G.2 Einordnung in das ISO/OSI Referenzmodell

7	Anwendungsschicht	*ftp, telnet, rlogin, rsh, ...*
6	Darstellungsschicht	-
5	Sitzungsschicht	
4	Transportschicht	*TCP, UDP*
3	Netzwerkschicht	*IP*
2	Verbindungsschicht	*Ethernet*
1	physikalische Schicht	

1 Adressierung

Grundprinzipien

- einheitliche Adressierung, unabhängig von Netzwerk-Technologie (Ethernet, Token-Ring)
  - ➥ virtuelles Netz: ***Internet***
- Netzwerkeinheitlich (über Router) - transparent für Benutzer
- Aufbau einer IP-Adresse:
  - ◆ 4 Byte: **a.b.c.d**, unterteilt in Netzwerk- und 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

2 Routing (2)

Direktes Routing

- Der Absender verpackt das IP-Datagramm in ein Paket des physischen Netzes (Encapsulation)
- Die Ziel-IP-Adresse wird in eine Adresse des physischen Netzes abgebildet
- Das Paket wird mittels der Netzwerk-Hardware übertragen
- häufigstes Verfahren: **ARP** *(address resulution protocol)* (auf Broadcast-Basis)

Routing-Protokolle

- **Autonomes System** (AS): Menge von Netzen und Routern einer Administrationseinheit
(z. B. alle Netze im deutschen Wissenschaftsnetz WiN)

- Autonome Systeme tauschen Informationen über Netze und zugehörige AS-Pfade untereinander aus (*Exterior Gateway Protocol — EGP,* speziell *Border Gateway Protocol* — **BGP**)

- Innerhalb eines AS werden Routing-Informationen mit *Interior Gateway Protocols* ausgetauscht (vor allem **RIP** und **OSPF**)

2 … *Protocol-Port* Konzept (2)

- für eine Reihe von Standard-Diensten (route-Protokoll, rwho, ...) sind Portnummern festgelegt (UNIX → Datei **`/etc/services`**)
- UDP agiert als Multiplexer/Demultiplexer auf IP
- UDP bietet sonst keine zusätzliche Funktionalität gegenüber IP - UDP-Pakete können
  - verloren gehen
  - in falscher Reihenfolge beim Empfänger ankommen

2 Rechnernamen / IP-Adressen (2)

- dezentrale Abbildung über ***Nameserver***
  - ◆ Beispiel: **faui04.informatik.uni-erlangen.de**
- Abbildung Rechnernamen ↔ IP-Adressen (Schnittstelle)
- Funktionen: ***gethostent(3)***
  gethostbyname(3)
  gethostbyaddr(3)
- Include-Datei: `<netdb.h>`
  Struktur *hostent* :

```
struct hostent {
    char *h_name;          /* official name of host */
    char  **h_aliases;     /* alias list */
    int   h_addrtype;      /* host address type (AF_INET) */
    int   h_length;        /* length of address */
    char  **h_addr_list    /* list of addresses from
                              name server */
};
```

G.8 spezielle Funktionalität

- Broadcast-Pakete
  - nur bei Datagramm-Sockets möglich
  - kann von allen Rechnern auf dem Netz empfangen werden
  - netzwerkunabhängige Programmierung kompliziert, da zuerst die angeschlossenen Netzwerke erfragt werden müssen

- Out of band data
  - dringende Daten, die bevorzugt empfangen werden sollen
  - nur bei Stream-Sockets
  - senden mit ***send(2)***, empfangen mit ***recv(2)***
  - bewirkt beim Empfänger das Signal **SIGURG**