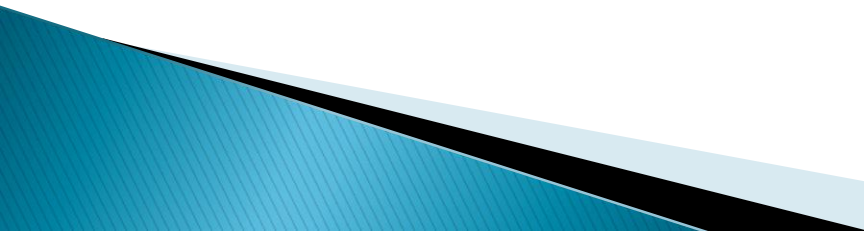
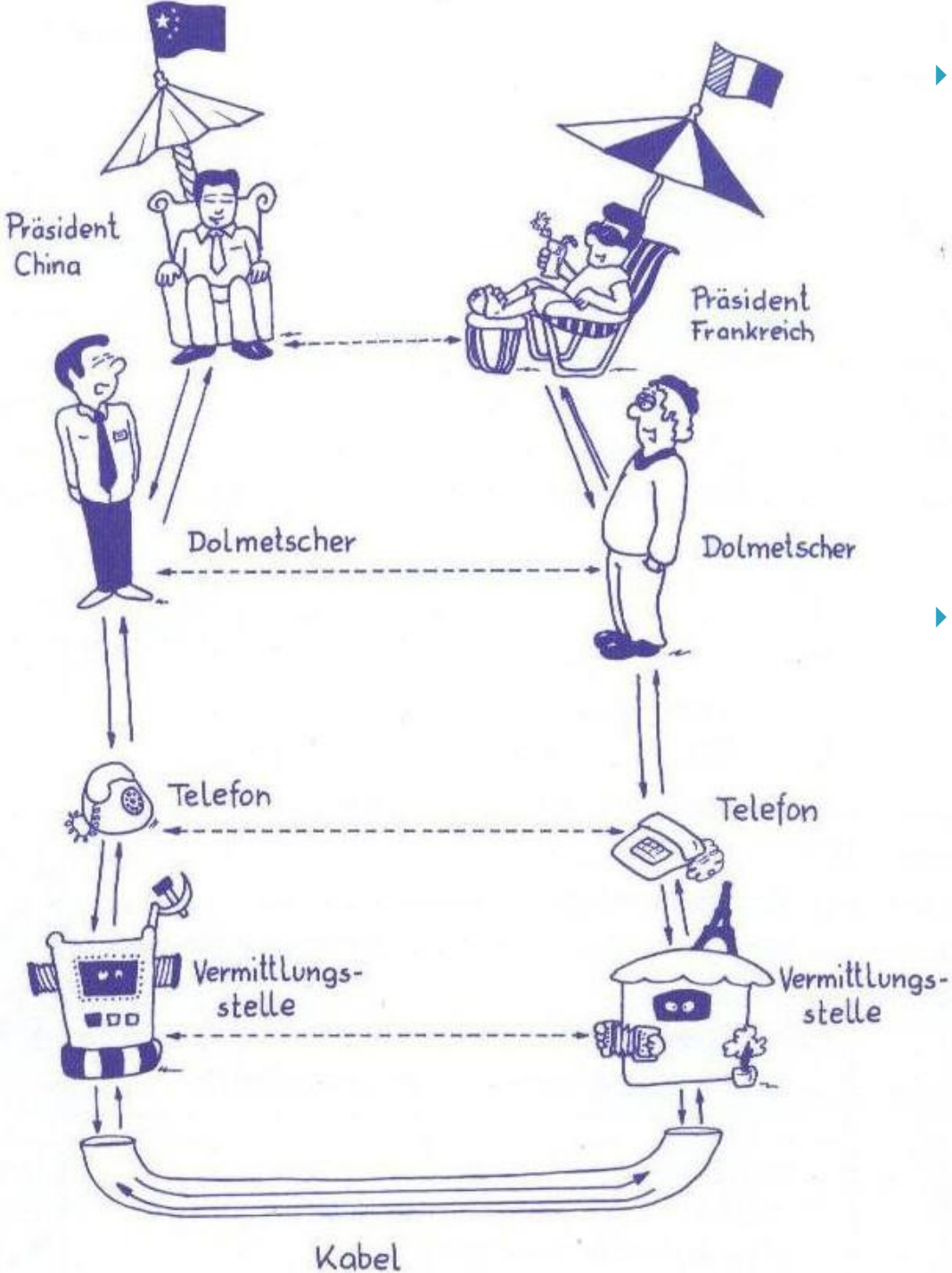


Netzwerke

Protokolle

Schichtenmodelle

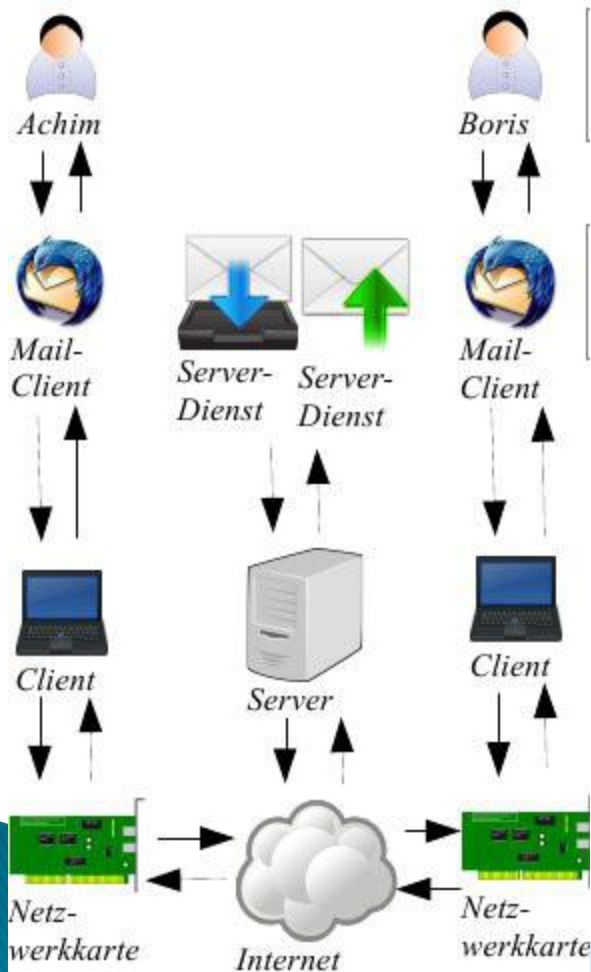
- ▶ Kommunikation findet in Computernetzen – wie im echten Leben – auf verschiedenen Ebenen statt.
 - ▶ Um zu beschreiben, wie die Datenkommunikation in Netzwerken prinzipiell funktioniert, verwendet man so genannte Schichtenmodelle
 - ▶ Schichtenmodelle gliedern komplexe Problemstellungen in einzelne Schritte, die nacheinander ausgeführt werden und aufeinander aufbauen
- 



- ▶ Nach welchen Prinzipien funktioniert die Kommunikation?
 - ▶ Jede Schicht kommuniziert nur mit der darüber / darunter liegenden Schicht (horiz. Kommunikation)
 - ▶ Schichtenmodelle beschreiben nicht, wie die Abläufe innerhalb der Schicht zu regeln sind.
- ▶ Vorteile eines Schichtenmodells?
 - ▶ Abläufe innerhalb der Schicht sind austauschbar (Spanisch statt Englisch, Mail statt Telefon)
 - ▶ Gegenüberliegende Seiten müssen sich nur auf einen Ablauf einigen.
 - ▶ Komplexes Problem der Kommunikation wird in Teilprobleme zerlegt

Das TCP / IP Referenzmodell

- ▶ Kommunikation in einem Netzwerk findet ebenfalls in mehreren Schichten statt
- ▶ Beispiel: Weg einer Email vom Sender zum Empfänger



Anwendungsschicht:

Funktion: Schnittstelle zwischen User und PC
Protokolle: z.B. HTTP, FTP, SMTP, POP,...

Transportschicht:

Funktion: Verbindungsaufbau, Zuverlässigkeit, Fehlerkorrektur
Protokolle: z.B. TCP, UDP

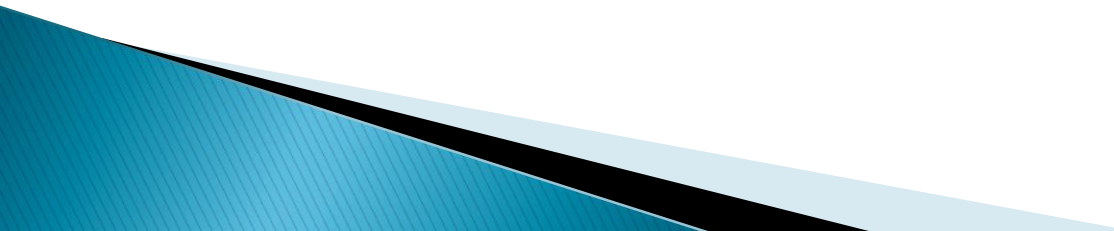
Internetschicht:

Funktion: Wegewahl (Routing) der Pakete
Protokolle: IP

Netzzugangsschicht:

Funktion: physikalische Zugriff
Protokolle: Ethernet, TokenBus, TokenRing, ...

TCP (Transmission Control Protocol)

- ▶ TCP (Transmission Control Protocol) ist dafür verantwortlich, dass Nachrichten, die gesendet werden, auch beim Empfänger ankommen.
 - ▶ TCP teilt die Originalnachricht einer Anwendung (z.B. eine Emailnachricht) in kleine Pakete auf.
- 

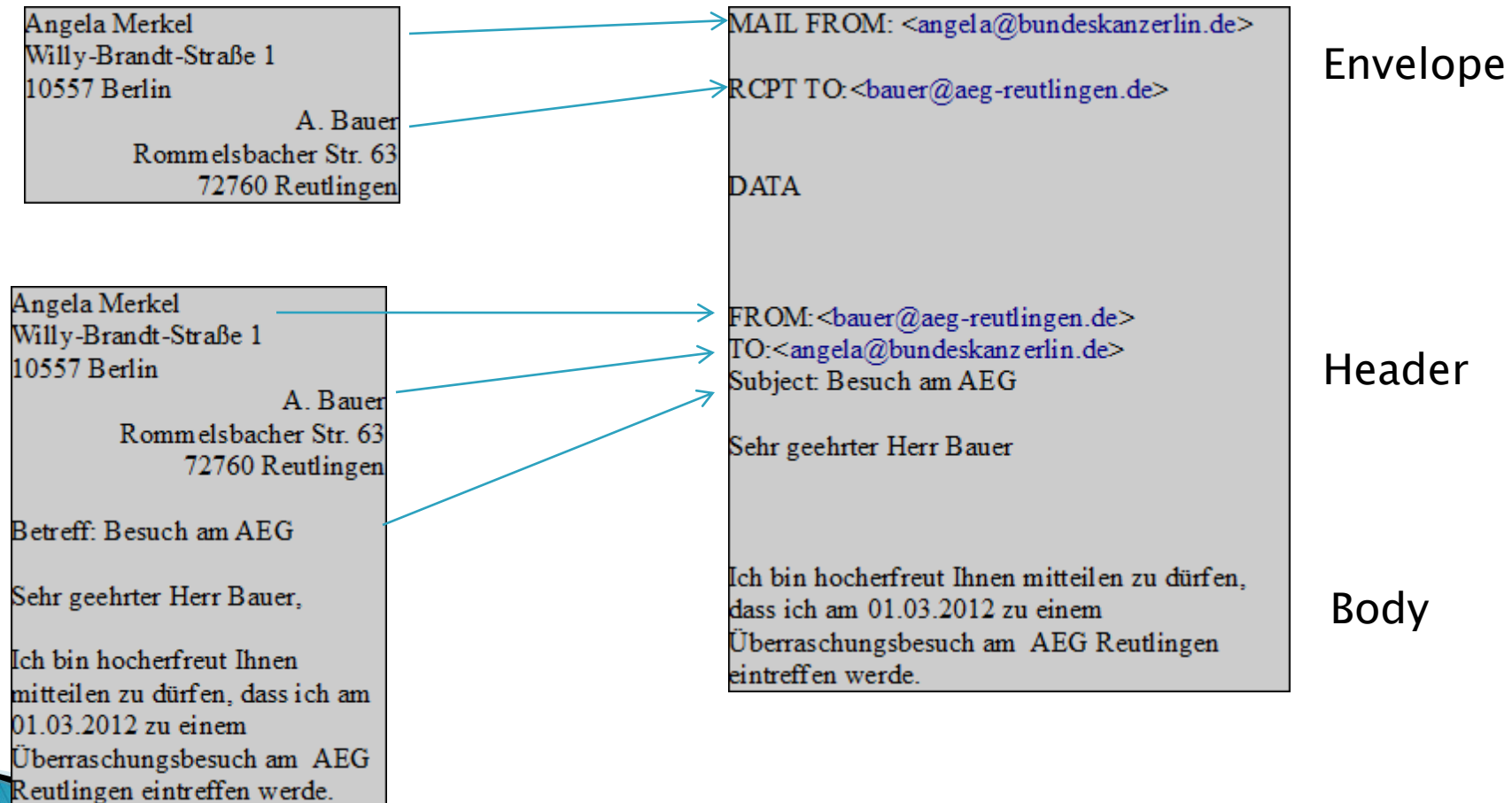
TCP (Transmission Control Protocol)

- ▶ TCP ordnet – jedem Paket zwei Portnummern zu (Portnummer der Senderanwendung, Portnummer der Empfängeranwendung).
 - Port = Kennzeichnung des Anwendungsprozesses. Webserver erwarten Anfragen z.B. auf Port 80.
- ▶ TCP nummeriert jedes ausgehende Paket
- ▶ Beim Empfänger wird kontrolliert, ob alle Pakete korrekt angekommen sind und diese wieder zusammengesetzt.

IP – Internet Protocol

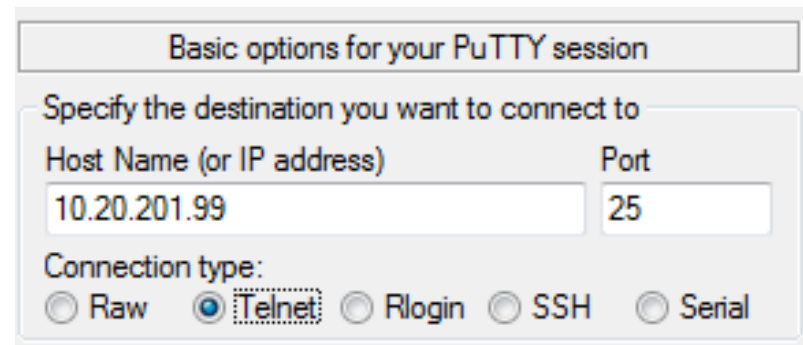
- ▶ Das Internet Protocol IP sorgt dafür, dass die Pakete ihren Weg vom Quellrechner zum Zielrechner finden.
 - IP versieht jedes Paket mit der IP-Adresse des Senders und der IP-Adresse des Empfängers.
 - IP versieht jedes Paket mit einer TTL (Time-to-Live), d.h. einer maximalen Lebensdauer. Ist diese abgelaufen, so wird das Paket verworfen.

Email – Vergleich mit Brief



Email – Zu Fuß bzw. per Hand

1. Starte das Programm *putty*.
2. Gib unter *HostName* die Adresse des Mailservers ein und unter Port wahlweise 25 oder 110, je nach dem ob du Emails abholen oder versenden willst.
3. Wähle unter *Connection type* Telnet aus.
4. Mit einem Klick auf *open* kannst du die Verbindung zum Mailserver herstellen.



Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address)	Port
10.20.201.99	25

Connection type:

Raw Telnet Rlogin SSH Serial

Email – Zu Fuß bzw. per Hand

- ▶ Starte die Eingabeaufforderung (cmd)
- ▶ Gib den Befehl „telnet“ gefolgt von der IP-Adresse des Servers und der Portnummer für das jeweilige Protokoll ein.
(telnet 10.20.201.99 25)
- ▶ Nun kannst du die Befehle der Prokollle verwenden.
- ▶ Versuche, deinem Nachbarn eine Email zu schicken.