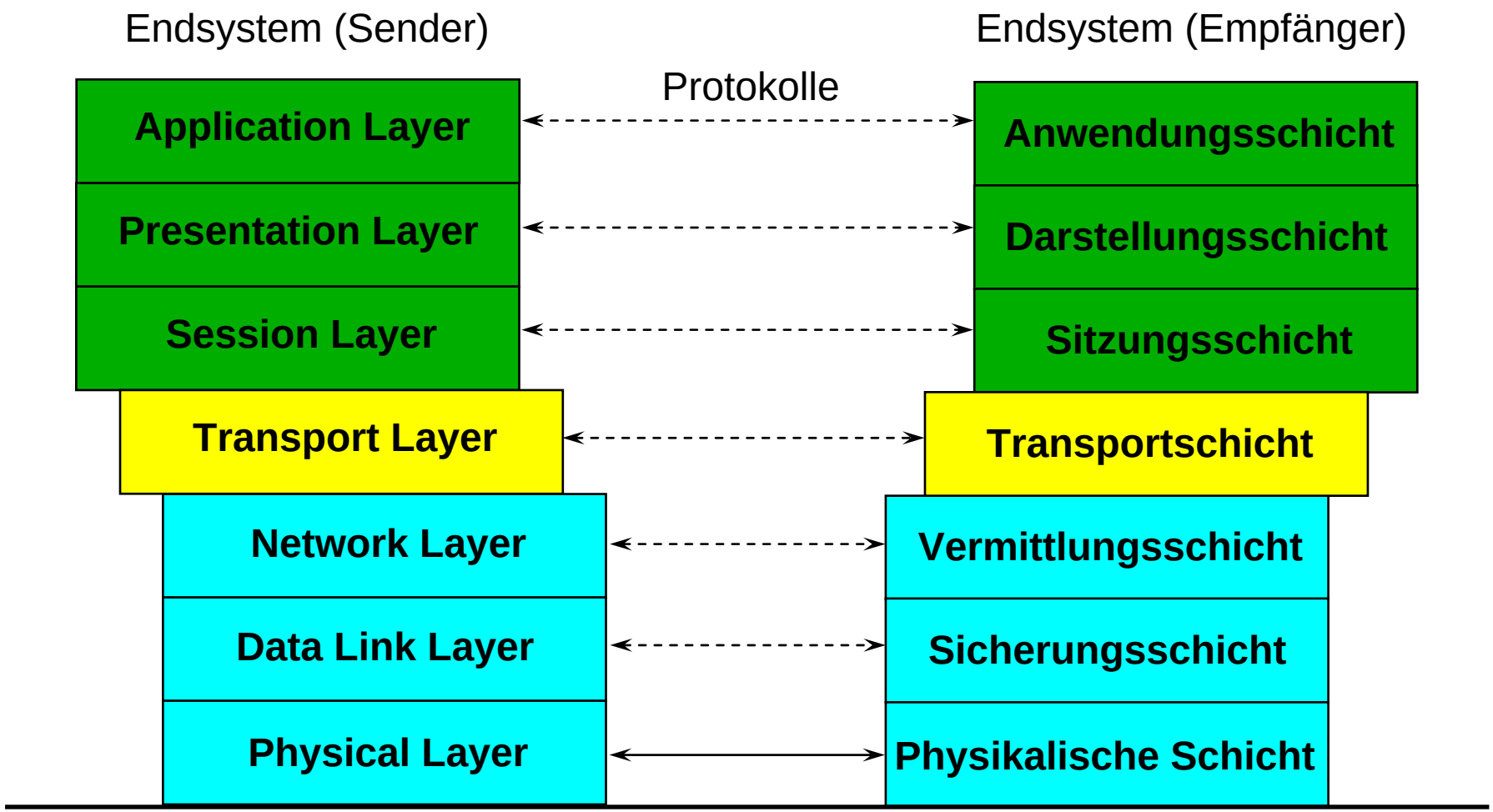
Heterogene Netze mit TCP/IP

Stand: 12.02.2006

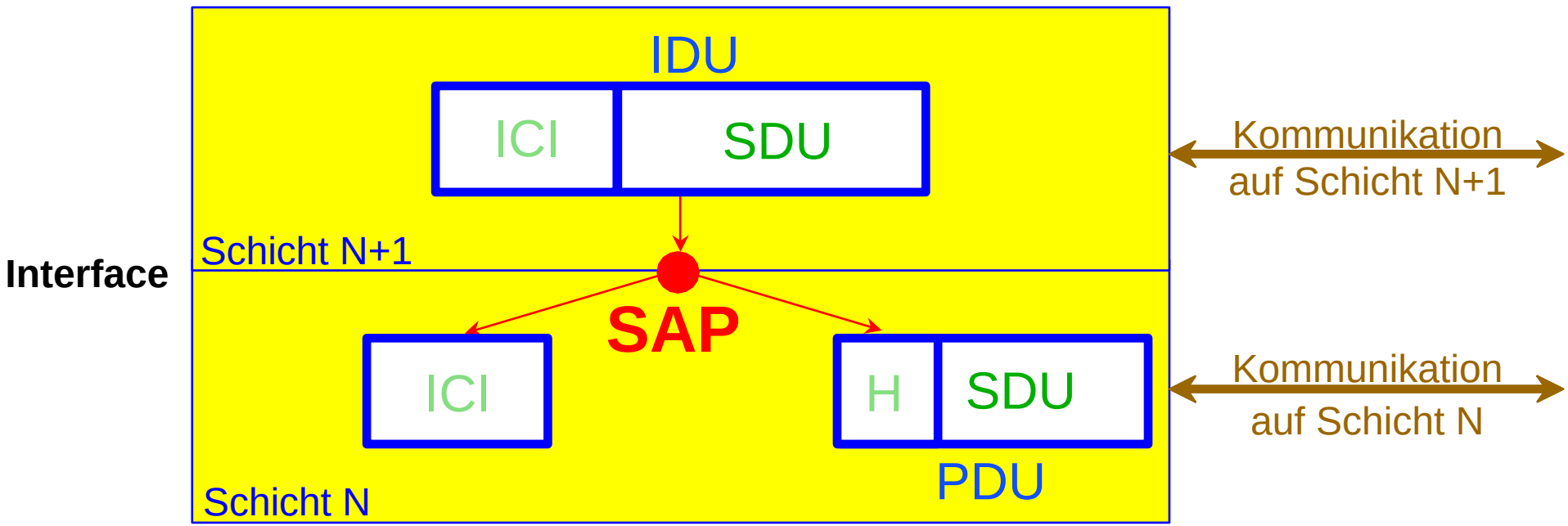
Kapitel 1

Grundlagen

ISO/OSI-Modell - Schichten



ISO/OSI-Modell - Schnittstelle



SAP	**Service Access Point**	**ICI**	**Interface Control Information**
IDU	**Interface Data Unit**	**SDU**	**Service Data Unit**
PDU	**Protocol Data Unit**	**H**	**Header**

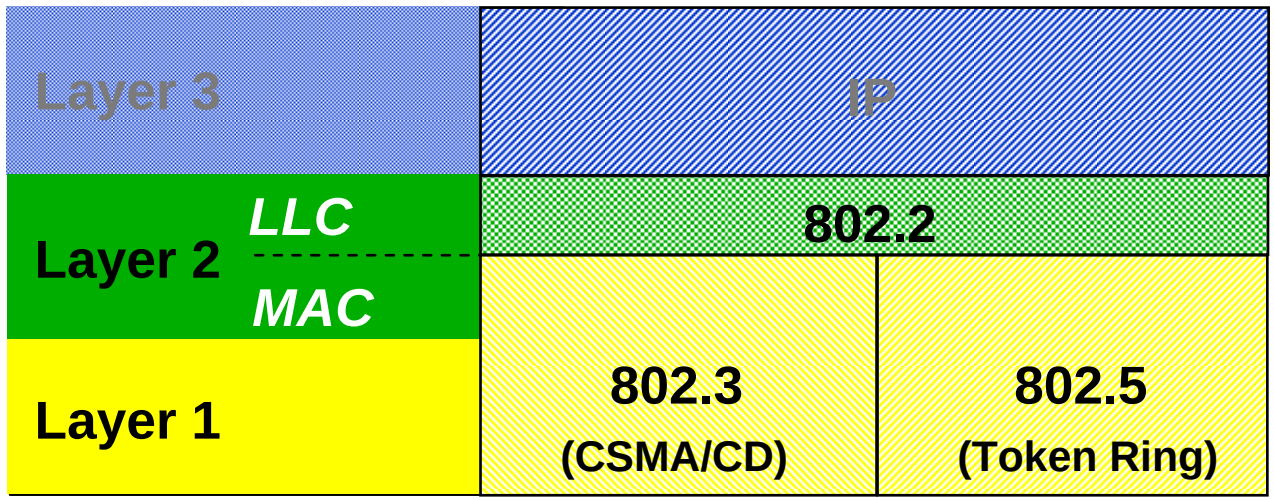
Standards der Arbeitsgruppe 802

802.1	Umfeld, LAN-/MAN-Management
802.1d	Transparent-/ SRT-Bridging
802.2	**Logical Link Control**
802.3	***CSMA/CD*) ("Ethernet")***
802.4	Token Bus
802.5	**Token Ring**
802.6	Distributed Queue Dual Bus (DQDB)
802.7	Broadband LANs
802.8	Multimode Fiber Optic Media
802.9	Integrated Services LAN
802.10	Std. for Interoperable LAN/MAN Security (SILS)
802.11	**Wireless LANs**
802.12	Demand Priority LAN > 10 MB ("VGanyLAN")
802.14	CATV-based Broadband Connectivity Networks
802.15	Wireless Personal Area Network (WPAN) - z.B. Bluetooth
802.16	**Worldwide Interoperability for Microwave Access** (WiMax)

http://standards.ieee.org/getieee802/portfolio.html

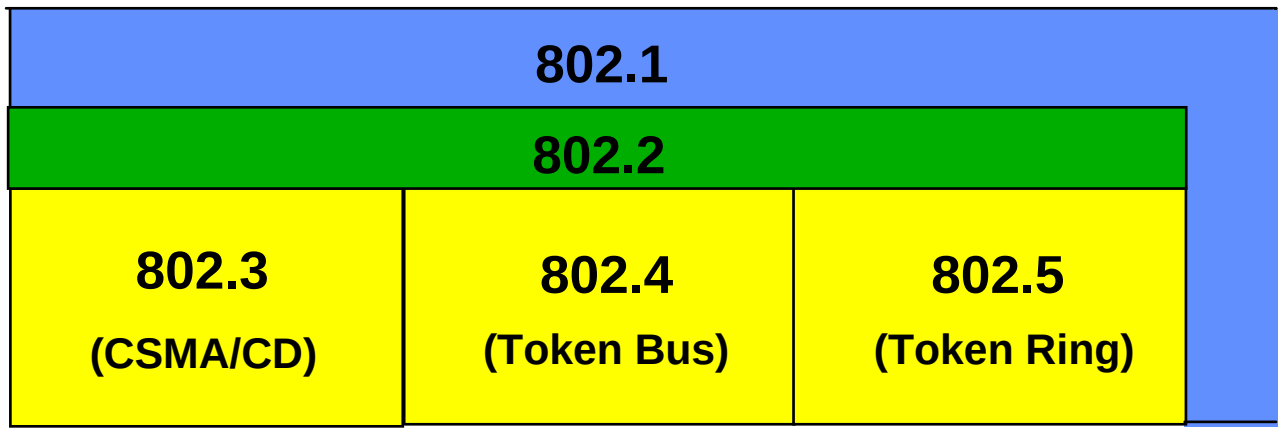
*) **C**arrier **S**ense **M**ultiple **A**ccess with **C**ollision **D**etection

IEEE-Standards, MAC und LLC



MAC **Medium Access Control**
LLC **Logical Link Control**

IEEE-Standards
802.1, 802.2, 802.3, 802.4, 802.5



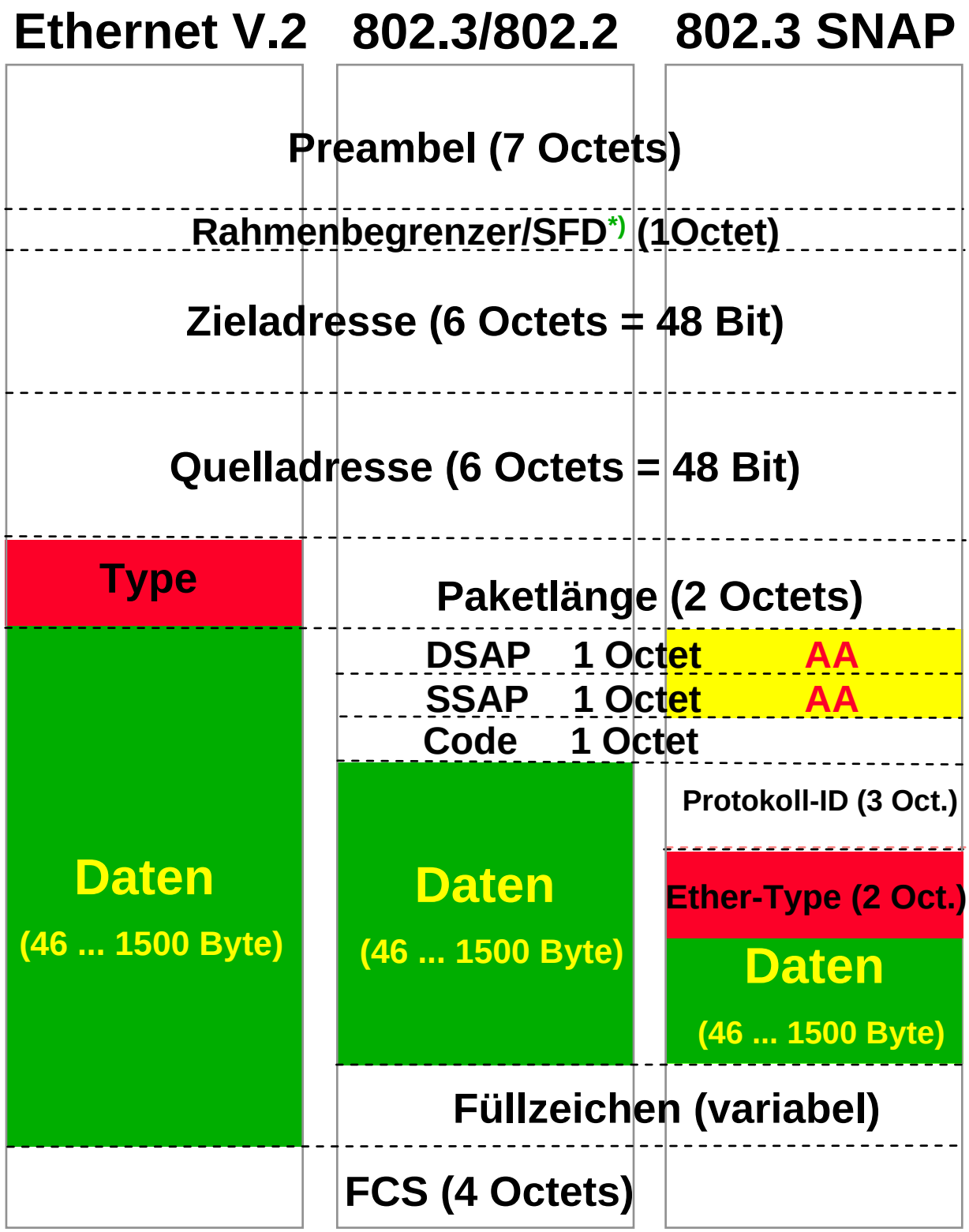
IEEE 802.3 (CSMA/CD) Standard Aktivitäten

802.3	**CSMA/CD (Ethernet): 10Base5**	
802.3a	**10Base2 (Cheapernet)**	
802.3b	10Broad36	
802.3e	1Base5 Starlan	
802.3i	**10Base-T**	
802.3j	10Base-F	
802.3u	**100Base-T** (“100 Mbit-Ethernet”)	
802.3x	Full Duplex/ Flow Control	
802.3z	Gigabit Ethernet	*(7/1998)*
802.3 ab	1000BASE-T	*(6/1999)*
802.3 ac	VLAN Tag	*(9/1998)*
802.3 ae	10Gb/s Ethernet	*(6/2002)*

IEEE 802.11 (WLAN) Standard Aktivitäten

- 802.11a 54 Mbps, 5 GHz - keine ETSI-Zulassung! (9/1999)
- **802.11b 11 Mbps, 2.4 GHz** (9/1999)
- 802.11d „World Mode“ (u.a. Roaming zwischen Ländern) (6/2001)
- 802.11e Quality Of Service
- **802.11g Higher Data Rate (> 20 Mbps)** (6/2003)
- **802.11i Authentication und Sicherheit** (inkl. WPA) (6/2004)

Ethernet- vs. 802.3-Pakete



*) SFD = Start Frame Delimiter

802.3 Paket (Aufbau)

Darstellung lt. IEEE 802.3 Standard:
Anordnung der Bits/ Bytes in Übertragungsreihenfolge
(höchstwertigstes Byte und niederwertigstes Bit werden zuerst übertragen)

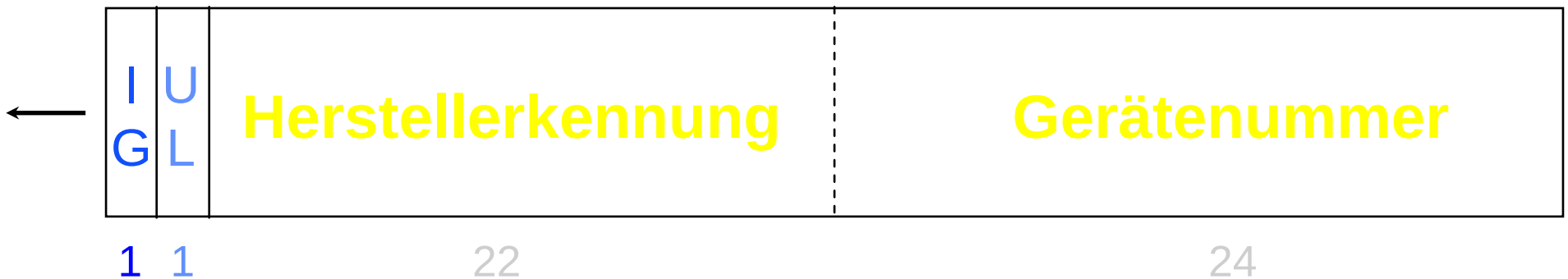
Herstellerkennungen (Auswahl):

00-00-5A	S&K	08-00-2B	DEC
08-00-02	3Com	AA-00-04	DECnet
08-00-09	HP	00-AA-00	Intel

Wichtige Herstellerkennungen

00-00-0C	**Cisco**
00-00-1D	**Cabletron**
00-00-24	**Olicom**
00-00-63	**HP** (LanProbe)
00-00-65	**Network General**
00-00-81	**Synoptics**
00-00-A2	**Wellfleet**
00-00-AC	**Conware**
00-00-B0	**RND**
00-00-C0	**SMC** (früher: WD)
00-00-F4	**Allied Telesis, Inc.**
00-00-F8	**DEC**
00-02-04	**Novell** NE3200
00-20-AF	**3COM Corporation**
00-60-09	**Cisco** Catalyst 5000 Ethernet switch
00-60-2F	**Cisco**
00-60-3E	**Cisco** 100Mbps interface
00-60-5C	**Cisco**
00-60-70	**Cisco** routers (2524 and 4500)
00-60-83	**Cisco** 3620/3640 routers
00-60-8C	**3Com**
00-60-97	**3Com**
00-60-B0	**Hewlett-Packard**
00-80-00	**Multitech Systems Inc**
00-80-16	**Wandel & Goltermann**
00-80-5F	**Compaq Computer Corporation**
00-80-63	**Richard Hirschmann Gmbh & Co**
00-80-C7	**Xircom, Inc.**
00-80-C8	**D-Link**
00-A0-00	**Bay Networks Ethernet switch**
00-DD-00	**Ungermann-Bass** IBM RT
00-DD-01	**Ungermann-Bass**
00-DD-08	**Ungermann-Bass**
00-E0-14	**Cisco Ethernet switch**
08-00-05	**Symbolics** LISP machines
08-00-06	**Siemens Nixdorf** PC clone
08-00-0F	**SMC** (Standard Microsystems Corp.)
08-00-1B	**Data General**
08-00-1E	**Apollo**
08-00-36	**Intergraph** CAE stations
08-00-38	**Bull**
08-00-39	**Spider Systems**
08-00-3E	**Motorola** VME bus processor modules
08-00-7C	**Vitalink** TransLAN III

802.3 Paket-Aufbau
Multicast-Adressen



Xxxx1 - XX - XX - XX - XX - XX

Adressen, die im **ersten Byte** einen **ungeraden** Wert haben, sind Multicast-Adressen

z.B.	09-00-09-00-00-01	HP-Probe
	AB-00-00-XX-XX-XX	DECnet Broadcast
	CF-00-00-00-00-00	Ethernet-Loop-Back
	FF-FF-FF-FF-FF-FF	**Ethernet Broadcast**

Wichtige Typfelder

00-00	***...***	***05-DC***	***IEEE802.3 Length Field*** *(05-DD ... 05-FF nicht vergeben!)*
06-00			**Xerox NS IDP**
08-00			**DOD Internet Protocol (IP)**
08-06			**Address Resolution Protocol (ARP)**
0B-AD			**Banyan Systems**
0B-AF			**Banyon VINES Echo**
60-00			**DEC unassigned, experimental**
60-01	**...**	**60-08**	**DEC**
80-05			**HP Probe protocol**
80-35			**Reverse Address Resolution Protocol (RARP)**
80-38	**...**	**80-42**	**DEC**
80-7D	**...**	**80-80**	**Vitalink**
80-9B			**EtherTalk (AppleTalk over Ethernet)**
80-F3			**AppleTalk Address Resolution Protocol (AARP)**
80-FF	**...**	**81-03**	**Wellfleet Communications**
81-37	**...**	**81-38**	**Novell, Inc.**
90-00			**Loopback (Configuration Test Protocol)**
90-01	**...**	**90-03**	**3Com** (früher: Bridge Communications)

Wichtige DSAPs/ SSAPs

00	Null SAP
02	Individual LLC Sublayer Mgmt Function
03	Group LLC Sublayer Mgmt Function
06	**ARPANET Internet Protocol (IP)**
42	**IEEE 802.1 Bridge Spanning Tree Protocol**
80	Xerox Network Systems (XNS)
98	**ARPANET Address Resolution Protocol (ARP)**
AA	**Sub-Network Access Protocol (SNAP)**
BC	Banyan VINES
E0	Novell Netware
F0	IBM NetBIOS
F4/ F5	IBM LAN Management
FE	ISO Network Layer Protocol
FF	Global SAP

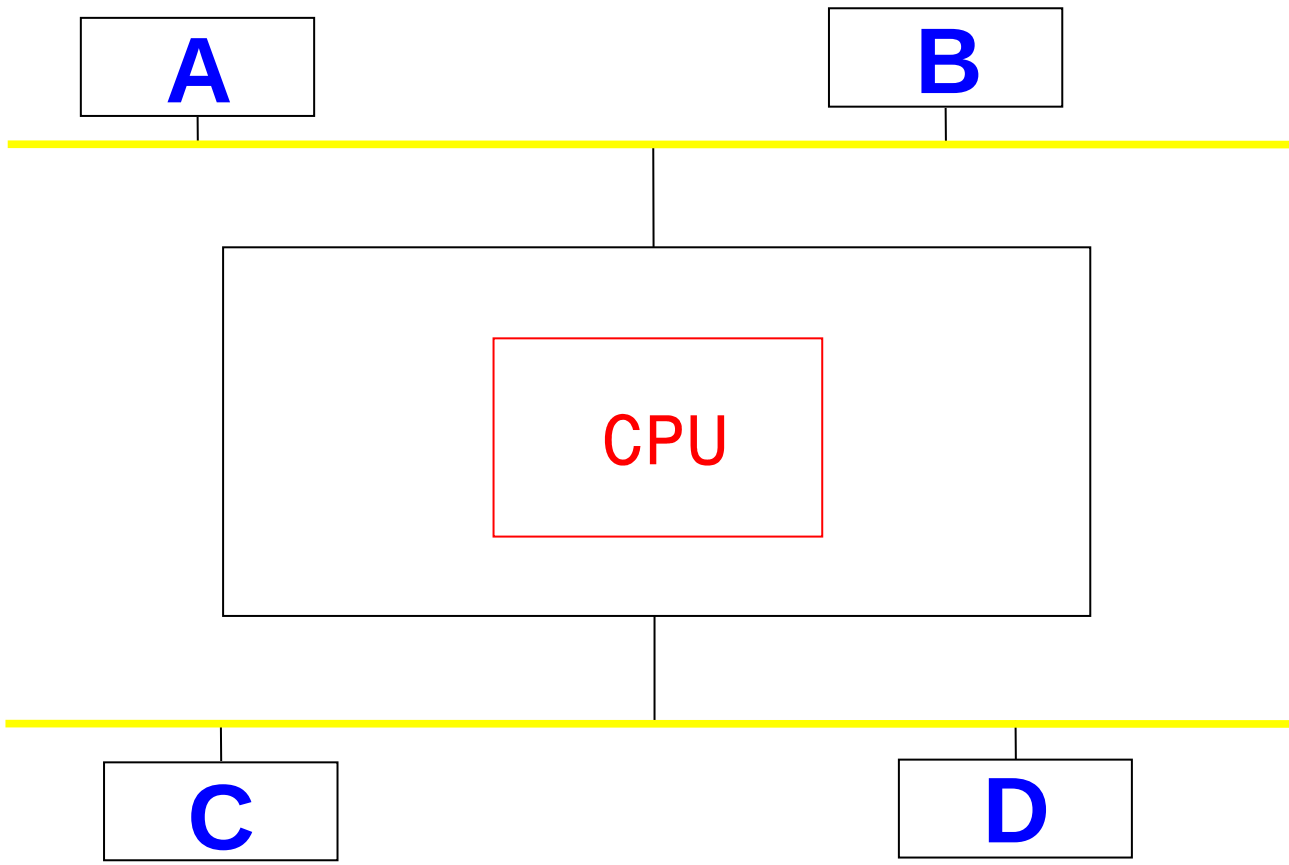
Komplette Listen zu Herstelleradressen, Typ-Feldern und DSAP/ SSAP

www.cavebear.com/CaveBear/Ethernet/**vendor.html**	**Herstellerkennung**
www.cavebear.com/CaveBear/Ethernet/**type.html**	**Typ-Felder**
www.cavebear.com/CaveBear/Ethernet/**multicast.html**	**Multicast-Pakete** (Adresse + Typ)
wwwhost.ots.utexas.edu/ethernet/enet-numbers/**ieee-oui-list.txt**	**Herstellerkennung** (von IEEE mit Anschrift - aber nicht ganz so umfangreich)
wwwhost.ots.utexas.edu/ethernet/enet-numbers/**ieee-lsap-list.txt**	**DSAP/ SSAP** (bei IEEE registriert)
wwwhost.ots.utexas.edu/ethernet/enet-numbers/**cisco-lsap-list.txt**	**DSAP/SSAP** (von CISCO installiert)

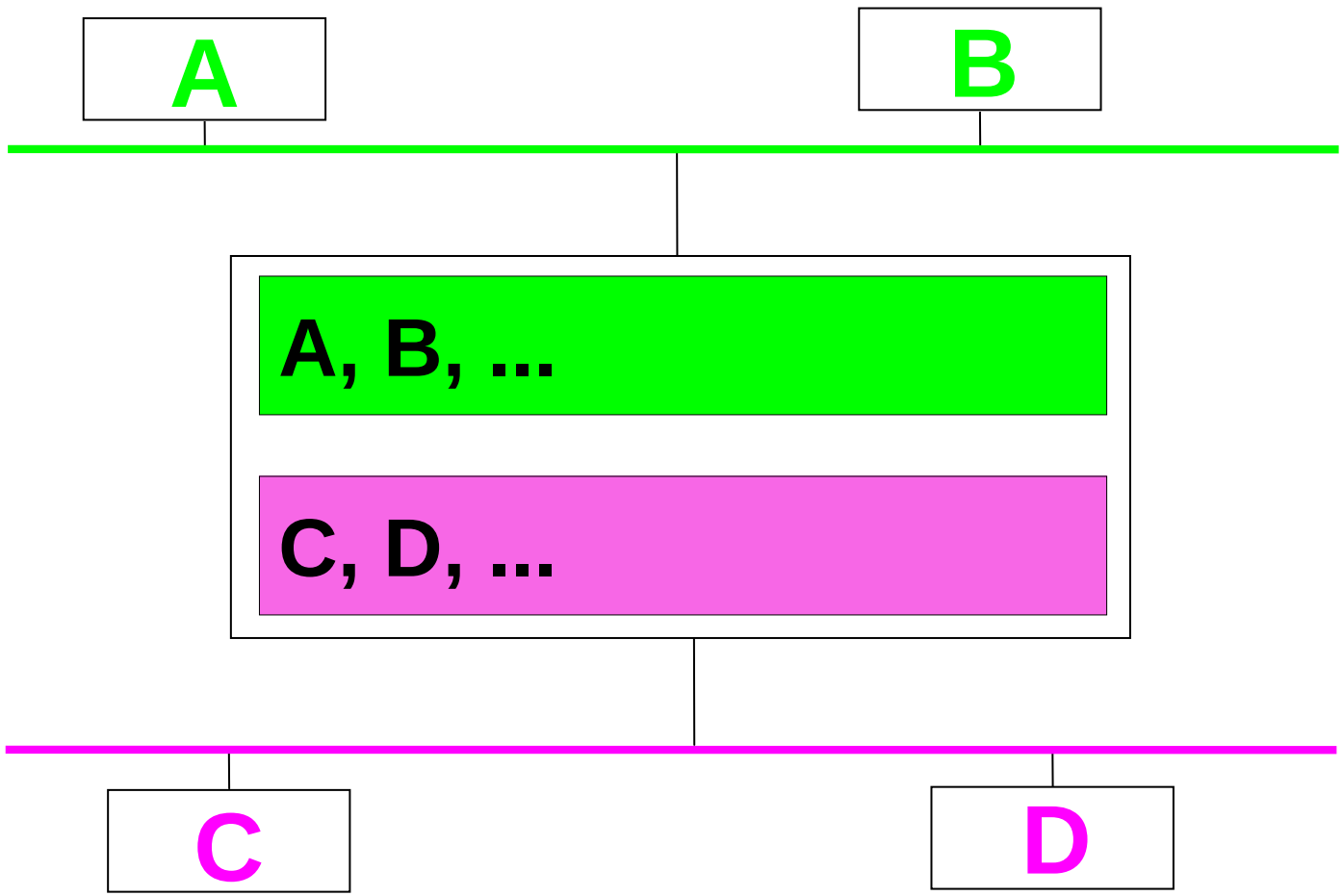
Bridges (Begriffe/ Bauweisen)

- ISO-Layer-2-Bridge
- MAC-Layer-Bridge
- Intelligente Bridge
- Filtering Bridge
- Local Bridge
- Remote Bridge
- Multiport-Bridge
- Spanning-Tree-Bridge
- Routing-Bridge
- Source-Routing-Bridge
- Switch
- Hub

Bridge - Arbeitsweise (1)



Bridge - Arbeitsweise (2)

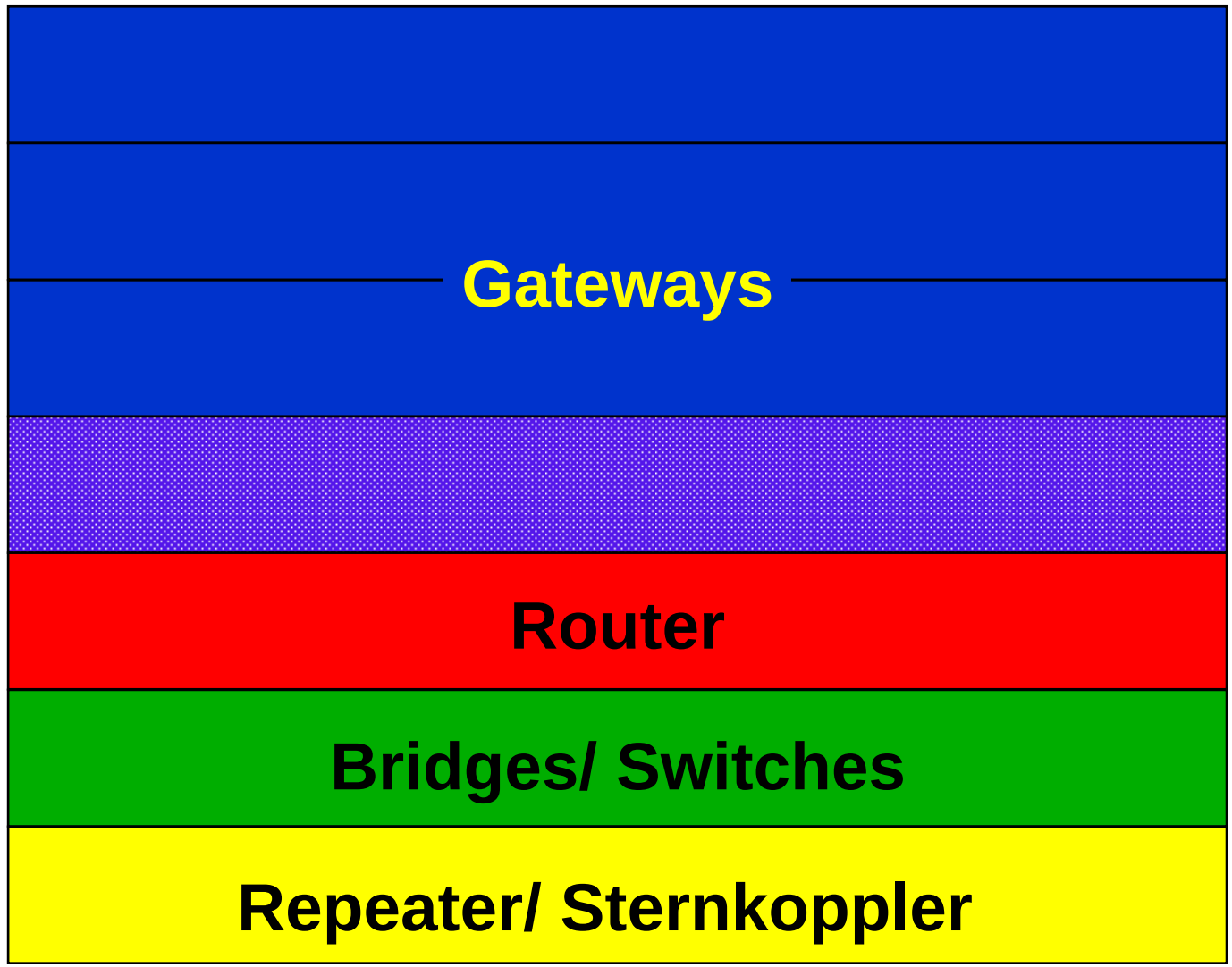


Bridges (Begriffe)

- **Filtering Rate**
  - ➔ Anzahl der Pakete, die sich eine Bridge “anschauen” kann
- **Forwarding Rate**
  - ➔ Anzahl der Pakete, die eine Bridge weiterreichen kann

  Achtung: Häufig Summe für beide/alle Übertragungsrichtungen!

Transitsysteme im OSI-Modell



Transitsysteme im OSI-Modell (Aufgaben)

Repeater: (Hub)

- Regeneriert und verstärkt das elektrische Signal.
- Führt keine "Bitinterpretation" durch.

Bridge/ Switch:

- Nimmt physikalische Trennung von Netzen vor („Collision-Domain").
- Führt Fehler- und Lasttrennung (auf Basis von MAC-Adressen) durch.
- Hat meist Mechanismen zum Filtern implementiert.
- Rudimentäre Mechanismen zur Wegefindung sind u.U. vorhanden ("Routing Bridge")

Router:

- Entkoppelt die (Sub-) Netze auf logischer (Protokoll-) Basis aufgrund von Layer 3-Adressen (z.B. IP-Adressen).
- Steuert den Verkehr zwischen Netzen ("Wegefindung").
- Arbeitet protokollabhängig!

Gateway:

- Nimmt eine Umwandlung von Diensten vor.
- Security-Mechanismen möglich (z.B. "Firewall", "Proxy").

Protokollarten

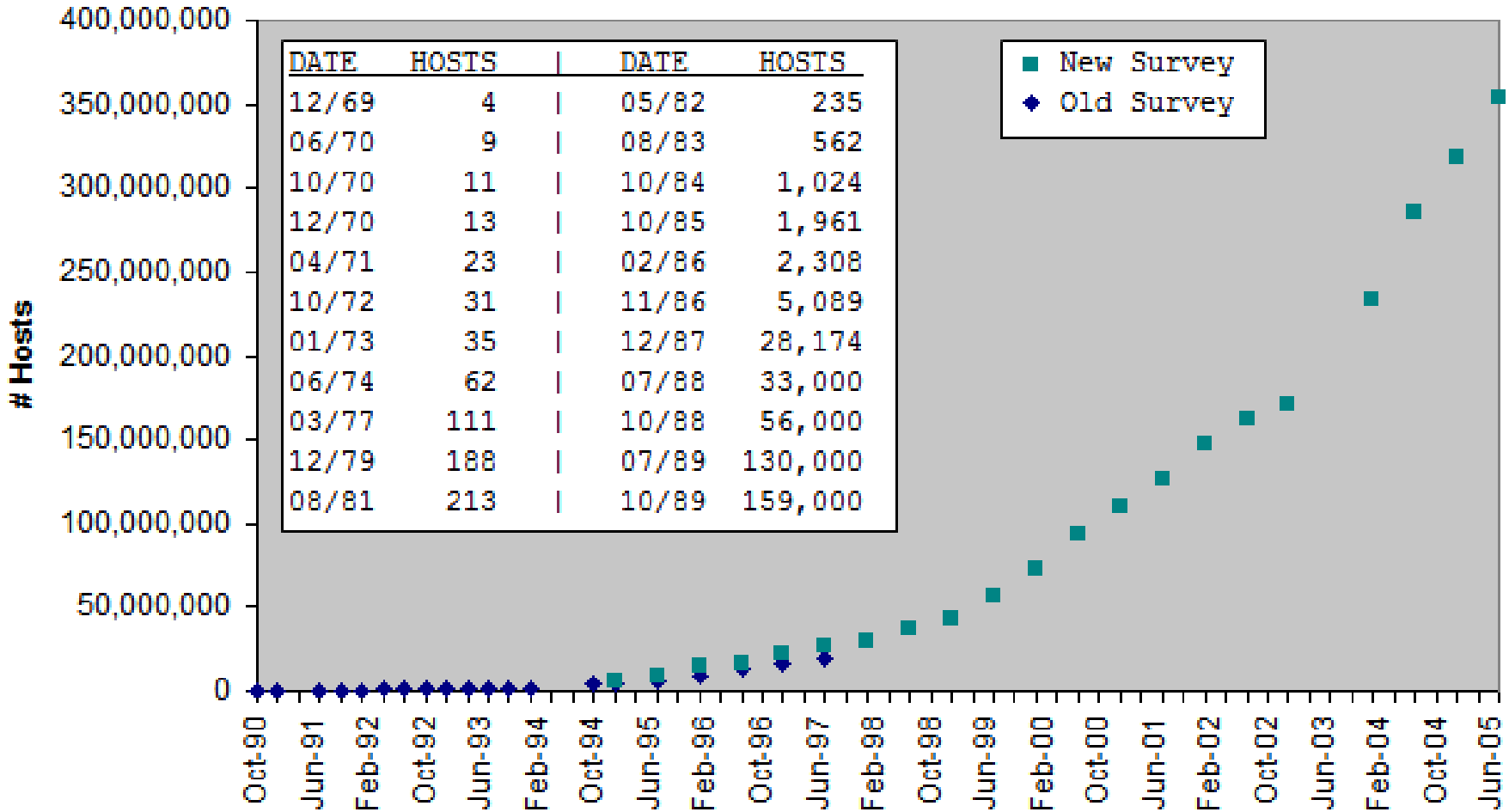
- **Verbindungsorientiert** (connection-oriented)
  logische Verbindung zwischen Kommunikations-Partnern

- **Verbindungslos** (connectionless, datagram-service)
  keine logische Verbindung
  Pakete werden unkontrolliert übertragen

TCP/IP-History (Überblick)

1969	erste Arbeiten an einem paketvermittelnden Rechnernetz
1972	das **ARPANET** wird der Öffentlichkeit vorgestellt
1973	“Ethernet is born”
1975	die DCA (Defence Communications Agency) übernimmt die Federführung im ARPANET
1976	Grundsteinlegung zu TCP/IP durch die IFIP (International Federation Of Information Processing)
1979	**D**EC, **I**ntel und **X**EROX (**DIX**-Group) entwickeln gemeinsam das Ethernet weiter
1980	Ethernet Version 1.0 wird veröffentlicht Bercley UNIX (BSD 4.1) wird entwickelt und enthält TCP/IP
1983	Das ARPANET wird endgültig von NCP auf TCP/IP umgestellt Aufteilung des ARPANET in MILNET und ARPANET
1985	Einführung von TCP/IP in kommerzielle Anwendungen
1991	mehr als 1000 Hersteller unterstützen TCP/IP
1993	mehr als 10000 Hersteller unterstützen TCP/IP
1994	**WWW** wird offizielles Projekt von CERN, die W3-ORG wird ins Leben gerufen
1996	das Internet umfasst ca. 15 Mio. Anschlüsse
2001	im November wird die **5 Mio. DE-Domain** vergeben - pro Minute werden 2 Domains vergeben (90 000/ Monat) - Start .DE am 5.11.1986

TCP/IP-History (Rechnerentwicklung)



DATE	HOSTS	DATE	HOSTS
12/69	4	05/82	235
06/70	9	08/83	562
10/70	11	10/84	1,024
12/70	13	10/85	1,961
04/71	23	02/86	2,308
10/72	31	11/86	5,089
01/73	35	12/87	28,174
06/74	62	07/88	33,000
03/77	111	10/88	56,000
12/79	188	07/89	130,000
08/81	213	10/89	159,000

Quelle: Hobbes Timeline: http://www.zakon.org/robert/internet/timeline

RFCs, MIL-Specs u.a.

RFC: **Request For Comments**

Arbeitspapiere, Protokollspezifikationen und Kommentare der Internet-Community

(Veröffentlicht durch das Stanford Research Institut: *www.rfc-editor.org*
Jahresende 2005: 4326 RFCs)

MIL-STD: **Ausführliche Beschreibung und Implementierungs-anweisung wichtiger DoD-Protokolle**

IEN: **Internet Experimental/ Engineering Notes**

Vorläufer der RFCs

Standardisierungsprozess / RFCs (1)

- **Offener Prozess**
- **Entwicklung durch Arbeitsgruppen der Internet Engineering Task Force (IETF)**
- **Entscheidung durch Internet Engineering Steering Group (IESG)**
- **Veröffentlichung in RFC**

Achtung:

Nicht jeder RFC beschreibt einen Standard („STDxxxx“)!

Auflistung aller Standards in **STD 1** (z.Z. RFC 3700)

Standardisierungsprozess/ RFCs (2)

Standardisierungsstufen (STD)

- Internet Draft (i.A. Arbeitsgruppe)
- Proposed Standard
- Draft Standard
- Internet Standard

Standardisierungsprozess/ RFCs (3)

Keine Standards:

- Experimental
- Informational (FYIxxxx)
- Best Current Practice (BCPxxxx)
- RARE*) Technical Reports (RTRxxxx)
- Historic

*) RARE = Reseaux Associes pour la Recherche Europeenne

RFCs - How to Publish an RFC

- **Independent Submissions**

  ... anyone can write an RFC and independently submit it to the RFC Editor for possible publication ...

- **Formatting Hints**

  [...] The rules for formatting RFCs were set in the days when most text editors basically handled ASCII text files. Life is more difficult for those who must use Microsoft Word to produce their RFCs. [...]

Wichtige Protokolle und deren RFCs

- **RFC 768** **UDP** (August 1980)
- **RFC 791** **IP** (September 1981); ersetzt: RFC 760
  Vorgänger: IEN 123
- **RFC 792** **ICMP** (September 1981); ersetzt: RFC 777
  Vorgänger: RFC 760; IEN 123
- **RFC 793** **TCP** (September 1981)
- **RFC 821** **SMTP** (August 1982); ersetzt: RFC 788
  Vorgänger: RFC 780, 772 (Sept. 1980)
- **RFC 826** **ARP** (November 1982)
- **RFC 854** **TELNET** (Mai 1983); ersetzt: RFC 764
- **RFC 959** **FTP** (Oktober 1985); ersetzt: RFC 765
  Vorgänger: RFC 542, 354, 265/264, 172, 114 (April 1971)

Kapitel 2

Internet Protocol (IP)

Internet Protocol (IP)

RFC 791 - STD 5 - MIL-Std. 1777

- **setzt auf dem Data Link Layer (Ethernet, TR etc.) auf**
- **nutzt (Ethernet-) Typefield: 08-00**
- **besitzt eine 802.2 DSAP/SSAP-Definition: 06**
- **ist Datagram-Service**
- **ermöglicht Verbindungen zwischen Netzen**
- **bietet Datentransport von einer Quell- zu einer Zieladresse**

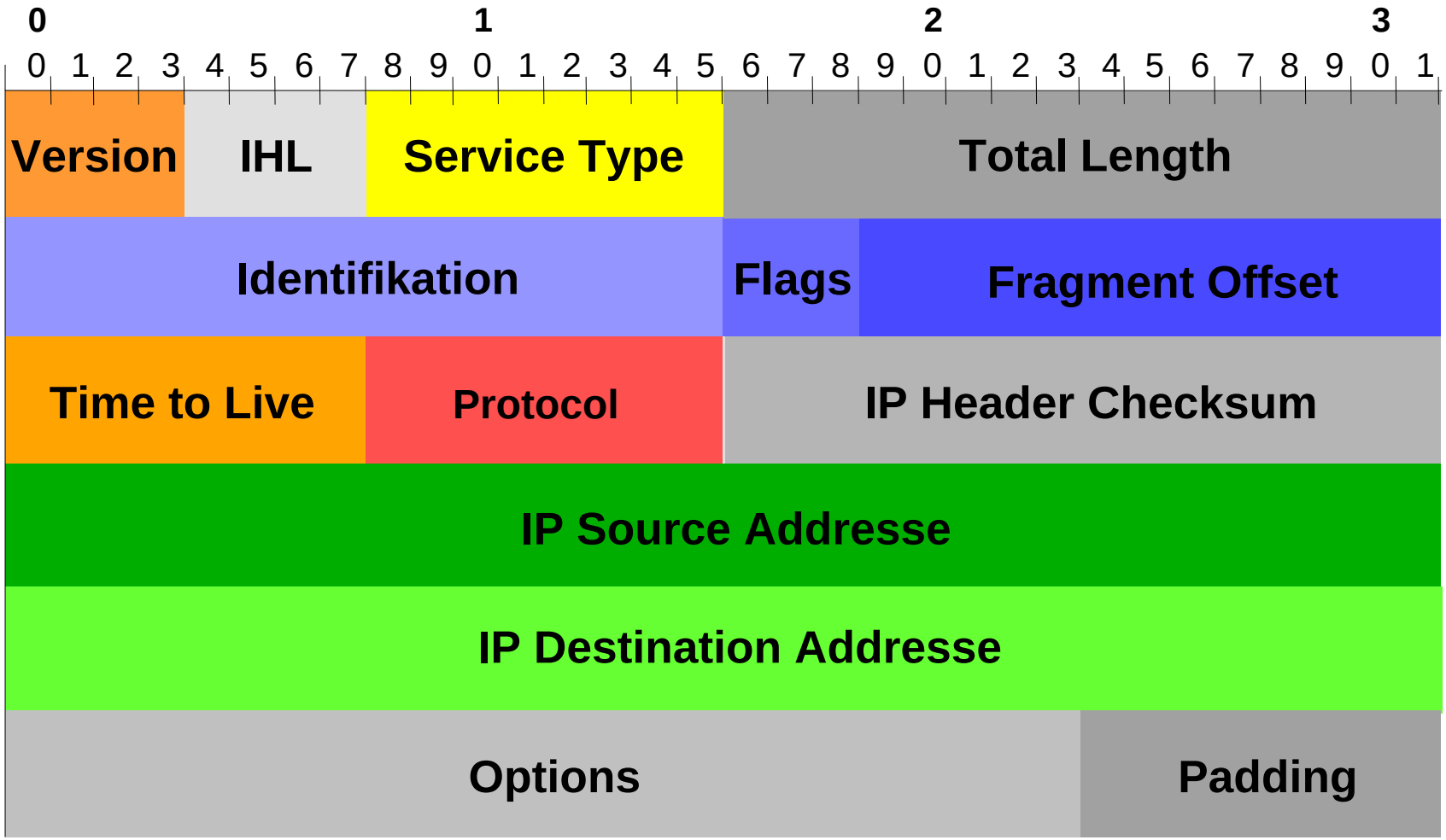
IP - Wichtige RFCs

RFC 791	**IP-Protokoll (STD 5)**
RFC 815	**IP over X.25 Networks**
RFC 894	**IP over Ethernet-Networks**
RFC 948	**IP over 802.3 Networks**
RFC 1051	**IP over Arcnet-Networks**
RFC 1055	**IP over Serial Lines (“SLIP”)**
RFC 1088	**IP over Netbios Networks**
RFC 1577	**IP over ATM Networks (“Classical IP”)**

IP - Eigenschaften

- **Datagram-Service (ungesichert!)**
- **Definition/ Adressierung höherer Protokolle**
- **Adressfunktion**
- **Routing zwischen Netzen**
- **Fragmentierung von Datenpaketen**
- **Wahl von Übertragungsparametern**

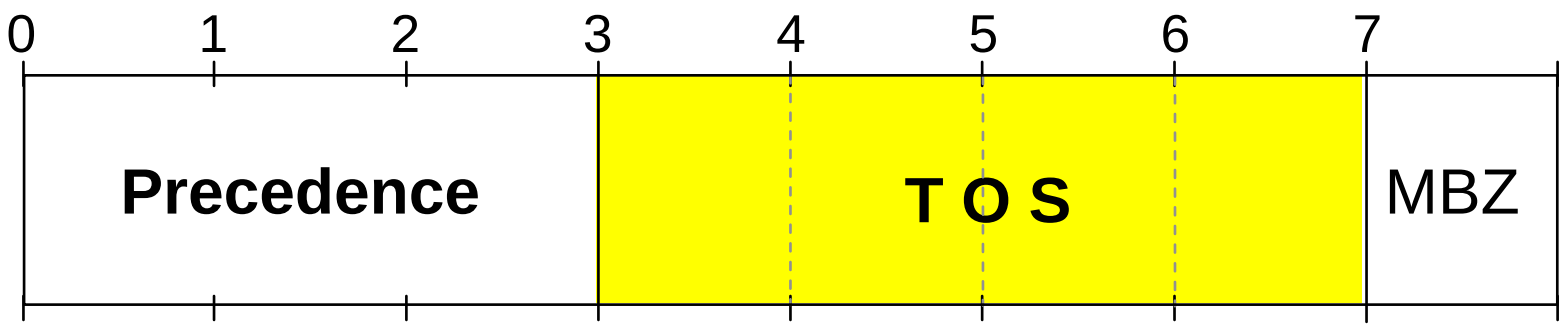
IP - Header



Service-Type (Neu-Definition)

RFC 1349

- **ersetzt RFC 791**
- **TOS (Type Of Service)**
- **4 Bit-Feld wird als Wert interpretiert**



Precedence = Vorrangssteuerung

MBZ = Must Be Zero

IP - TOS

Werte

0000	Default
0001	Minimize Monetary Cost
0010	Maximize Reliability
0100	Maximize Throughput
1000	Minimize Delay
1111	Maximize Security

IP - TOS

Default Werte bei verschiedenen Diensten

TELNET	**1000**	**minimize delay**
FTP Control	**1000**	**minimize delay**
FTP Data	**0100**	**maximize troughput**
SMTP (Command Phase)	**1000**	**minimize delay**
SMTP (Data Phase)	**0100**	**maximize troughput**
SNMP	**0010**	**maximize reliability**
ICMP	**0000**	*aber: request = response*

IP - Fragmentierung
Warum Fragmentierung

- **Hardware-/ Software-Beschränkungen, Definition des Protokolles, Beschränkung durch Norm**
  (z.B. Topologie-Übergang)
- **Maßnahmen zur Fehlerreduktion**
- **zum Erhöhen der “Zugangsgerechtigkeit” auf Datenkanal (Begrenzung der Zugriffszeit)**

IP - Fragmentierung
max. Paketlänge auf verschiendenen Netzen

Medium	Bit	Byte
• **Token Ring** (16 Mbit/s)	143928	**17997**
• **Token Ring** (4 Mbit/s)	36008	**4501**
• **Ethernet**	12144	**1518**
• **X.25** (Maximum)	8192	**1024**
• **X.25** (Standard)	1024	**128**

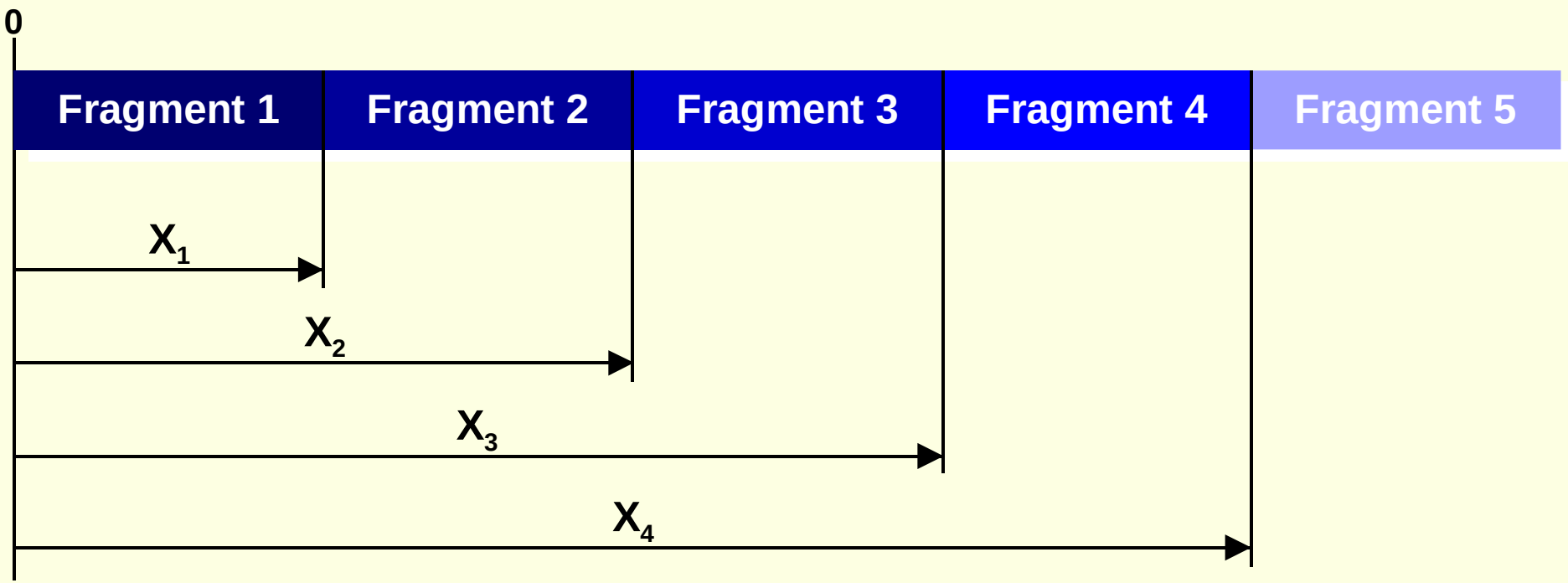
IP - Fragmentierung

Fragment Offset

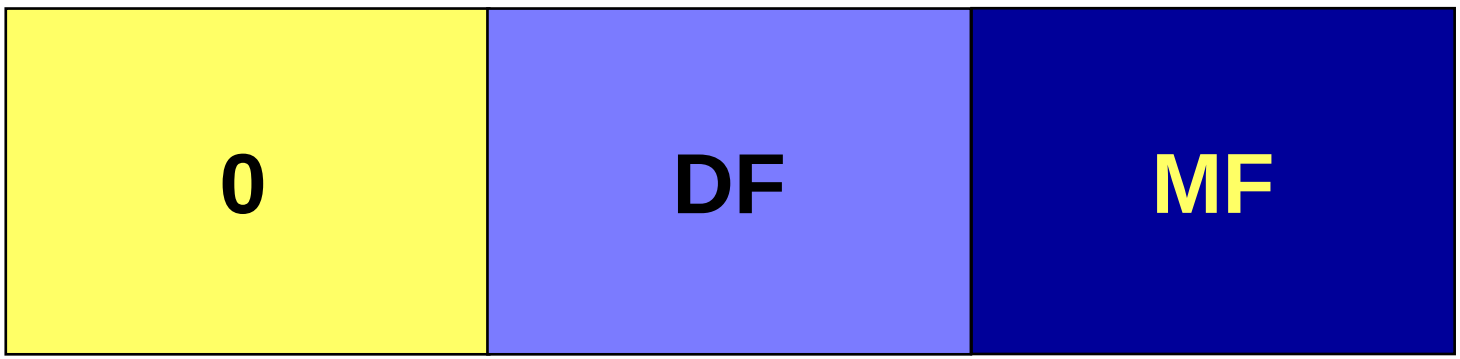
- Gibt die Länge relativ zum Beginn des Datenbereichs im ursprünglichen Datagram an
- Ermöglicht dem Empfänger mehrere Fragmente in richtiger Reihenfolge zusammenzusetzen
- Bei vollständigen Datagramen (keine Fragmentierung) und beim ersten Fragment hat der Fragment Offset immer den Wert 0

IP - Fragmentierung

Fragment Offset



IP - Fragmentierung
Flags



DF (Don’t Fragment): 0 = May Fragment
1 = Don’t Fragment

MF (More Fragment): 0 = Last Fragment
1 = More Fragment

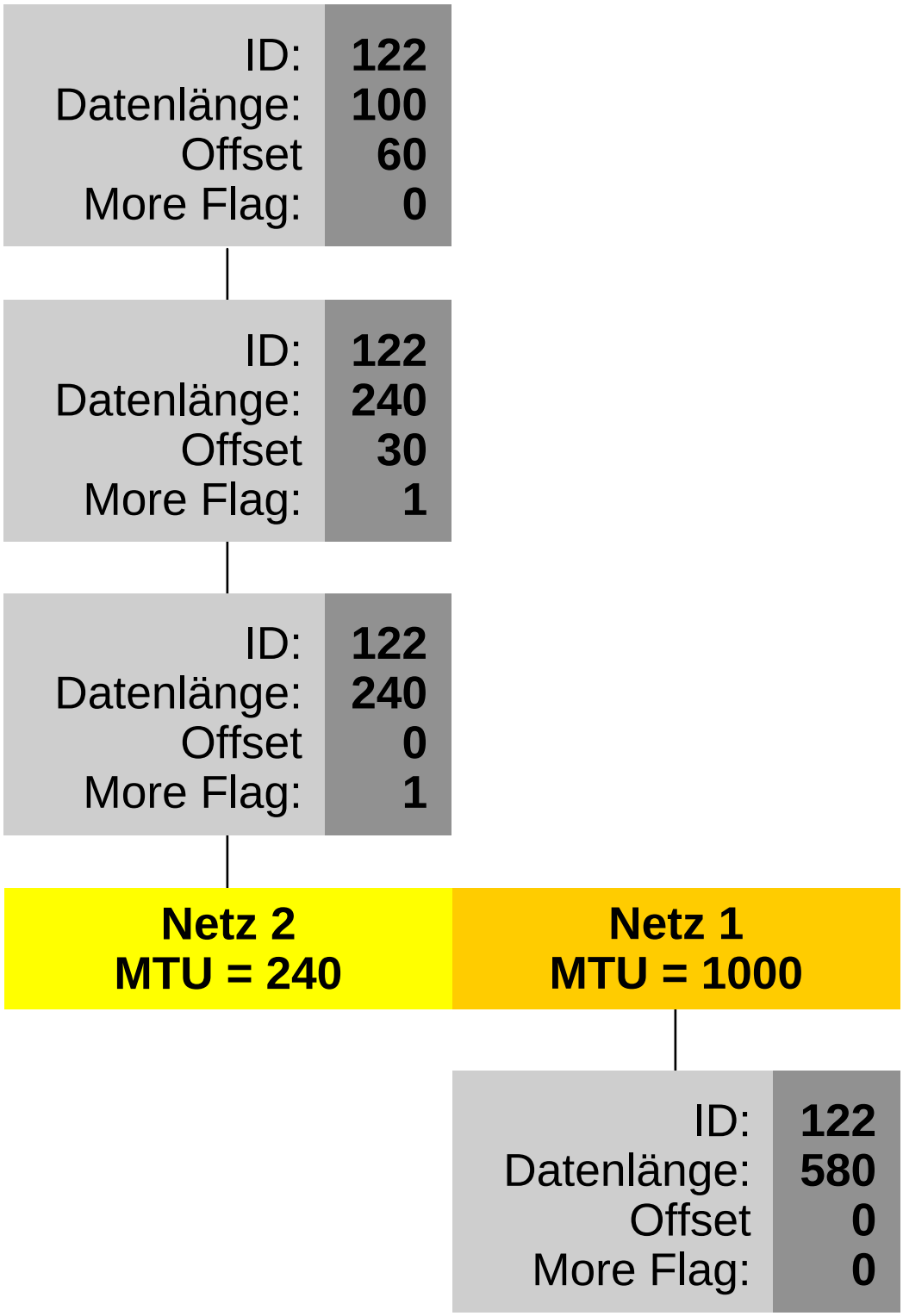
IP - Fragmentierung

- **Veränderte Felder im Header**
  - ➔ Gesamtlänge
  - ➔ Flags (MF)
  - ➔ Fragment-Offset
  - ➔ IP-Header-Prüfsumme
  - ➔ Optionen

IP - Fragmentierung / Reassemblierung

- **Identische Felder bei Reassemblierung**
  - ➔ Zieladresse
  - ➔ Quelladresse
  - ➔ Protokoll-Typ
  - ➔ Identifikation

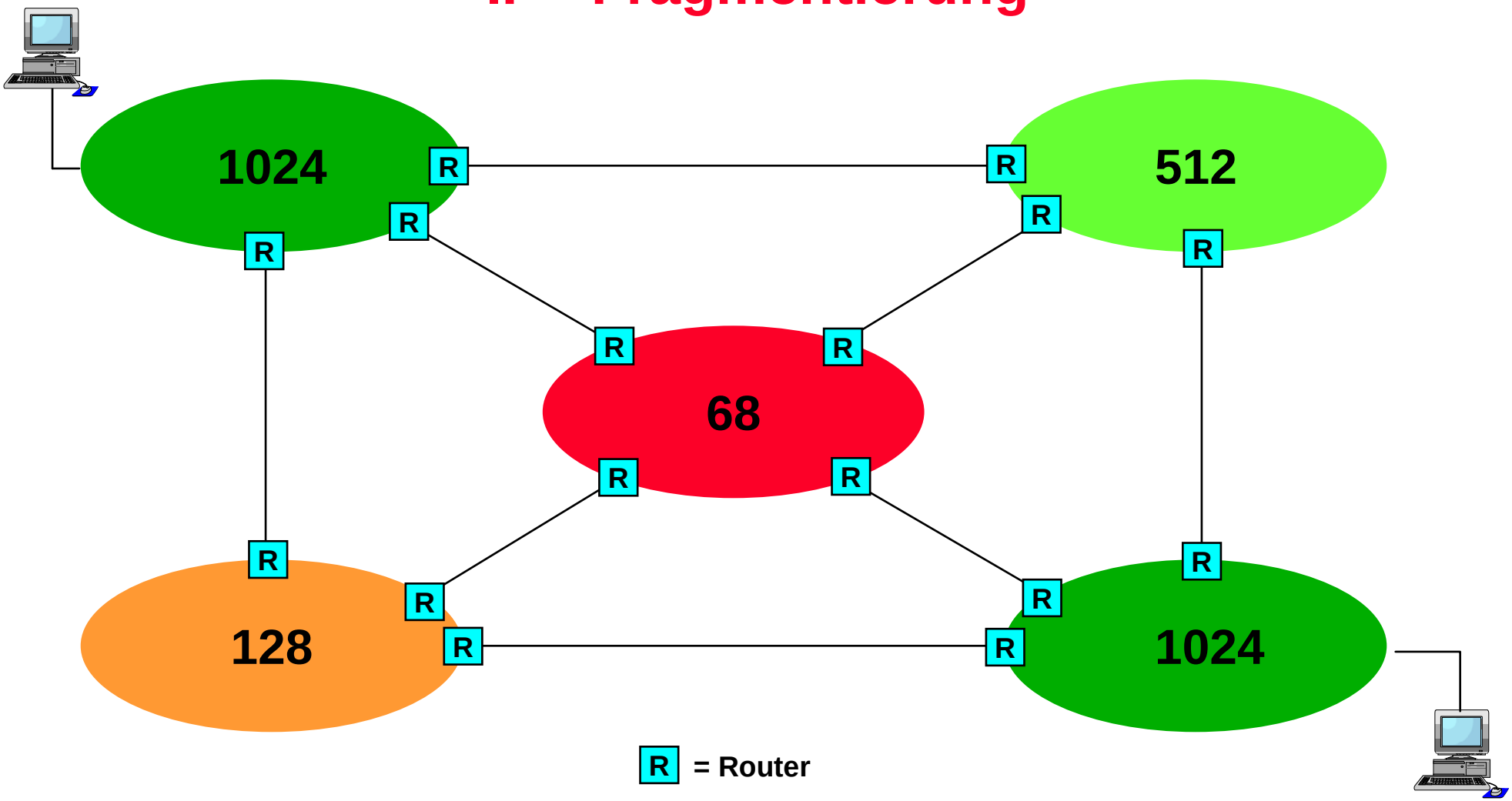
IP - Fragmentierung



Hinweis: Fragment-Offset hat 8 Byte als Einheit!

Stand: 12.02.2006

IP - Fragmentierung



MERKE:

Der Zusammenbau (Reassemblierung) fragmentierter Pakete erfolgt nur beim Empfänger (Endgerät) !

IP - Lebenszeit

- **Problem**
  - ➔ beim Routen (durch vermaschte Netze), können Datagrame/ Fragmente ziellos und unendlich lange kreisen
    (z.B. falsche Routingtabelle)

*Konsequenz: **Ressourcen werden vergeudet***

IP - Lebenszeit

- **Lösung**
  - ➔ TTL-Feld (**T**ime **T**o **L**ive)
    - Reduzierung des Wertes in jedem Router
    - Bei Erreichen des Wertes “0”, wird Paket vernichtet (nicht weitergereicht)

Ausgewählte IP-Protokollnummern

01	**ICMP**	**Internet Control Messsage Protocol**
04, 94	IP in IP	capsulation
06	**TCP**	**Transmission Control Protocol**
08	EGP	Exterior Gateway Protocol
09	IGP	any private interior gateway protocol
17	**UDP**	**User Datagram Protocol**
29	ISO-TP4	ISO-Transport-Protocol Class 4
50	**ESP**	**Encapsulating Security Payload (IPsec)**
51	**AH**	**Authentication Header (IPsec)**
88	IGRP	Interior Gateway Routing Protocol (CISCO)

IP - Optionen

- **Optionale Services**
  - ➔ Security (16 Security Level)
  - ➔ Loose Source Routing
  - ➔ Strict Source Routing
  - ➔ Record Route
  - ➔ Stream ID
  - ➔ Internet Timestamp
  - ➔ No Operation
  - ➔ End of Option List

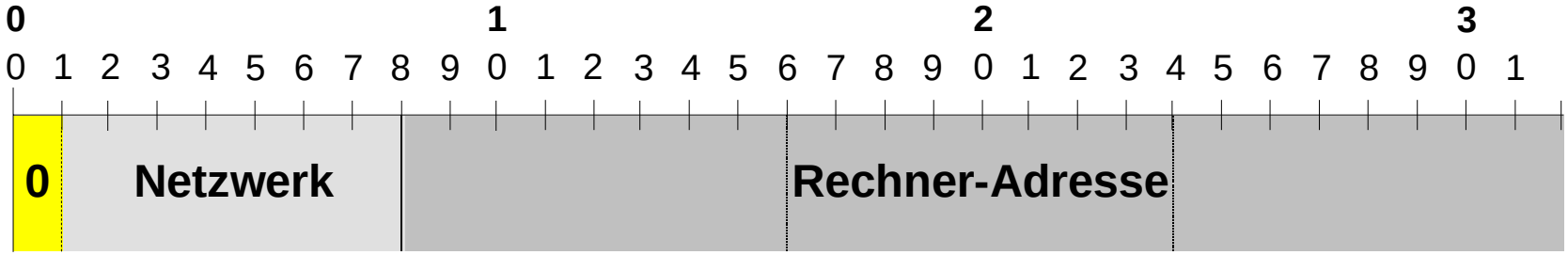
Kapitel 3

IP- Adressierung/ IP-Subnetting

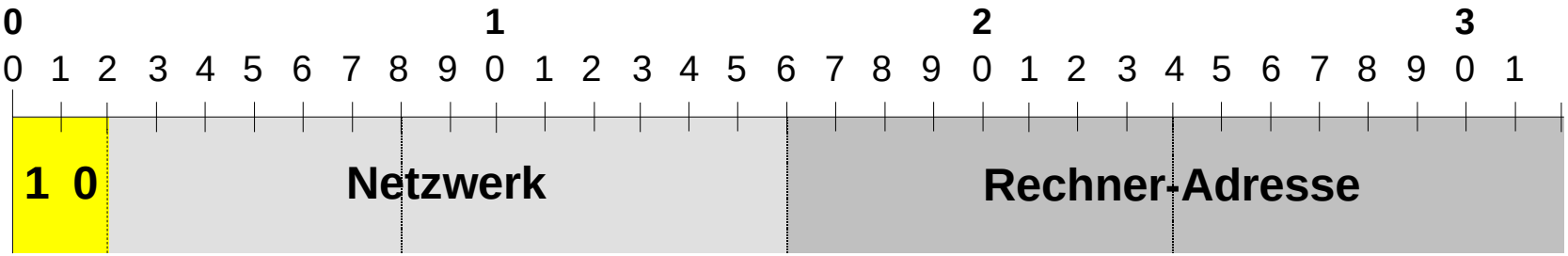
IP Adressen (Aufbau)

198	.	71	.	191	.	1		*dezimal*
1100 0110		0100 0111		1011 1111		0000 0001		*dual*
C6	:	47	:	BF	:	01		*hex*

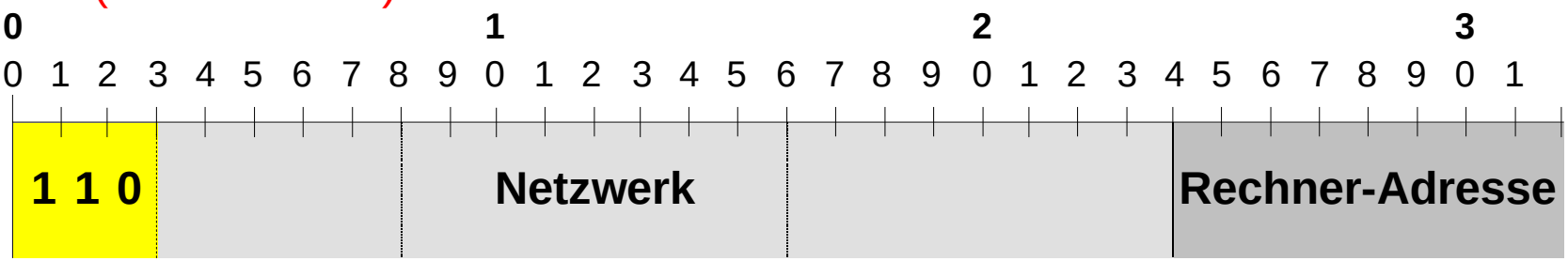
Class A (Wert 0-127)



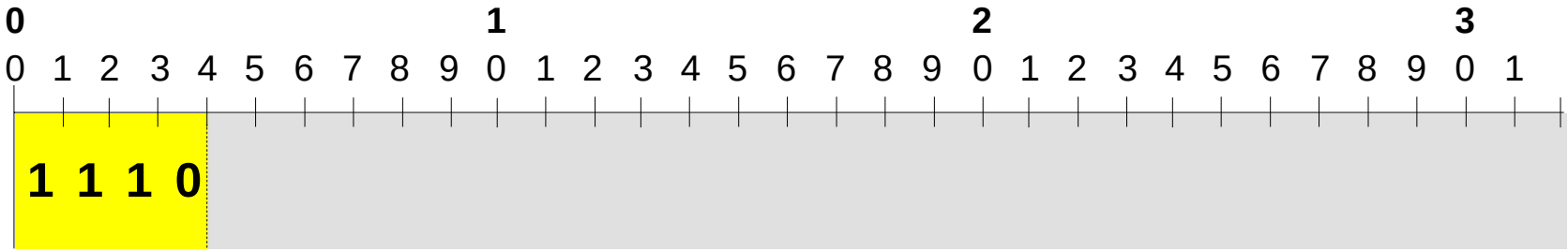
Class B (Wert 128-191)



Class C (Wert 192-223)

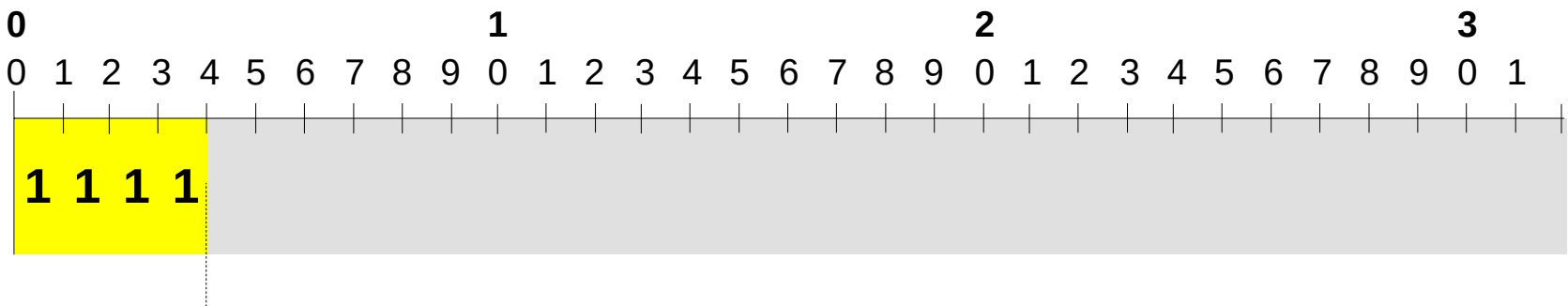


Class D (Wert 224-239)



Multicast-Adressen

Class E (Wert 240-255)



undefiniertes Format

Adress-Klassen sind definiert in RFC 1020 bzw. 1166 (Juli 1990) [Internetnumbers]

Ausgewählte IP-Multicast-Adressen

224.0.0.0	Base Address (reserved)
224.0.0.1	All *Systems* on this subnet
224.0.0.2	All *Routers* on this subnet
224.0.0.5	OSPF - All Routers
224.0.0.9	RIP-2
224.0.0.10	IGRP-Routers
224.0.1.8	SUN NIS (‘Yellow Pages’)
224.0.1.24	microsoft-ds
224.0.2.2	SUN RPC (NFS)

Adressen mit besonderer Bedeutung

127.x.x.x	**Local Host (127.0.0.1)**
255 (im Host-Teil)	**All-One-Broadcast**
255.255.255.255	**All Hosts on *this* net**
0 (im Host-Teil)	**All-Zero-Broadcast** (veraltet)
0 (im Netz-Teil)	**This Net**

Private Adressen
(nach RFC 1918)

10.0.0.0	-	10.255.255.255	ein Class A-Netz
172.16.0.0	-	172.31.255.255	16 Class B-Netze
192.168.0.0	-	192.168.255.255	256 Class C-Netze

vgl. auch: „Special-Use IPv4 Addresses“ (RFC 3330)

z.B.:
169.254.0.0 Link Local (falls DHCP nicht funktioniert)

IP - Adressen / - Subnetz-Masken

IP-Adresse

198	.	71	.	191	.	1
1100 0110		0100 0111		1011 1111		0000 0001
C6	:	47	:	BF	:	01

Subnetz-Maske ***255.255.255.000***

11111111	11111111	11111111	00000000

IP - Subnetting mit erweiterter Subnetz-Maske

		1. Octet	2. Octet
IP	**126.xxx.**xxx.xxx	0111 1110	**.xxxx xxxx**
SN	**255.128.**000.000	1111 1111	.1000 0000

auch: **126.x.x.x/ 9**

IP = IP-Adresse
SN = Subnetz-Maske

Subnetting Varianten

- **RFC 950** (altes/ ursprüngliches Verfahren: **classful routing**)
  - ➔ Unterstes und oberstes Netz (alle Bit auf „0“ bzw. alle Bit auf „1“) können nicht genutzt werden
    - ⇩„0“ = eigenes Subnetz
    - ⇩„1“ = Broadcast-Adresse
  - ➔ 2^n-2 Subnetze

- **RFC 1878** („Modern software will be able to utilize all definable networks“)
  - ➔ Unterstes und oberstes Netz (alle Bit auf „0“ bzw. alle Bit auf „1“) *können* genutzt werden
  - ➔ 2^n Subnetze

IP - Subnetting

Interne Vorgehensweise des Rechners

① **Eigene Adresse** ∧ **Eigene SN-Maske** ——— **Ergebnis A (eigenes Netz)**

② **Ziel Adresse** ∧ ***Eigene*** **SN-Maske** ——— **Ergebnis B (Ziel-Netz)**

Wenn A = B ⇨ Destination in selbem Netz
Wenn A ≠ B ⇨ Destination in anderem Netz

Anmerkung: ∧ = logisches UND

MERKE:

Die Default-Subnetzmaske kann nur in Richtung *mehr Netze* modifiziert / überschrieben werden !

Kapitel 4

IP über serielle Leitungen (SLIP, PPP, PPPoE)

Serial Line IP (SLIP)
RFC 1055

- **keine Fehlererkennung/ -korrektur**
- **nur Punkt-zu-Punkt-Verbindungen**
- **keine Adressinformationen**
  - ➔ **Adresse des Partners muss bekannt sein**
- **keine Protokollidentifikation (“Type-Field”)**
  - ➔ **Multiprotokollübertragung über eine Leitung nicht möglich**
- **Daten werden in “Framing Characters” eingepackt**

END:	**192**	ESC:	219
ESC END:	219 220	ESC ESC:	219 221

- **Kompression für TCP/IP-Header in RFC 1144 definiert**

Point To Point Protocol (PPP)

RFC 1661/ 1662 - STD 51
RFC 2153 (Vendor Extensions)

- **Verbindungsaufbau auf Layer 2 (HDLC-basierend bzw. asynchron)**
- **Fehlerkorrektur**
- **Adressinformationen**
  ➔ multipointfähig (derzeit nicht genutzt)
- **Protokoll-Feld**
  ➔ multiprotokollfähig (auf einer Leitung)
- **feste maximale Paketlänge (1500 Byte)**
- **echte Datenkomprimierung (optional)**
- **Testen der Leitungsqualität (optional)**

Point To Point Protocol (PPP)

Paketaufbau (synchron/ asynchron)

Flag	Address	Control	Protocol	DATA (Information)	FCS	Flag
0111 1110	1111 1111	0000 0011	16 bit	≤ 1500 Byte	16 bit	0111 1110

Point To Point Protocol (PPP)

Ausgewählte Protokoll-Nummern

- 80-21 IP
- 80-27 DECnet
- 80-2B IPX
- 80-3F Netbios
- 80-57 IPv6
- 80-FD Compression Control Protocol

- C0-21 Link Control Protocol
- C0-23 Password Authentication Protocol
- C0-25 Link Quality Report
- C2-25 RSA Authentication Protocol

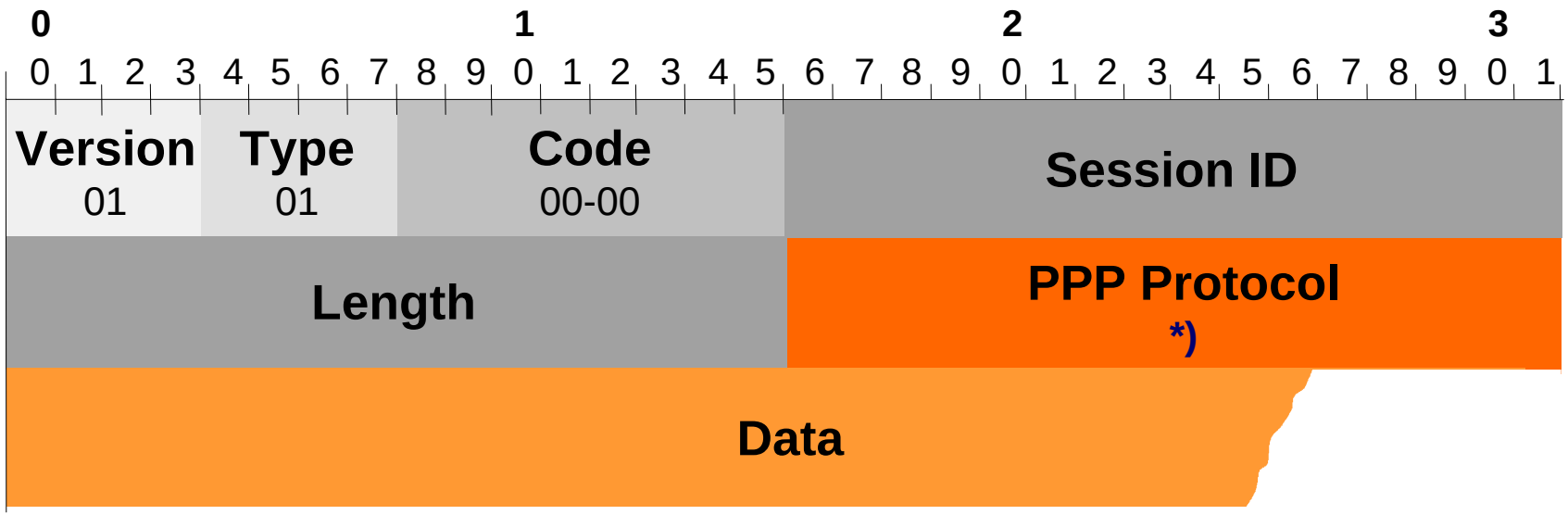
PPP over Ethernet (PPPoE)

RFC 2516

- **PPP-Pakete werden in Ethernet Pakete „eingepackt“**
- **(Ethernet-) Typefields: 88-63 (Discovery Stage), 88-64 (Session Stage)**
- **max. MTU: 1492 (PPPoE-Header + PPP-Protocol-ID)**
- **zweistufiges Konzept:**
  - **➔Server-Suche und Server-Auswahl (Discovery-Stage) „stateless“ bis zum Aufbau einer PPP-Verbindung**
  - **➔Verbindungsaufbau (Session Stage)**

PPP over Ethernet (PPPoE)

Paketaufbau (Session Stage)



*) = C0-21 (Link Control Protocol)

Kapitel 5

IP Next Generation (IPng)
IP Version 6 (IPv6)

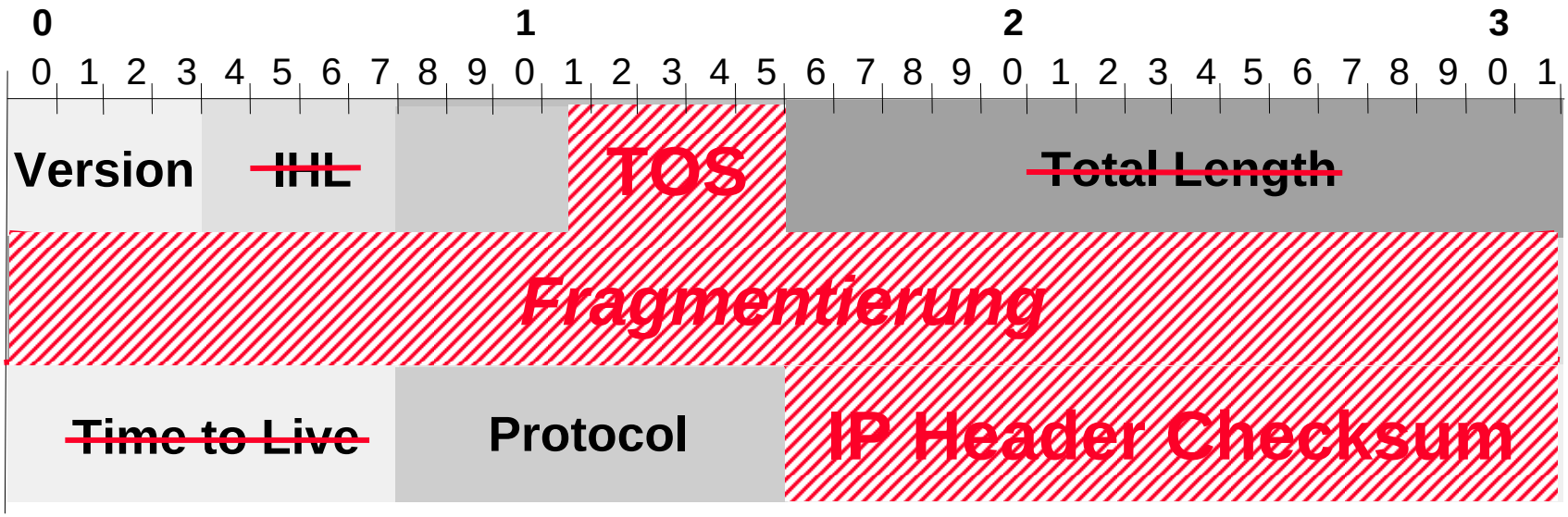
IPv6 (IPng) - Neuer Adressbereich

- **Adressbereich umfasst 128 Bit/ 16 Byte**
  [vgl.: 32 Bit/ 4 Byte bei IP v.4]
  - ➔ **3,4 * 10^{38} Adressen**
  - ➔ theoretisch:
    - ➲ 6,66*10^{23} (genau: 665.570.793.348.866.943.898.599) Adressen/ m^2
    - ➲ 666 Billiarden Adressen/ mm^2
    - ➲ 6,5*10^{28} Adressen pro Mensch
  - ➔ praktisch (worst case):
    - ➲ ca. 1000 Adressen/ m^2

IPv6 (IPng) - Neue Eigenschaften

- **Reduzierung des Header-Overheads durch Weglassen von nicht benötigten Feldern**
- **Erweiterungs-Header (optional)**
- **Fragmentierung nicht mehr in den Routern**
  minimale Transportgröße: 1280 Byte/ „Path MTU Discovery“-Funktion
- **Security-Features (Authentifizierung, Verschlüsselung)**
- **Priorisierung/ Realtime-Fähigkeiten („Traffic Class“/ “Flow Label”)**
- **Nutzdatenanzeige (“Payloadlength”)**
- **“Jumbo-Payload”- Feld (> 65535 Byte)**
- **automatische Systemkonfiguration („Neighbor Discovery“)**
- **Mobile IP**

Veränderungen im IPv6-Header (zu IPv4)

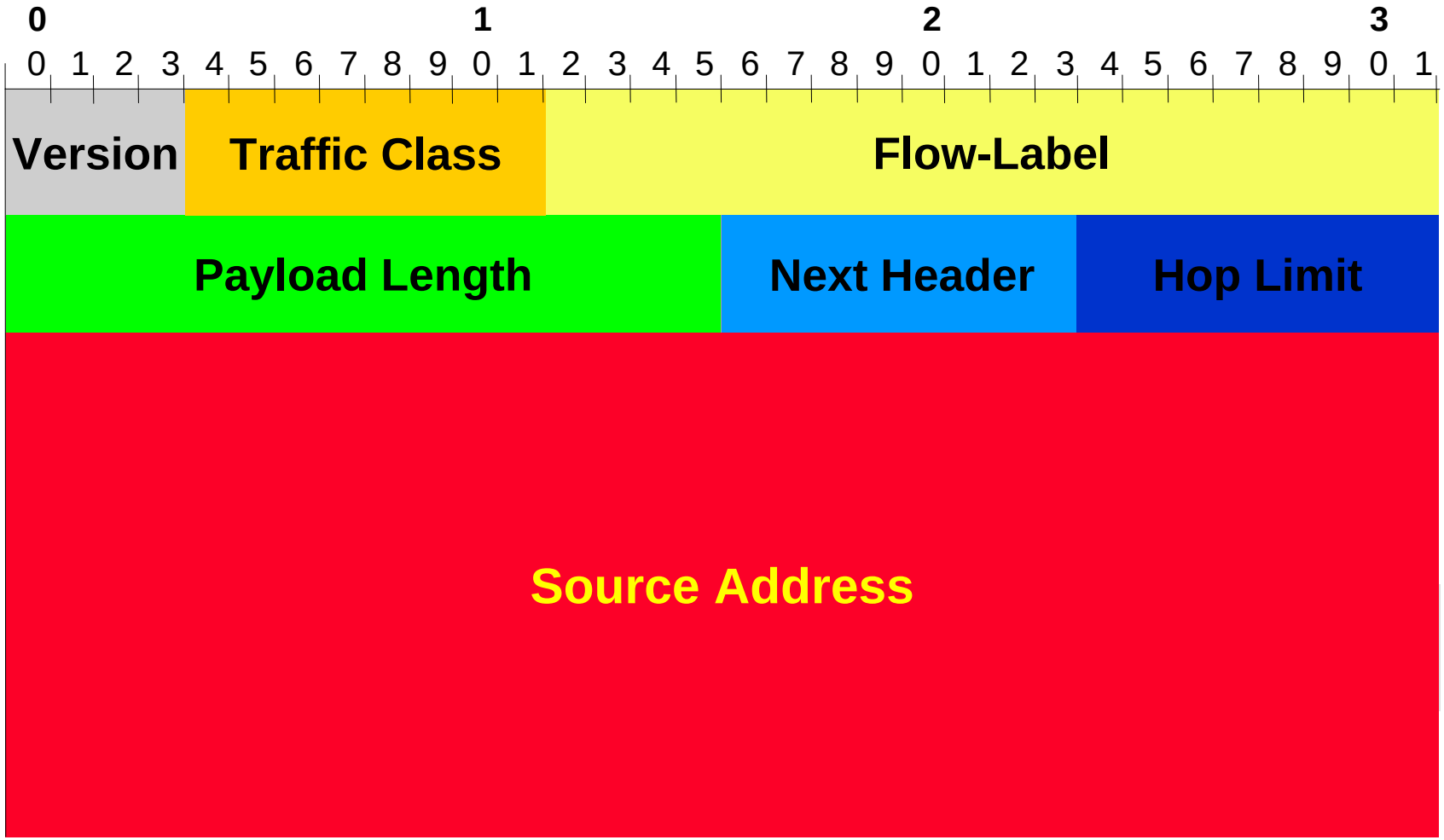


Feld entfällt ersatzlos

Feld bekommt anderen Namen/ Bedeutung

IPv6 Basis Header

(Ausschnitt - ohne „Destination Address“)



IPv6 - Erweiterungs-Header

- Routing Header **(Source Route)** - **Next Header = 43**
- Fragmentation Header **(nur Host)** - **Next Header = 44**
- Authentication Header - **Next Header = 51**
- ESP-Header - **Next Header = 50**

IP Standard-Paket

IP Paket mit verschiedenen Headern

IPv6 - Adressschema und Adressarten

- Präfix (3 Bit)
- öffentlicher Bereich (45 Bit)
- lokaler Bereich (80 Bit)

- ‘Anycast Address’ (“mehrfache” Adresse)
- Multicast Adressen
  (keine Broadcast Adressen mehr)

IPv6 Adress-Aufteilung

FP Format Prefix (001)

TLA Top Level Aggregator (Public Transport Topology)

NLA Next Level Aggregator (Provider)

SLA Site Level Aggregator (Subnet)

Local (inkl. Interface [48 Bit])

IPv6 - RFCs

- RFC 1881 Address Allocation Management
- RFC 1883 Specification (➔ RFC 2460 - DRAFT)
- RFC 1884 Addressing (➔ RFC 2373)
- RFC 1887 Address Allocation
- RFC 1897 Testing Address Allocation (➔ RFC 2471)
- RFC 1825 Security Architecture (➔ RFC 2401)
- RFC 1826 IP Authentication Header (➔ RFC 2402)
- RFC 1827 IP Encapsulation Security Payload (➔ RFC 2407)
- RFC 1828 IP Authentication Using Keyed MD5
- RFC 1829 The ESP DES-CBC Transform

- RFC 2401 - 2411: IPsec

Kapitel 6

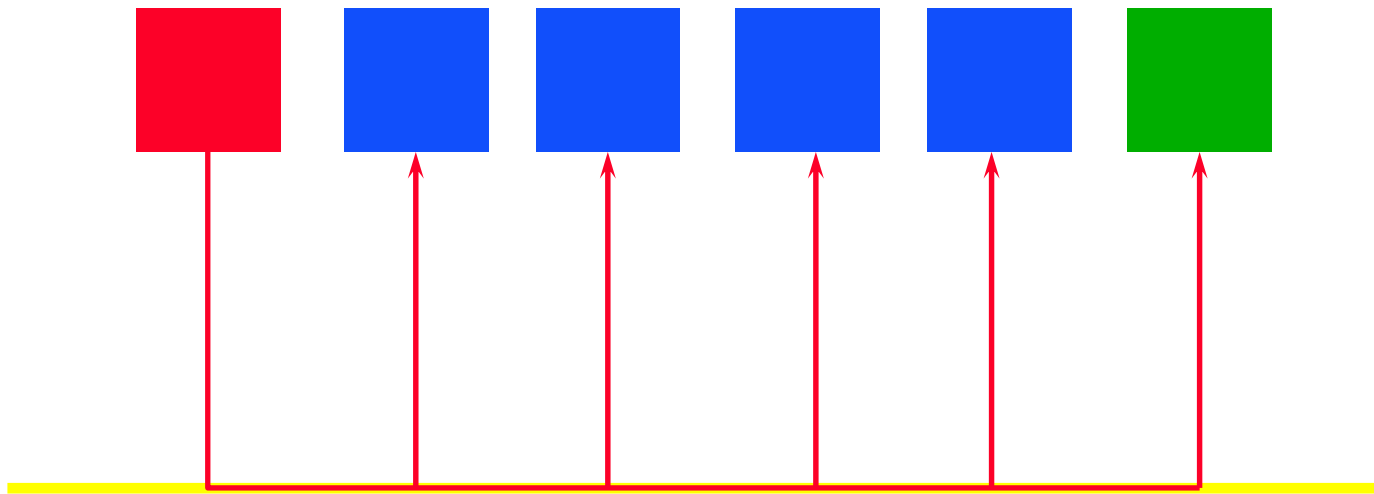
Address Resolution Protocol (ARP)

Adress Resolution Protocol (ARP)

RFC 826 - STD 37

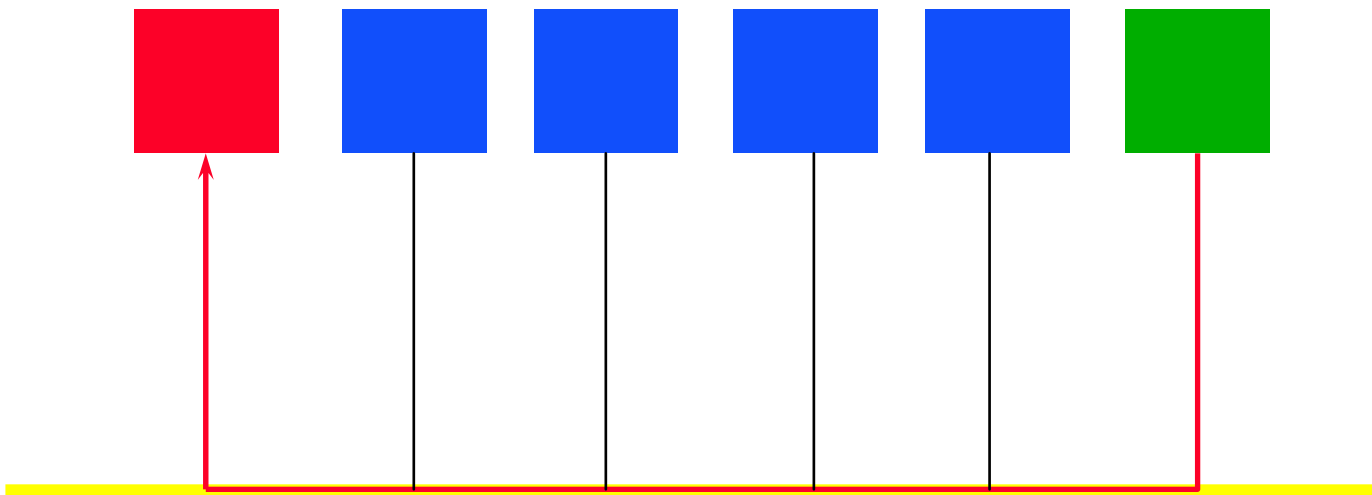
- **setzt auf dem Data Link Layer (Ethernet, TR etc.) auf**
- **nutzt (Ethernet-) Typefield: 08-06**
- **besitzt keine offizielle Definition (bei IEEE) in 802.2 (DSAP/SSAP)**
- **ist ein Datagram-Service**
- **Aufgabe: Zuordnung von Ebene 3 (IP-) Adressen zu Ebene 2 (physikalische) Adressen**

ARP - Request



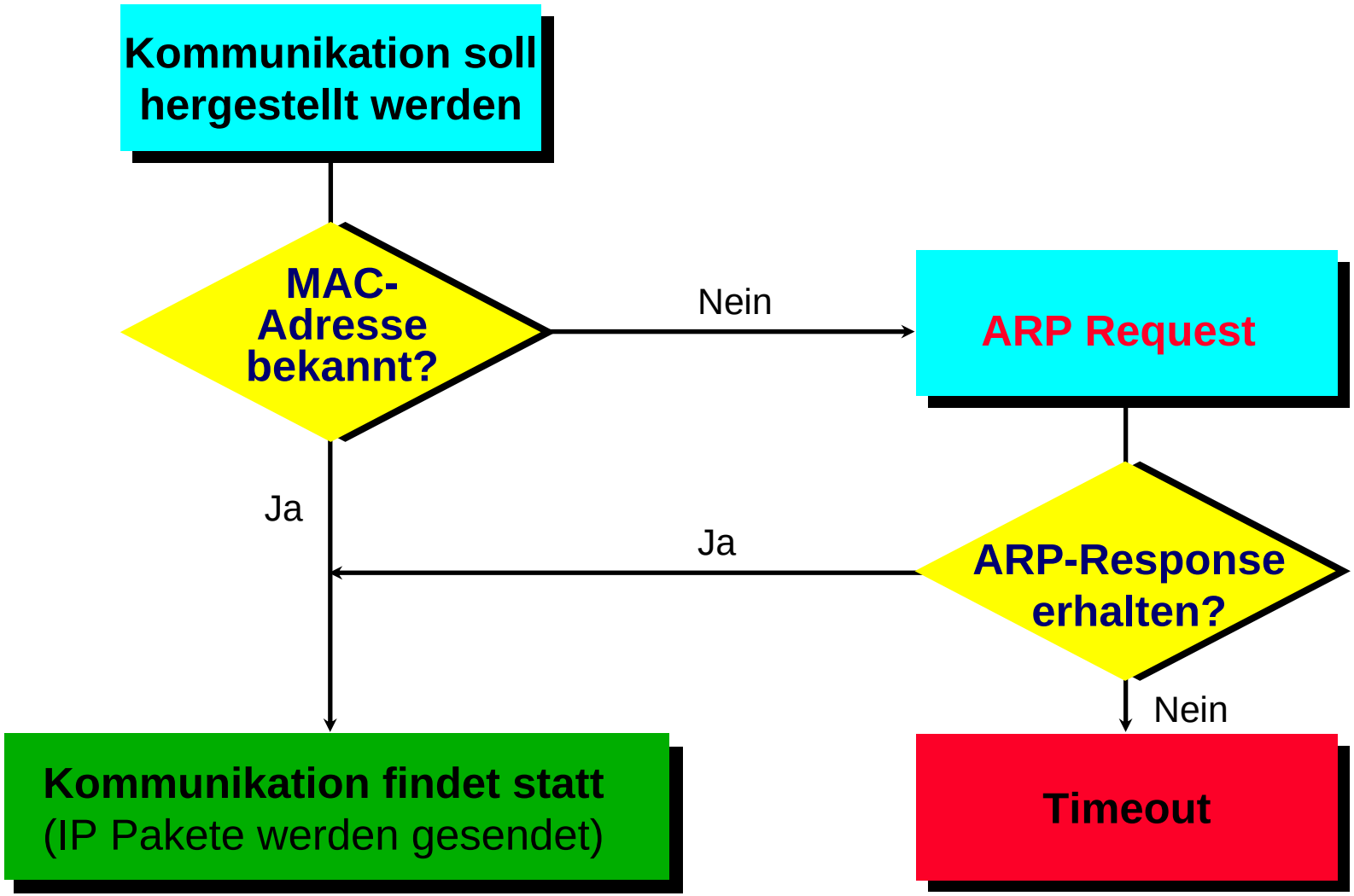
Broadcast: “Wer kennt die Ebene 2 Adresse von GRÜN?”

ARP - Response (1)

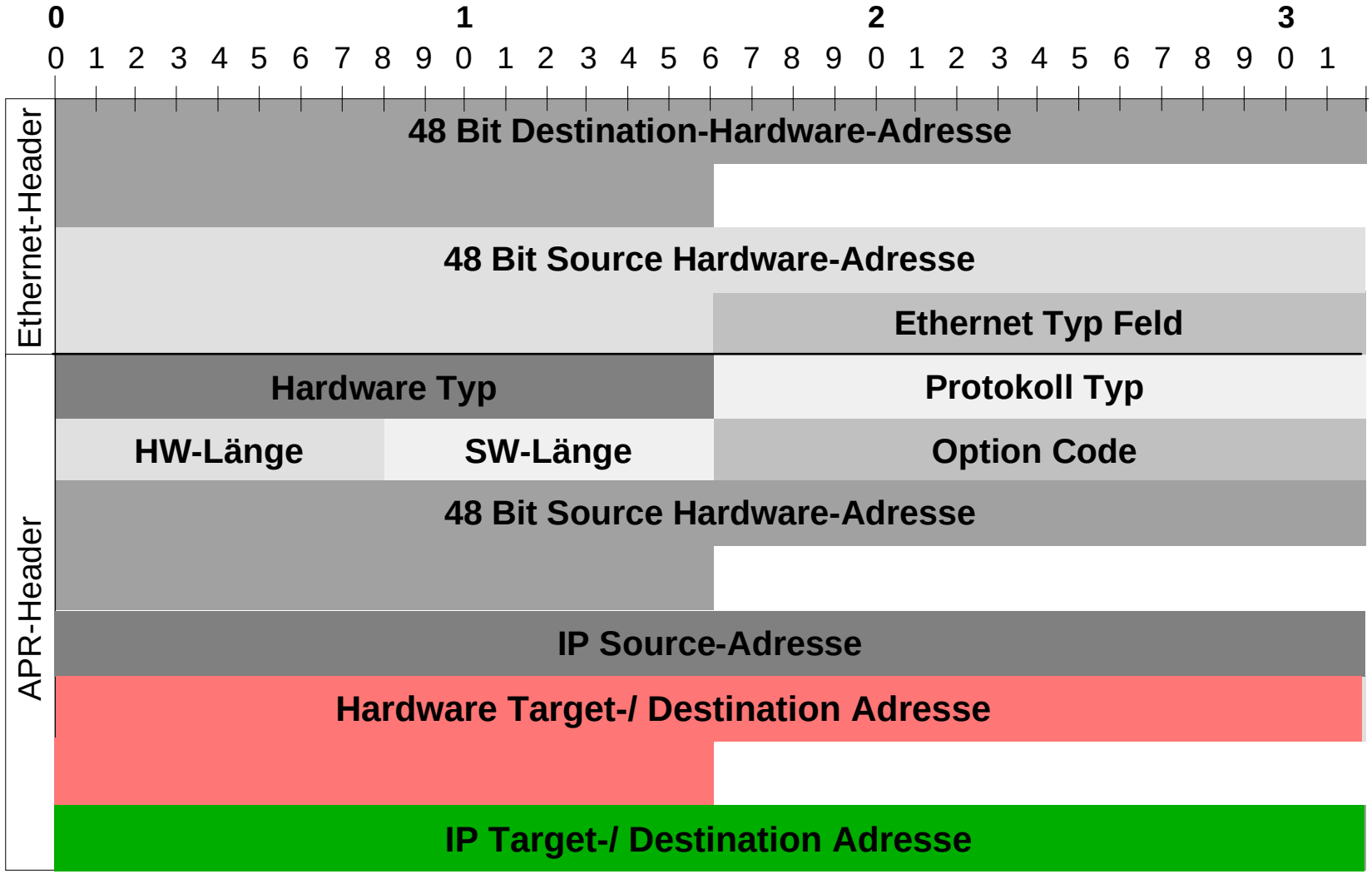


Gerichtete Antwort (Unicast):
“Hier ist die gesuchte (**meine**) Ebene 2 Adresse”

ARP - Ablaufdiagramm



ARP - Datenformat



ARP - Hardware Typ

Netztyp	Bezeichnung
1	Ethernet (10 Mbit/s)
2	Experimental Ethernet (3Mbit)
3	Amateur Radio
4	Proteon Token Ring
5	Chaos Net
6	IEEE 802 Networks
7	ARCnet

ARP - Protokoll Typ

(vgl. Ethernet “Type-Field”)

Wert (hexadezimal)	Bezeichnung
0600	**XNS**
0800	**IP**
0806	**ARP**

ARP - Felder

- **Hardware-Länge**
  Definiert Länge der Hardware-Adresse (Ethernet = 6 Byte)

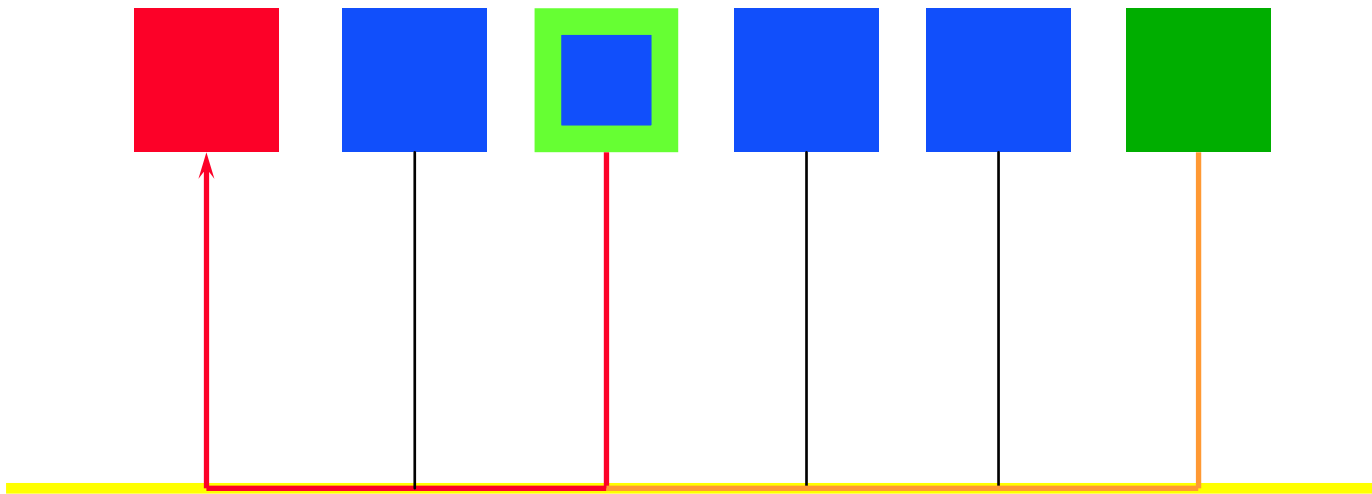
- **Software-Länge**
  Definiert Länge der Protokoll-Adresse (IP = 4 Byte)

- **Option Code**
  1 = ARP Request
  2 = ARP Reply

ARP - Adressfelder

- **Hardware-Source-Adresse**
  Hardware-Adresse des Senders
- **Protokoll-(IP)-Source-Adresse**
  IP-Adresse des Senders
- **Hardware-Target-/Destination-Adresse**
  Hardware-Adresse des Empfängers/ Ziels
- **Protokoll-(IP)-Target-/Destination-Adresse**
  IP-Adresse des Empfängers/ Ziels

ARP - Response (2)



❶ Unicast: “Hier ist die gesuchte Ebene 2 Adresse”

❷ Unicast: “Hier ist die gesuchte (**meine**) Ebene 2 Adresse”

ARP - Befehl

- **arp -a**
  ARP-Cache anzeigen

- **arp -s <IP-Adr.> <HW-Adr.>**
  Zuordnung IP-Adr./HW-Adresse

- **arp -s <IP-Adr.> <HW-Adr.> PUB**
  zugeordnete HW-Adresse wird als ARP-Response ausgegeben

- **arp -d <IP-Adr.>**
  Eintrag wird gelöscht

Gratuitous ARP

- **Host schickt eine Anfrage mit eigener IP-Adresse (als Target-Adresse) *unaufgefordert* ins Netz**
  - ➔ Feststellung ob eigene IP-Adresse mehrfach vorhanden ist
  - ➔ Update der ARP-Tabellen in den anderen Rechnern

Reverse Address Resolution Protocol (RARP)

Reverse Address Resolution Protocol (RARP)

RFC 903 - STD 38

- **setzt auf dem Data-Link-Layer (Ethernet/ TR) auf**
- **nutzt (Ethernet-) Type-Field: 80-35**
- **besitzt keine 802.2-Definition (DSSAP/ SSAP)**
- **Zuordnung von HW-Adressen (Ebene 2) zu IP-Adressen (Ebene 3)**
- **Aufbau wie ARP-Paket**

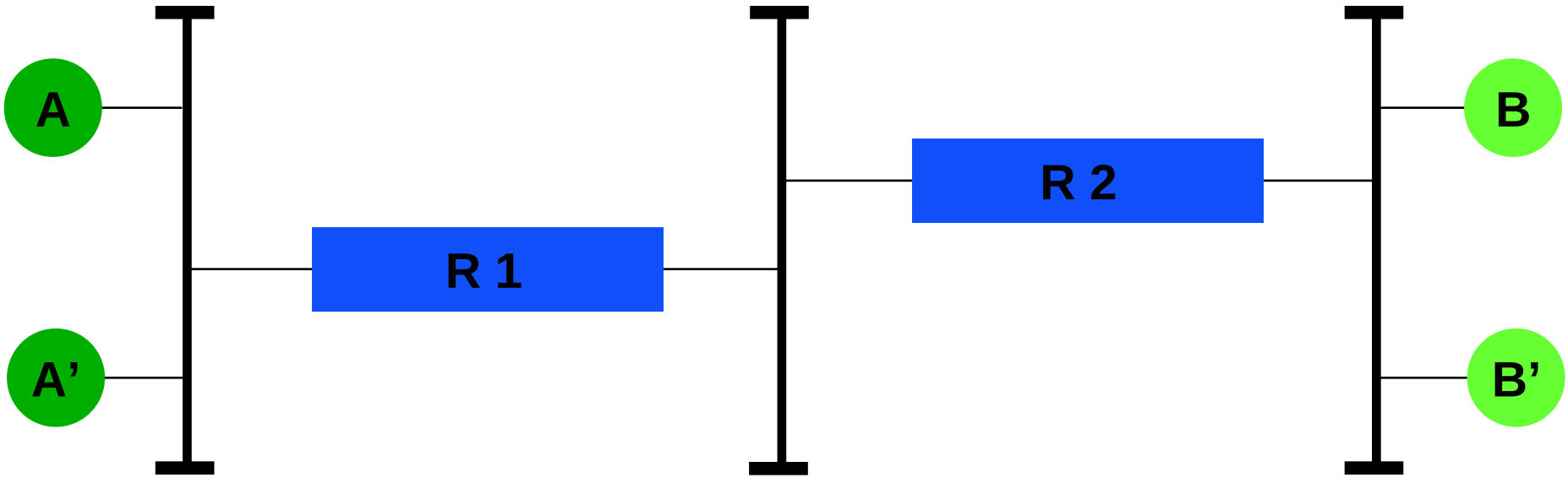
Ausnahme: **Option Code**
3 = RARP Request
4 = RARP Reply

♦ **Funktionalität heute i.a. durch BootP abgedeckt**

Kapitel 7

IP - Routing

Routing auf Backbone

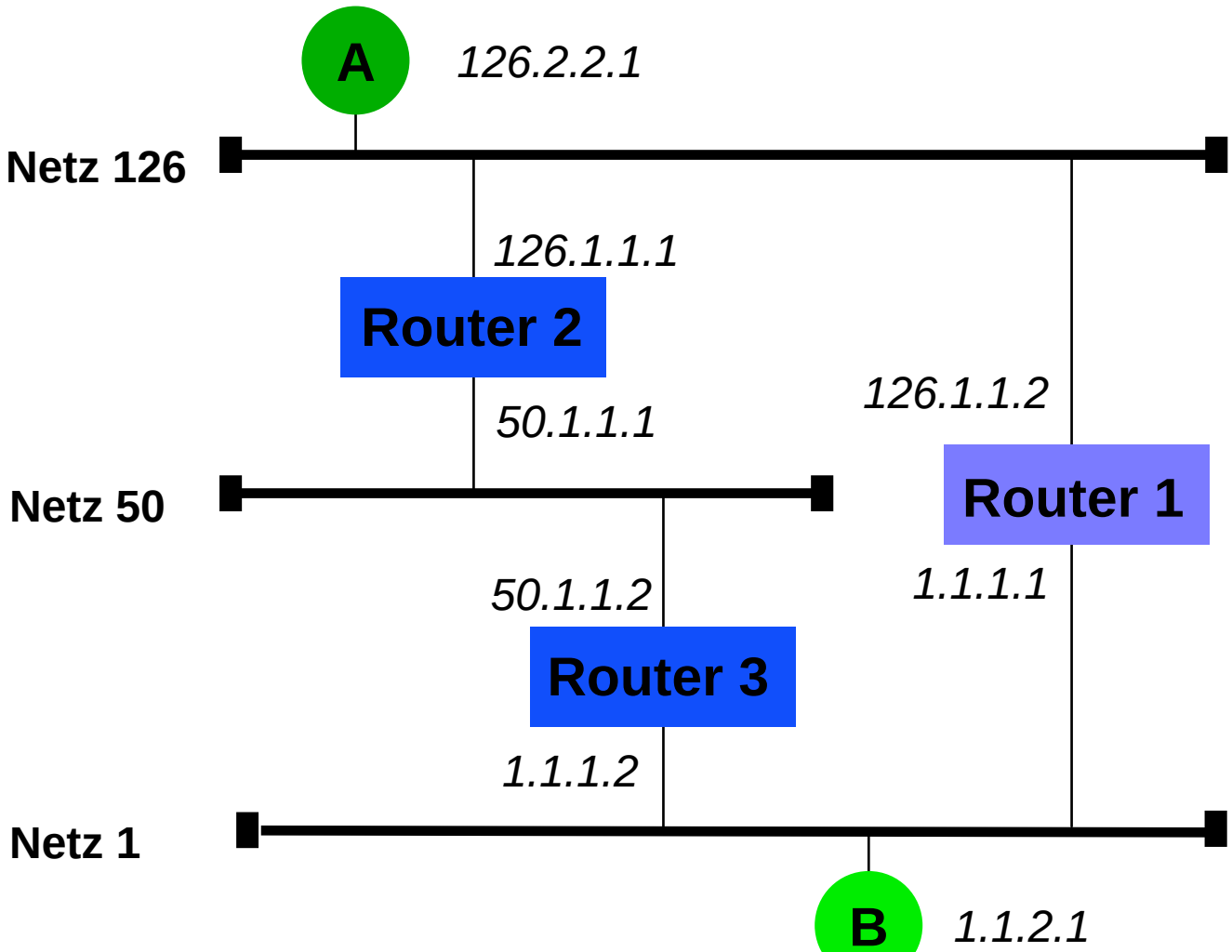


Zeitpunkt	*Data Link Layer*		*Network Layer*	
	Sender	Empfänger	Sender	Empfänger
t_1	A	R1	A	B
t_2	R1	R2	A	B
t_3	R2	B	A	B

MERKE:

Beim Einsatz von Routern geht die Transparenz auf Layer 2 vollständig verloren !

Routing in vermaschtem Netz



Routing - Verfahren

- statisches Routing
- dynamisches Routing
- default Routing

IP-Optionen

- Source-Route
  - ➔ Loose Source-Route
  - ➔ Strict Source-Route
- Record Route

MERKE:

Ein IP-Router hat keine Geräte-Adresse, sondern nur seine Schnittstellen(karten) zu den angeschlossenen Netzen!

und

Diese muss aus demselben Adressbereich stammen wie die Adressen der angeschlossenen, zu routenden Rechner !

Proxy ARP

Proxy ARP

- Kein Protokoll, sondern Programm (Prozess) auf Router
- Leitet ARP-Anfragen an Routing-Table weiter
- Erspart (temporär) Routing-Einträge auf Hosts
- Belastet den Router durch notwendige zusätzliche ARP-Bearbeitung

Kapitel 8

Internet Control Message Protocol (ICMP)

Internet Control Message Protocol (ICMP)

RFC 792 - STD 5

- **setzt direkt auf dem Internet Protokoll (IP) auf**
- **nutzt IP-Protokoll-Nr.: 01**
- **es dient dem Informationsaustausch der Endgeräte über den aktuellen Status der Ebene 3 (IP)**
- **es gibt Error-Meldungen und Info-Meldungen.**
  - **Error-Pakete beinhalten, neben der Fehlermeldung, auch immer den Header und die ersten 64 Bit des den Fehler verursachenden Paketes.**
  - **Info-Meldungen basieren auf einem Request-/ Response-Verfahren**

ICMP- Fehlermeldungen

- Destination Unreachable
- Redirect Message
- Source Quench
- Time Exceeded
- Parameter Problem

ICMP - Destination Unreachable-Meldung (Auswahl)

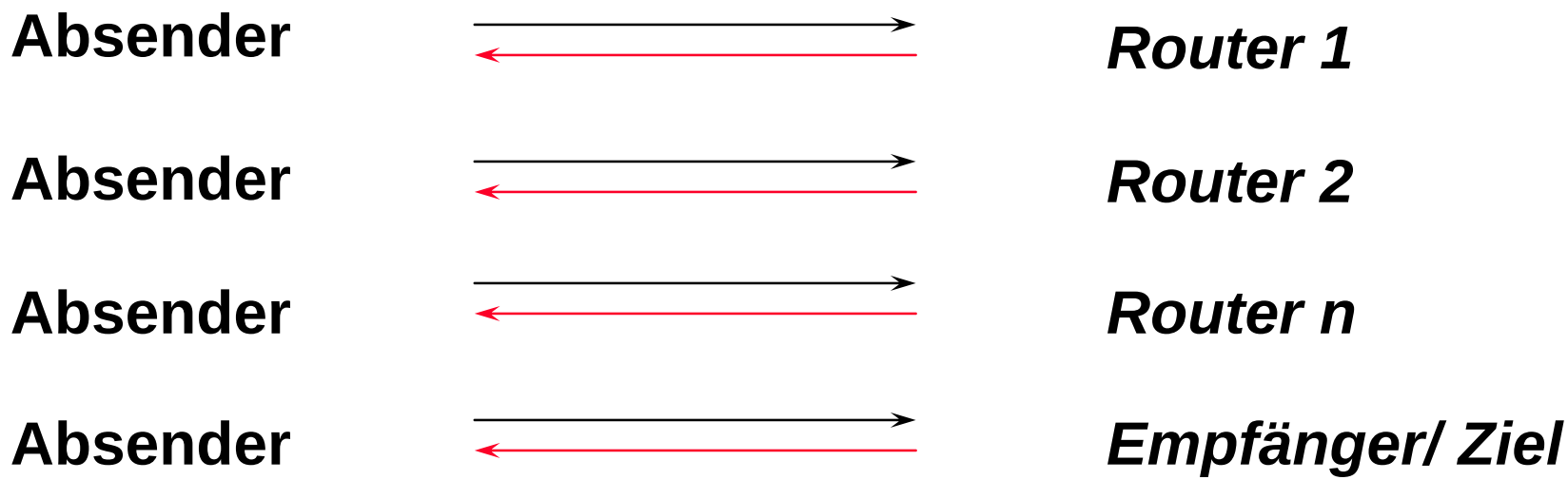
- **Net/ Host Unreachable** *Router*
- **Communication with Destination Network/ Host is Administratively Prohibited** *Router*
- **Destination Network/ Host Unreachable for Type of Service** *Router*
- **Fragmentation Needed and DF Set** *Router*
- **Source-Route Failed** *Router*

- **Protocol/ Port Unreachable** *Host*

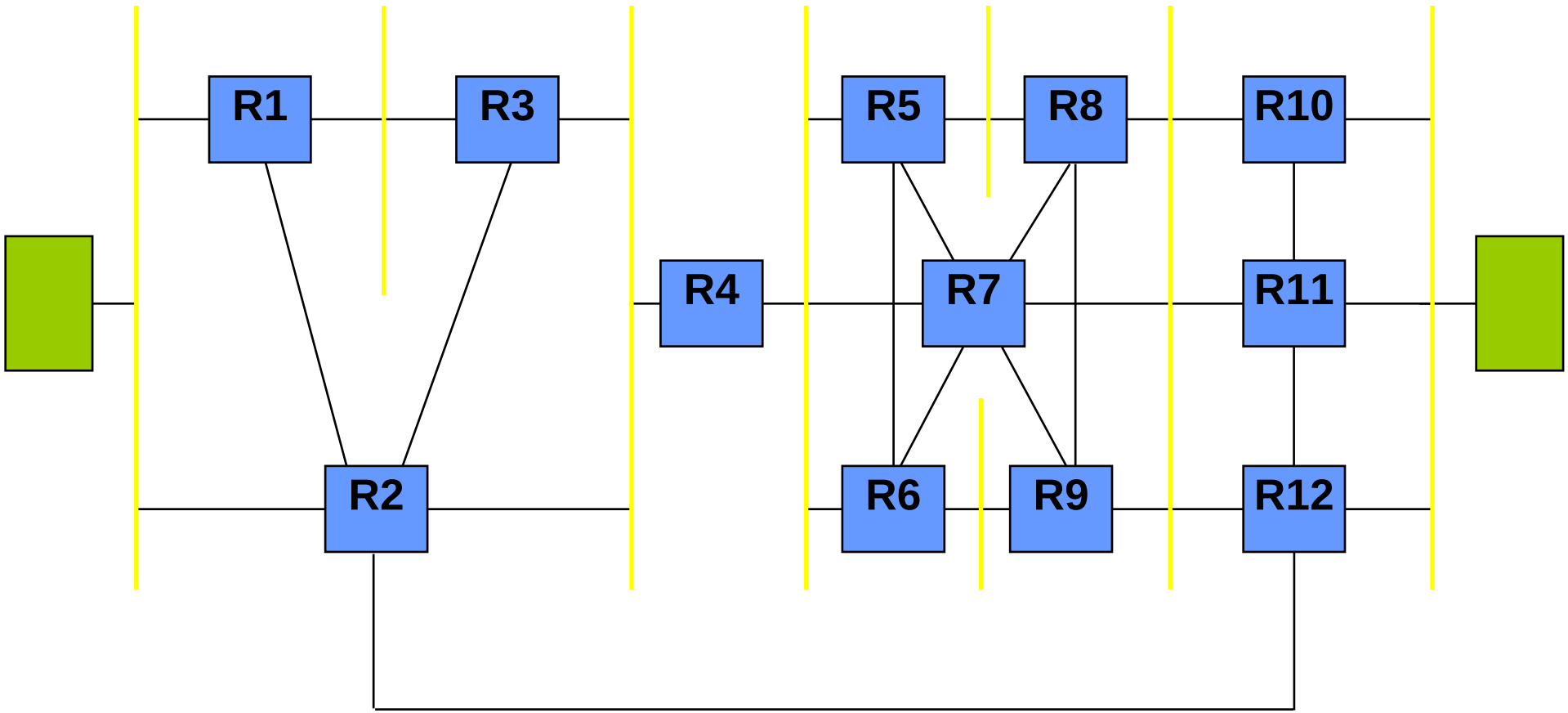
ICMP- Info-Meldungen

- Echo
- Information
- Timestamp
- Address Mask
- Trace Route

IP / ICMP “Trace-Route”
„Klassische“ Methode



Trace Route in dynamischen Netzwerken



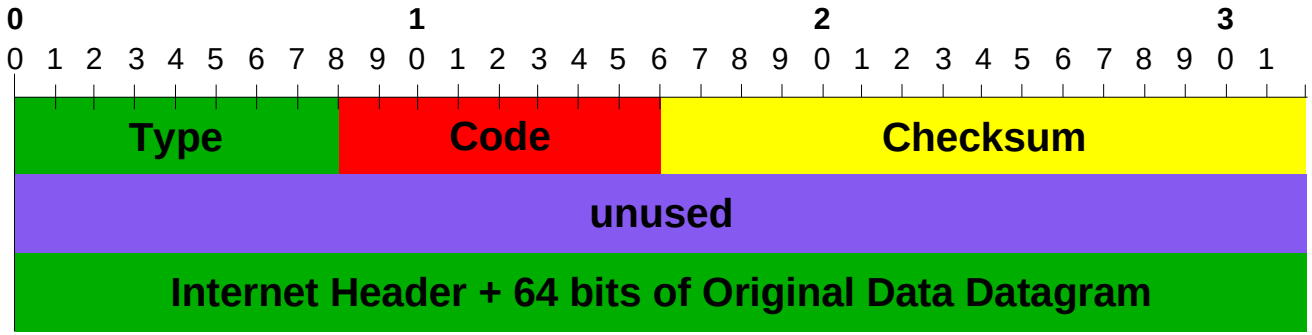
IP/ ICMP “Trace-Route”
Neue Methode

⟶ IP-Paket “Trace Route” (OHC wird incrementiert)

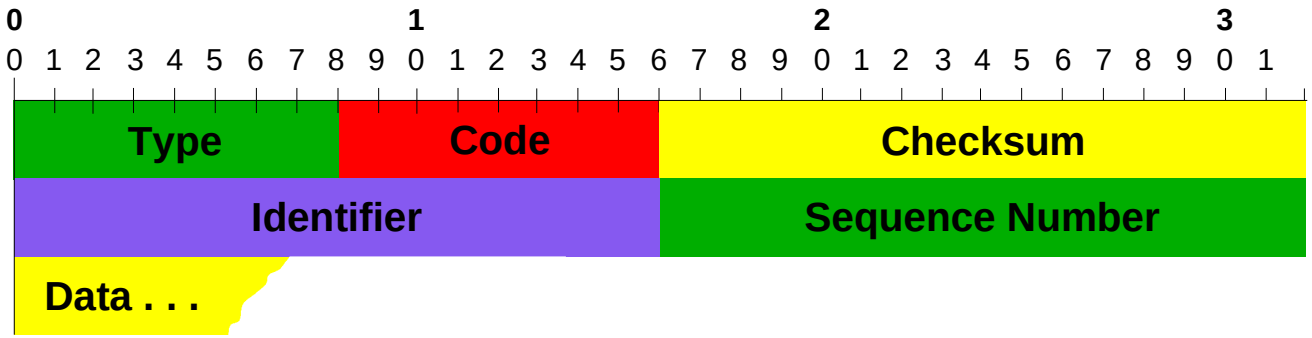
⟵ ICMP-Message “Trace Route” (1, 2, ..., n) (RHC wird incrementiert)

OHC = Outbound Hop Count

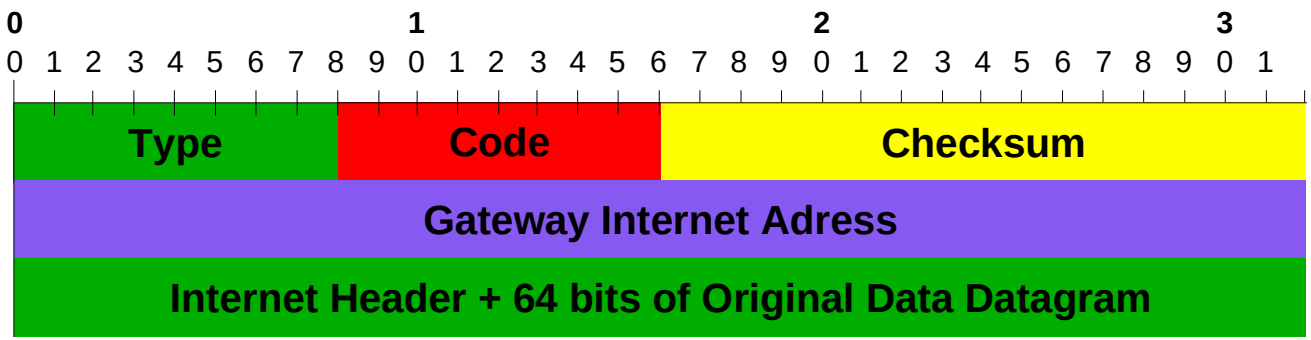
RHC = Return Hop Count



Destination Unreachable Message



Echo or Echo Reply Message *(“Ping”)*



Redirect Message

ICMP - Messages
Type Numbers (Auswahl)

00	Echo Reply
02	Destination Unreachable
04	Source Quench
05	Redirect
08	Echo Request
11	Time Exceed
12	Parameter Problem
30	Traceroute

37 - 255 “reserved”

Kapitel 9

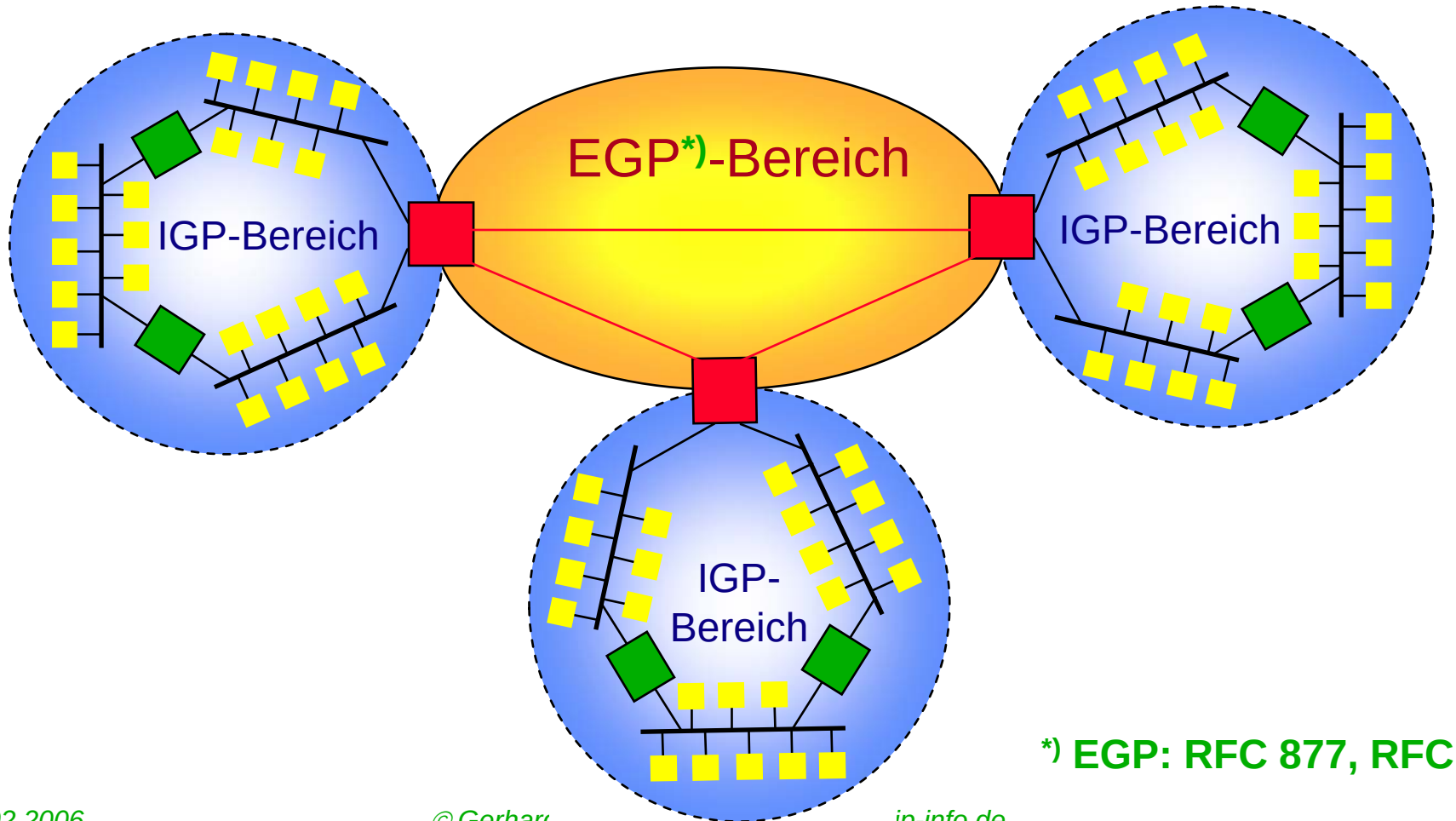
Routing Protokolle

Aufgabe der Routing Protokolle

"The goal of a routing protocol is very simple:
It is to supply the information that is needed to do routing."

C. Hedrick: RFC 1058 - Routing Information Protocol,
Juni 1988 , Seite 3

Arten von Routing Protokollen



*) EGP: RFC 877, RFC 904

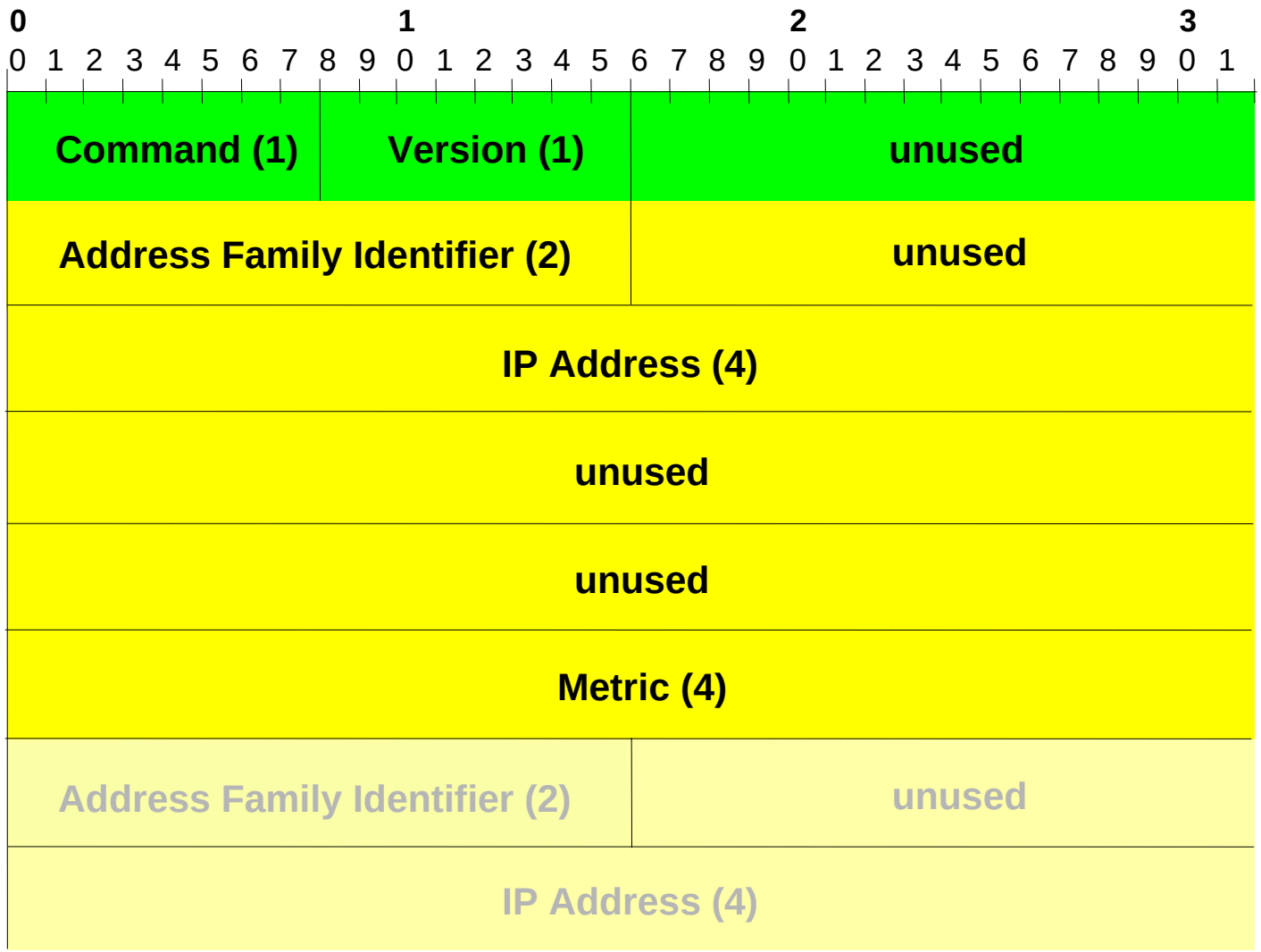
Routing Information Protocol (RIP)

Routing Information Protocol (RIP)

RFC 1058 - STD 34

- **kein MIL-Standard**
- **setzt auf dem Datagram-Transport-Dienst des UDP auf**
- **nutzt UDP-Port 520**
- **stammt ursprünglich aus der XNS-Protokoll-Familie**
- **ist Bestandteil des BSD 4.3-UNIX (routed-Daemon)**
- **gehört zu der Familie der Distance-Vektor-Protokolle** (Bellman-Ford-Algorithmus)

RIP - Paketaufbau



RIP - Paketaufbau

Bedeutung der Felder

- Command Feld: 1 = Request
2 = Response
- Address Family Identifier: 2 = IP
- IP-Adress: Ziel-Netz bzw. -Rechner
- Metric (=Hops): Entfernung bis Ziel
(Länge: 4 Bit = max. 15 Hops)

Länge des Paketes: max. 512 Byte
(~ 25 Info-Felder)

RIP
Routing Tabelle/ Routing Updates

❶ **Regelmäßige Routing-Updates** (alle 30 sec)

❷ **Überprüfen**, ob

neue “Metric” < alte “Metric”

⇒ ***JA:*** Wert übernehmen - Update des Eintrags beendet

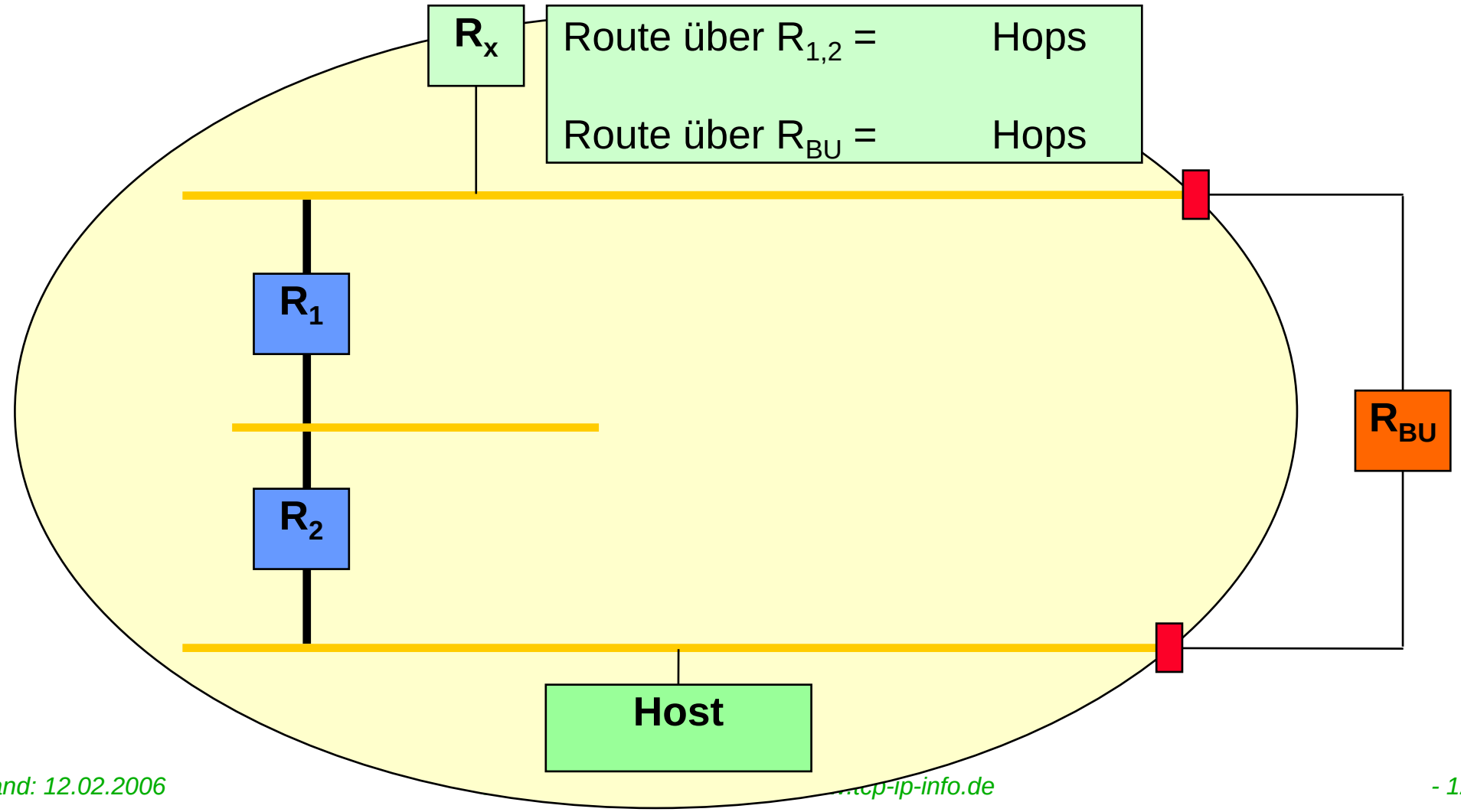
⇒ ***NEIN:*** Wert beibehalten und

❸ **Überprüfen**, ob Routing-Update von dem Router kam, der den letzten Eintrag erstellt hat

⇒ ***JA:*** Wert ***auf jeden Fall*** übernehmen (auch wenn größer) Update des Eintrags beendet

⇒ ***NEIN:*** Update des Eintrags beendet

RIP (Hopcount)



Split Horizon

- **Verhindert Rückrouten (reverse route)**
  - ❶ Updates, die über eine bestimmte Schnittstelle gesendet werden, berichten nicht über Routen, die über diese Schnittstelle gelernt wurden
  - ❷ Updates, die über eine bestimmte Schnittstelle gesendet werden kennzeichnen jedes über diese Schnittstelle erlernte Netzwerk als nicht erreichbar
    (Split Horizon with poisoned reverse)
  - ➔ **spart Ressourcen**
  - ➔ **verhindert Routing-Schleifen**

Classful Routing
nach RFC 950

- **Es werden keine Subnetzmasken zusammen mit der Ziel-Adresse weiter gemeldet**

  ❶ **Zieladresse befindet sich direkt in dem mit dem Router verbundenen Netzwerk:**

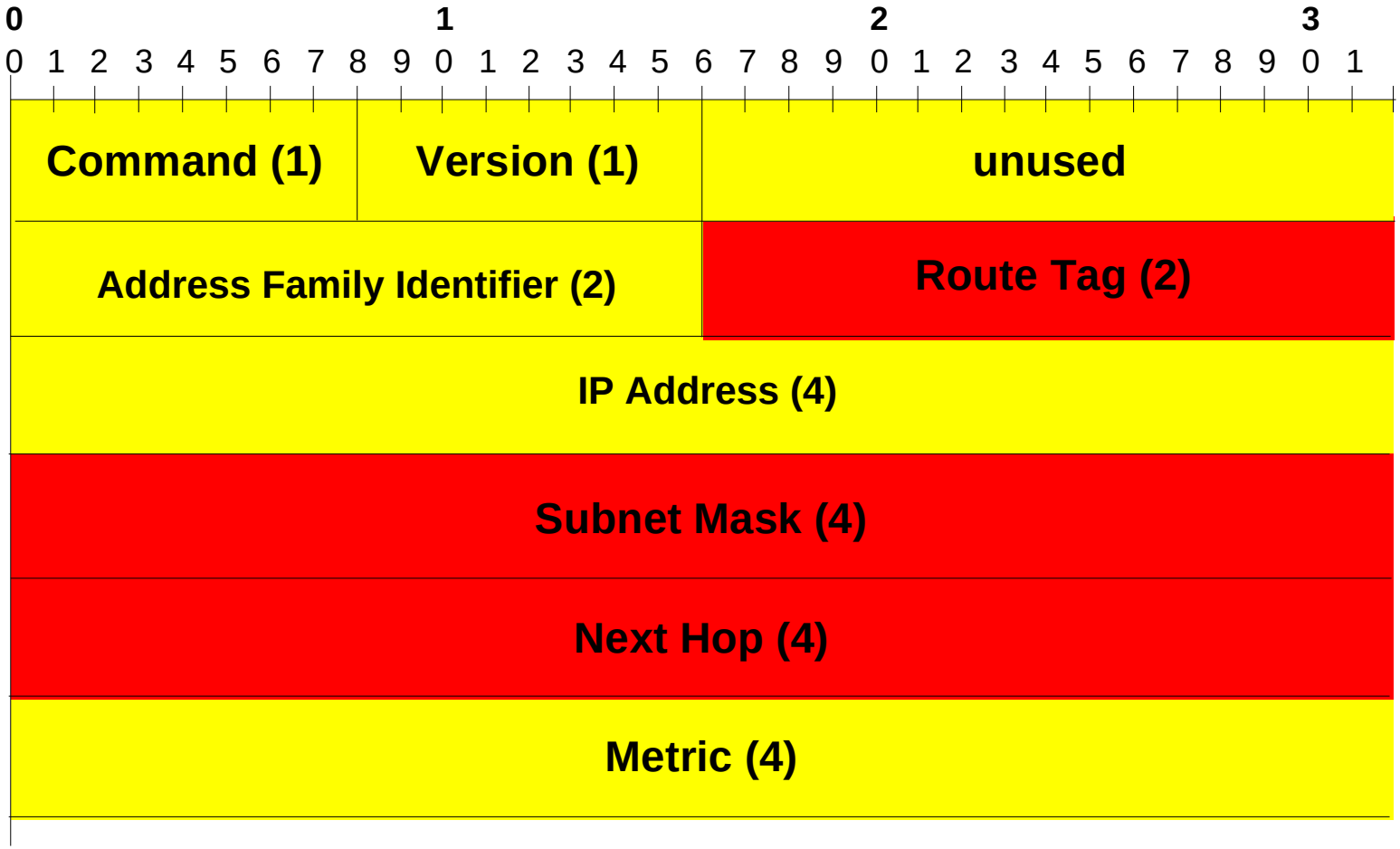
  ➔ **Subnetzmaske der NIC wird verwendet**

  ❷ **Zieladresse befindet sich in „Remote-Netzwerk“:**

  ➔ **Default-Subnetzmaske wird verwendet**

➔**Unterstes und oberstes Subnetz -** alles „0“ **(Hauptnetz-Netzwerknummer)** bzw. alles „1“ **(Broadcast des Hauptnetzes) - können nicht genutzt werden**

RIP-2 (STD 56)
Paketaufbau



= RIP v.1-Feld

= neues Feld (RIP-2)

Open Shortest Path First (OSPF)

Open Shortest Path First (Version 2)
OSPF 2

RFC 2328 - STD 54

- **Erweiterung von OSPF (RFC 1131)**
- **setzt auf IP auf (IP-Protokoll-Nr.: 89)**
- **Interior Gateway Protocol**
- **Link State Protocol**
- **Virtuelle Topologie (Autonomous System = AS)**
  ➔ alle Router haben identische Datenbank
- **Dynamisches Routing Protokoll**

OSPF 2 - Eigenschaften/ Funktionalitäten

- **Routing-Updates nur bei Topologieänderungen**
- **Routing-Updates über IP-Multicasts**
- **Jeder Router berechnet** (s)**einen Baum** (mit sich selbst als Root)
- **Unterschiedliche Routen je nach Type Of Service**
- **Load-Balancing bei Routen mit gleichen “cost”**

OSPF 2
Areas

- **Bildung von “Areas” möglich (Topologie wird verborgen)**
  - ➔ Reduzierung des Routing-Verkehrs
  - ➔ Routing innerhalb der Area wird nur durch Topologie der Area selbst bestimmt
  - ➔ unterschiedliche Topologie-DBs innerhalb eines AS

- **Authentifizierung (“Trusted Router”) innerhalb eines AS durch “Router-Id”**

Kapitel 10

Transmission Control Protocol (TCP)

Transmission Control Protocol (TCP)

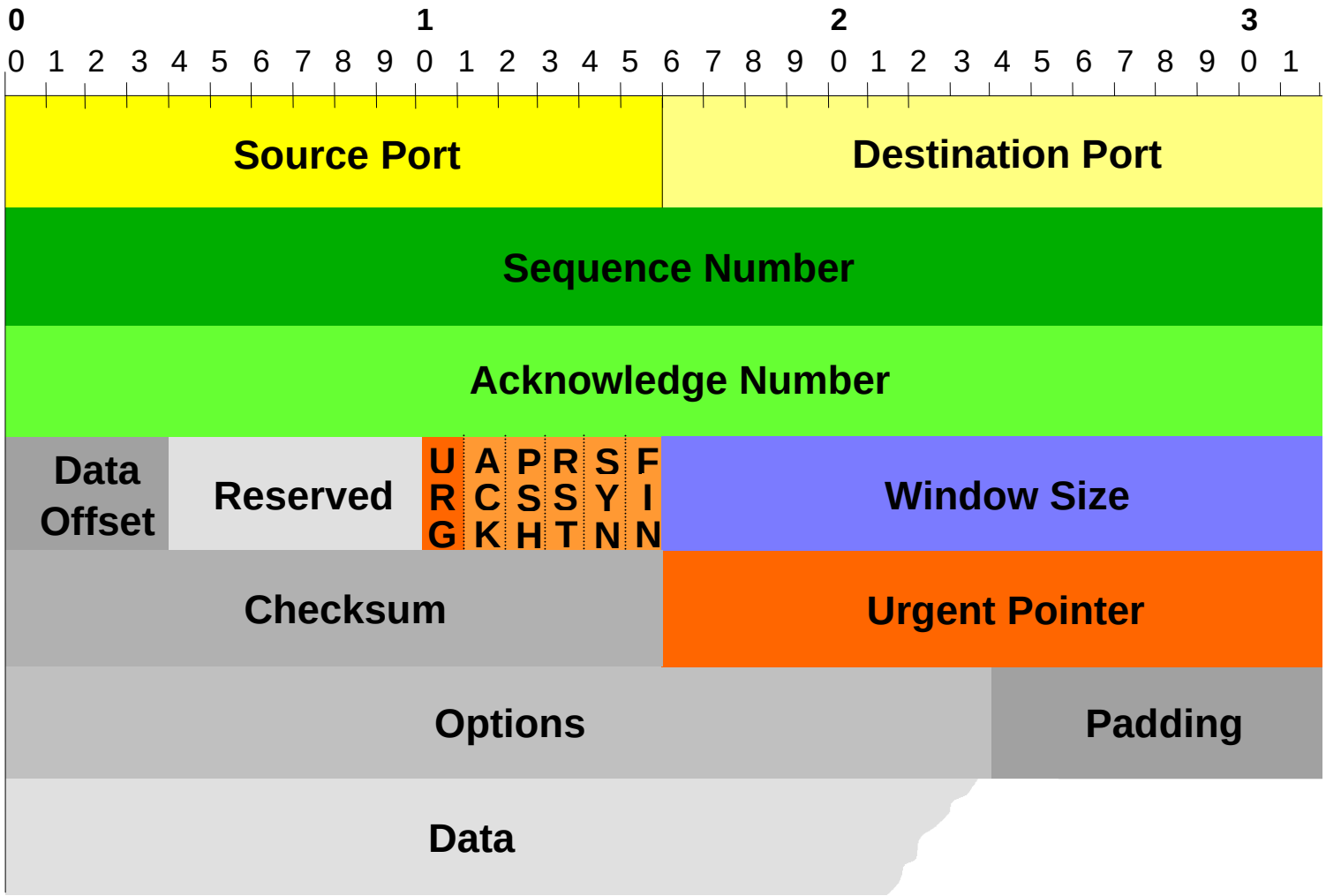
RFC 793 - STD 7 - MIL-Std. 1778

- **setzt direkt auf dem Internet Protokoll (IP) auf**
- **nutzt IP-Protokoll-Nr.: 06**
- **garantiert eine fehlergesicherte, zuverlässige Transport-Verbindung zwischen zwei Rechnersystemen (Ende zu Ende Kontrolle)**

TCP - Eigenschaften

- **Multiplexing**
- **End To End Controle**
- **Verbindungsmanagement** („Three-Way-Handshake“)
- **Flusskontrolle** („Sliding-Window-Mechanism“)
- **Zeitüberwachung**
- **Fehlerbehandlung**

TCP - Header



TCP - Multiplexmechanismus (1)

- **Port**
  Zuordnung der Pakete zur nächsthöheren Ebene

- **Socket**
  Eindeutige Adressierung einer TCP-Verbindung
  (IP-Adresse + Port-Nr.)

- **Well Known Port/ Socket**
  (Registrierte) Port-Nr. für (Standard-)Applikationen
  z. B. FTP: 21/ 20
  TELNET: 23
  SMTP: 25
  (vgl. „services“-Dateien)

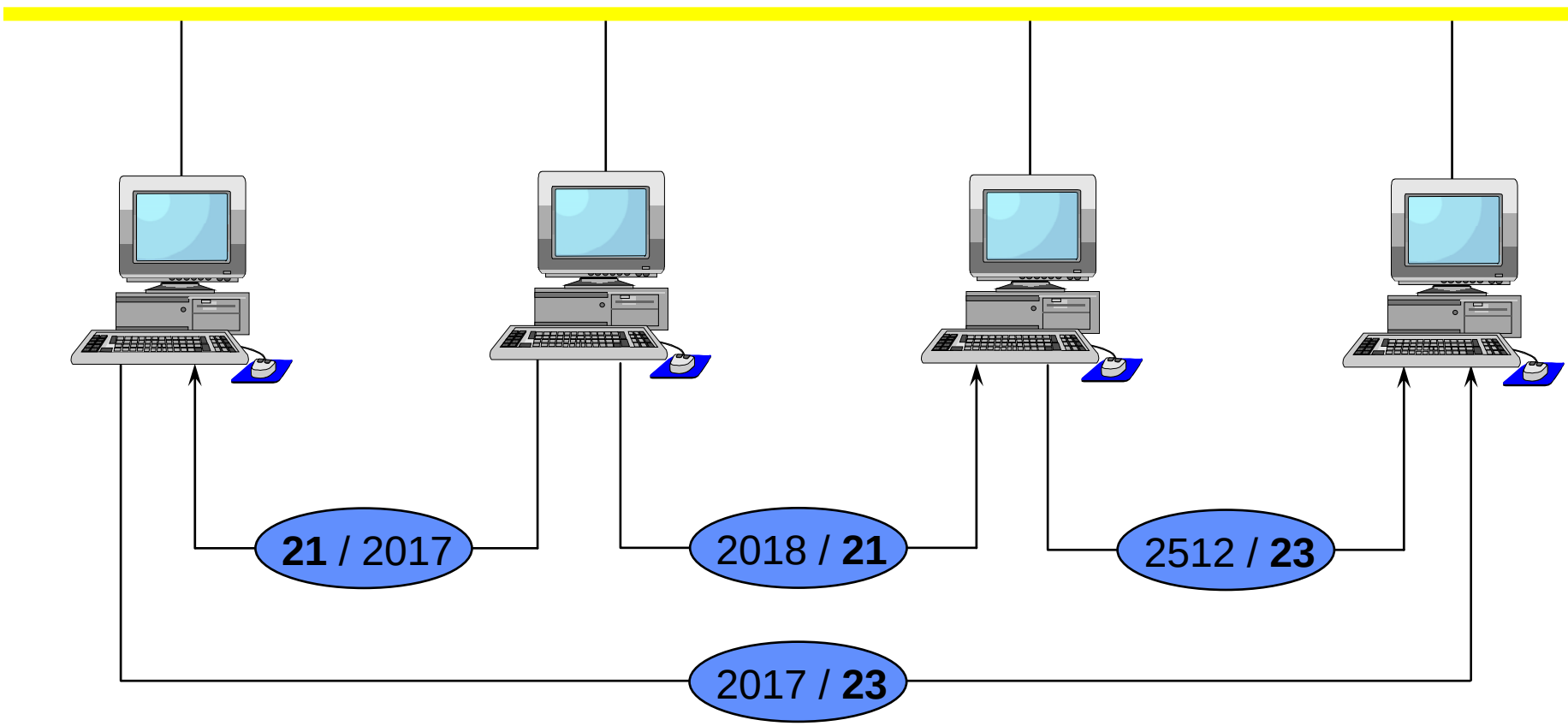
TCP - Well-Known-Ports (Auswahl)

20	**FTP-Data**	
21	**FTP**	(Steuerleitung)
23	**TELNET**	
25	**SMTP**	(Simple Mail Transfer Protocol)
43	nicname	(Who Is)
53	**domain**	(DNS)
66	**sql*net**	(Oracle SQL*NET)
67/68	**BOOTP**	(Server/Client)
70	**gopher**	
80	**WWW-HTTP**	
110	**POP3**	
111	**sunrpc**	(“NFS” - SUN Remote Procedure Calls)
137/ 138/ 139	**netbios**	(name-/ datagram-/ session service)
161/ 162	**SNMP**	(SNMP/ SNMP-Trap)
443	**https**	
512/ 514	**exec/ cmd**	(rexec/ rsh)

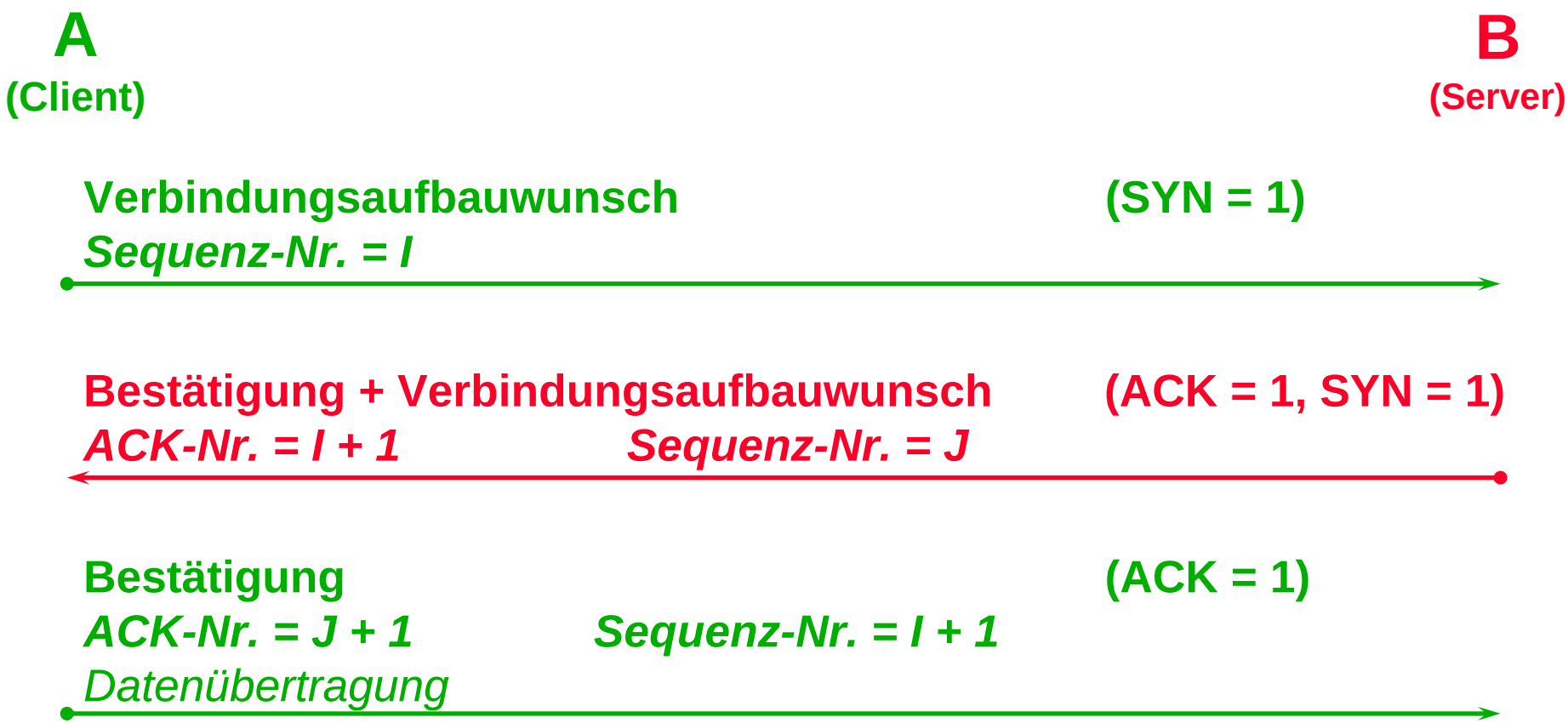
TCP - Well-Known-Ports (Auswahl)
Besonderheiten

513/ tcp	**login**	(rlogin; *nur* **TCP**-Port!)
513/ udp	**who**	(rwho/ ruptime; *nur* **UDP**-Port!))
ab 1024:	***„High-Ports“***	
1352	**Lotus Notes**	
1416	**Novell LU 6.2**	
1525	**orasrv**	(Oracle)
1527	tlisrv	(“)
1529	coauthor	(“)
1986-1999	cisco	(u.a. licensemanager, snmp-rcp-port)
1989	mshnet	(MHSnet system)
2784	www-dev	(world wide web - development)

TCP - Multiplexmechanismus (2)



TCP - Verbindungsaufbau
(Three-Way-Handshake)



TCP - Verbindungsabbau

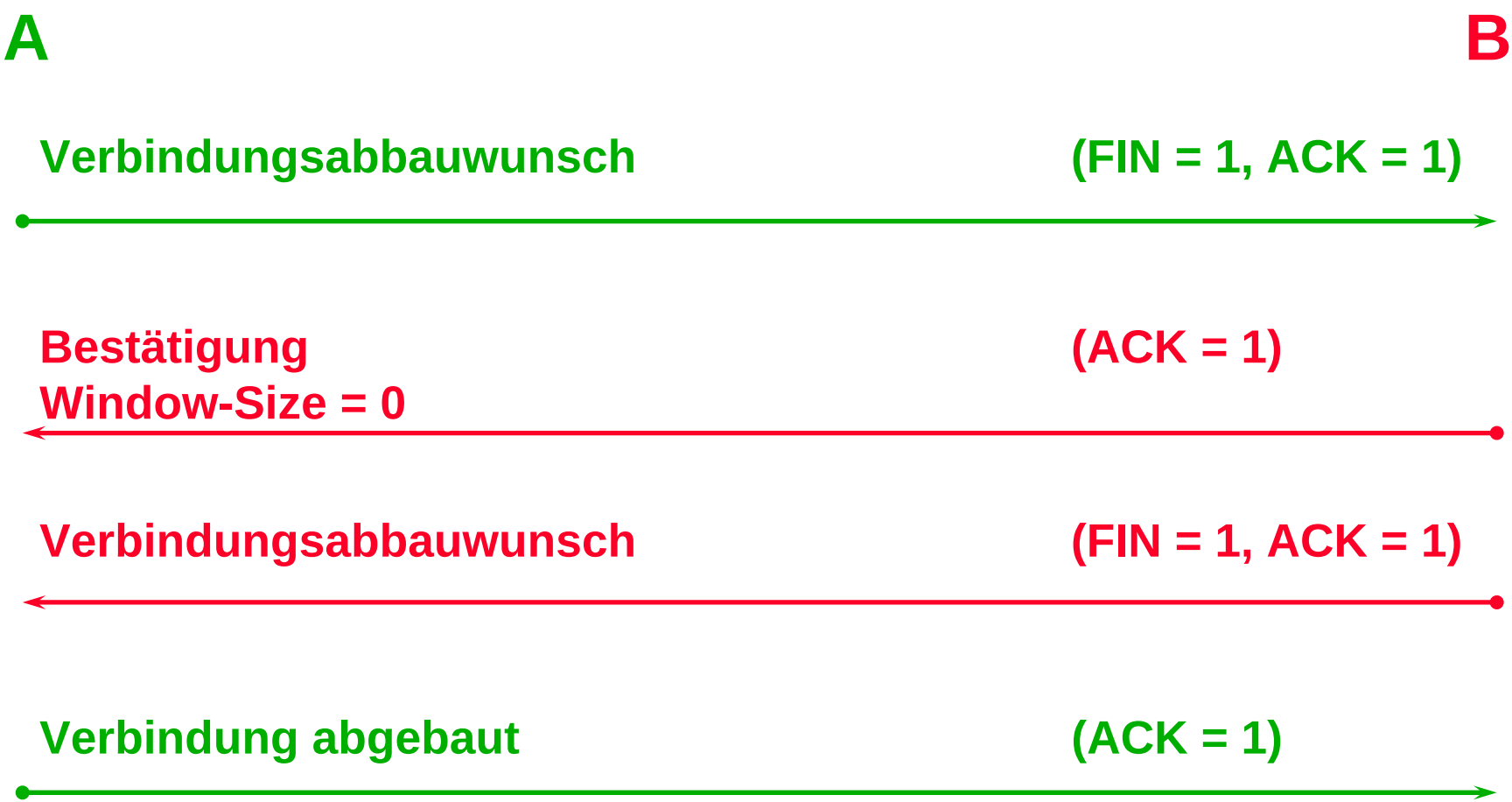
- **Alle Daten, die zur Übertragung anstehen, werden übermittelt und der Empfänger über den Verbindungsabbau informiert**

  ➔ gesicherter Abbau mit einem Three-Way-Handshake

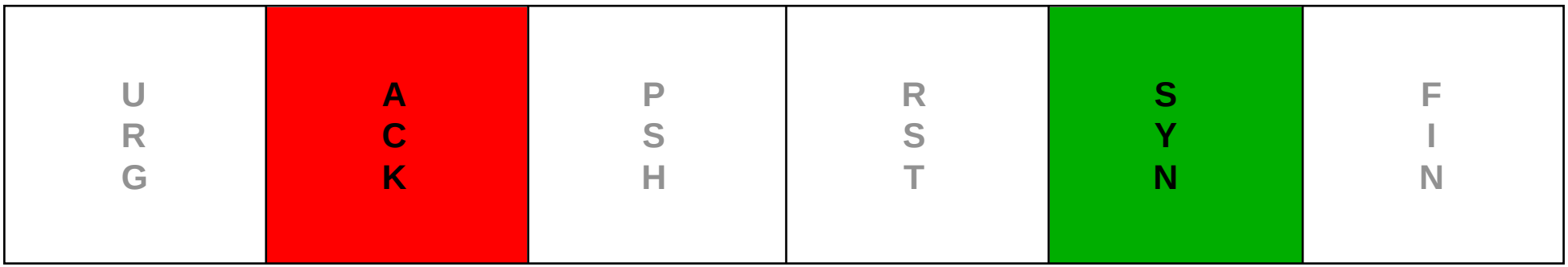
  ➔ alle Daten werden übermittelt bevor die Verbindung endgültig abgebaut werden kann

  ➔ nach einer gewissen Wartezeit wird die Verbindung abgebaut

TCP - Verbindungsabbau



TCP - Flags (SYN, ACK)



SYN zeigt an, dass eine Verbindung aufgebaut (synchronisiert) werden soll

ACK bestätigt den Empfang von Daten (acknowledgement)

TCP - Flags (RST, FIN)

RST zeigt an, dass der Sender die Verbindung abbauen will (reset). Wird auch als Antwort auf ungültige Pakete gesendet.

FIN bestätigt, dass die Verbindung endgültig abgebaut ist (final) und keine weiteren Daten folgen (➔ Last Packet Flag).

TCP - Flags (PSH, URG)

PSH teilt dem Empfänger mit, dass die Daten sofort an die höhere Schicht weitergereicht werden müssen (push)

URG zeigt an, dass der "Urgent-Pointer" berücksichtigt werden muss. Dieser kennzeichnet das Ende von Vorrangsdaten

TCP - Flusssteuerung

Problem:

- **werden die Pakete schneller gesendet, als sie der Empfänger verarbeiten kann, hat dies Konsequenzen**
  - ➔ neu ankommende Segmente müssen verworfen werden
  - ➔ daraus resultieren Sendewiederholungen, die die Datenübertragung verlangsamen und Sender und Empfänger zusätzlich belasten

Lösung:

- **Der Empfänger teilt dem Sender durch den Sliding-Window-Mechanismus mit, wie viele Segmente er (noch) aufnehmen kann**

TCP - Verbindungsmanagement

- **Daten können beim Transport**
  - ➔ verloren gehen
  - ➔ verfälscht werden (defekte Pakete)
  - ➔ durcheinander gebracht werden (falsche Reihenfolge)
  - ➔ verzögert werden
  - ➔ dupliziert werden

TCP - Sendewiederholung

- **Unter folgenden Umständen muss ein Datenpaket erneut gesendet werden:**
  - ➔ das Paket ist beschädigt und wird vom Empfänger vernichtet
  - ➔ das Segment geht (bereits) im Netz verloren

- **TCP arbeitet mit dem sog. PAR - Mechanismus**
  (**P**ositive **A**cknowledgement with **R**etransmission)
  - ➔ ACK n+1
    alle Daten bis zur Sequenznummer “n” werden bestätigt
    (als nächstes wird das Segment n+1 erwartet)

TCP - Retransmission Timer

- **Segment wird wiederholt, wenn der Retransmission Timer vor Eintreffen der Empfangsbestätigung abläuft**
- Problem
  - ➔ *Anfangswert zu niedrig:*
    zu viele Sendewiederholungen (**Duplikate!**)
  - ➔ *Anfangswert zu hoch:*
    verlorenes Segment wird zu spät wiederholt

⇨ **Die TCP-Spezifikationen *schreiben* einen dynamischen Retransmission Timer *vor* (RFC 2988)**

TCP - Retransmission Timer

- **Basis Algorithmus (Begriffe)** - nach RFC 2988 (Nov. 2000)

  – Retransmission Timeout (***RTO***)
  – Round-Trip Time (***RTT***)
  – Smoothed Round-Trip Time (*SRTT*) [= gemittelte RTT]
  – Round-Trip Time Variation (*RTTVAR*) [= Abweichung]

  – Anfangswert des RTO zwischen 2,5 sec und 3 sec
  – danach:

$$RTO < SRTT + 4*RTTVAR$$

TCP - Duplikatbehandlung

- **Der Empfänger kann Original und Duplikat nicht voneinander unterscheiden**
- **Der Empfänger nimmt an, dass seine Bestätigung verloren gegangen ist und bestätigt erneut**
- **Der Sender ignoriert, wenn Segmente mehrmals bestätigt werden**
- **Duplikate können auch nach dem Verbindungsabbau eintreffen und werden dann ignoriert**

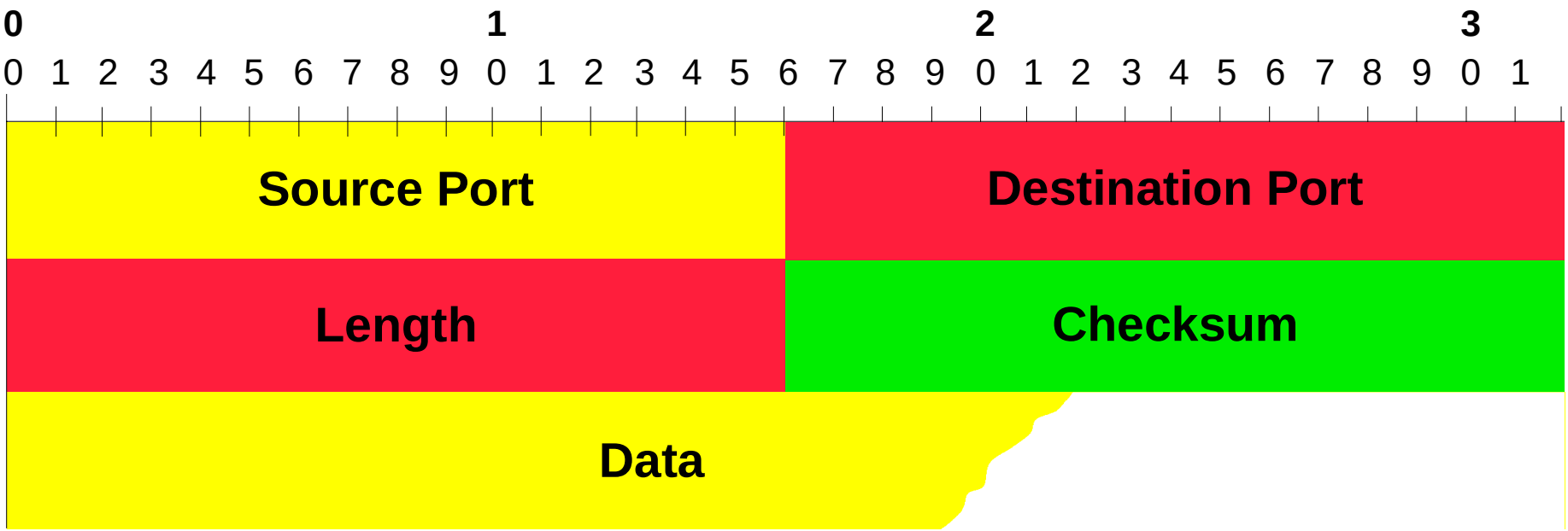
Wichtige TCP - Timer

- **Retransmission Timer**
  - ➔ nach Ablauf werden Daten neu geschickt
- **Give Up Timer**
  - ➔ max. Zeit, die der Sender bis zur Bestätigung seiner Pakete wartet
- **Reconnection Timer**
  - ➔ min. Zeit zwischen Abbau und Aufbau einer Verbindung
- **Retransmit-Syn Timer**
  - ➔ min. Zeit zwischen erfolglosem Verbindungsaufbau und erneutem Connection Request
- **Window Timer**
  - ➔ max. Zeit zur Umstellung der Window-Size

Kapitel 11

User Datagram Protocol (UDP)

UDP - Header



User Datagram Protocol (UDP)

RFC 768 - STD 6

- **Kein MIL-Standard**
- **setzt direkt auf dem Internet Protokoll (IP) auf**
- **IP-Protokoll-Nr.: 17**
- **Datagram Service zwischen Rechnern (keine virtuelle Verbindung)**

UDP - Eigenschaften

- **Transport Protokoll ohne “End to End”- Kontrolle**
  - ➔ **Kein Verbindungsmanagement**
    (keine aktiven Verbindungen!)
  - ➔ **Keine Flusskontrolle**
  - ➔ **Kein Mulitplexmechanismus**
  - ➔ **Keine Zeitüberwachung**
  - ➔ **Keine Fehlerbehandlung**

Dienste auf UDP

Dienst	UDP-Portnummer
IEN 116	42
DNS	53
RIP	520
BootP	67, 68
TFTP	69
sunrpc (NFS)	111
SNMP/ SNMP-TRAP	161, 162

Vergleich der Layer-4-Protokolle TCP und UDP

Eigenschaft	TCP	UDP
Ende zu Ende Kontrolle	ja	nein
Zeitüberwachung der Verbindung	ja	nein
Flow-Control (über das Netz)	ja	nein
Reihenfolgerichtige Übertragung	ja	nein
Erkennung von Duplikaten	ja	nein
Fehlererkennung	ja	einstellbar
Fehlerbehebung	ja	nein
Addressierung der höheren Schichten	ja	ja
Three-Way-Handshake	ja	nein
Größe des Headers	20 - 60 Byte	8 Byte
Geschwindigkeit	langsam	schnell
Belastung der Systemressourcen	normal	gering

Kapitel 12

Teletype Network (TELNET)

TELNET

RFC 854 - STD 8 - MIL-Standard 1782

- **Setzt auf dem gesicherten Transport Service von TCP auf**
- **TCP/UDP Port 23**
- **Remote Login-Dienst**

TELNET - Problematik

- Vielzahl von Terminal-Typen
- Verbindung zu Rechnern verschiedener Hersteller
- Unterschiedliche Übertragungseigenschaften

TELNET - Arbeitsweise

Lösung des Problems

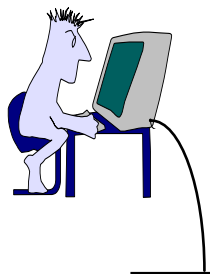
- **Beim TELNET wirken drei Funktionsgruppen zusammen:**
  - ➔ Network Virtual Terminal (NVT)
  - ➔ TELNET-Kommandos
  - ➔ Optionen

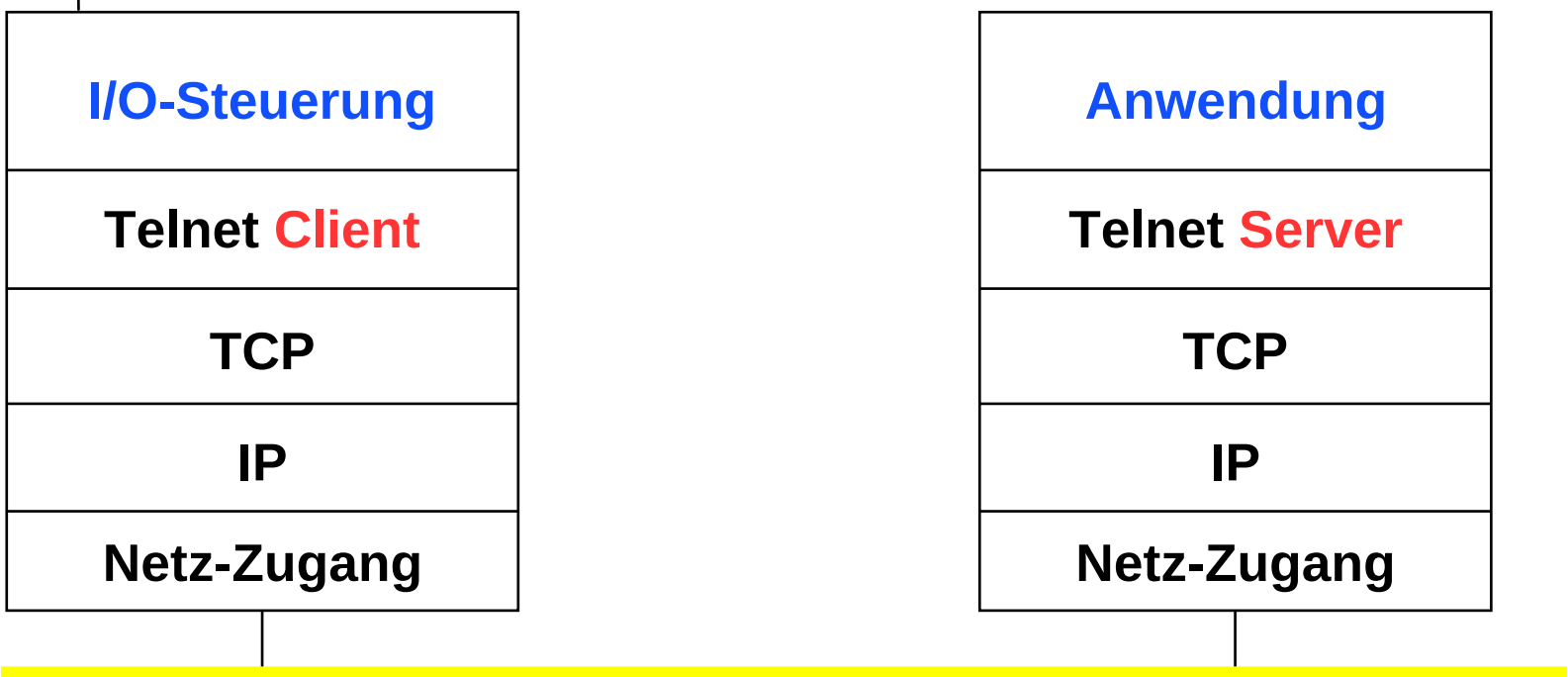
- **TELNET verwendet keinen eigenen Protokoll-Header, sondern verpackt die Steuerzeichen in dem Datenstrom**
  - ➔ Das **I**nterpret **A**s **C**ommand (**IAC**) (Hex FF) wird unmittelbar vor die Kommandodaten gestellt

TELNET - Network Virtual Terminal

- Fiktive Ein-/Ausgabe-Einheit mit bekannten Eigenschaften
- “Drucker” zur Anzeige von Ausgabedaten
- Tastatur zur Dateneingabe
- 7 Bit ASCII in 8 Bit Wort (per default)
- Unbegrenzte Zeilen- und Seitenlänge
- Steuerfunktionen
- “Drucker” für Steuerzeichen

TELNET - Network Virtual Terminal (Modell)





TELNET - Lokale Kommandos

- **Lokale Kommandos werden nicht über das Netz übertragen**
  - ➔ **Erase Character:** Löscht das letzte eingegebene Zeichen
  - ➔ **Erase Line:** Löscht die letzte Eingabezeile

TELNET - Remote-Kommandos (Auswahl)

- **Remote-Kommandos werden durch vorgestelltes <IAC> übertragen**

➔ **Interrupt Process:** (dez. 244)	Bewirkt den Abbruch des laufenden TELNET-Prozesses. Erzwingt Abbau der bestehenden Verbindung
➔ **Abort Output:** (dez. 245)	Datenausgabe wird abgebrochen. Prozess bleibt bestehen
➔ **Are You There:** (dez. 246)	Überprüft Prozess-Prozess-Kommunikation. Bewirkt Signal
➔ **Break:** (dez. 243)	Darstellung der Break-Taste
➔ **Go Ahead:** (dez. 249)	Signal zum Richtungswechsel bei Halbduplex-Übertragung

TELNET - Aushandeln von Optionen

- **Regeln:**
  - ➔ Aufforderung zum Einschalten kann zurückgewiesen werden
  - ➔ Aufforderung zum Ausschalten von Optionen muss akzeptiert werden
  - ➔ es dürfen nie Optionen ausgehandelt werden, die sich bereits in der gewünschten Stellung befinden
  - ➔ Optionen werden erst nach Bestätigung gültig
  - ➔ Optionen treten unmittelbar nach der Bestätigung in Kraft

TELNET - Aushandeln von Optionen Befehle

- **WILL** Der **Sender** zeigt an, dass er eine Option **ein**schalten möchte
  Antwort: DO oder ***DONT***
- ***WONT*** Der **Sender** zeigt an, dass er eine Option ***aus***schalten möchte
  Antwort: *DONT*
- **DO** Der Sender zeigt an, dass der **Empfänger** eine Option **ein**schalten soll
  Antwort: WILL oder ***WONT***
- ***DONT*** Der Sender zeigt an, dass der **Empfänger** eine Option ***aus***schalten soll
  Antwort: *WONT*

TELNET - Optionen

• **Extended ASCII**	**(dez. 17)**	(RFC 698)
• **Binary Transmit**	**(dez. 0)**	(RFC 856)
• (local) **Echo**	**(dez. 1)**	(RFC 857)
• **Suppress GA**	**(dez. 3)**	(RFC 858)
• **Terminal Speed**	**(dez. 32)**	(RFC 1079)
• **Terminal Type, X.3 PAD**	**(dez. 24)**	(RFC 1091)
• **Extended Options List**	**(dez. 255)**	(RFC 861)

TELNET - Terminal-Typen

(aus „Assigned Numbers“ - Auswahl)

DEC-DECWRITER-I
DEC-DECWRITER-II
DEC-GIGI
DEC-GT40
DEC-GT40A
DEC-GT42
DEC-LA120
DEC-LA30
DEC-LA36
DEC-LA38
DEC-VT05
DEC-VT100
DEC-VT101
DEC-VT102
DEC-VT125
DEC-VT131
DEC-VT132
DEC-VT200
DEC-VT220
DEC-VT240
DEC-VT241
DEC-VT300
DEC-VT320
DEC-VT340

IBM-1050
IBM-2741
IBM-3101
IBM-3101-10
IBM-3151
IBM-3179-2
IBM-3180-2
IBM-3196-A1
IBM-3275-2
IBM-3276-2, -3, -4
IBM-3277-2
IBM-3278-2, -3, -4, -5
IBM-3278-2E, -3E, -4E, -5E
IBM-3279-2, -3
IBM-3279-2E, -3E
IBM-3477-FC, -FG
IBM-5081
IBM-5151
IBM-5154
IBM-5251-11
IBM-5291-1
IBM-5292-2
IBM-5555-B01, -C01
IBM-6153
IBM-6154
IBM-6155
IBM-AED

PERKIN-ELMER-550
PERKIN-ELMER-1100
PERKIN-ELMER-1200

TELEVIDEO-910
TELEVIDEO-912
TELEVIDEO-920
TELEVIDEO-920B
TELEVIDEO-920C
TELEVIDEO-925
TELEVIDEO-955
TELEVIDEO-950
TELEVIDEO-970
TELEVIDEO-975

TEKTRONIX-4006
TEKTRONIX-4010
TEKTRONIX-4012
TEKTRONIX-4013
TEKTRONIX-4014
TEKTRONIX-4023
TEKTRONIX-4024
TEKTRONIX-4025
TEKTRONIX-4027
TEKTRONIX-4105
TEKTRONIX-4107
TEKTRONIX-4110
TEKTRONIX-4112
TEKTRONIX-4113
TEKTRONIX-4114
TEKTRONIX-4115
TEKTRONIX-4125
TEKTRONIX-4404

Insgesamt: 326
(Stand: 1.5. 2001)

Kapitel 13

File Transfer Protocol (FTP)

File Transfer Protocol (FTP)

RFC 959 - STD 9 - MIL-Standard 1780

- **Setzt auf dem gesicherten Transport Service von TCP auf**
- **TCP/UDP Port 21 und (ggf.) 20**
- **File-Transfer-Dienst**

FTP - Problematik

- **unterschiedliche Architekturen:**
  Prozessoren, Betriebssysteme, ...
- **unterschiedliche Datenformate:**
  Bitanordnung, ASCII, EBCDIC, ...
- **unterschiedliche Dateistrukturen:**
  zeilenorientiert, record-orientiert, seitenorientiert, ...
- **unterschiedliche Übertragungsweisen:**
  stream, asynchron, blockmode, ...

FTP - Arbeitsweise

Lösung des Problems

- **Konsens zwischen Systemen erfolgt**
  - ➔ **durch Reduzieren der individuellen Eigenschaften auf Optionen von allgemeiner Bedeutung**
  - ➔ **nicht durch Transformation auf ein Meta-Format (“Network Virtual File”)**

FTP-Session (Prinzipdarstellung)

❶ **Aufbau einer Steuerleitung/ -verbindung (Port 21) durch Client**

❷ **Austausch von Befehlen und Parametern**

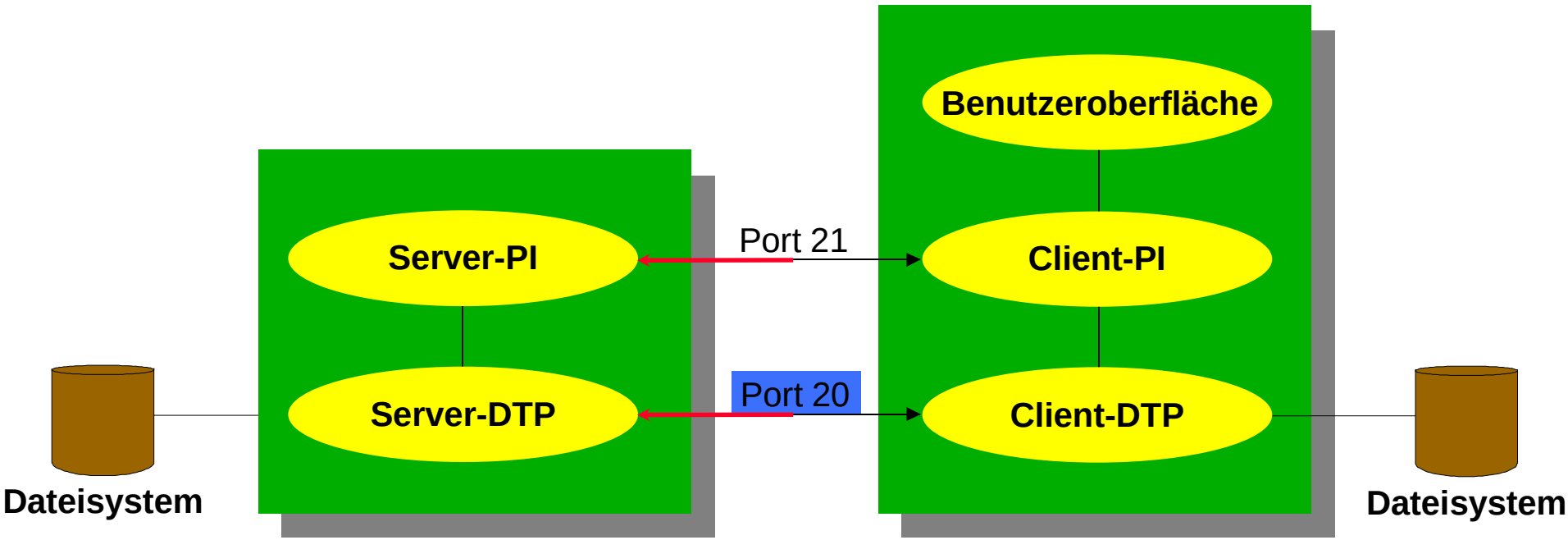
① **Aufbau einer Datenleitung/ -verbindung (Port 20) durch Server**

② **Datenübertragung**

③ **Abbau der Datenverbindung**

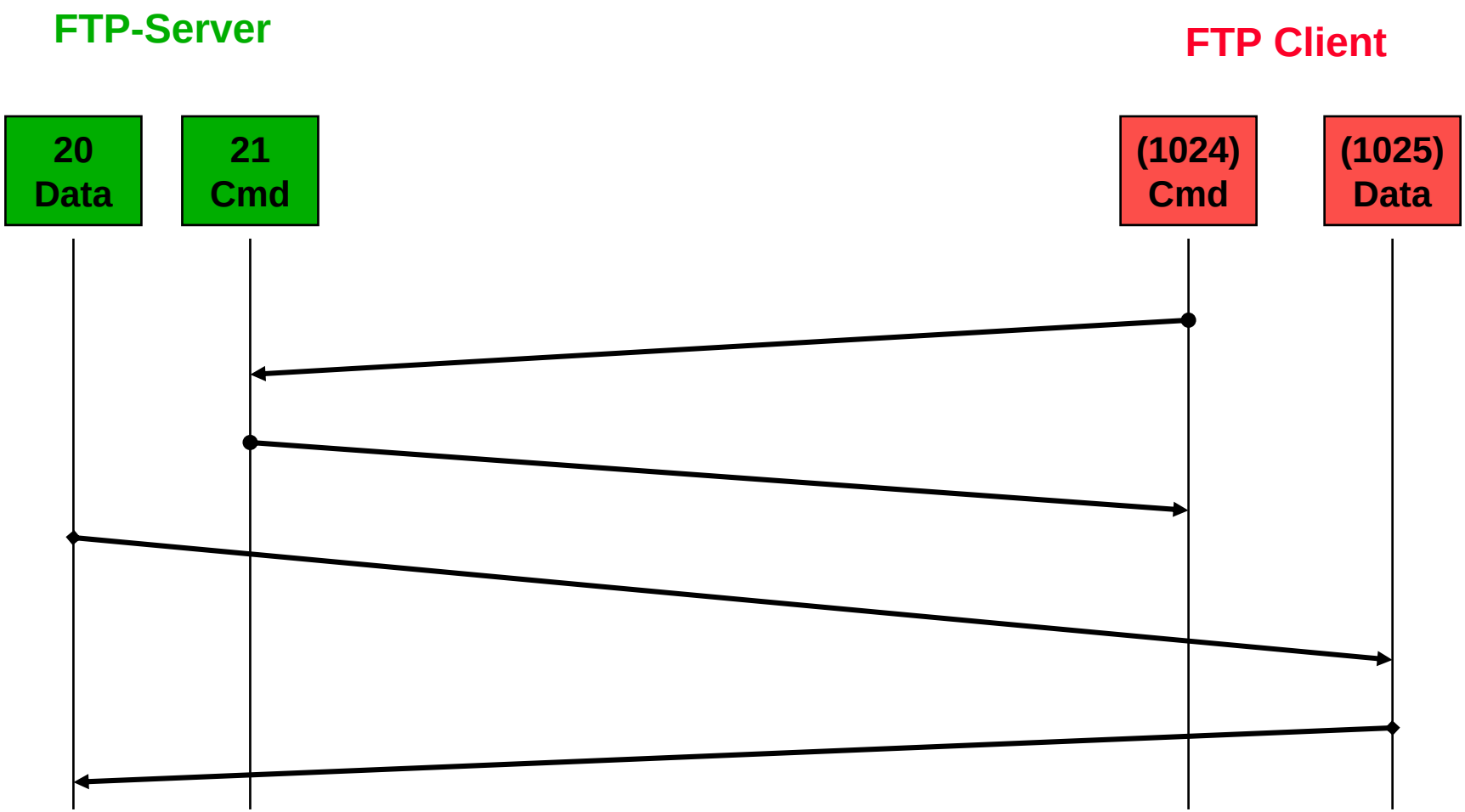
❸ **Abbau der Steuerleitung**

Das FTP - Modell

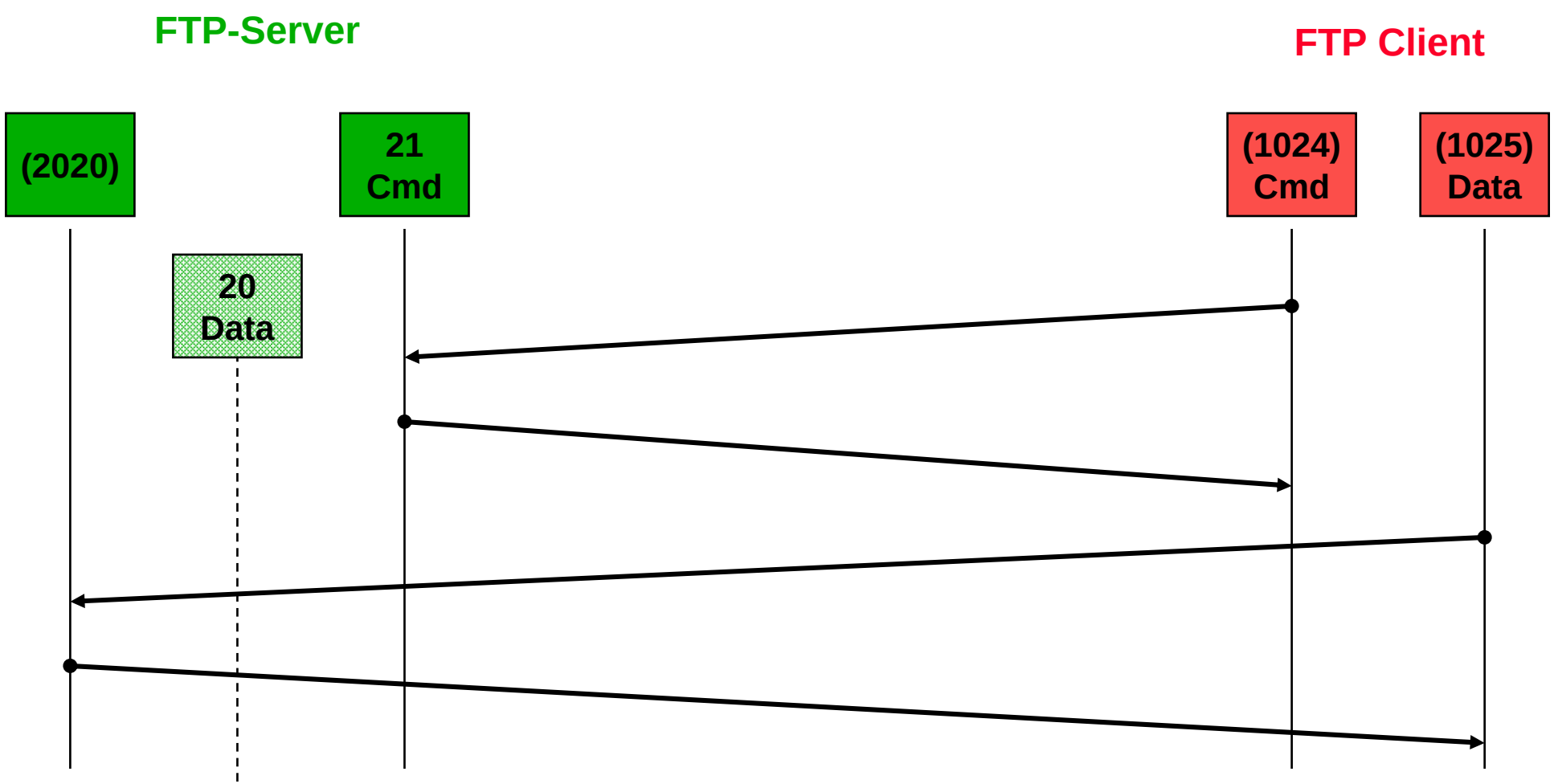


PI = **P**rotocol **I**nterpreter
DTP = **D**ata **T**ransfer **P**rocess

Active FTP



Passive FTP



FTP - Transfer Parameter

- **TYPE** (representation type)
  - ➔ **A** **ASCII**
  - ➔ **E** **EBCDIC**
  - ➔ **I** **image**
  - ➔ **L** <*byte size*> **local byte-size**
- **STRU** (structure)
  - ➔ **F** **file,** no record structure
  - ➔ **R** **record structure**
  - ➔ **P** **page structure**

Wichtige FTP - Befehle

- **dir, ls** Inhaltsverzeichnis anzeigen
- **cd** Inhaltsverzeichnis wechseln
- **pwd** Name des aktuellen Inhaltsverz. anzeigen
- **bin** bzw. **ascii**
  Übertragungsmodus binär/ ascii
- **hash** Übertragung grafisch darstellen (mit #####)
- **get** bzw. **put (mget** bzw. **mput)**
  eine Datei (ein komplettes Verzeichnis) holen bzw. senden

Kapitel 14

Simple Mail Transfer Protocol (SMTP)

Simple Mail Transfer Protocol (SMTP)

RFC 821 - STD 10 - MIL-Standard 1781

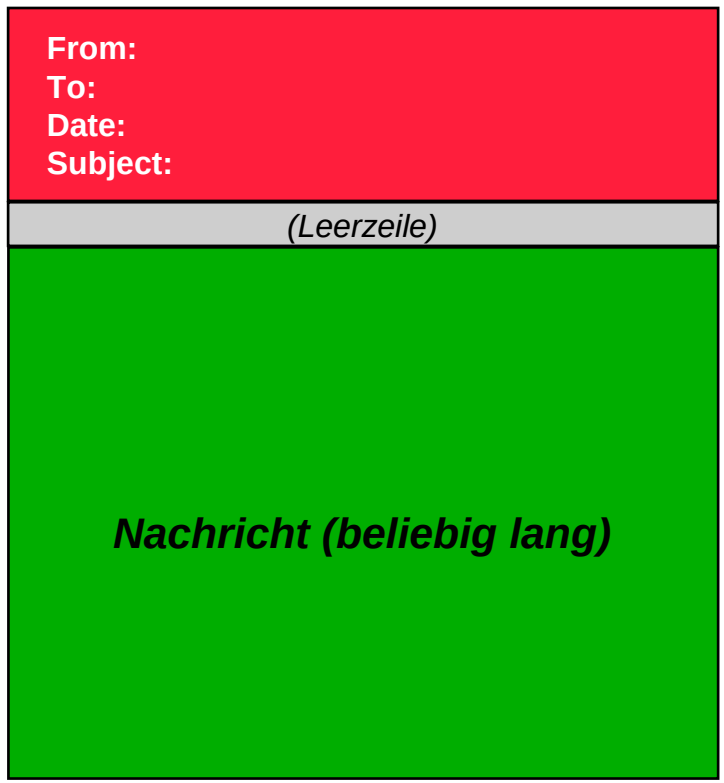
- **Setzt auf dem gesicherten Transport Service von TCP auf**
- **TCP/UDP Port 25**
- **E-Mail-Dienst**

SMTP - Message-Format

STD 11

- **definiert in RFC 822 (“822-Message-Format” bzw. „The format of ARPA Internet text messages“)**
- **verwendet 7-Bit-ASCII-Zeichensatz (ausschließlich!)**
- **Message setzt sich zusammen aus Header und Body**

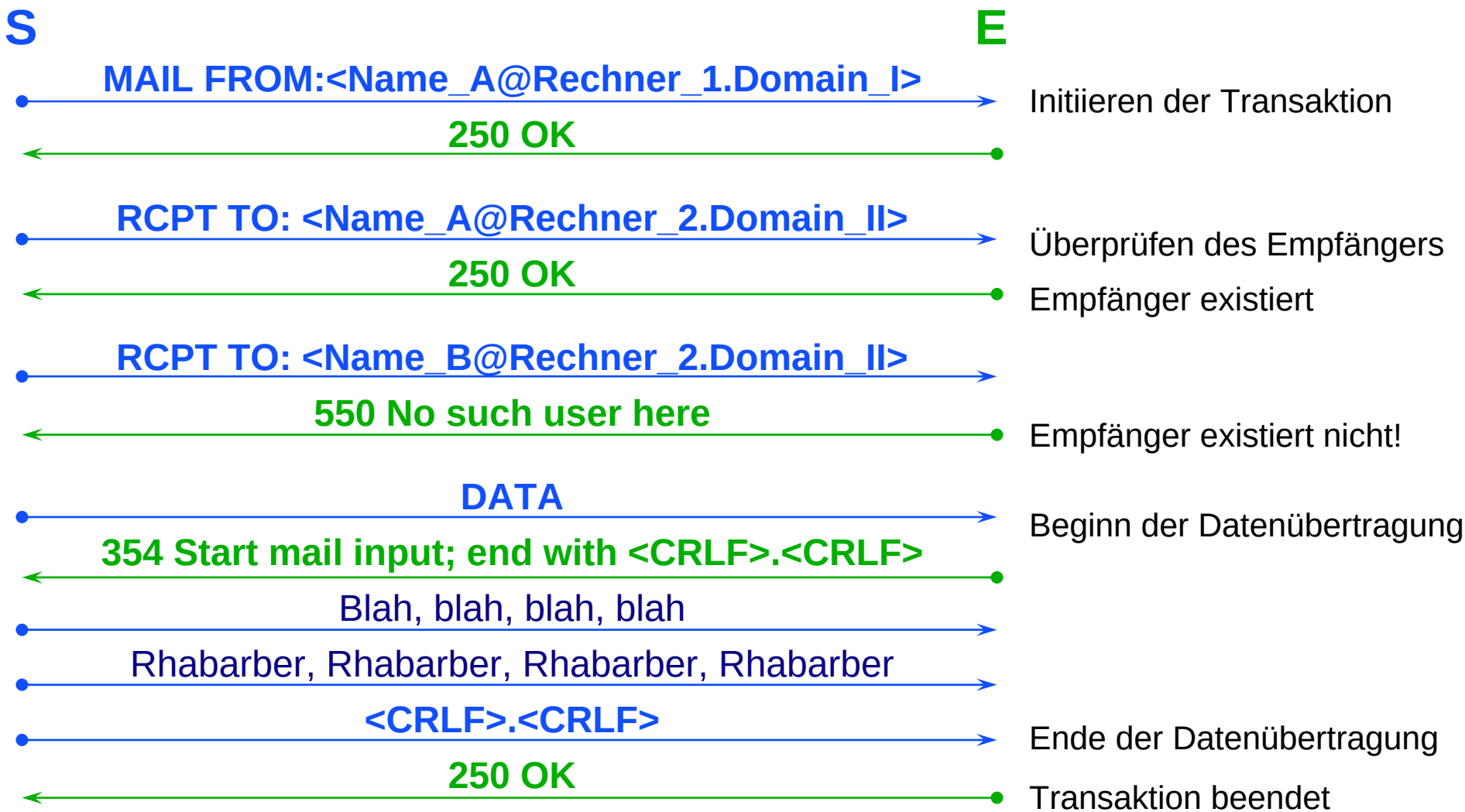
SMTP - Message-Format



Message Header

Message Body

SMTP - Übertragung



SMTP - Kommandos (Auswahl)

- **MAIL:** leitet die Transaktion ein mit der Identifikation des Absenders
- **RCPT:** *receipient* - identifiziert den/ die Empfänger
- **VRFY:** *verify* - sucht zu einem vorgegebenen Namen den zugehörigen Pfad
- **EXPN:** *expand* **-** interpretiert einen Namen als Mailing-Liste und löst diesen auf
- **SEND:** kommuniziert direkt mit dem Terminal des Empfängers
- **SOML:** *send or mail* - kommuniziert mit dem Terminal bzw. der Mailbox, wenn das Terminal nicht erreichbar ist
- **SAML:** *send and mail* - kommuniziert mit Terminal und Mail-Box

Post Office Protocol - Version 3 (POP3)

RFC 1939 - STD 53

- **Setzt auf dem gesicherten Service von TCP auf**
- **TCP/UDP-Port 110**
- **ermöglicht einem Client das „Abholen“ von E-Mail von einem Mail-Server**
- **User-Authentisierung erfolgt über Username/ Password**
- **unterstützt keine Veränderung der Mail auf dem Server (abgeholte Mail wird i.a. gelöscht)**
  (im Gegensatz zu: IMAP4 [**Internet Message Access Protocol**] - RFC 2060)

Kapitel 15

Name-Services

Name-Services - Aufgabe

- dienen der Zuordnung Rechnername → IP-Adresse
- werden im einfachsten Fall durch eine lokale Datei (/etc/hosts, hosts.txt etc.) realisiert
- können, je nach Ausprägung, recht komplex aufgebaut sein und vielartige Informationen weiterreichen

Internet Name Server
IEN 116

Internet Name Server

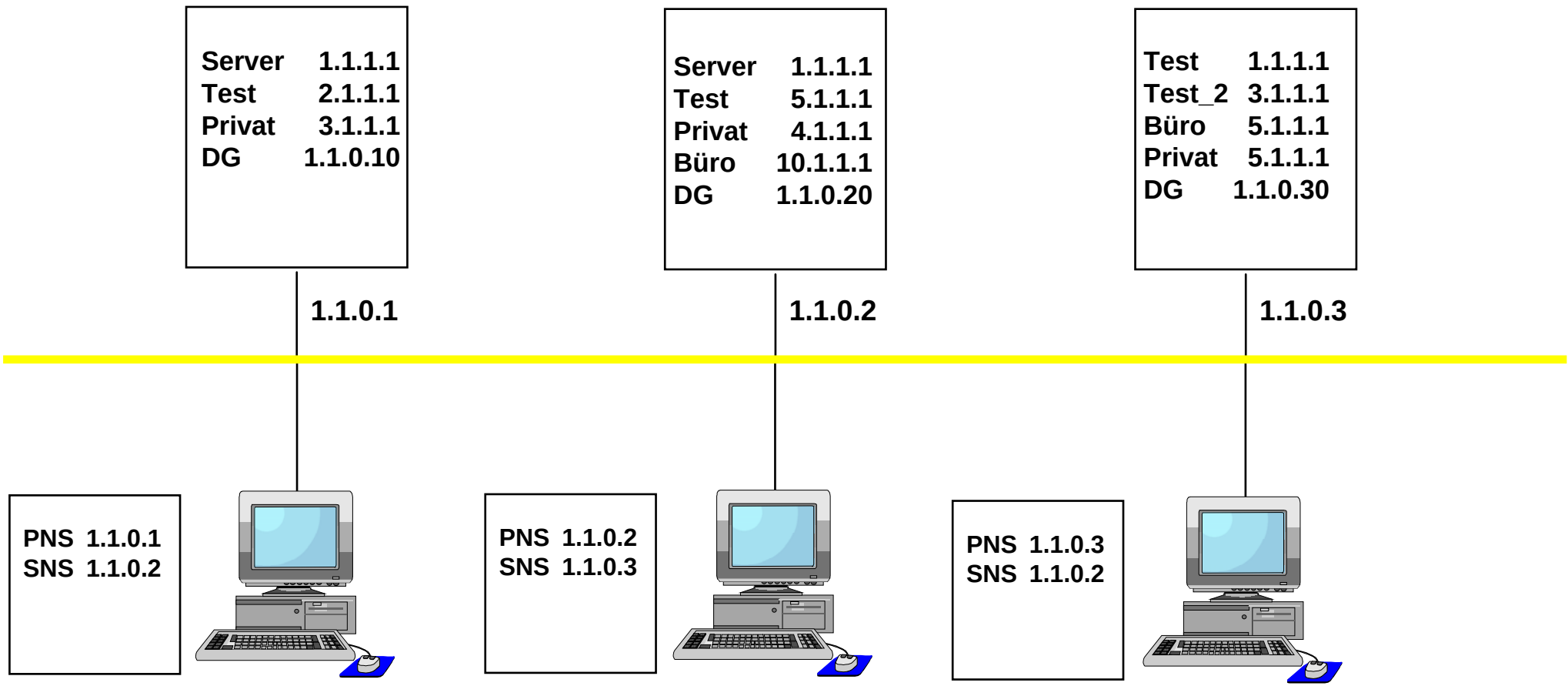
IEN 116 (August 1979)

- **kein MIL-Standard**
- **setzt auf dem Datagram-Transport-Service von UDP auf**
- **UDP/TCP Port 42**
- **Zuordnen von Hostnamen zu IP-Adressen**

IEN 116-Internet-Name-Service Funktionsweise (1)

- **Name-Server sind unabhängig voneinander**
- **kein hierarchisches System (flache Topologie)**
- **Wildcards optional**

IEN 116-Internet-Name-Service - Funktionsweise (2)



Domain Name System/ Service (DNS)

DNS

RFC 1033 - Administrators Operations Guide
RFC 1034 - Concepts and Facilities
RFC 1035 - Implementation and Specification

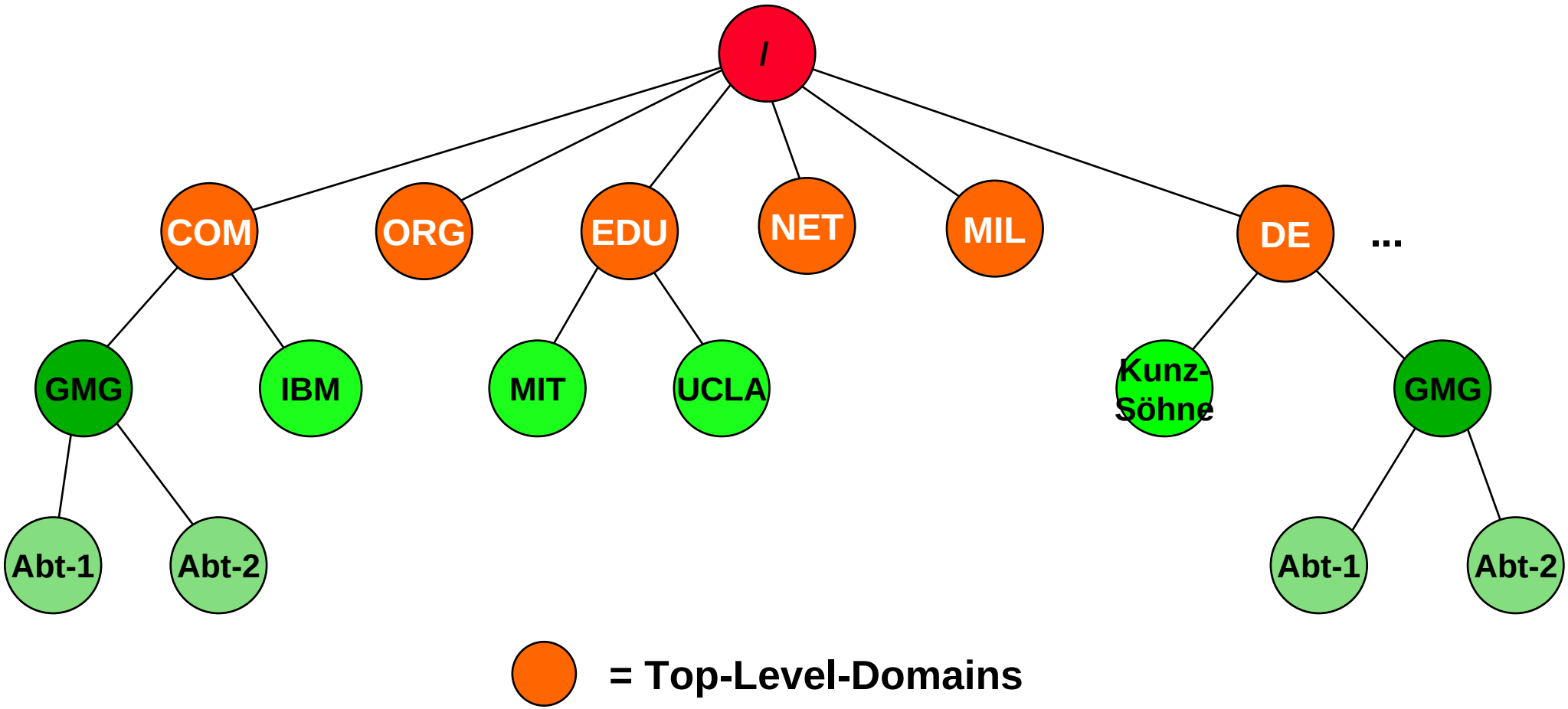
STD 13

- **kein MIL-Standard**
- **setzt auf dem Datagram-Transport-Service von UDP auf**
- **UDP/TCP Port 53 (UDP und TCP!)**
- **Zuordnen von Hostnamen zu IP-Adressen**

DNS - Funktionsweise

- **basiert auf einer verteilten Datenhaltung**
- **basiert auf einem hierarchischen Modell**
- **kennt verschiedene DNS-Servertypen**
- **verfügt, neben der Namen-IP-Zuordnung, über zusätzliche - teilweise optionale - Möglichkeiten**

DNS - Funktionsweise (hierarchisches Modell)



Top Level Domains (TLD)

- gTLD (Generic TLD):
  - z.B.: com, org, net
- ccTLD (Country Code TLD)
  - z.B.: de, ch, uk, us
- sTLD (Sponsored TLD)
  - z.B.: job, info, mobi, post, mail, travel, xxx

DNS - Servertypen

- **Primary Name Server (Master)**
  - ➔ Enthält Datenbank mit authorisierten Daten
  - ➔ Ort der Datenpflege
- **Secondary Name Server (Slave)**
  - ➔ Enthält Datenbank mit authorisierten Daten
  - ➔ Holt sich regelmäßig Updates von Master
- **Caching Server**
  - ➔ Merkt (“cacht”) sich nur Daten (nicht authorisiert)
  - ➔ verwirft “gecachte” Daten nach vorgegebener Zeit (TTL-Feld mit 32 Bit Länge)

DNS - Funktionalitäten und Funktionsweisen

- **Auflösen von Namen (in IP-Adressen)**
- **Auflösen von IP-Adressen (in Namen - optional)**
- **Übermitteln von weiteren Informationen**
- **Beantworten von Anfragen mit Name, Verweis auf anderen Server oder Error (Name nicht bekannt)**
- **Beantworten von Anfragen nur mit Name oder Error, da Anfrage ggf. an einen anderen Server weitergereicht wurde (rekursive Anfrage - optional)**

DNS - Query-Types (Auswahl)

- **A** Address (Rechneradresse)
- **NS** Name-Server
- **CNAME** Canonical Name (Zuordnung von Nicknames)
- **HINFO** Rechner- (Host-) Information (CPU, Betriebssystem)
- **SOA** Start Of Authority (Update von PNS-Daten)
  - ➔ **SERIAL** Änderungen in Datensatz (“Versionspflege”)
  - ➔ **REFRESH** Zeit zwischen Updatepolls
  - ➔ **RETRY** Zeitdauer bis zum Wiederholen eines fehlgeschlagenen REFRESH
  - ➔ **EXPIRE** Zeit bis zum Löschen eines Eintrages (nach fehlgeschlagenem REFRESH)
- **MX** Mail Exchange Server
- **WKS** Well Known Services (TCP/ UDP-Dienste <256)

DNS - Einschränkungen

- **Namen (Labels): max. 63 Byte**
- **Rechnernamen: max. 255 Byte**
- **TTL: positive Werte einer vorzeichen-behafteten 32 bit Integer Zahl**
- **UDP Nachricht: max. 512 Byte**

MERKE:

Ein DNS-Server muss sich nicht in der Domain befinden, für die er Informationen bereithält!

Kapitel 16

UDP Bootstrap Protocol (BootP)

Dynamic Host Configuration Protocol (DHCP)

BootP

BOOTP

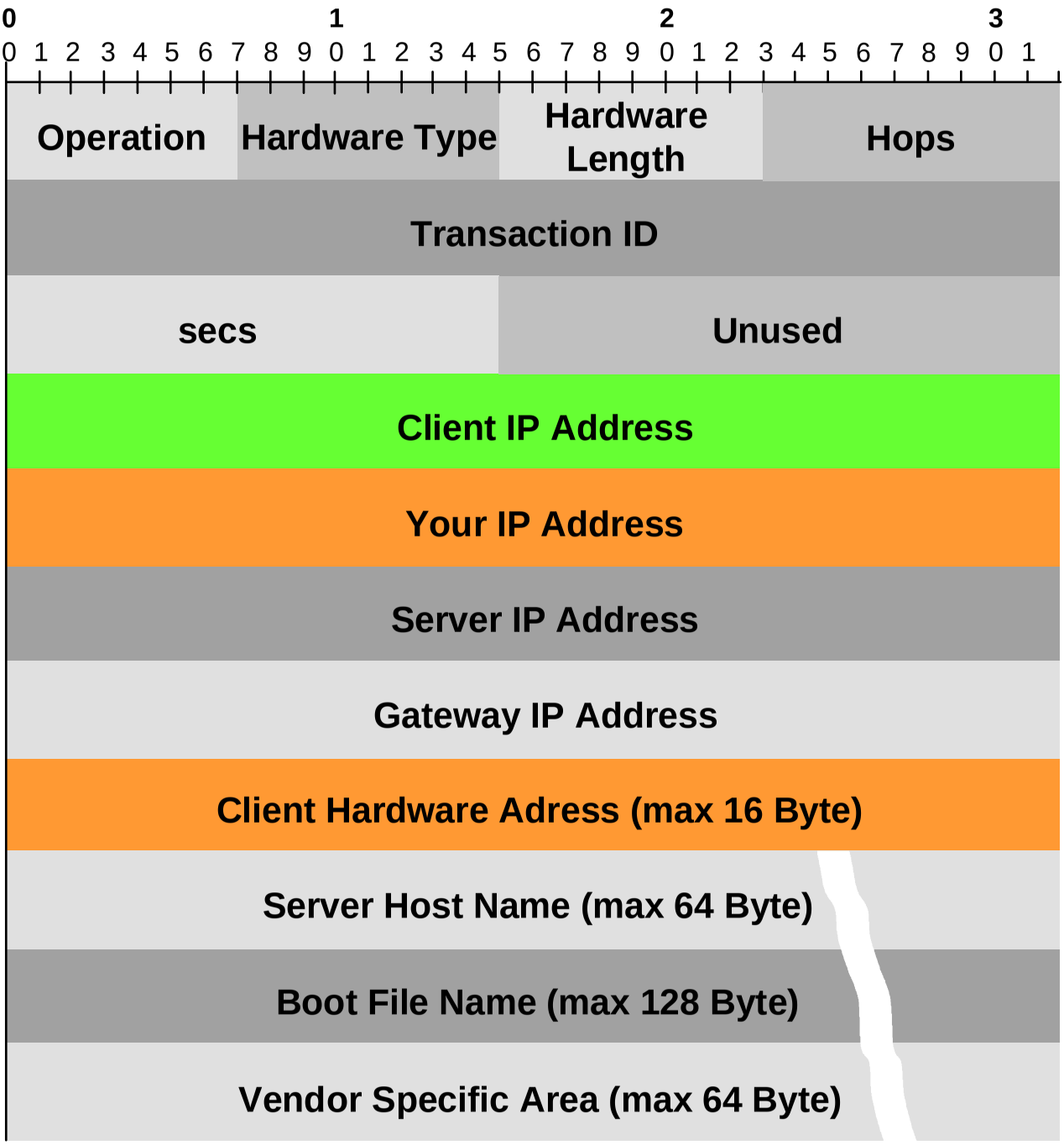
RFC 951, RFC 1542, 2132 (Vendor Specific Extensions)

- **kein MIL-Standard**
- **setzt auf dem Datagram-Transport-Service von UDP auf**
- **UDP/TCP Port 67 (Client ➔ Server)**
  68 (Server ➔ Client)
- **Umwandlung von Ebene 2-Adressen in IP-Adressen**
- **Übertragen von Informationen, die zum Booten notwendig sind (Vendor Specific Extensions)**

BOOTP - Funktionsweise

- **BOOTP-Request erfolgt gerichtet oder per IP-Broadcast**
- **wird ein BOOTP-Request nicht beantwortet, erfolgt eine erneute Anfrage**
- **um das Netzwerk nicht mit Paketen zu überschütten (“flooding”), wird eine dem Ethernetverhalten ähnliche Backoff-Strategie empfohlen.**
- **durch das “secs”-Feld, kann eine Antwortpriorität erreicht werden**
- **das Booten über Router (“Gateways”) hinweg ist optional und benötigt einen BOOTP-Relay-Agent**

BOOTP - Datenformat



BOOTP - Vendor Specific Area (Auswahl)

- **Time-of-Day** **aktuelle Zeit**
- **Subnet-Mask** **IP-Subnetz-Maske**
- **Router** **IP-Adresse von Routern**
- **Time-Server** **IP-Adresse eines Time-Servers**
- **IEN116-Server** **IP-Adresse eine IEN 116 Name-Servers**
- **Domain Server** **IP-Adresse eines Domain-Name-Servers**
- **LPR-Server** **IP-Adresse eines BSD-Print-Servers**
- **Hostname** **Name des Client (local station)**
- **Boot Size** **Größe des Boot-Files** (in 512 Byte Blocks)
- **Extensions Path** **Definiert TFTP-File, das als VSA interpretiert wird**
- **End (255h)** **Ende der Vendor Specific Area**

DHCP

DHCP

RFC 2131

- **nutzt BOOTP bzw. setzt auf BOOTP auf**
- **Paketaufbau identisch zu BOOTP**

  ➔ **Ausnahme:**

  “Vendor Specific Extensions” ➔ “Options” (RFC 2132)
  - **Minimumlänge des VSA-Feldes: 312 Byte**
  - **definiertes “Magic Cookie”** (99.130.83.99)
- **automatisches Zuweisen von IP-Adressen auf Zeit bzw. unendlich** (32 Bit-Wort = 1 sec - 136 Jahre)
- **manuelle Vergabe von IP-Adressen möglich**
- **DHCP-Server muss BOOTP-Clients bedienen können**

DHCP-Messages

- **DHCPDISCOVER** Broadcast von Client, zur Suche verfügbarer Server
- **DHCPOFFER** Server teilt Client Konfigurationsparameter mit
- **DHCPREQUEST** Client fordert angebotene Parameter von Server an bzw. bestätigt Parameter/ verlängert "Lease"
- **DHCPACK** Server bestätigt Client die Richtigkeit der Adresse
- **DHCPNACK** Server teilt Client mit, dass Adresse nicht verwendet werden kann
- **DHCPDECLINE** Client teilt Server mit, dass Adresse schon genutzt wird
- **DHCPRELEASE** Client teilt Server mit, dass Adresse nicht weiter benötigt wird

Automatische Adressvergabe durch DHCP

- **Client sucht DHCP-Server; ggf. Vorschläge" für Netzwerk-Adresse und Gültigkeitsdauer (DHCPDISCOVER)**
- **DHCP-Server antworten mit IP-Adresse (DHCPOFFER)**
- **Client sucht sich eine Antwort aus und antwortet allen Servern (DHCPREQUEST) - "Server Identifier Option" muss gesetzt sein**
- **Der ausgesuchte Server reserviert die vorgeschlagene Adresse und schickt Konfigurations-Parameter - ggf. vorher Test der Adresse durch ICMP-Echo Request (DHCPACK)**

  Alle anderen Server wissen, dass ihr "Angebot" abgelehnt wurde und die vorgeschlagene IP-Adresse wieder frei verfügbar ist

Kapitel 17

Trivial File Transfer Protocol (TFTP)

Trivial File Transfer Protocol (TFTP)

RFC 1350 - STD 33

- **kein MIL-Standard**
- **setzt auf dem Datagram-Transport-Service von UDP auf**
- **UDP/TCP Port 69**
- **einfacher File-Transfer-Dienst ohne Login-Prozedur (“Poor Man’s File Transfer”)**
- **wird (meist) für Netz-Boot-Vorgänge eingesetzt**

TFTP - Funktionsweise

- **TFTP verfügt über fünf Funktionen:**

  ➔ **Read Request (RRQ):**
  fordert File von Remote-Rechner an

  ➔ **Write Request (WRQ):**
  sendet File zu Remote-Rechner

  ➔ **Data (DATA):**
  kennzeichnet den eigentlichen Datenstrom

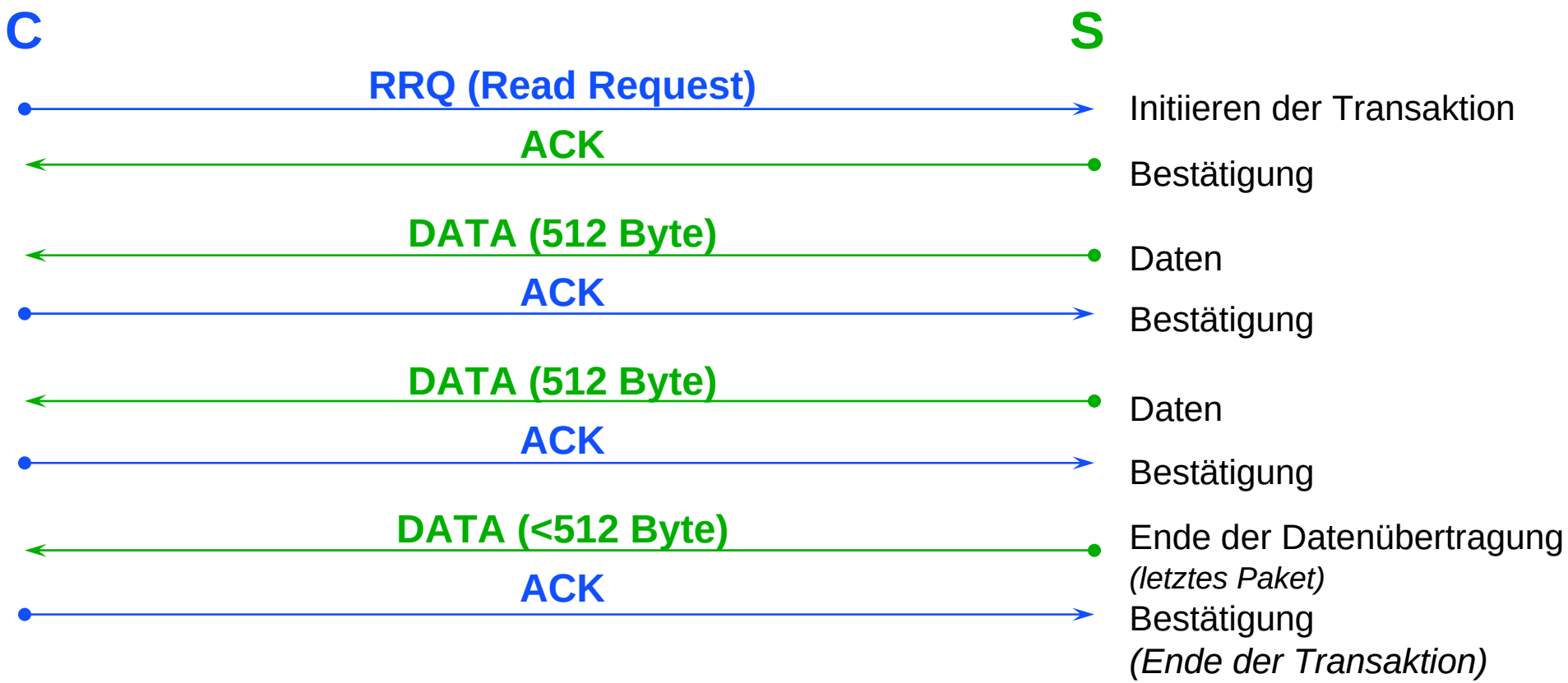
  ➔ **Acknowledgement (ACK):**
  bestätigt empfangene Pakete

  ➔ **Error (ERROR):**
  zeigt Übertragungsfehler an

TFTP - Übertragungsmechanismus

- Verbindungsaufbau mit RRQ bzw. WRQ
- Pakete werden in festem Format (512 Byte) übertragen
- Pakete < 512 Byte zeigen Ende der Übertragung an
- jedes gesendete Paket wird einzeln bestätigt
- ERROR verursacht Übertragungsabbruch - kein Retransmit!

TFTP - Übertragungsmechanismus



Kapitel 18

Die “R”-Utilities
rlogin, rcp, rsh/rexec

Die “R” Utilities
- rlogin, rcp, rsh/ rexec -

- **setzten auf dem gesicherten Transport Service von TCP auf**
- **TCP/UDP Ports:**
  512 (rsh), 513 (rlogin), 514 (rexec)
- **erlauben ein *login*, ein *copy* (rcp), so wie das Ausführen fremder Dateien (*shell-scripts*) auf einem fremden Rechner**
- **es ist keine aktive Identifizierung und Authetifizierung notwendig**

R-Utilities - Zugriffsmechanismen

- **zwei Dateien zur Freigabe von Zugriffsberechtigungen**
  - ➔ **.rhosts im Home-Verzeichnis des Anwenders**
  - ➔ **hosts.equiv (unter /etc)**
- **Freigabe bezogen auf Rechner- und Usernamen**
- **Anwender “root” muss immer Password eingeben**

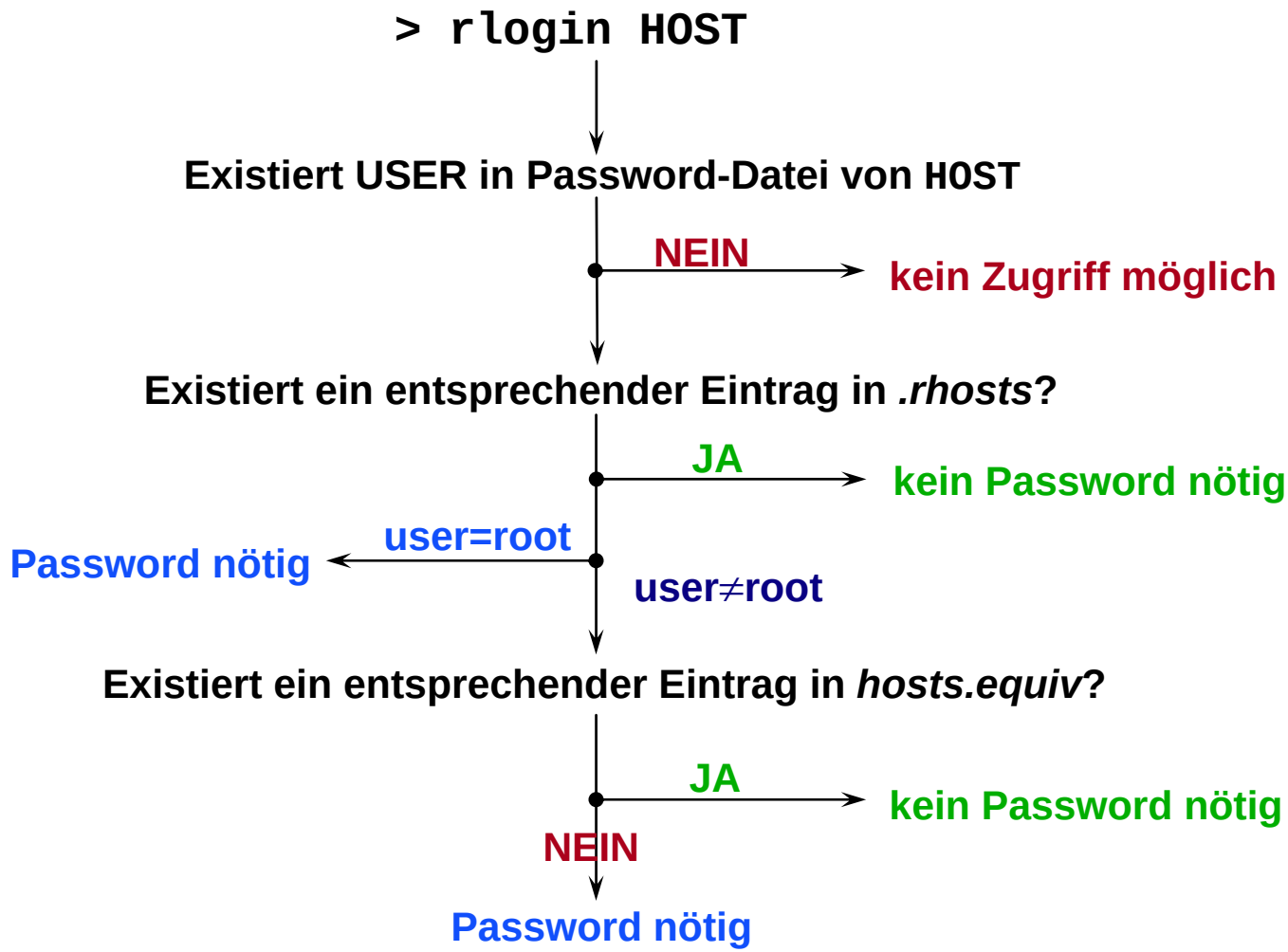
R-Utilities - Authorisierungsdateien (Einträge)

+
von jeder Maschine/ alle Benutzer von allen Maschinen

<hostname>
von der Maschine <hostname> mit eigener Kennung

<hostname><username>
angegebener <username> von <hostname> unter eigener Kennung/ allen Kennungen

R-Utilities - Ablaufdiagramm für Zugriff



Kapitel 19

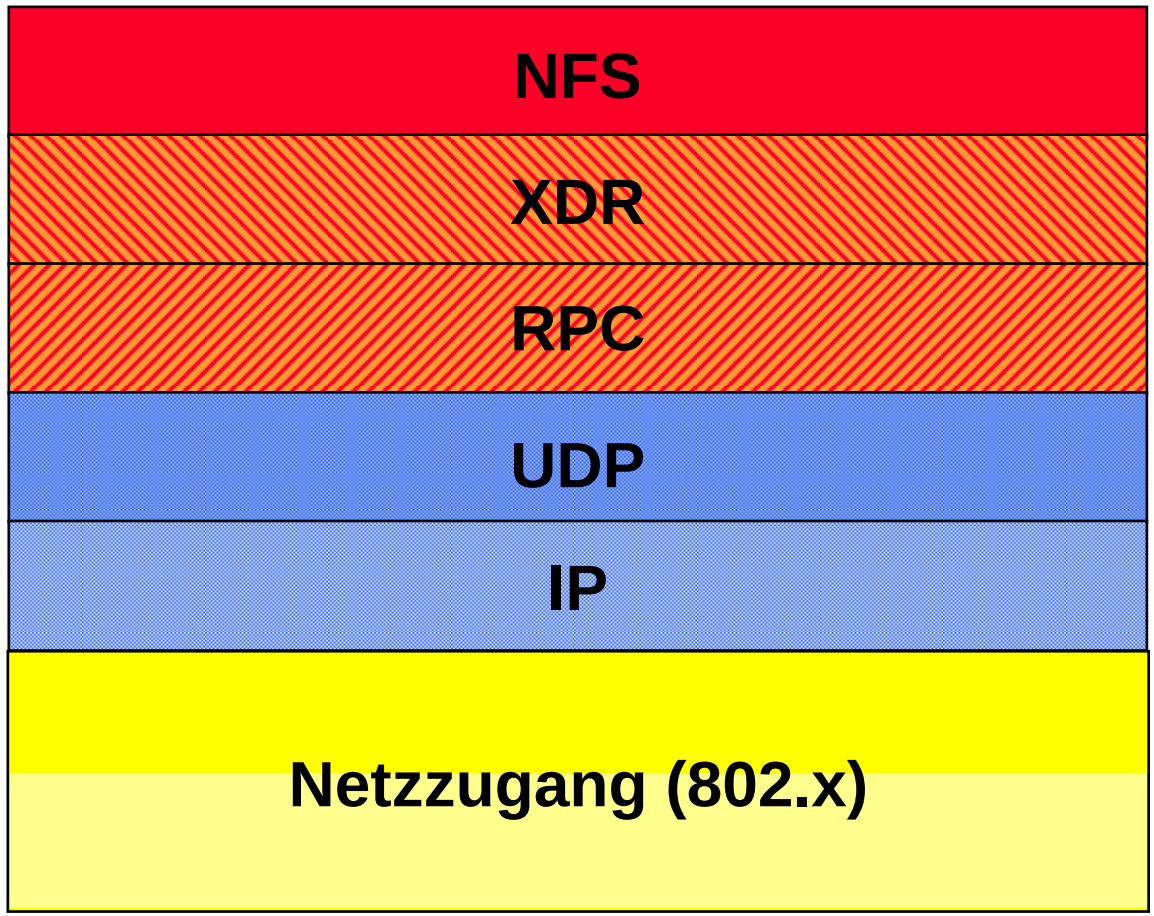
Network File System (NFS)

Network File System (NFS)

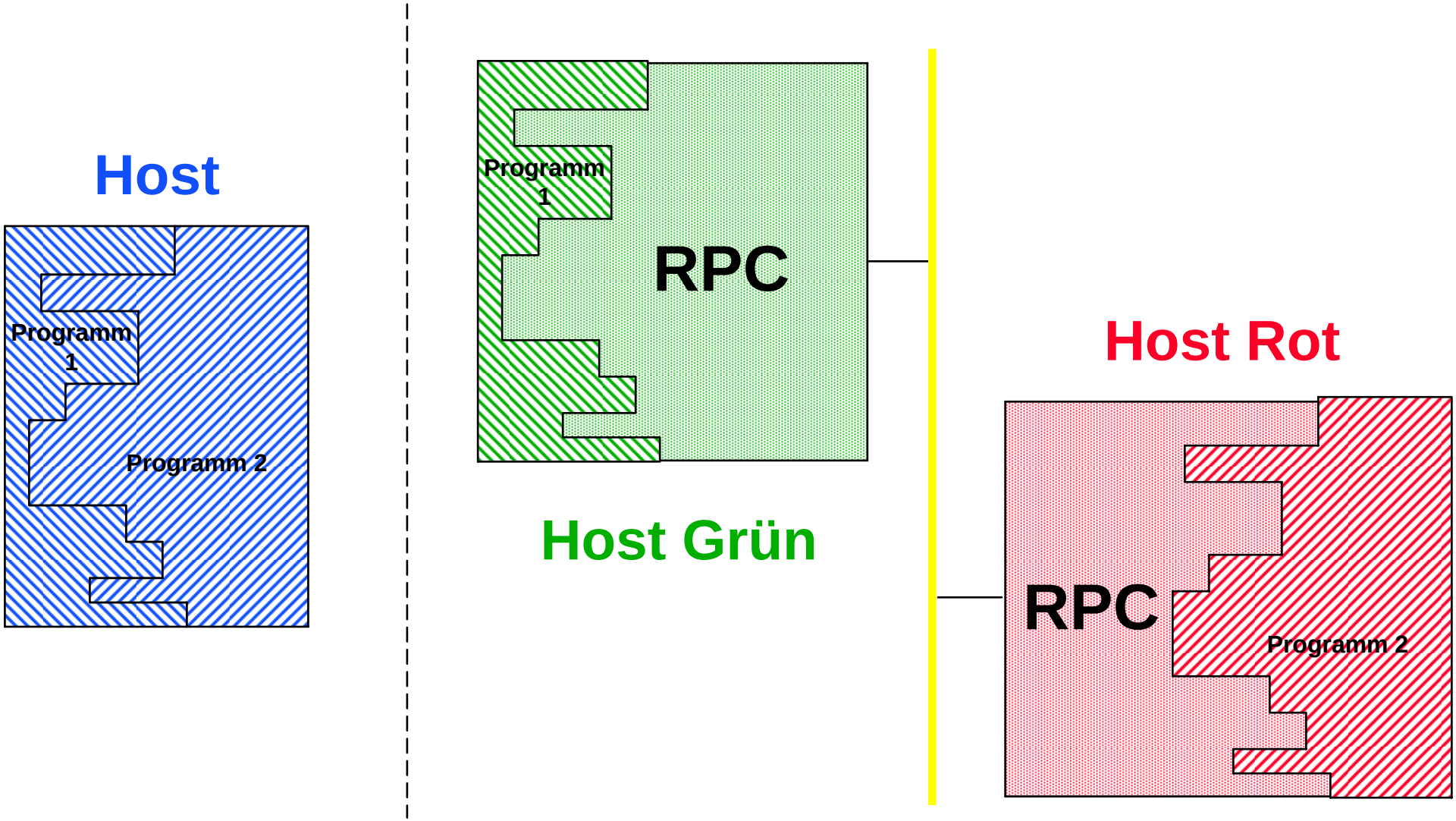
RFC 3010

- **kein MIL-Standard**
- **setzt auf XDR (“External Data Representation” - RFC 1014/ RFC 1832) und den SUN-RPCs (RFC 1057) auf**
- **UDP/TCP Port 111 (RPC)**
- **erlaubt den Zugriff auf ein “Netzwerklaufwerk” mit 80% der lokalen Performance**
- **ist eine “stateless” Verbindung**
- **explizite Freigabe auf dem Server** (/etc/exports)

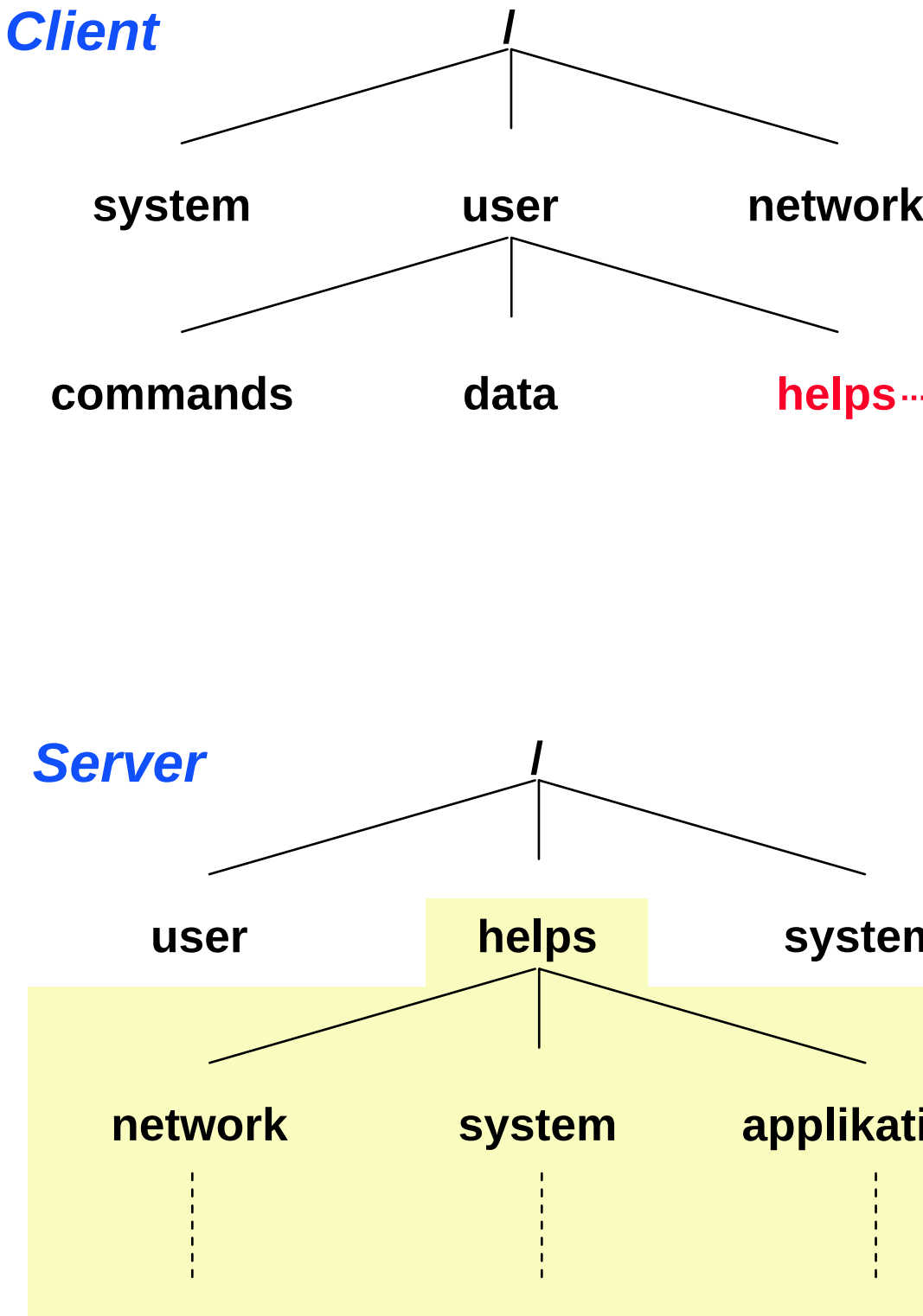
NFS im OSI-Modell



NFS - Remote Procedure Calls



NFS - Mount



Befehl:

```
mount -t nfs -o ro,soft server:/helps /user/helps
```

Kapitel 20

Internet
und
Netzwerk-Sicherheit

Begriffserklärungen

- **Internet** Anzahl aller (öffentl.) Rechner, die IP nutzen
- **WWW** World Wide Web (Anwendung, Dienst)
- **HTML** Hypertext Markup Language („Programmier“sprache)
- **HTTP** Hypertext Transfer Protocol (Dienst, Protokoll)
- **URI** Unified Ressource Identifier (Verweis auf Dokument)
- **URL** Unified Ressource Locator
  (Verweis auf Dokument incl. Protokollangabe)

  Bsp.: http://home.t-online.de/home/gerhard.glaser

World Wide Web (WWW)

- **1989: Ursprung in einem Projekt des CERN, Genf**
  war gedacht als einfaches System zur Kommunikation zwischen Physikern (Nutzung von Hypertextdokumenten)
- **1990: zeilenorientierte Oberfläche (Line-Mode-Browser)**
- **1993: Browser mit grafischer Benutzeroberfläche**
- **1994: die W3-Organisation (www.w3.org) wird ins Leben gerufen; Aufgabe: Weiterentwicklung und Standardisierung**

➔**Vorteil der WWW-Oberfläche:**
Integration vieler traditionelle Internet-Dienste
(FTP, News, Gopher etc.)

Hypertext Transfer Protocol (HTTP)

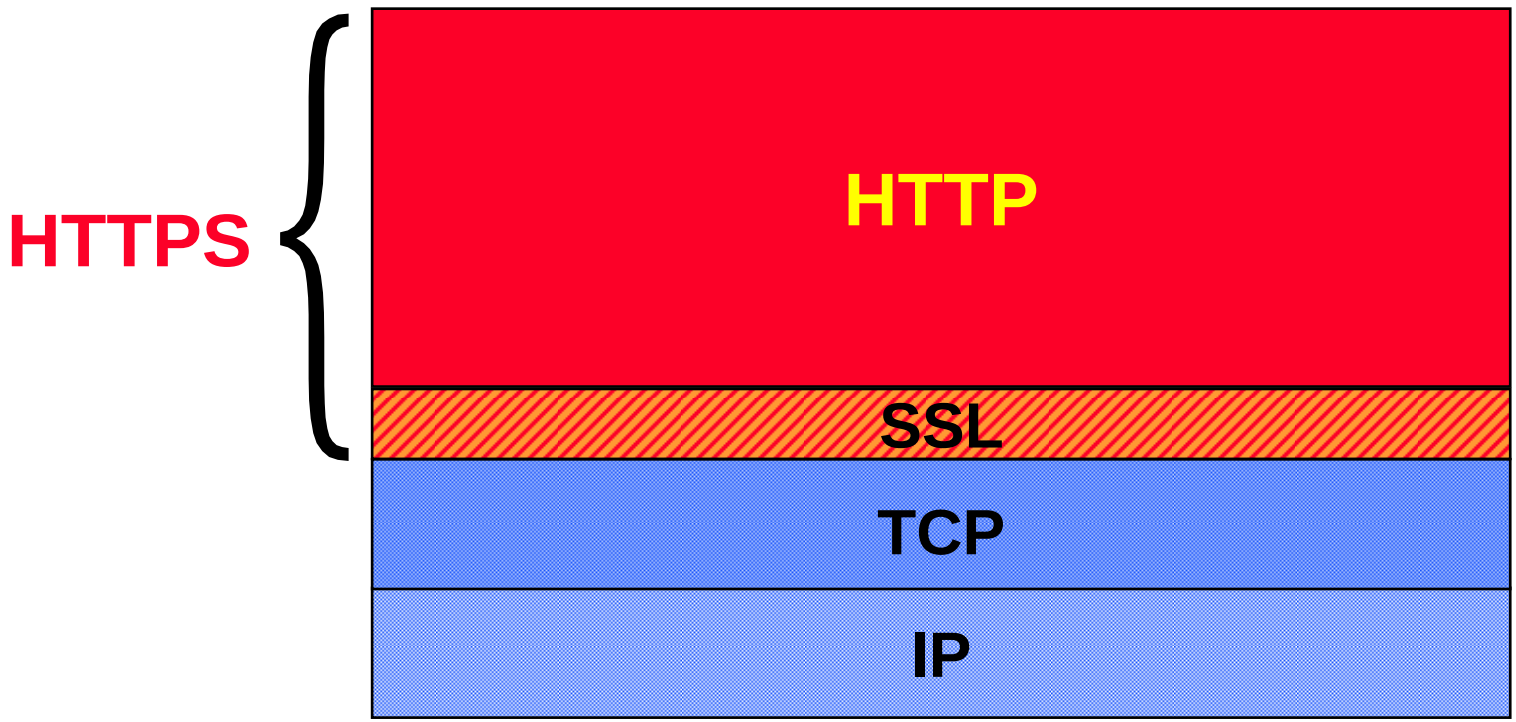
RFC 1945 HTTP 1.0 (1996)
RFC 2616 HTTP 1.1 (1997)

- **setzt auf dem gesicherten Transport Service von TCP auf**
- **TCP-Port 80 (veränderbar)**
- **basiert auf einem Request-/ Response-Verfahren zur Abfrage von Dokumenten**

❶ Verbindungsaufbau
❷ **Anforderung (Request)**
URI, protocol version, request modifier, client information
❸ **Antwort (Response)**
message protocol version, success-/ error-code, server information, “data”
❹ Verbindungsabbau

HTTPS (Secure HTTP)

- **Nutzt SSL** (Secure Socket Layer) **für verschlüsseltes HTTP (Port-Nr. 443)**
- **Verschlüsselung erfolgt über Zertifikate** (werden im Browser abgelegt)



HTTP Status Codes (Überblick)

1xx: Informational
Request received, continuing process

2xx: Success
The action was successfully received, understood, and accepted

3xx: Redirection
Further action must be taken in order to complete the request

4xx: Client Error
The request contains bad syntax or cannot be fulfilled

5xx: Server Error
The server failed to fulfill an apparently valid request

HTTP Status Codes - nach RFC 2616 - (1)

100 Continue
101 Switching Protocols

200 OK
201 Created
202 Accepted
203 Non-Authoritative Information
204 No Content
205 Reset Content
206 Partial Content

300 Multiple Choices
301 Moved Permanently
302 Found
303 See Other
304 Not Modified
305 Use Proxy
307 Temporary Redirect

HTTP Status Codes - nach RFC 2616 - (2)

400	**Bad Request**
401	Unauthorized
402	Payment Required
403	Forbidden
404	Not Found
405	Method Not Allowed
406	Not Acceptable
407	Proxy Authentication Required
408	Request Time-out
409	Conflict
410	Gone
411	Length Required
412	Precondition Failed
413	Request Entity Too Large
414	Request-URI Too Large
415	Unsupported Media Type
416	Requested range not satisfiable
417	Expectation Failed

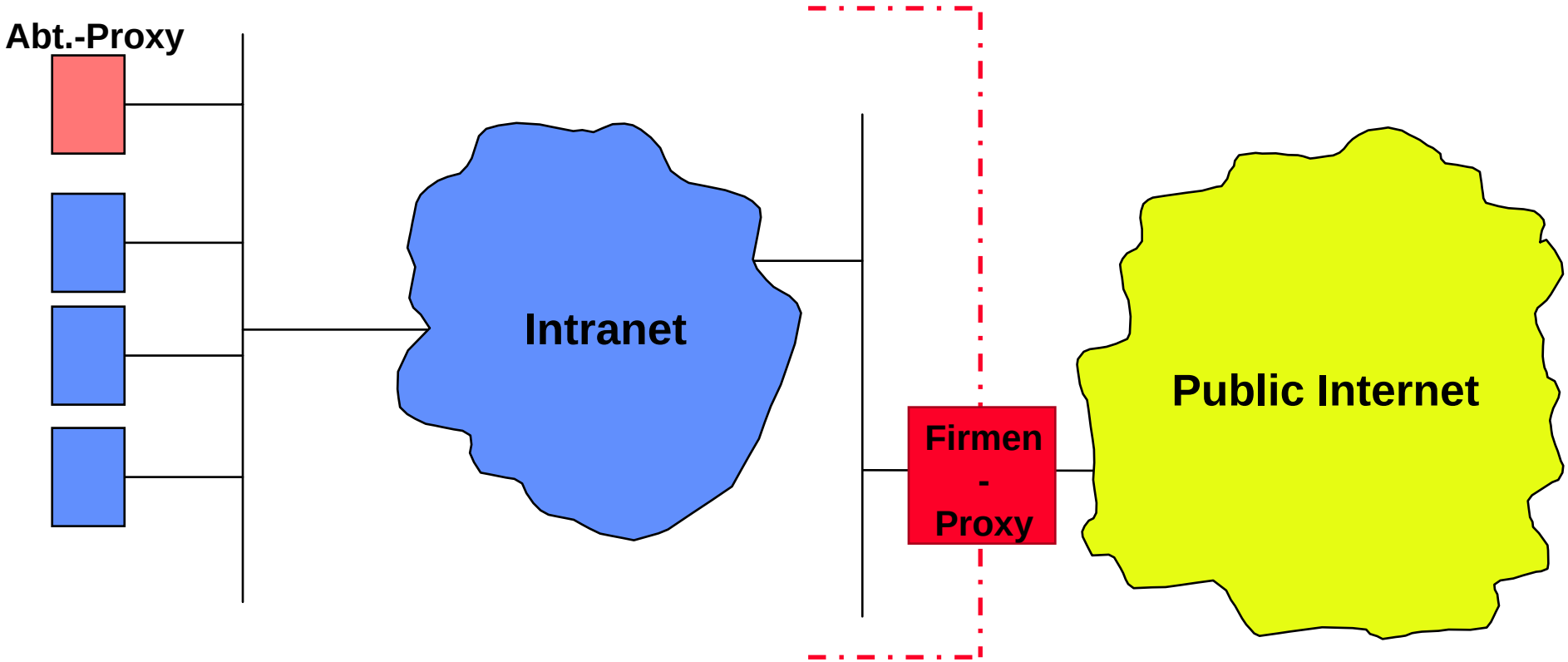
HTTP Status Codes - nach RFC 2616 - (3)

500	**Internal Server Error**
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Time-out
505	HTTP Version not supported

Proxy-Server

- Ein zwischengeschaltetes Programm (Rechner), das sowohl als Client als auch als Server arbeitet
- Anfragen werden bearbeitet ggf. übersetzt und dann weitergereicht
- Proxies dienen dazu, Zugriffe zu steuern (“Firewall”) und Anfragen für nicht unterstützte Protokolle weiterzureichen
- Die physikalische Ausprägung (Rechner) verfügt i.a. noch über eine Caching-Funktionalität
- Können kaskadiert werden

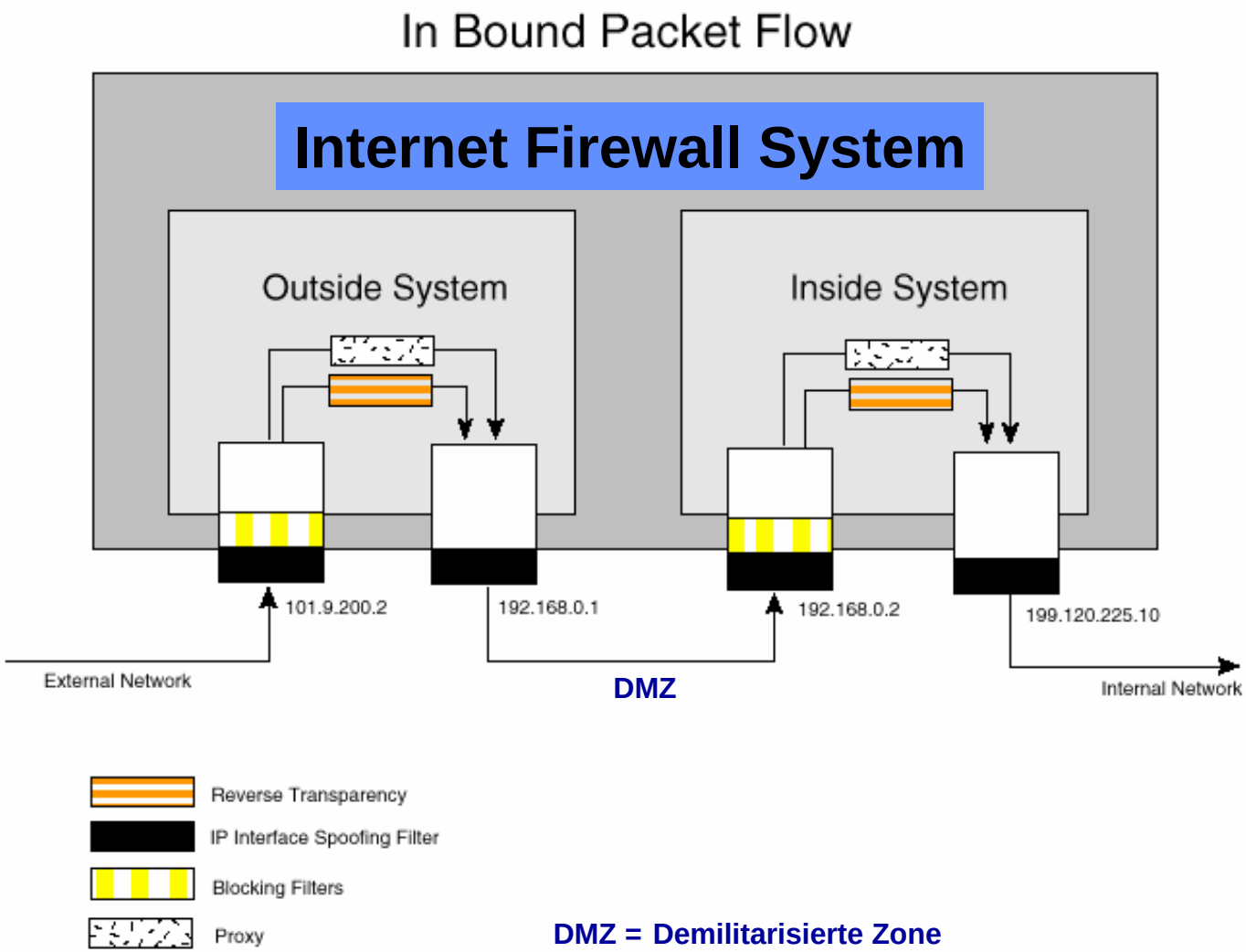
Proxy-Server



Socks-Server
(vs. Proxy-Server)

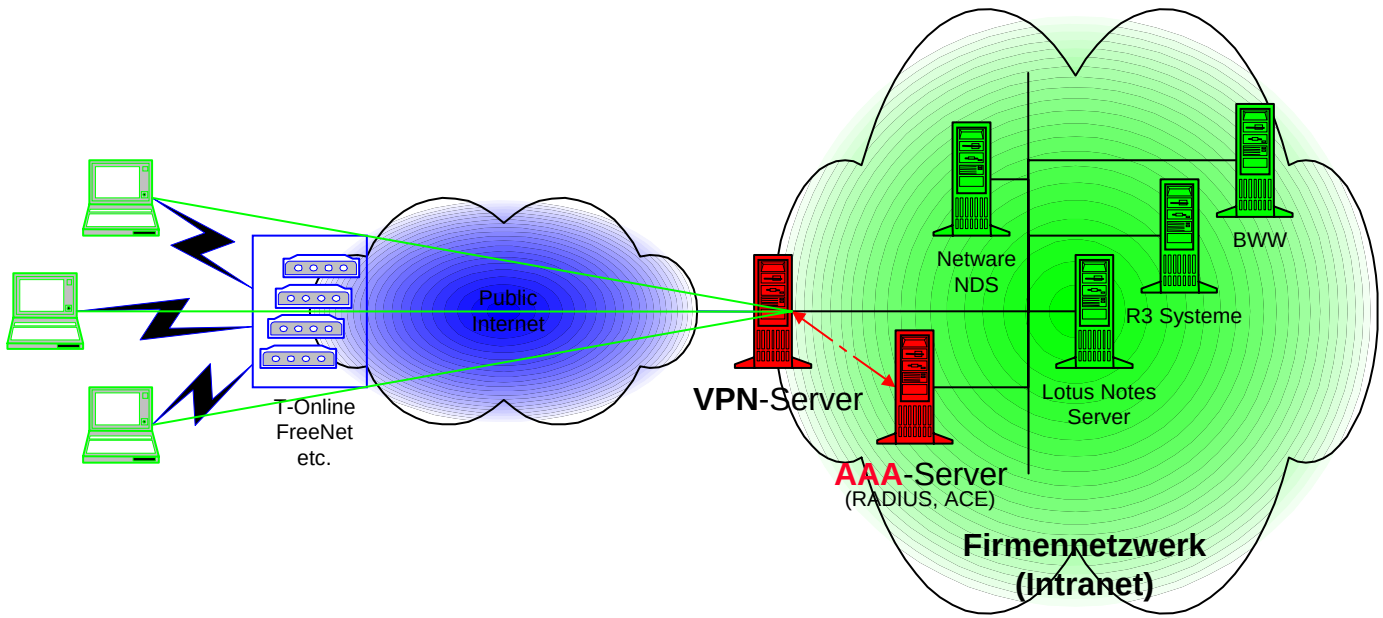
- **nutzt TCP/UDP Port 1080 für alle Dienste („Tunnel“)**
  (Proxy nutzt Port des Dienstes - z.B. Port 80)
- **Dienst/ Anwendung muss „socksifiziert“ sein**
  (Dienst/ Anwendung unterstützt normalerweise Proxy-Funktion - transparenter Proxy möglich)
- **Socks muss Anwendung nicht unterstützen**
  (Proxy muss Anwendung unterstützen)
- **Anwender ist für Socks freigeschaltet - oder nicht**
  (Proxy kann separat für jeden Dienst freigeschaltet werden)

Firewall



DMZ = Demilitarisierte Zone

Virtual Private Networks (VPN)



AAA-Server: Authentication, Authorization, Accounting

Tunneling Protokolle (Auswahl)

- **IPsec**
- PPTP (Microsoft „alt“)
- **L2TP keine Verschlüsselung**
  - ➲ **IPsec secured L2TP (L2TP/ IPsec)** (Microsoft „neu“)

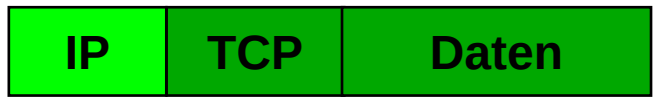
- **SSL (neu!)**

IPsec - Eigenschaften

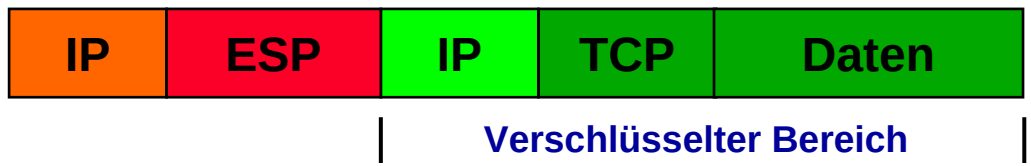
- **Layer 3**
- **Paketintegrität (Hash-Based Message Authentication Code)**
- **Paketauthentifizierung** („Abfallprodukt“ des HMAC)
- **Paketverschlüsselung**
- **Schutz vor Replay-Angriffen**
- **IP-Tunneling (ausschließlich IP)**
- **Schlüsselmanagement (IKE)**

IPsec - Tunnel- und Transport-Modus

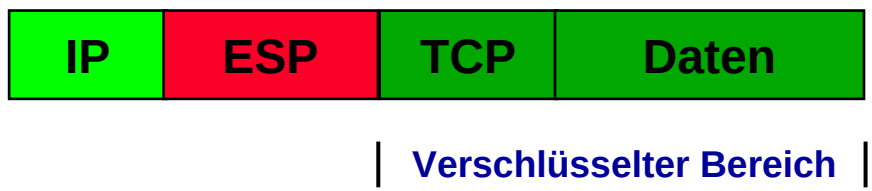
Original-Paket



IPsec-Tunnel-Modus

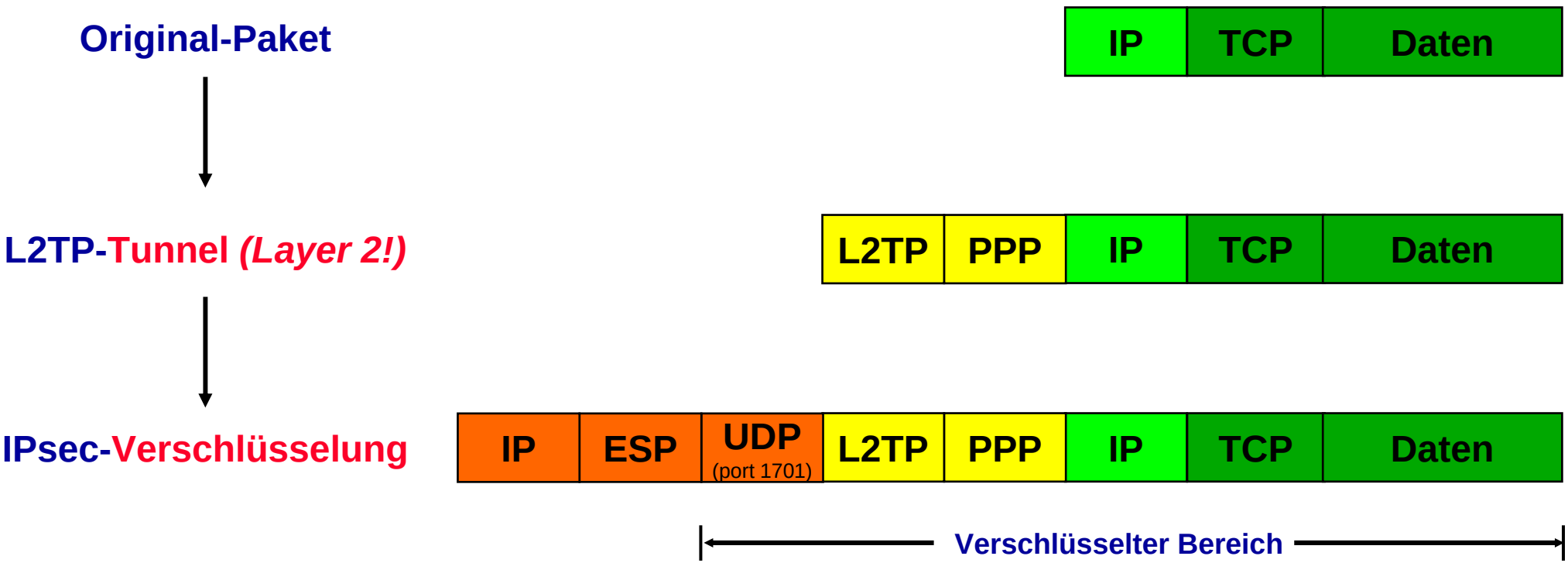


IPsec-Transport-Modus



IPsec secured L2TP (L2TP/ IPsec)

RFC 3193
Microsoft Knowledge Base: Q265112



Kapitel 21

Simple Network Management Protocol (SNMP)

Netzwerk-Management Funktionen nach OSF/ EMA
(Enterprise Management Architecture)

- **Configuration Management (“Change Management”)**
- **Fault Management**
- **Performance Management**
- **Security Management**
- **Accounting Management**

Simple Network Management Protocol (SNMP)

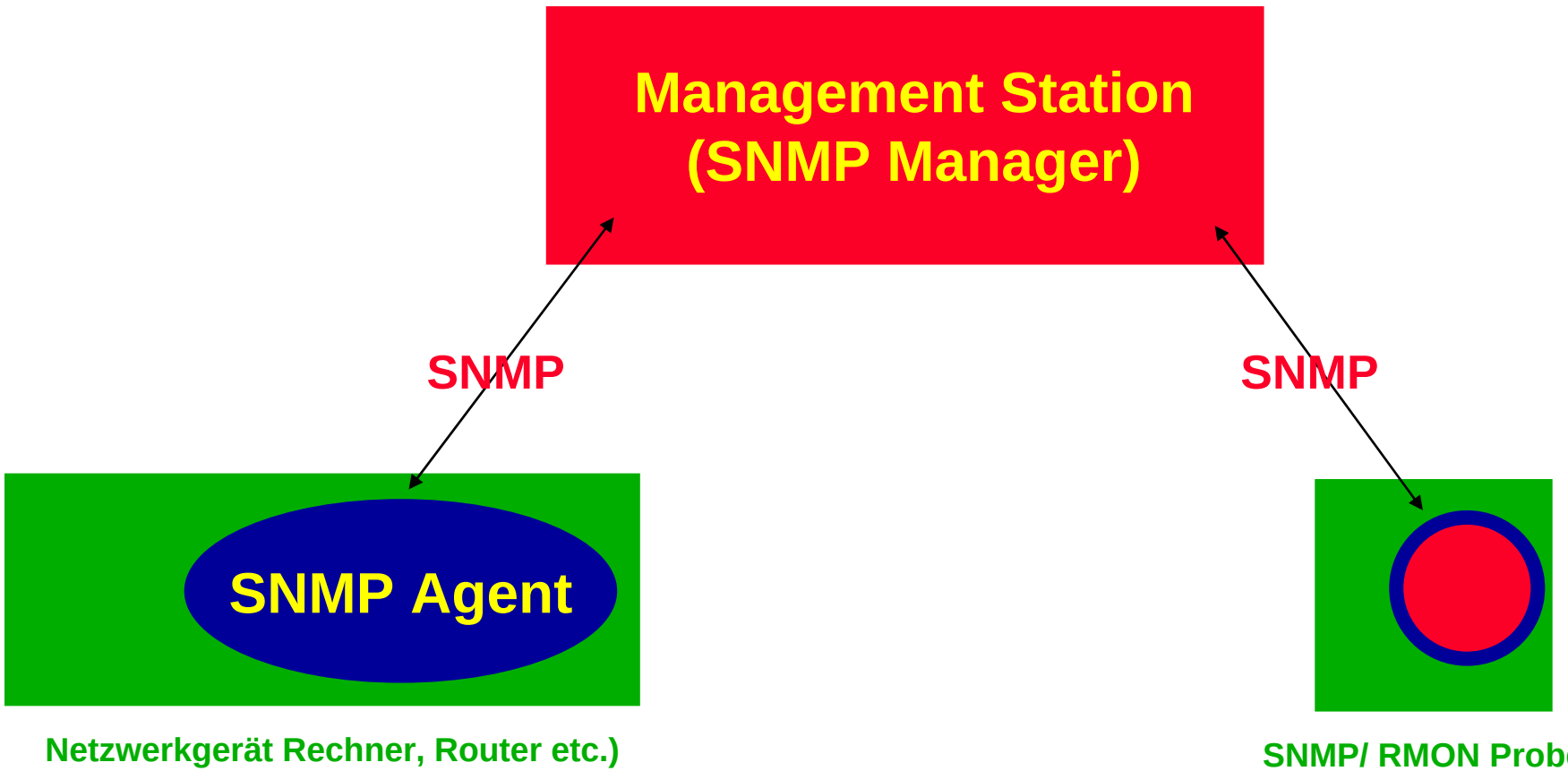
RFC 1157 - STD 15
RFC 3411 - 3418 - STD 62

- **kein MIL-Standard**
- **setzt auf dem Datagram-Transport-Service von UDP auf**
- **UDP/TCP Port 161 und 162 (für Traps)**
- **dient zur Vereinheitlichung der erfaßten Daten (Variablen) und zur Übertragung derselben**
- **erlaubt herstellerspezifische “Ergänzungen”**

SNMP - Funktionsweise

- es werden Variablen (“Managed Objects”) definiert, die hierarchisch in der “Management Information Base” (MIB) abgelegt werden
- auf den zu überwachenden Geräten läuft ein Agent, der die Informationen einsammelt
- die Informationen werden von dem Management-Server abgefragt und u.U. gesetzt
  - ➔ get *<Variable>* holt die spezifizierte Variable
  - ➔ get-next holt nächste Variable im Datenmodell
  - ➔ get-bulk holt mehrere Variablen gleichzeitig
  - ➔ set *<Variable>* setzt eine Variable
- in besonderen Fällen (Traps) werden Daten an den Management-Server gesendet
  - ➔ event *<Variable>*

SNMP – Aufbau



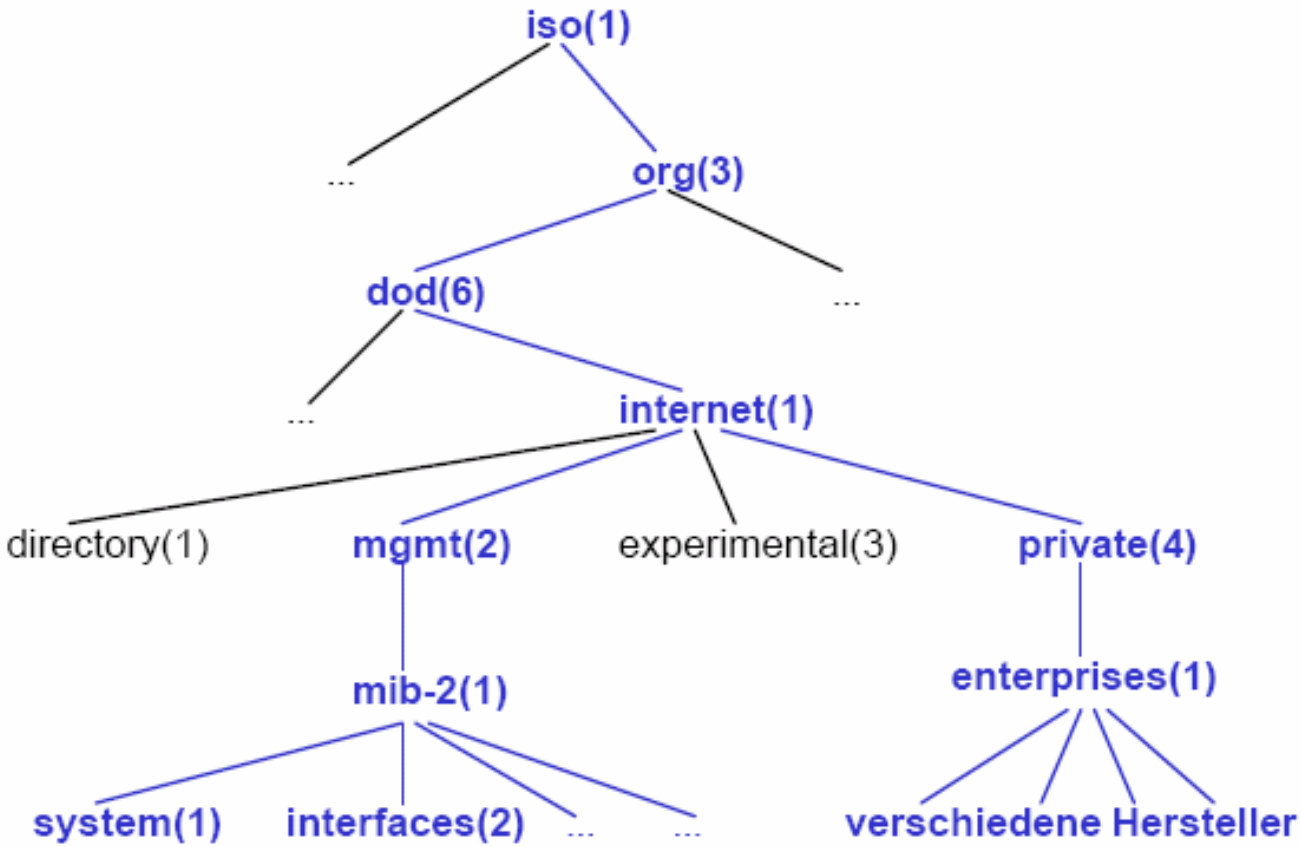
SNMP - Aufbau

- **Manager:** **Überwacht, Konfiguriert (polling:** Regel**)**
- **Agent:** **Sammelt/ ändert Daten (traps:** Ausnahme**)**

- **RMON:** ***Rem*ote *Moni*toring**
  Verlagerung der Intelligenz von Manager auf Probe

  ⇒ **Manager: zum Netz (Managed Objects) hin**
  ⇒ **Agent: zur Management-Station hin**

SNMP – Aufbau (MIB)



SNMP

wichtige RFCs (allgemeine Definitionen)

- RFC 1157 SNMP (STD0015)
- RFC 1643 Definitions of Managed Objects for the Ethernet-like Interface Types (STD0050)
- RFC 3411 Architecture for Describing SNMP Management Frameworks
- RFC 3412 Message Processing and Dispatching for SNMP
- RFC 3413 SNMP Applications
- RFC 3414 User-based Security Model for SNMP
- RFC 3415 View-based Acces Control Model for SNMP
- RFC 3418 Management Information Base (MIB) for SNMP
- RFC 2576 Coexistence between SNMP v1, SNMP v2 and SNMP v3

SNMP - weitere wichtige RFCs (1)

- **RFC 1213** **MIB II - STD 17**
- **RFC 2011 - 2013** Updates zu 1213
- **RFC 1515** **MAU MIB**
- **RFC 2233** **IF-Group**
- **RFC 2665** **Ether-like MIB** (Ethernet/ 802.3)
- **RFC 1694** **SMDS MIB**
- **RFC 2515** **ATM**
- **RFC 1696** **Modem MIB**
- **RFC 2127** **ISDN MIB**
- **RFC 2108** **Repeater MIB**
- **RFC 1493** **Bridge-MIB**
- **RFC 1749** **Station Source Routing MIB**
- **RFC 1850** **OSPF MIB**

SNMP - weitere wichtige RFCs (2)

- **RFC 1513 Token Ring RMON**
- **RFC 1748 Token Ring MIB**
- **RFC 1749 Station Source-Routing MIB**
- **RFC 1666 SNA NAU MIB**
- **RFC 1747 SNA SDLC MIB**
- **RFC 1559 DECnet MIB** (27.12.93)
- **RFC 1742 Apple Talk MIB**
- **RFC 2790 Host Ressources MIB**
- **RFC 2248 Network Service Monitoring MIB**
- **RFC 2789 Mail Monitoring MIB**
- **RFC 1611 DNS-Server MIB** (HISTORIC)
- **RFC 1628 UPS MIB**
- **RFC 2819 RMON MIB - STD 59**

SNMP - wichtige MIBs (1)

System MIB
(in RFC 3418)

- **Variable 1.3.6.1.2.1.1.x**
- **enthält Informationen über den Rechner (Management - Teil)**
  (für Gesamtrechner = Host-Ressources MIB - RFC 2790)
  - ➔ **Systembeschreibung**
  - ➔ **Standort**
  - ➔ **Betreuer/ Verantwortlicher**
  - ➔ **“UpTime”**

SNMP - wichtige MIBs (2)

Interface Table (in RFC 1213)

- **Variable 1.3.6.1.2.1.2.2.1.x**
- **enthält Informationen über das/ die Interfaces**
  - ➔ **Anzahl**
  - ➔ **Typ** (z.B. X.25, Ethernet, 802.3, 802.5, FDDI, PPP, ISDN etc.)
  - ➔ **MTU**
  - ➔ **Geschwindigkeit**
  - ➔ **Physical Address**
  - ➔ **Status (administrativ/ operational)**

SNMP - wichtige MIBs (3)

Address Translation Table (in RFC 1213)

- **Variable 1.3.6.1.2.1.3.1.1.1 - 3**
- **enthält Informationen über den ARP-Table**
  - ➔ **Art des Eintrags** (statisch/ dynamisch)
  - ➔ **Physical Address**
  - ➔ **Net Address (z.B. IP-Adresse)**

SNMP - wichtige MIBs (4)

IP - MIB
(in RFC 1213)

- **Variable 1.3.6.1.2.1.4.x**
- **enthält Informationen über das IP-Protokoll**
  - ➔ **Routing (ja/ nein)**
  - ➔ **Default TTL**
  - ➔ **Subnet-Maske**
  - ➔ **Broadcast-Adresse**
  - ➔ **Routing Tabelle**
  - ➔ **Routing Maske**
  - ➔ **Next Hop**
  - ➔ **Errors etc.**

SNMP - wichtige MIBs (5)

Übersicht

- **icmp**
- **tcp**
- **udp**
- **transmission** (ca. 575 Variablen!)
- **snmp**
- **ospf**
- **privat** (z.B.)
  - ➔ **bay networks/ wellfleet**
  - ➔ **cisco**
  - ➔ **3com**
  - ➔ **fore**
  - ➔ **novell**
  - ➔ **qms**

Kapitel 22

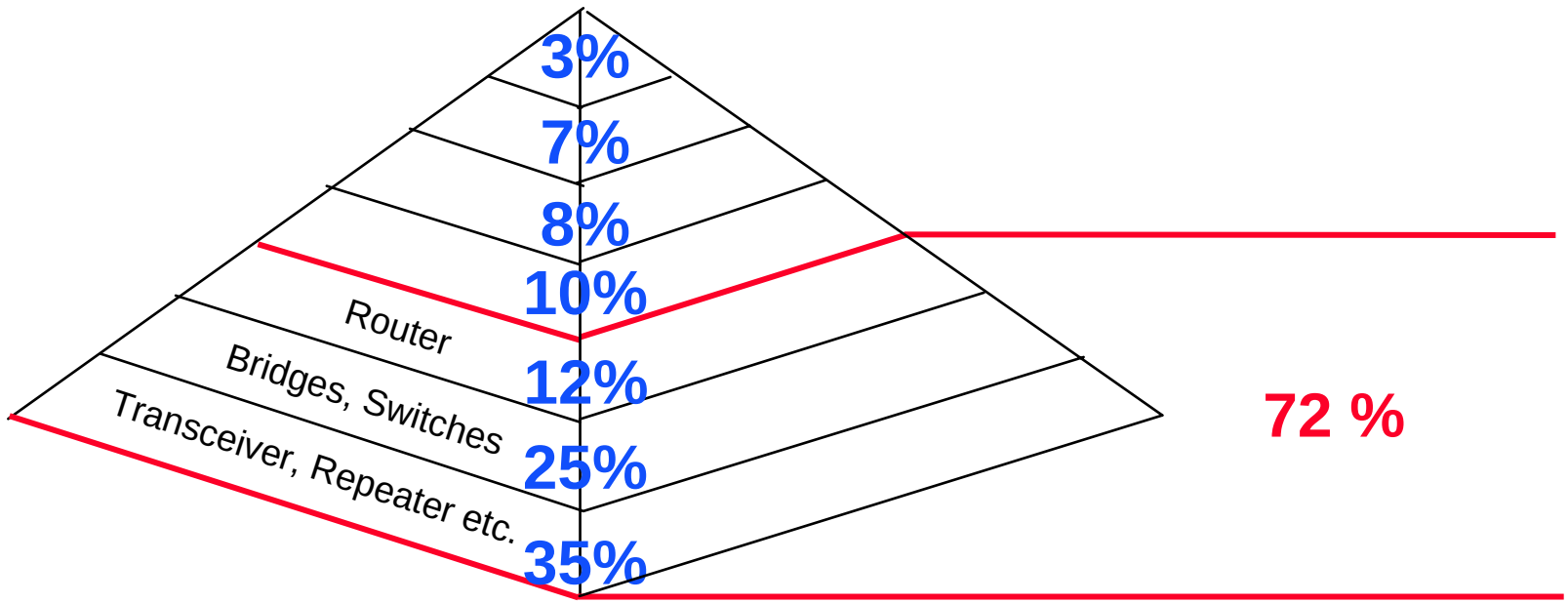
Trouble-Shooting

Trouble-Shooting

- **Wichtige Quelle:**

  ➔ **RFC 2151 (Juni 1997):**
  „A Primer On Internet and TCP/IP Tools and Utilities“

Fehlerursachen



Layer 1: fehlerhafte Kabel; elektrische Störungen; Kollisionen etc.
Layer 2: 802.x-Inkompatibilitäten; falsch konfigurierte Hardware-Adressen; Broadcaststorms
Layer 3: falsch konfigurierte IP-Adressen, Subnetzmasken und Broadcast-Adressen; falsche Routing-Tabellen/ Einträge; Protokollinkompatibilitäten
Layer 4 - 7: unkorrekt implementierte Protokolle

Trouble-Shooting - „eingebaute“ Tools/ Befehle (1)

- **arp** — **Zeigt/ modifiziert den ARP-Cache**
  - ➔ -a — Darstellen aller Einträge
  - ➔ -d — Löschen von Einträgen
  - ➔ -s — Setzen von Einträgen
    -s PUB — Antwortet auf Anfragen

Trouble-Shooting - „eingebaute“ Tools/ Befehle (2)

- **netstat** **Zeigt eine (Karten-) Statistik**
  - ➔ -a alle Verbindungen
  - ➔ -e Ebene 2 (Ethernet-) Statistik
  - ➔ -p [Protokoll] über TCP oder UDP
  - ➔ **-r** **Routingtable**
  - ➔ -s Statistik (ausführlich)
  - ➔ interval [sec] automatischer Update (in sec)

Trouble-Shooting - „eingebaute“ Tools/ Befehle (3)

- **route** **Zeigt/ modifiziert die Routingtabelle und routingspezifische Einträge**

Syntax: route [*command* [dest.] [MASK netmask] [GW]]

➔ *command*	PRINT
	ADD
	DELETE
	CHANGE
➔ dest.	Zieladresse für die der Eintrag gelten soll
➔ MASK	Subnetzmaske
➔ GW	Gateway (Router) für dest.

Trouble-Shooting - „eingebaute“ Tools/ Befehle (4)

- **ping** **Testet die Erreichbarkeit eines IP-Rechners**

➔	-t		unbegrenzt
➔	-n	*<count>*	Anzahl von Pings
➔	-l	*<size>*	Paketgröße (Vorsicht!)
➔	-f		don’t fragment
➔	-i	*<TTL>*	TTL-Wert setzen/ vorgeben
➔	-v	*<TOS>*	TOS-Wert setzen
➔	-j	*<host-list>*	Loose Source Routing
➔	-k	*<host-list>*	Strict Source Routing
➔	*-w*	*<timeout>*	Wartezeit in ms
➔	-R		Trace Route (nicht Windows-Betriebssysteme)

Trouble-Shooting - „eingebaute“ Tools/ Befehle (5)

- **Tracert** **Trace Route*)**

Syntax: tracert [-d] [-h *max_hops*] [-j *host-list*] [-w *ms*] Name

- ➔ -d keine Hostnamen (nur IP-Adressen) anzeigen
- ➔ -h TTL-Feld
- ➔ -j Loose Source Routing
- ➔ -w Time Out (in ms)

*) nur Windows Betriebssysteme

Probe vs. Analyzer

- fest installiert
- auf eine Topologie festgelegt - nur “Punktmessungen”
- Daten müssen über das Netz übertragen werden
- bei Ausfall des Netzes keine Überwachung möglich bzw. nur “outbound”
- benötigt Managementstation (muss konfiguriert werden)
- nutzt (proprietäre!) MIBs
- kann permanent verschiedene wichtige Segmente “remote” überwachen

- unabhängig von Standort
- ist für verschiedene Netztopologien (gleichzeitig) einsetzbar - Messen von Topologieübergängen möglich
- Daten werden lokal bearbeitet und gespeichert
- funktioniert unabhängig von Netz
- ist unabhängig von einer Managementstation (sofort einsetzbar)
- großer Funktionsumfang
- i.a. physische Anwesenheit notwendig (u.U. sinnvoll beim Troubleshooting)

Probe vs. Analyzer (Fazit)

- **Probe** zur ständigen (Fern-) Überwachung des Netzwerkes, zur Erstellung von Statistiken zur Netzwerkplanung und zur "Früherkennung" von Fehlern

  Zum Erweitern des "Horizontes" eines (proprietären) Managementsystems

- **Analysator** zur Fehlerbehebung vor Ort bzw. zum Ermitteln von Statistiken, wenn Übertragung über das Netzwerk nicht mehr möglich ist.

  Zum Messen von Verzögerungszeiten (**latency**) bei Bridges, Routern etc.

Wichtige Eigenschaften von Analysatoren (1)

- **auf das Einsatzgebiet abgestimmte Hardware (Achtung bei reinen “Software-Analysatoren”!)**
- **schnelle Hard- und Software (max. Anz. Paketen/s)**
- **genügend Filter**
  - **➔ Adressen von Layer 2 und Layer 3**
  - **➔ beliebige Byte-/ Bitkombinationen**
  - **➔ Umwandeln von “vom Netz gefischten” Paketen**
  - **➔ einfach konfigurierbar**
- **verfügbare Protokollfamilien (nachrüstbar/ selbst erweiterbar)**
- **“Postprocessing”!**

Wichtige Eigenschaften von Analysatoren (2)

- **Suchmöglichkeiten im Puffer (z.B. “Strings” an beliebiger Stelle)**
- **Speichermöglichkeit von ausgewählten Paketen**
- **Generieren von Paketen (im Hintergrund, mit Anzeige!)**
  - **➔ Zufalls-Pakete („random“)**
  - **➔ fehlerhafte Pakete (falsche FCS)**
  - **➔ Umwandeln von “vom Netz gefischten” Paketen**
  - **➔ selbst generierte Pakete**
- **aussagekräftige Statistiken (mit wählbarer Integrationszeit)**

Wichtige Eigenschaften von Analysatoren (3)

- einfache (Macro-) Programmiersprache
- Einfache Bedienbarkeit (über Funktionstasten) - ohne Handbuch!!!
- gut ablesbare Anzeige (farbig?!)
- Dual-Port ?!
- Ausbaubarkeit (Hardware und Software)
- (proprietäres) Datenformat/ Exportfilter

Kostenlose Tools

- **Ethereal**
  Netzwerk-Protokoll-Analyzer für Windows
  Download: http://www.ethereal.com/

- **Nmap**
  Der Portscanner (UNIX, LINUX, DOS)
  Download: http://www.insecure.org/nmap/

- **Advanced Portscanner** bzw. **Advanced LAN-Scanner**
  Windows-Portscanner
  Downloads:
  http://www.radmin.com/radmin/utility/pscanner.php/ bzw. http://www.radmin.com/radmin/utility/lscan.php

- **Neotrace**
  Grafisches Traceroute-Frontend für Windows
  Download: http://www.zdnet.de/downloads/prg/e/0/de0DE0-wc.html

Wichtige Adressen im Internet

IANA (Internet Assigned Numbers Authority):	**www.iana.org**
Assigned Numbers:	**www.iana.org/numbers.html**
Wichtige Organisationen:	**www.iana.org/implinks.htm**
IPv4 Address Space:	**www.iana.org/assignments/ipv4-address-space**
RFCs (RFC Editor):	**www.rfc-editor.org/**
RFCs (Direktaufruf):	**ftp.isi.edu/in-notes/rfc*<Nr.>*.txt**
Internet Standards (STD) :	**ftp.rfc-editor.org/in-notes/std/std1.txt**
Official Internet Protocol Standards:	**www.rfc-editor.org/rfcxx00.html**
DE-NIC:	**www.denic.de/**
IPv6:	**www.computermethods.com/ipng/**
802-Standards:	**standards.ieee.org/catalog/802info.html**

Literatur

Hein, Mathias: TCP/IP im Einsatz - mitp (Datacom)
ISBN 3-8266-4094-2

Hunt, Craig: TCP/ IP Netzwerk- Administration - O'Reilly,
ISBN 3-8972-1110-6

Doyle, Jeff: Routing TCP/IP - Markt und Technik (Cisco Press - CCIE #1919)
ISBN 3-8272-533-3

Dittler, Hans Peter: IPv6 - das neue Internet Protokoll - dpunkt-Verlag;
ISBN 3-932588-18-5

Lipp, Manfred: VPN - Virtuelle Private Netzwerke - Addison-Wesley
ISBN 3-8273-1749-5

Tanenbaum, Andrew S. - Computer Networks *(engl.)* - Prentice Hall
ISBN 0-13-066102-3