiedlichen Elemente eines Wireless Networks und wie arbeiten sie miteinander?

4. Was passiert wenn man innerhalb eines WLan-Netzwerkes den Standort wechselt?
5. Was sind die 5 Eckpfeiler des 5G Atoms und welche Teilbereiche gibt es?
6. Was bedeutet "Multiple Access" im Rahmen des IEEE 802.11?
7. Was sind nennenswerte Unterschiede zwischen ARP und DNS?
8. Worin unterscheiden sich Router und Switches?
9. Wie funktioniert der Link Layer und was ermöglicht dieser? Skizziere!
10. Wie unterscheiden sich TDMA, FDMA, CDMA? Skizziere.
11. Wie funktioniert BGP? Gibt es äquivalente Protokolle?
12. Welche Arten der Datenübertragungen gibt es und welche grundlegenden Protokolle werden dafür implementiert? *Tip: Uni-Cast sendet von 1 zu 1 weiterem.*
13. Wie funktioniert SDN?
14. Was sind Grundfunktionalitäten der Applikationsschciht. Welche Protokolle werden hier implementiert?
15. Wie unterscheiden sich Data Plane und Control Plane? Gibt es unterschiedliche Variationen des Control Plane?
16. Wie wird ein IPv6 Paket behandelt, sollte man lediglich IPv4 weitersenden können?
17. Welche Versionen von HTTP existieren und was sind Unterschiede zwischen den einzelnen Versionstypen? Ist HTTP ein sicheres Protokoll?

6.3.2 Berechnungen

1. I seriously fucking hope this wont come.
2. Welche Arten der IP-Adressierungen gibt es? Wie werden Geräte adressiert?

6.3.3 Wahr oder Falsch

1. IEEE 802.11 ist ein Transportprotokoll.
2. Mithilfe der Datenrate-adaption kann man Datenqualität optimieren. Das SNR verstärkt diese Optimierung.
3. DAS SNR ist eine relative Maßzahl von Geräusch und Signal. Je höher das SNR, desto schlechter ist es.

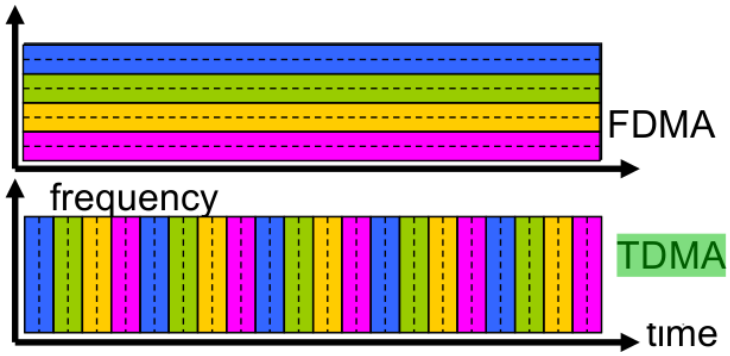
4. Die Verarbeitung der MAC Addresse gehört zur Vermittlungsschicht

5. Das 802.3 ist eine veraltete Version des 802.11.

6. Der Link Layer wird abstrahiert in 2 Unterebenen, Dataplane und Controlplane.

7. P2P-Protokolle, also PPPs, verwenden Switches zur Übertragung.

8. TDMA und FDMA sind zeitlich gesehen gleich schnell mit Datenübertragungen.
   Prinzipiell Ja. Wir betrachten zunächst beide Implementierungsmöglichkeiten:



   Der einzige Unterschied zwischen FDMA und TDMA ist dessen Platzkomplexität. FDMA benötigt mehr Frequenzen, wodurch das System, per User, außerordentlich groß sein muss. TDMA kann auf einem einzigen Kanal mehrere User unterhalten, jedoch auf Kosten der Qualität. Siehe auch:

   `https://www.taitradioacademy.com/topic/the-difference-between-fdma-and-tdma-1/`

9. Die Abkürzung 'AS' steht für "automated Services" und steht im Zusammenhang mit Routerregionen.
   Falsch, AS steht für 'Autonomous System'. Diese jedoch stehen tatsächlich in Zusammenhang mit Routerregionen.

10. BGP dient dem Forwarding von Paketen.
    Richtig, wobei es allgemein den Pakettransit, durch Ermitteln des nächstgünstigsten Nachbars, unterstützt.

11. Bei "Multicast" sendet genau 1 Sender an beliebig viele Empfänger.
    Richtig. Normalerweise mittels UDP. .

12. SDN implementiert Congestion Control.
    Falsch, SDN implementiert lediglich Flow-based forwarding. Es gibt jedoch eine Erweiterung, das SDTCP, welches Congestion control implementiert.

13. Es ist möglich mittels SNMP auf Geräte im gleichen Netzwerk zuzugreifen.
    Richtig, man erinnere sich an Timmse's Präsentation des SNMP auf Druckerzugriff im Rahmen der x'ten LV-Einheit. Anhand verschiedenen Befehlen kann man Geräte direkt ansprechen, soweit man im gleichen Netzwerk ist. Jedoch muss man dies per Gerät freigeben.

14. Die Begriffe "Openflow" und "Flooding" beschreiben exakt das gleiche.
    Falsch, Flooding beschreibt den Vorgang, in welchem ein Router alle unterliegenden Geräte abfragt (ähnlich wie Broadcast). Openflow ist Teil des SDN als Kommunikationsprotokoll. Es gibt Zugriff auf Hardwarekomponente des Switches oder ROuters zur Bearbeitung eintreffender Netzwerkpakete (Auch *Forwarding Plane* genannt).

15. Die Firewall, etwa im OpenFlow, vergleicht IP-Adressen und TCP/UDP Port Nummern. Anhand dessen werden Daten akzeptiert oder gesperrt.
    Richtig.

16. Forwarding ist ein äquivalenter Ausdruck für Routing.
    Nein. Forwarding beschreibt wohin ein Packet gesendet wird, während Routing die Route betrachtet welche das Packet vom Sender zu Empfänger hintergelegt hat.

17. Das Hot Potatoe Routing nutzt Flooding zur Datenweitergabe.
    Falsch. Anhand iBGP wird bereits vornherein ermittelt, wer der nächstgünstigste Nachbar zum übertragen ist. Zu diesem wird dann gesendet.

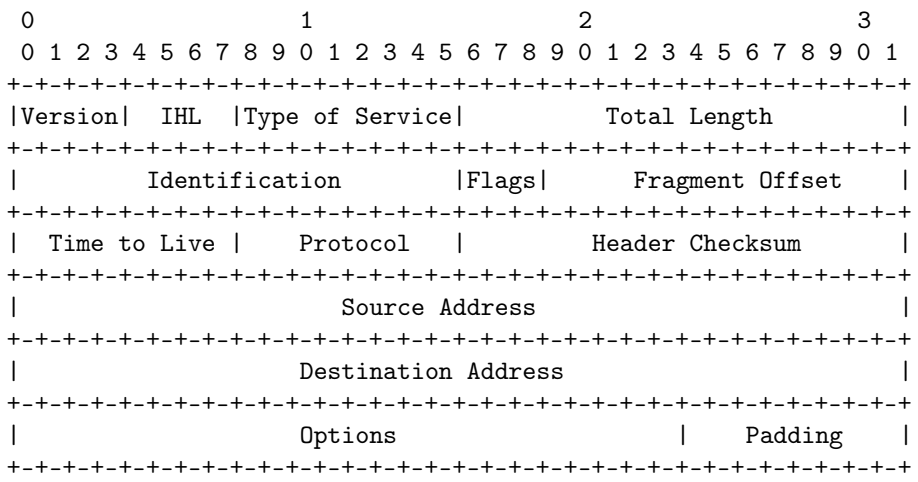
18. Der Networklayer implementiert die Funktionen des Forwarding und des Routing.
    Falsch, das wird von der Vermittlungsschicht übernommen. Der Netzworklayer dient dem Zustellen zur richtigen MAC-Addresse

19. Eine IPv4-Addresse deckt einen Adressbereich von $2^(32)$ Addressen ab.
    Richtig, wobei viele davon vorreserviert sind, wie etwa 255.255.255.255 (Broadcast).

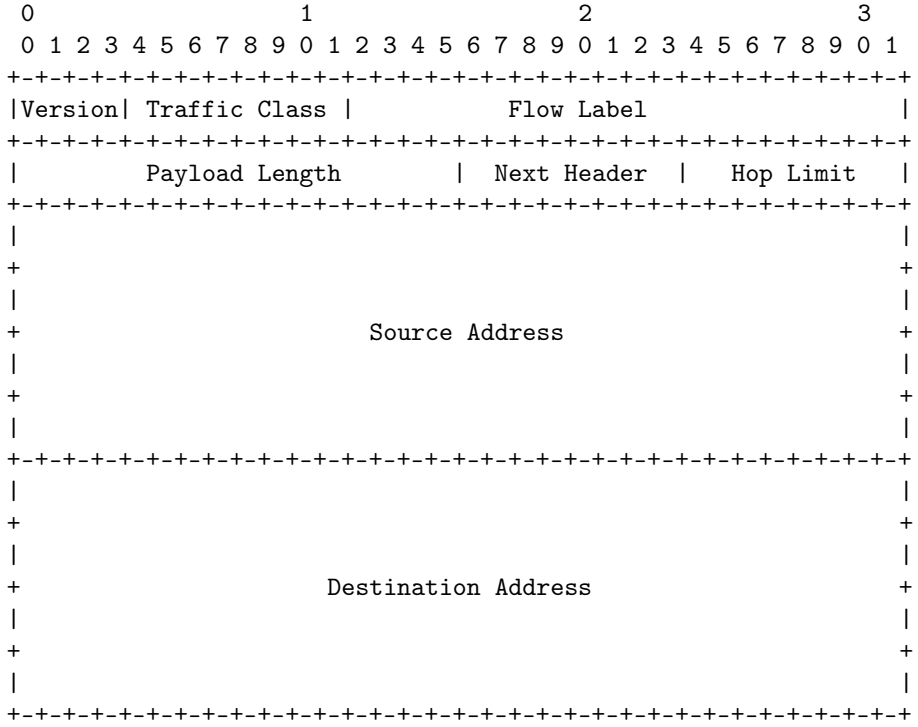
20. Jeder einzelne Router, welche ein Paket durchläuft, prüft die Checksumme des Paketes auf dessen Richtigkeit.
    Richtig. Bei Verwurf des Paketes wird auch kein neues angefordert. Alternativ, wenn die Time-To-Life ausläuft wird es ebenso verworfen.

21. IPv6 und IPv4 unterscheiden sich lediglich durch Header, welche in IPv6 neu implementiert wurden: Priority, Flow Label und Next Header-Field.
    Falsch, es gibt weitere Felder die geändert wurden.

IPv4 Header-felder:



IPv6 Header-Felder:

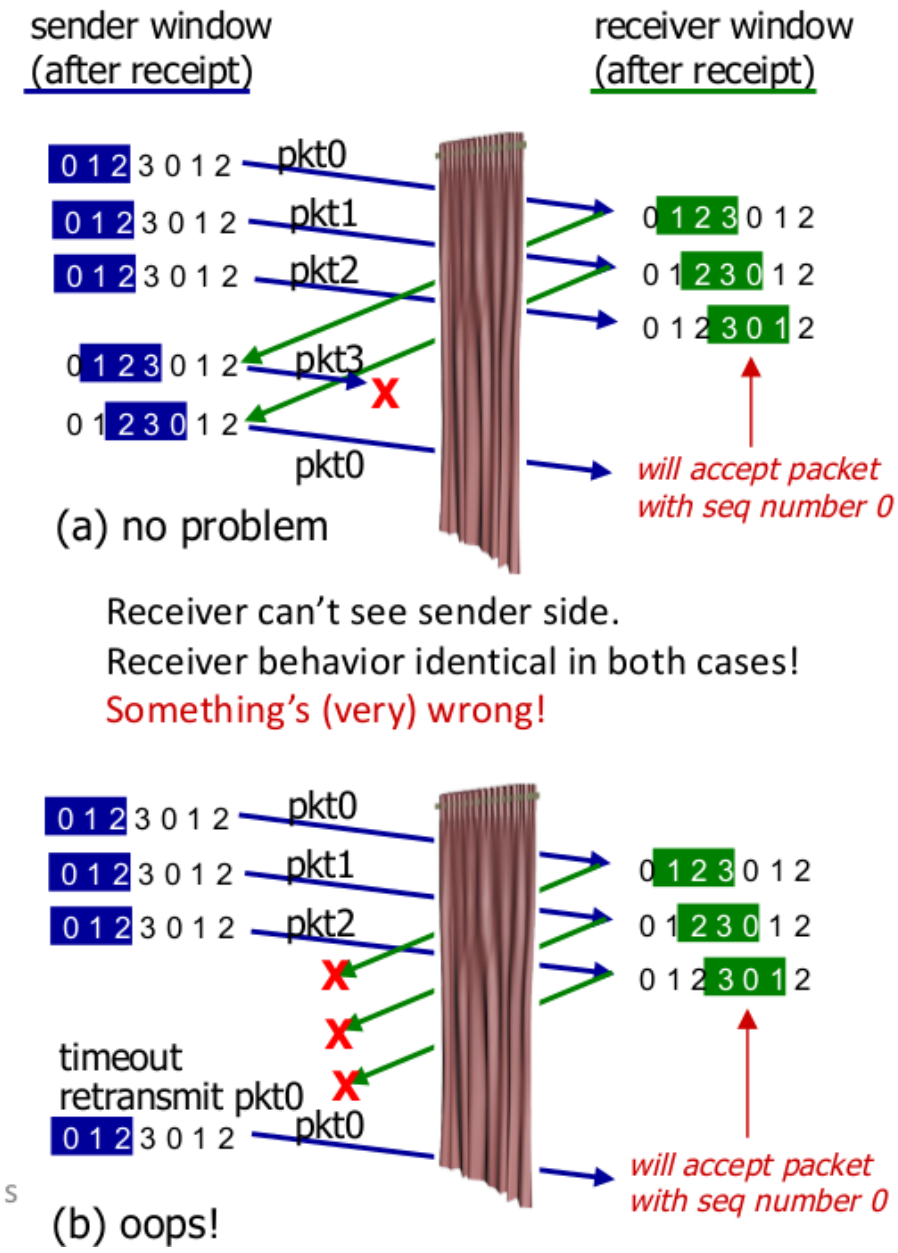


22. TCP implementiert Multicast.
    Falsch, TCP ist ein direktes P2P Protokoll. Multicast wird mittels UDP

implementiert, aufgrund des Abysmalen Overheads des TCP.

23. Existieren Bitfehler in einem Datenpaket, so können diese mithilfe der Checksumme erkannt werden.
    Richtig, die Checksumme wird in jedem Router neuberechnet.

24. Bei Selective Repeat existieren Fehlermöglichkeiten, welche den Algorithmus außer Kraft setzen.
    Ja, es gibt die Möglichkeit dass bei ACK-Verlust die Synchronisation von Klient und Server flöten geht.



25. Für Go-Back-N gilt: Es wird solange das gesamte Window neugesendet, bis die ACKs aller Pakete empfangen wurden.
Falsch. Es werden lediglich die Pakete gesendet, welche keine ACK zurücklieferten. Anschließend wird das Window nach vorne verschoben. Siehe:

`http://www.ccs-labs.org/teaching/rn/animations/gbn_sr/`

26. Die Abfrage von IP-Adressen mittels DNS ist sowohl rekursiv als auch iterativ ausführbar.
    Wahr.

27. Aufgerufene Adressen werden werden im Schnitt für 2 Tage gecached.
    Wahr, wobei diese Zeit konfiguierbar ist.

28. HTTP-Fehlercodes beginnend mit 5 beschreiben Klient-Fehler.
    Falsch, dies sind Server-Fehler. Klient-Fehler sind 4xx.

7 Sources

- https://www.elektronik-kompendium.de/sites/net/0503281.html
- http://www.ipv6-portal.de/tools/subnet-tabelle.html
- https://talvindersingh1992.wordpress.com/2013/07/07/multiple-access-techniques-tdma-fdma-c
- VO-Folien
- Wikipedia