

Grundlagen der Informatik für Ingenieure I

7. Rechnernetze - Einführung

7.1 Geschichte

7.2 Standardprotokolle

7.3 Struktur und Funktionsweise

7.3.1 Hardware

7.3.2 Struktur

7.3.3 Client-Server-Modell

7.3.4 Adressierung und Routing

7.4 Protocol-Layers

7.5 Dienste / Höhere Protokolle etc.

7.6 WWW

7.7 Sonstiges

7.1 Geschichte

7.1.1 Entwicklung

- Initiative 1968/69:
Forschungsbehörde im US-Verteidigungsministerium - ARPA
((Defense) Advanced Research Project Agency)
 - ◆ Universitäten entwickeln ein Computer-Netz (ARPANET)
 - ◆ 1970: 4 Computer verbunden
1971: 13, 1972: 23, ..., 1976: 63
- 80er-Jahre:
 - ◆ Aufteilung in militärisches und wissenschaftliches Netz
 - ◆ neue Technik um mehrere Computer-Netze miteinander zu verbinden
 - Internet
 - ◆ neue Netze von Forschungsorganisationen, NASA, ...
 - ◆ Netze in Europa
 - ◆ 1981: 213 Computer, 1984: 1024, 1989: 159.000

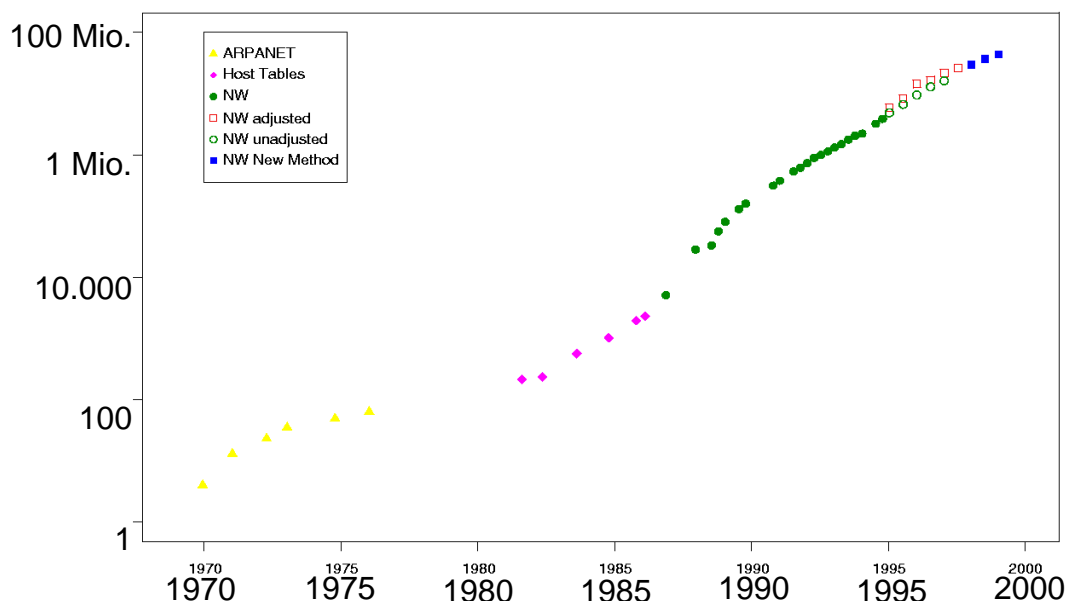
7.1.1 Entwicklung (2)

■ 90er-Jahre

- ◆ Internet bislang nur zu Forschungszwecken
- ◆ Basis-Software für das World Wide Web wird entwickelt
- ◆ erste kommerzielle Netzwerke mit Internet-Technologie
- ◆ Unternehmen schließen sich an
- ◆ große Zahl kommerzieller Netz-Provider kommt auf den Markt
- ◆ zunehmend für Privathaushalte interessant
- ◆ Zusammenwachsen von Telefon, Rundfunk, Fernsehen und Computernetzen beginnt
- ◆ neue Anwendungsbereiche für Internet-Technologie (z. B. Haustechnik)
- ◆ 1990: 300.000 Computer, 1995: 6 Mio., 2000: 100 Mio. ?

7.1.2 Wachstum

Rechner im Internet 1969 - 1999



★ 1 Milliarde WWW-Seiten durch Suchmaschinen indiziert (geschätzt: 550 Mrd ex. tatsächlich)

7.1.3 Geschwindigkeit

■ typische Geschwindigkeit von Internetverbindungen

	Weitverkehr	lokale Netze
1969:	10 kBit/s	
1975:	50 kBit/s	100 kBit/s
1980:	250 kBit/s	10 MBit/s
1985:	1 MBit/s	34 MBit/s
1990:	2 MBit/s	100 MBit/s
1999:	6 GBit/s	1 GBit/s

7.1.4 Anwendungen

- 70er-Jahre: Electronic Mail
Dateitransfer
interaktiver Zugang zu Rechnern
- 80er-Jahre: News (schwarze Bretter, Diskussionsforen)
frei zugängliche Dateiarchive
- 1990/1991: Software zur automatischen Suche in Dateiarchiven
Elektronische Bibliothekskataloge
- 1992/1993: Software für World Wide Web entsteht
- 1994: WWW-Datenmenge wächst um Faktor 1.600
- ab 1995: Übertragung von Bildern und Ton
Videokonferenzen

7.2 Standardprotokolle

■ Spezifiziert in RFCs (Request for Comment)

<http://www.faqs.org/rfcs/rfc-index.html>

◆ UDP	(User Datagram Protokoll)	RFC 768	August	1980	4 Seiten
◆ IP	(Internet Protokoll)	RFC 791	September	1981	51 Seiten
◆ ICMP	(Itn. Contr. Message Prot.)	RFC 792	September	1981	22 Seiten
◆ TCP	(Transmission Contol Prot.)	RFC 793	September	1981	85 Seiten
◆ ARP	(Adress Resolution Prot.)	RFC 826	November	1982	10 Seiten
◆ RARP	(Reversed Adress ...)	RFC 903	Juni	1984	4 Seiten

7.3 Struktur und Funktionsweise

7.3.1 Hardware

■ Medien

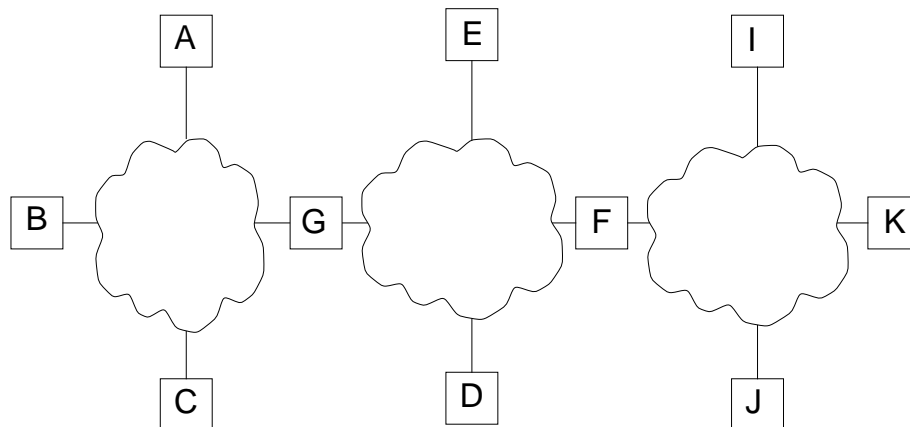
- ◆ Ethernet (10/100 MBit/s, 50-500m)
Bus-System mit CSMA/CD
(Carrier Sense Multipel Access/Collision Detection)
Gigabit-Ethernet über Glasfaser
- ◆ Token Ring (4/16 MBit/s, 150-300m zw. Repeatern); (IBM)
- ◆ FDDI (100 MBit/s, ca. 100km; Fiber Distributed-Data Interface)
- ◆ ATM (155/622/+ MBit/s, variabel; Asynchronous Transfer Mode)

■ Verbindung von Netzen

- ◆ Gateways / Router / Switches

7.3.2 Struktur

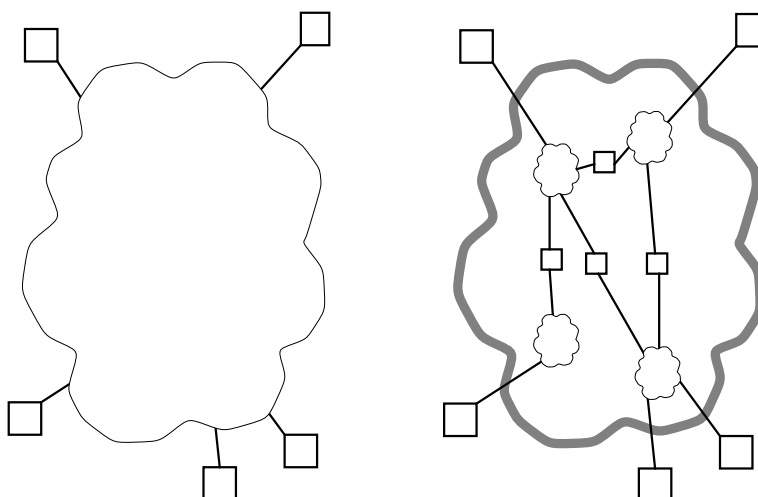
- Internet = Verbund vieler Einzelnetze zu einem virtuellen Gesamtnetz
- Beispiel: Verknüpfung von drei Netzwerken



- G und F sind Gateways

7.3.2 Struktur (2)

- Benutzersicht Systemsicht



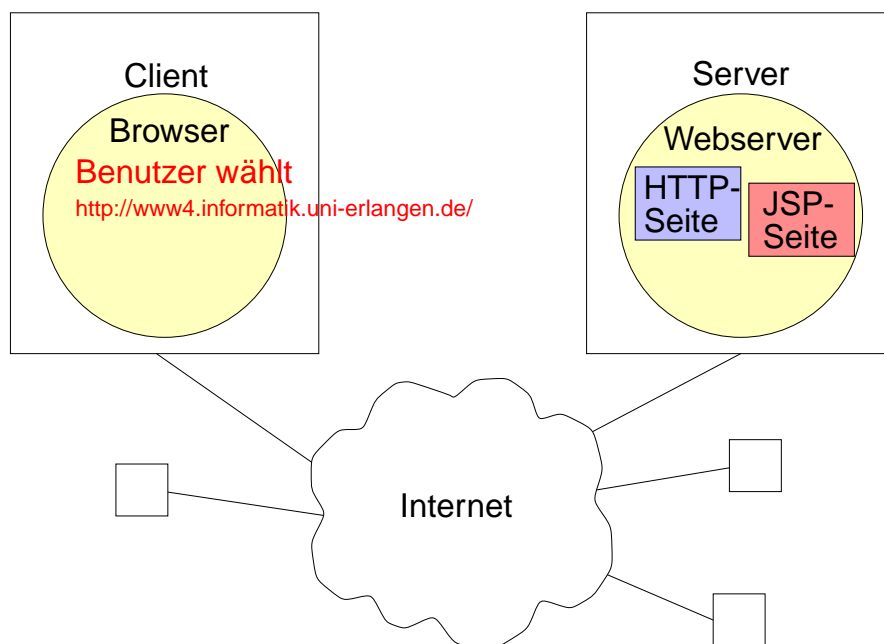
- ★ Für "normale" Anwendungsprogrammierung reicht Benutzersicht
- ★ Verstehen von Sicherheitsproblematiken erfordert Detailwissen [\(+ Administration\)](#)

7.3.3 Client-Server-Modell

- ★ Server: Programm, das einen Dienst (Service) anbietet, der über einen Kommunikationsmechanismus erreichbar ist
- Server
 - ◆ Akzeptieren Anforderungen anderer Rechner
 - ◆ Führen angebotenen Dienst aus
 - ◆ Schicken das Ergebnis zurück zum Sender der Anforderung
 - ◆ Server sind normalerweise als normale Benutzerprozesse realisiert
- Client
 - ◆ Ein Programm wird ein Client, sobald es
 - eine Anforderung an einen Server schickt und
 - auf eine Antwort wartet
- ★ Client-Server-Modell im Internet weit verbreitet

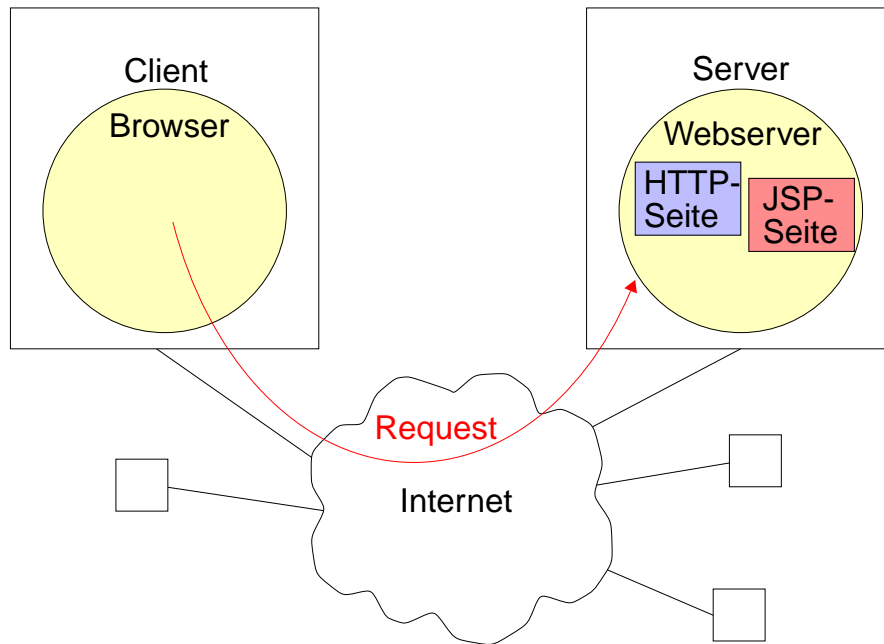
7.3.3 Client-Server-Modell (2)

- Beispiel: Webserver



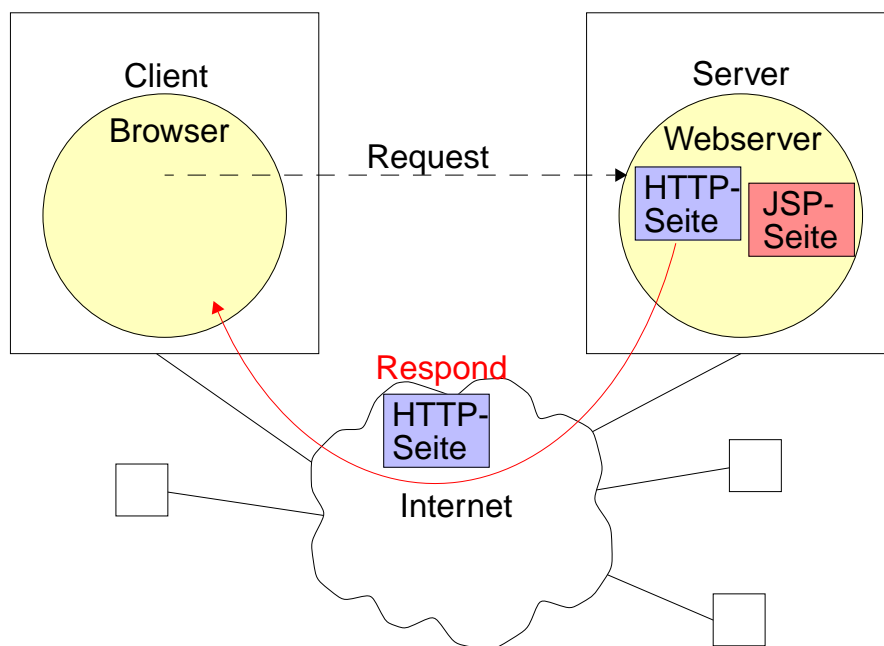
7.3.3 Client-Server-Modell (2)

■ Beispiel: Webserver



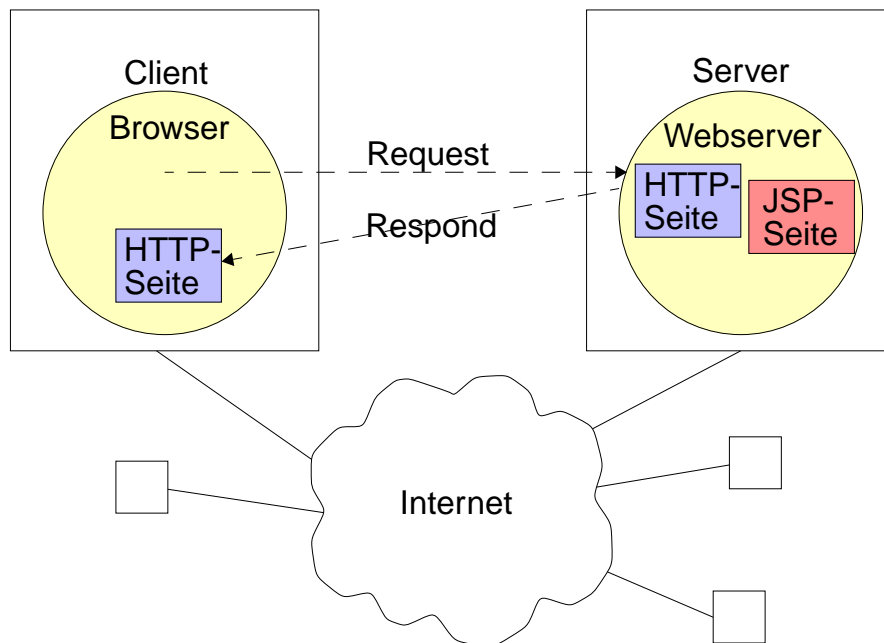
7.3.3 Client-Server-Modell (2)

■ Beispiel: Webserver



7.3.3 Client-Server-Modell (2)

■ Beispiel: Webserver

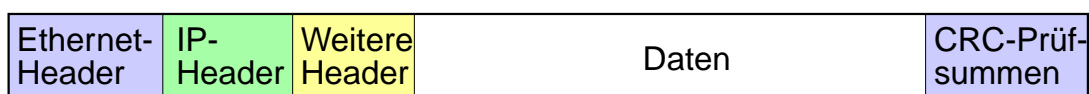


7.3.4 Adressierung und Routing

■ Internet Protocol – IP

- ◆ Unabhängig von verwendeter Hardware
- ◆ Ermöglicht einheitliche Adressierung aller Rechner
- ◆ Bildet virtuelles Netzwerk auf der Basis vieler physischer Netzwerke
- ◆ Paketorientiert: IP-Datagramm
- ◆ Unzuverlässige Datenübertragung
- ◆ Routing-Konzepte (mittels ICMP)

■ Realisierung:



7.3.4 Adressierung und Routing (2)

■ Aufbau von Ethernet- und IP-Header

0	16	31
Präambel (0 - 31)		
Präambel (32 - 63)		
Ziel-Adresse (0 - 31)		
Ziel-Adresse (32 - 47)		Quell-Adresse (0 - 15)
Quell-Adresse (16 - 47)		
Paket-Typ	Version (0x40), Kopflänge (5)	Diensttyp
Paketlänge (in Bytes)	Fragmentidentifikator	
Fragmentadresse	Max. Zahl an 'hops'	Protokoll
Prüfsumme	IP-Adresse der Quelle (0 - 15)	
IP-Adresse der Quelle (16 - 31)	IP-Adresse des Ziels (0 - 15)	
IP-Adresse des Ziels (16 - 31)	Optionen	
Optionen	Füllbyte	Daten (Bytes 0 - 1)
Daten (Bytes 2 - 5)		
...		
CRC-Prüfsumme		

- IP-Paket ist in Ethernet-Frame eingebettet

7.3.4 Adressierung und Routing (3)

■ IP-Adressen

- ◆ 4 Bytes: **a.b.c.d** (unterteilt in Netzwerk-/Rechneradresse)
 - a < 128: **a** = Netzwerk, **b.c.d** = Rechner (Class A Net)
große Organisationen
 - 127 < a < 192: **a.b** = Netzwerk, **c.d** = Rechner (Class B Net)
Organisationen mittlerer Größe
 - a > 191: **a.b.c** = Netzwerk, **d** = Rechner (Class C Net)
kleinere Organisationen

◆ Spezielle Adressen

- 127.0.0.1 Localhost
- 255 als Rechneradresse Broadcast
- a >= 224 Multicast

★ Problem: max ca. 4 Mrd Adressen möglich

- ➔ Adreßraum wird allmählich knapp
- ➔ IPv6 (128 Bit Adressen)

7.3.4 Adressierung und Routing (4)

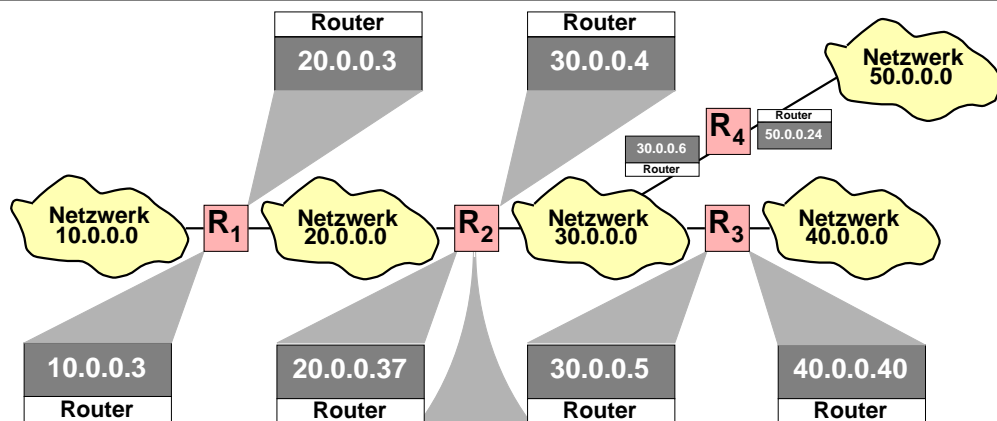
■ DNS – Domain Name System

- ◆ Abbildung von Rechnernamen auf IP-Adressen
- ◆ Erlaubt Benutzerfreundliche Namen
 - Beispiele: www.config.de
 fau40.informatik.uni-erlangen.de

■ Routing

- ◆ Finden eines Weges für ein IP-Paket von der Quelle zum Ziel
- ◆ Verwendung von Routing-Tabellen
 - Abbildung von Ziel-Netzwerk auf nächsten Rechner
 - Default Route
- ◆ 2 Arten:
 - direktes Routing
 - indirektes Routing

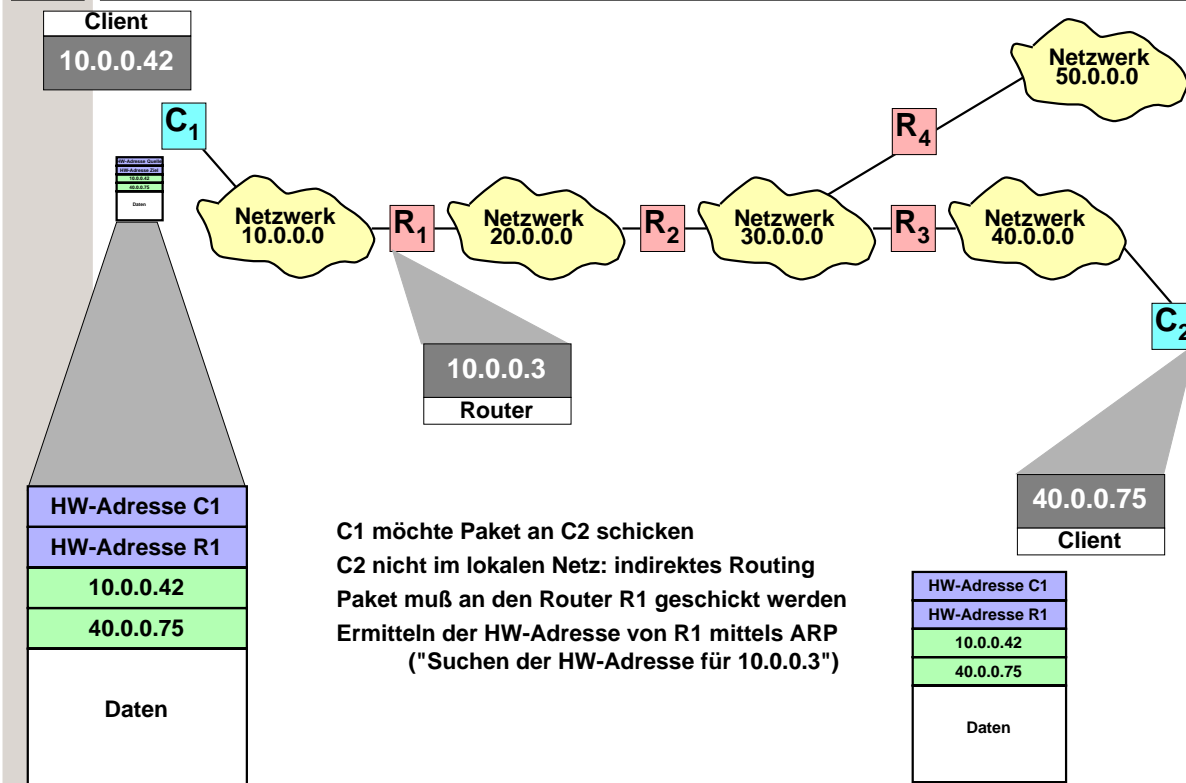
7.3.4 Adressierung und Routing (5)



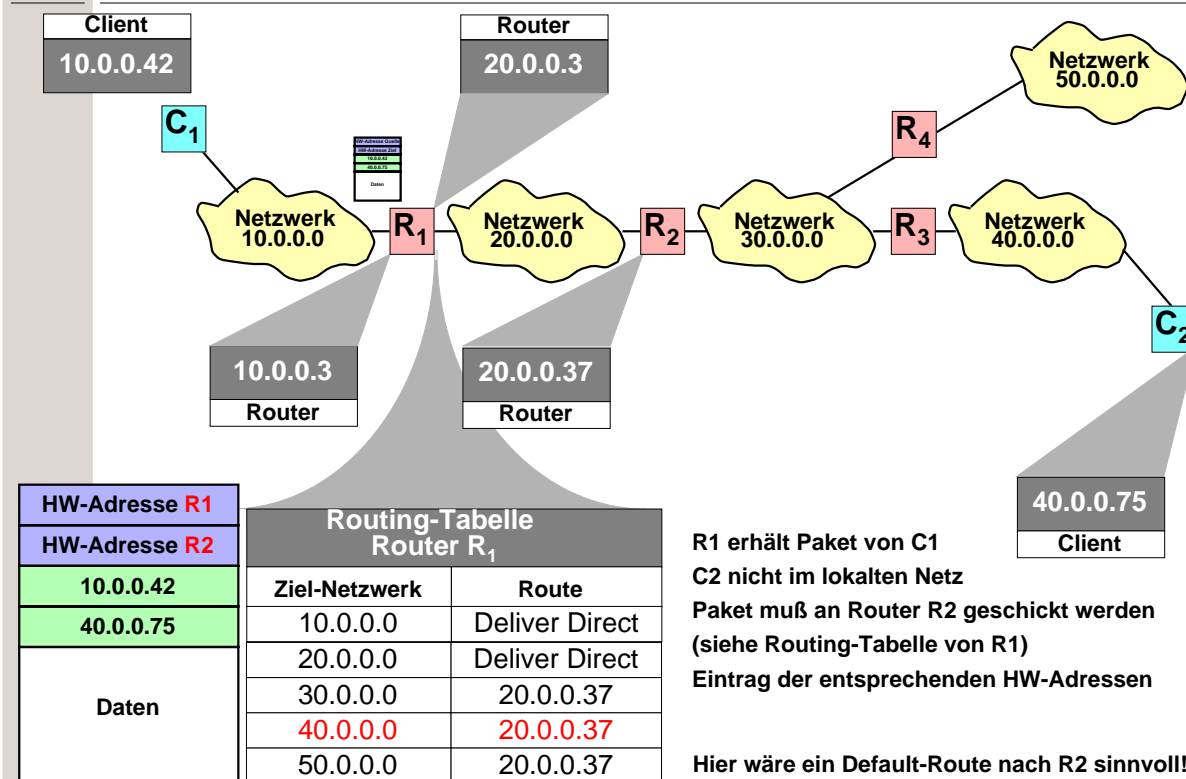
Router und Routing-Tabellen

Routing-Tabelle Router R ₂	
Ziel-Netzwerk	Route
10.0.0.0	20.0.0.3
20.0.0.0	Deliver Direct
30.0.0.0	Deliver Direct
40.0.0.0	30.0.0.5
50.0.0.0	30.0.0.6

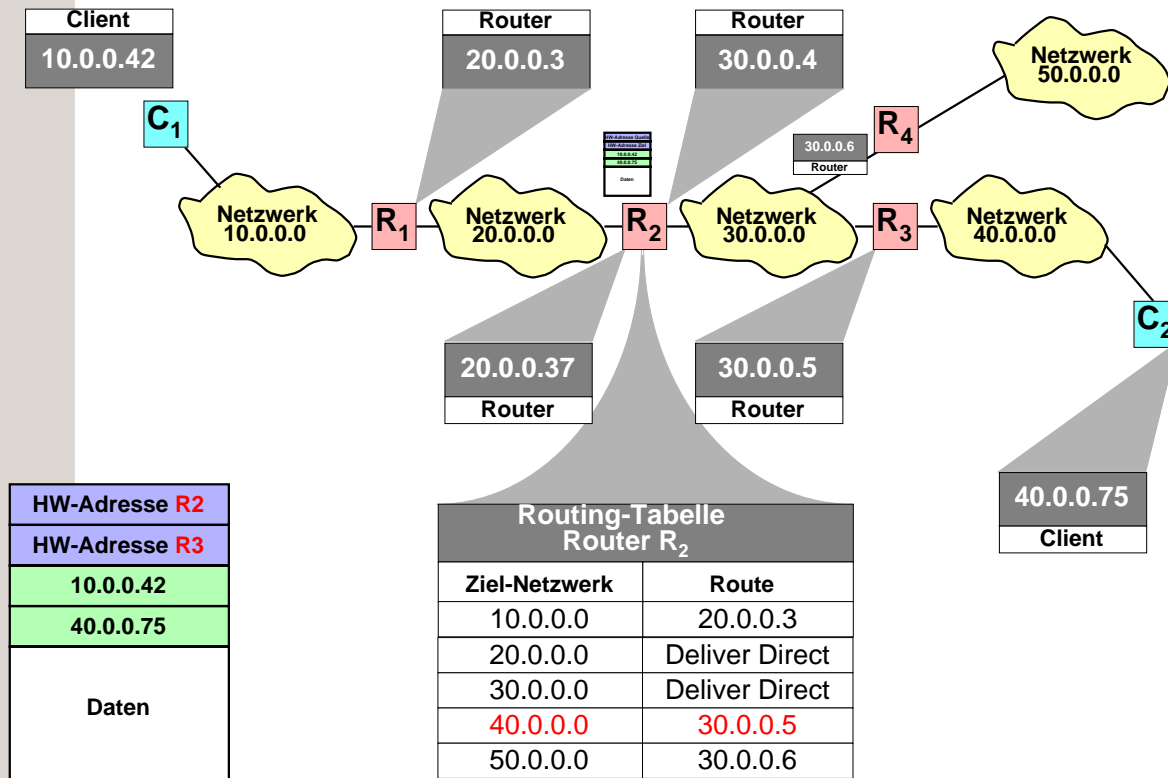
7.3.4 Adressierung und Routing (5)



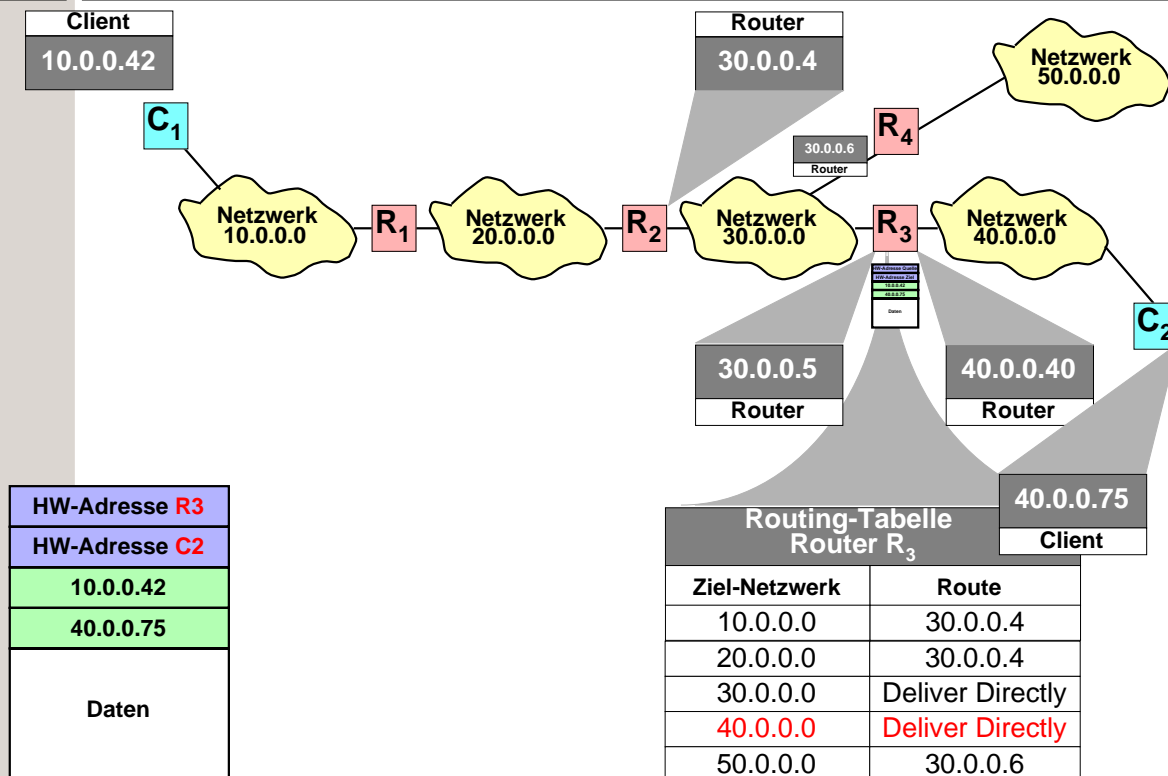
7.3.4 Adressierung und Routing (5)



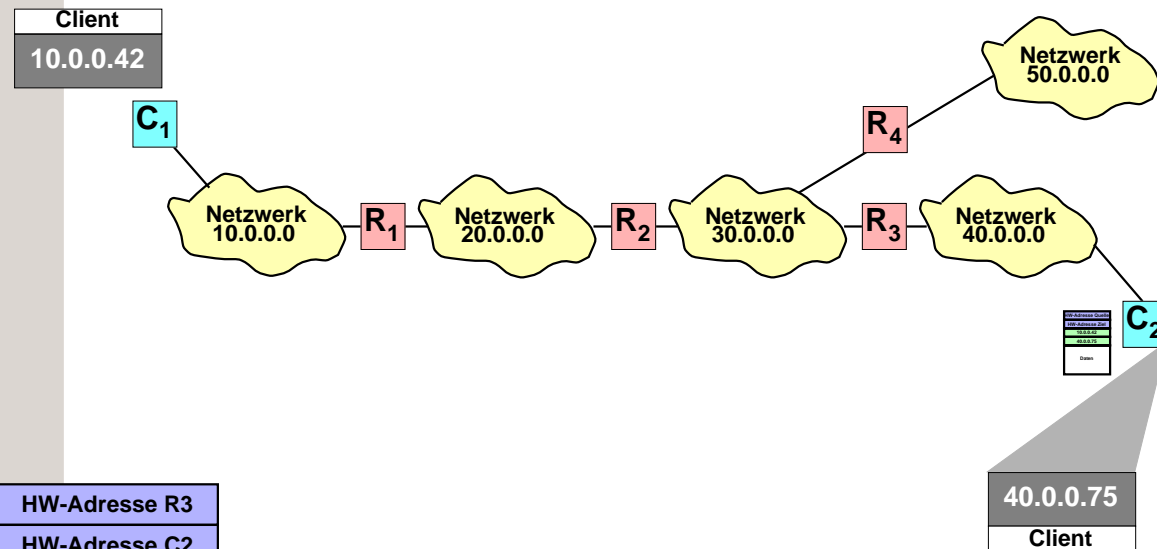
7.3.4 Adressierung und Routing (5)



7.3.4 Adressierung und Routing (5)



7.3.4.0 Adressierung und Routing (5)



HW-Adresse R3
HW-Adresse C2
10.0.0.42
40.0.0.75
Daten

7.3.4 Adressierung und Routing (6)

■ Route ansehen (UNIX):

```
tracert www4.informatik
tracert to www4.informatik.uni-erlangen.de (131.188.34.200), 30 hops max, 40 byte packets
 1  reliant.gate.uni-erlangen.de (131.188.30.8)  0.904 ms  0.981 ms  0.568 ms
 2  faui40 (131.188.30.40)  0.911 ms *  0.551 ms

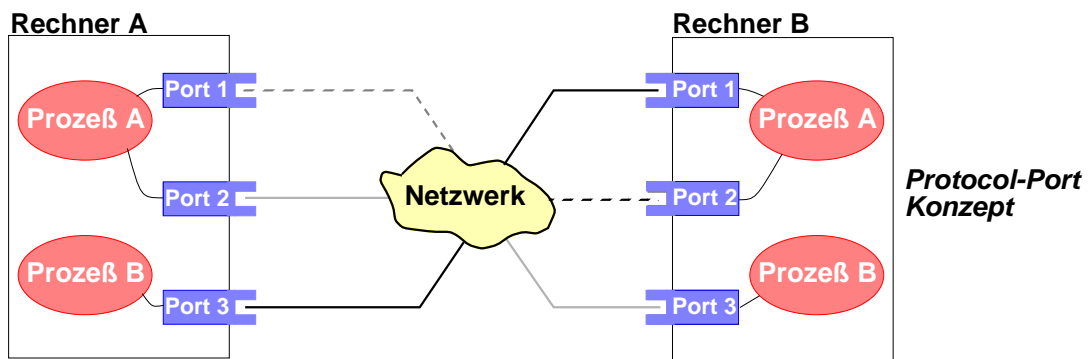
tracert www.sun.com
tracert to www.sun.com (64.124.140.181), 30 hops max, 40 byte packets
 1  reliant.gate.uni-erlangen.de (131.188.34.8)  0.718 ms  0.477 ms  0.44 ms
 2  suedstern.gate.uni-erlangen.de (131.188.21.65)  0.617 ms  0.487 ms  0.42 ms
 3  excelsior.gate.uni-erlangen.de (131.188.5.1)  3.296 ms  4.665 ms  3.265 ms
 4  ar-erlangen1.g-win.dfn.de (188.1.36.1)  0.273 ms  0.401 ms  9.783 ms
 5  cr-erlangen1.g-win.dfn.de (188.1.72.1)  0.423 ms  0.304 ms  0.326 ms
 6  cr-essen1.g-win.dfn.de (188.1.18.106)  41.563 ms  25.146 ms  24.231 ms
 7  cr-frankfurt1.g-win.dfn.de (188.1.18.89)  24.096 ms  24.031 ms  24.032 ms
 8  ir-frankfurt2.g-win.dfn.de (188.1.80.38)  41.244 ms  24.186 ms  24.298 ms
 9  ge9-0.prl.fra1.de.mfnx.net (216.200.116.97)  24.076 ms  24.181 ms  24.063 ms
10  so-0-1-0.crl.fra1.de.mfnx.net (216.200.116.213)  24.18 ms  24.204 ms  24.209 ms
11  pos9-0.crl.cdg2.fr.mfnx.net (64.125.31.161)  33.873 ms  33.924 ms  33.845 ms
12  so-5-0-0.crl.lhr3.uk.mfnx.net (64.125.31.154)  43.539 ms  43.507 ms  43.452 ms
13  so-7-0-0.crl.dca2.us.mfnx.net (64.125.31.186)  115.417 ms  115.504 ms  115.309 ms
14  so-3-0-0.mpr3.sjc2.us.mfnx.net (208.184.233.133)  181.442 ms  181.473 ms  181.46 ms
15  so-0-0-0.mpr4.sjc2.us.mfnx.net (64.125.30.2)  181.498 ms  181.497 ms  181.504 ms
16  so-1-0-0.cr2.sjc3.us.mfnx.net (208.184.233.50)  181.604 ms  181.66 ms  181.672 ms
17  pos1-0.er2a.sjc3.us.mfnx.net (208.185.175.198)  181.648 ms  181.643 ms  181.567 ms
```

- 17 "Hops" bis zum Ziel (d.h. ein Paket geht durch 16 Router)!
- Routen können sich dynamisch ändern

7.3.4 Adressierung und Routing (6)

■ Adressierung von Diensten

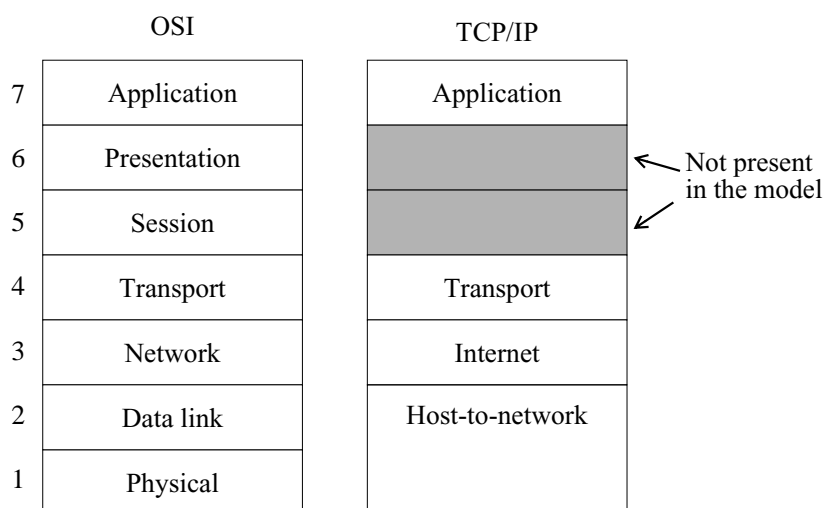
- ◆ IP: nur Adressierung von Rechnern
- ◆ Dienste / Services können über **Ports** angesprochen werden
 - Dienste verwenden normalerweise UDP oder TCP
 - ➔ UDP/TCP-Adresse = IP-Adresse + Portnummer des Dienste



- ◆ Für viele Standard-Dienste sind Portnummern festgelegt (/etc/services)

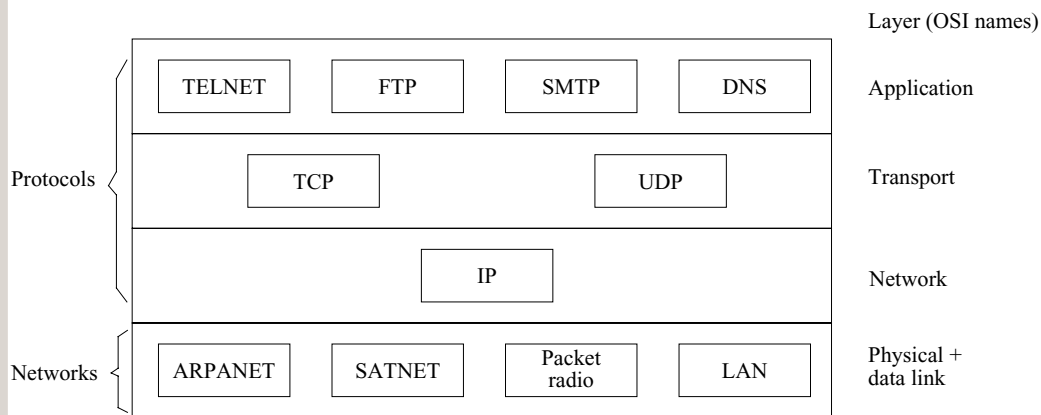
7.4 Protocol-Layers

7.4.1 OSI - TCP/IP - Reference-Modell



The TCP/IP reference model (nach Tanenbaum)

7.4.2 TCP/IP - Protocols



7.5 Dienste / Höhere Protokolle etc.

7.5.1 Grundlegende Protokolle (IP basiert)

- UDP (User Datagram Protocol)
 - ◆ Paketorientiert / Verbindungslos
 - ◆ Keine Fehlerbehandlungsmaßnahmen
 - ◆ Geeignet für Broadcast / Multicast
 - ◆ Kommunikation zwischen Ports (Kommunikationsendpunkte)
- TCP (Transmission Control Protocol)
 - ◆ Stromorientiert / Verbindungsorientiert
 - ◆ Garantiert korrekte Datenübertragung
 - ◆ Kommunikation zwischen Ports (Sockets, Mailboxen, Prozesse)
- ★ Diese Protokolle bilden die Grundlage fast aller Internet-Kommunikation

7.5.2 UDP-basiert Dienste

- NTP (Network Time Protocol)
 - ◆ Weltweite Synchronisation der Rechneruhren
 - LAN: im Bereich 1ms
 - WAN: im Bereich 10 bis 100ms
 - ◆ Updates ca. im 10-Minuten-Takt
 - ◆ Entfernung vom Zeit-Empfänger
 - Stratum-0: Empfänger selbst
 - Stratum-1: Rechner, der direkt mit Empfänger verbunden ist
 - Stratum-2: mit Stratum-1 Rechner verbunden, etc.
- TFTP (Trivial File Transfer Protocol; benutzt z.B. zum Booten)
- Syslog

7.5.3 TCP-basierte Dienste

- SMTP (Simple Mail Transfer Protocol)
 - ◆ Transportieren von E-Mails
- FTP (File Transfer Protocol)
 - ◆ Austauschen von Dateien zwischen Rechnern
- HTTP (Hypertext Transfer Protocol)
 - ◆ Übertragen von Web-Inhalten
- NFS (Network File System)
 - ◆ Transparenter Dateizugriff auf andere Rechner
 - ◆ Verwendung des RPC
- telnet (Terminal Emulation for TCP/IP Networks)
 - ◆ Einloggen in andere Rechner
 - ◆ Direkte Verbindung zu Diensten

7.5.3 TCP-basierte Dienste (2)

- rlogin, rsh, rcp (r = remote...)
 - ◆ Einloggen in andere Rechner
 - ◆ Kopieren von Daten von/zu anderen Rechnern
- SSH (Secure Shell)
 - ◆ Ersatz für rlogin, rsh, rcp
 - ◆ Starke Authentisierung
 - ◆ Verschlüsselung aller übertragenen Daten
 - ◆ Verschiedenen Algorithmen (IDEA, DES, RSA) wählbar
 - ◆ Verfügbar für Windows, Unix, Macintosh, and OS/2
- ...

7.6 WWW

- Ursprünglich statisch
 - ◆ HTML (Hypertext Markup Language)
 - ◆ HTTP (Hypertext Transfer Protocol)
- Client-Side Scripting
 - ◆ JavaScript
 - ◆ Applets
- Server-Side Scripting
 - ◆ SHTML (Server-Side Include)
 - Einfügen "statischer" Inhalte (Header, Footer, etc.)
 - ◆ CGI (Common Gateway Interface)
 - Oft zum Bearbeiten von Forms
 - Programmierung in C, C++, Perl, Java, ...

7.6 WWW (2)

- Server-Side Scripting (2)
 - ◆ PHP (Hypertext Preprocessor)
 - Mischung aus Java, C, Perl + Erweiterungen
 - Einfaches Erstellen dynamisch erzeugter Web-Inhalte
 - ◆ ASP (Active Server Page)
 - Microsoft-Spezifisch
 - Programmierung in VBScript or JScript
 - Einbettung direkt in HTML-Code
 - ◆ Servlets, JSP (Java Server Page)
 - Programmierung in Java
 - Einbettung direkt in HTML-Code
 - Aufrufe anderer Klassen, Beans etc. erlaubt

7.7 Sonstiges

- XML (Extensible Markup Language)
 - ◆ Aus SGML (Standard Generalized Markup Language) hervorgegangen
 - Dokumentenverwaltung
 - ◆ Definition eigener Tags
 - ◆ DTD (Document Type Definition)
 - legt fest, wie die XML-Tags interpretiert werden sollen
 - ◆ CSS (Cascading Style Sheets) / XSL (Extensible Style Language)
 - Trennung des Layouts vom Inhalt
 - Beschreiben, wie gewisse Elemente aussehen sollen
- PGP (Pretty Good Privacy)
- SSL (Secure Socket Layer)
- VPN (Virtual Private Network)