

# Grundlagen Rechnernetze und Verteilte Systeme

SoSe 2012

## Kapitel 0: Überblick und Einführung

Prof. Dr.-Ing. Georg Carle

Stephan M. Günther, M.Sc.

Nadine Herold, M.Sc.

Dipl.-Inf. Stephan Posselt

Fakultät für Informatik

Lehrstuhl für Netzarchitekturen und Netzdienste

Technische Universität München

## Kapitel 0: Organisatorisches, Überblick und Schichtenmodelle

- 1 Die Vorlesung im Überblick
  - Organisatorisches zur Vorlesung
  - Zusammenfassung der einzelnen Kapitel
  
- 2 Geschichte des Internets
  - Von der Entstehung bis zum heutigen Internet
  - Bedeutung des Internets für die Gesellschaft
  
- 3 Schichtenmodelle
  - Was sind Schichtenmodelle?
  - Wozu sind Schichtenmodelle gut?
  - Das ISO/OSI-Modell

## Organisatorisches zu Vorlesung und Übung

- ▶ Prof. Dr.-Ing. Georg Carle
- ▶ Email: [carle@net.in.tum.de](mailto:carle@net.in.tum.de)
- ▶ Raum: MI 03.05.054
- ▶ Sprechstunde: Mo 16:00 – 17:00 Uhr bzw. nach Vereinbarung



### Übungsleitung:



- ▶ Stephan M. Günther, M.Sc.
- ▶ Email: [guenther@net.in.tum.de](mailto:guenther@net.in.tum.de)
- ▶ Raum: MI 03.05.057
- ▶ Sprechstunde: jederzeit

- ▶ Nadine Herold, M.Sc.
- ▶ Email: [herold@net.in.tum.de](mailto:herold@net.in.tum.de)
- ▶ Raum: MI 03.05.044
- ▶ Sprechstunde: nach Vereinbarung



## Vorlesungsbetrieb

### Termine

- ▶ Mo 14:15 – 15:45, MI HS 1
- ▶ Di 10:15 – 11:45, MW 0001 (Gustav-Niemann-Hörsaal)

### Zentralübung

- ▶ Nach Ankündigung findet anstelle einer Vorlesung eine Zentralübung statt
- ▶ Inhalte sind Programmieraufgaben sowie Klausurvorbereitung

## Übungsbetrieb

### Tutorübungen

- ▶ Wöchentliche Übungsblätter
- ▶ Bearbeitung während der Tutorübung in Kleingruppen
- ▶ Keine Anwesenheitspflicht / Hausaufgaben

### Termine und Anmeldung zu den Tutorübungen

- ▶ Anmeldung über TUMOnline
- ▶ Anmeldung freigeschaltet ab morgen (17. April) um 19:00 Uhr
- ▶ Übungsbeginn in der 2. Vorlesungswoche (ab 23. April)

## Vorläufige Gruppenliste Übersicht

Tag	Start <sup>1</sup>	Ende	Raum	Tutor	#	Sprache
Mo	08:00	10:00	03.07.023	Felix Kuperjans	1	deutsch
Mo	12:00	14:00	MI HS 2	Alexander Winkler	2	deutsch
Mo	12:00	14:00	00.08.036	Mathias Kanzler	3	deutsch
Mo	16:00	18:00	00.08.036	Mathias Kanzler	4	deutsch
Mo	16:00	18:00	00.08.059	Alexander Kurtz	20	<b>english</b>
Di	08:00	10:00	03.07.023	Maurice Leclair	5	deutsch
Di	12:00	14:00	00.08.036	Alexander Winkler	6	deutsch
Di	12:00	14:00	00.08.059	Maurice Leclair	7	deutsch
Di	12:00	14:00	02.13.010	Benedikt Engeser	14	deutsch
Di	16:00	18:00	00.08.059	Alexander Kurtz	8	<b>english</b>
Di	16:00	18:00	00.08.036	Tim Wiese	17	deutsch
Mi	08:00	10:00	03.07.023	Sebastian Hofstetter	9	deutsch
Mi	10:00	12:00	03.07.023	Sebastian Hofstetter	10	deutsch
Mi	12:00	14:00	03.07.023	Benedikt Engeser	11	deutsch
Do	08:00	10:00	03.07.023	Markus Reiter	21	deutsch
Do	10:00	12:00	00.08.059	Andreas Herzog	12	deutsch
Do	10:00	12:00	00.08.036	Vedat Levi Alev	18	deutsch
Do	12:00	14:00	00.08.059	Markus Reiter	13	deutsch
Do	12:00	14:00	00.08.036	Johannes Naab	22	deutsch
Fr	10:00	12:00	00.08.059	Tim Wiese	15	deutsch
Fr	10:00	12:00	00.08.055	Felix Kuperjans	19	deutsch
Fr	14:00	16:00	00.08.036	Johannes Naab	16	deutsch

<sup>1</sup> s.t / c.t. wird vom Tutor in der ersten Übung entschieden

## Programmieraufgaben

Es gibt semesterbegleitend 4 Programmieraufgaben:

- ▶ Bearbeitung in Gruppen von **maximal 2 Personen**
- ▶ Anmeldung der Teams
  - ▶ bis spätestens 22. April
  - ▶ über diesen **Link**.
  - ▶ bzw. **Homepage zur Vorlesung ([www.net.in.tum.de](http://www.net.in.tum.de))** → Lehre → SoSe 12 → Vorlesungen → Informationen des Lehrstuhls → Formular
- ▶ Abgabe der Aufgaben über Subversion (Details folgen)
- ▶ Geplante Aufgaben:
  - 1 MATLAB-Aufgabe zu Kanalkodierung
  - 2 Implementierung von Traceroute in C
  - 3 Implementierung eines Client-/Server-Programms in C oder Java
  - 4 Implementierung eines Streaming Servers in C oder Java
- ▶ Abgabe der Programmieraufgaben via SVN (Versionsverwaltung)

Die Teilnahme an den Programmieraufgaben

- ▶ ist **freiwillig**
- ▶ aber **Bestandteil der Bonusregelung**. (→ Details folgen gleich)

## Tutorgruppen zu den Programmieraufgaben

Infolge des unterschiedlichen Vorwissens der einzelnen Vorlesungsteilnehmer bieten wir zwei spezielle Tutorgruppen an:

Tag	Start <sup>2</sup>	Ende	Raum	Tutor	#	Sprache
Mo	12:00	14:00	03.07.023	Hieu Dao	-	deutsch
Do	12:00	14:00	03.07.023	Hieu Dao	-	deutsch

- ▶ Beginn am Donnerstag, 3. Mai (danach wöchentlich)
- ▶ Keine Anmeldung notwendig
- ▶ [Konkrete Fragen / Probleme zu den Programmieraufgaben](#)
- ▶ [Kein](#) regulärer Tutorbetrieb (keine Tutoraufgaben)
- ▶ Bei Bedarf können weitere Gruppen bereitgestellt werden

---

<sup>2</sup>s.t / c.t. wird vom Tutor entschieden

## Modulprüfung

- ▶ am Semesterende
- ▶ schriftlich, 90 Minuten, 85 Punkte
- ▶ zugelassene Hilfsmittel:
  - ▶ 1 beidseitig **handschriftlich** beliebig beschriebenes A4-Blatt (Kopien sind nicht zulässig)
  - ▶ nicht-programmierbarer Taschenrechner
  - ▶ Wörterbuch für ausländische Studenten
- ▶ Anmeldung über TUMOnline (tba)

## Bonusregelung

- ▶ Erfolgreiche Bearbeitung von mind. 3 der 4 Programmieraufgaben
- ▶ Teilnahme an der Midterm-Prüfung
  - ▶ 45 Minuten, 15 Bonus-Punkte
  - ▶ Hilfsmittel wie in der Modulprüfung
  - ▶ Anmeldung über TUMOnline (tba)
- ▶ Der in der Midterm erzielte **Bonus wird nur dann angerechnet, wenn mind. 3/4 Programmieraufgaben erfolgreich** bearbeitet wurden.



## Vorlesungsmaterialien

Die Vorlesungsmaterialien bestehen aus:

- ▶ Vorlesungsfolien
- ▶ Tutorübungen und Lösungsvorschläge
- ▶ Programmieraufgaben
- ▶ Skript (bislang Kapitel 0 – 1)

Bereitstellung der Vorlesungsfolien über die

- ▶ [Homepage zur Vorlesung \(www.net.in.tum.de\)](http://www.net.in.tum.de) → Lehre → SoSe 12

Bereitstellung der übrigen Unterlagen und zusätzlichen Literaturlauszügen

- ▶ SVN bzw. [Moodle \(www.moodle.tum.de\)](http://www.moodle.tum.de).

Forum für organisatorische Ankündigungen und inhaltliche Ergänzungen über

- ▶ [Moodle](#).

Die Anmeldung zum Moodle-Kurs erfolgt

- ▶ automatisch nach der [Anmeldung zur Vorlesung über TUMonline](#).

# Zusammenfassung der einzelnen Kapitel

## Kapitel 1: Physikalische Schicht

- 1 Signale, Information und deren Bedeutung
  - ▶ Was sind Signale?
  - ▶ Entropie und Information
- 2 Klassifizierung von Signalen
  - ▶ Zeit- und Frequenzbereich
  - ▶ Abtastung, Rekonstruktion und Quantisierung
- 3 Übertragungskanal
  - ▶ Einflüsse des Übertragungskanals auf Signale
  - ▶ Kapazität eines Übertragungskanals (Modell)
- 4 Nachrichtenübertragung
  - ▶ Quellen- und Kanalkodierung
  - ▶ Impulsformung
  - ▶ Modulation
- 5 Übertragungsmedien
  - ▶ Elektromagnetisches Spektrum
  - ▶ Koaxialleiter
  - ▶ Twisted-Pair-Kabel
  - ▶ Lichtwellenleiter

# Zusammenfassung der einzelnen Kapitel

## Kapitel 2: Sicherungsschicht

### 1 Darstellung von Netzwerken als Graphen

- ▶ Netztopologien
- ▶ Adjazenz- und Distanzmatrix
- ▶ Shortest Path Tree und Minimum Spanning Tree

### 2 Verbindungscharakterisierung, Mehrfachzugriff und Medienzugriffskontrolle

- ▶ Serialisierungs- und Ausbreitungsverzögerungen
- ▶ Nachrichtenflussdiagramme
- ▶ ALOHA und Slotted ALOHA
- ▶ CSMA, CSMA/CD und CSMA/CA
- ▶ Token Passing

### 3 Rahmenbildung, Adressierung und Fehlererkennung

- ▶ Erkennung von Rahmengrenzen und Codetransparenz
- ▶ Adressierung und Fehlererkennung
- ▶ Fallstudie: IEEE 802.3u (FastEthernet)

### 4 Verbindungen auf Schicht 1 und 2

- ▶ Hubs, Bridges und Switches
- ▶ Collision und Broadcast Domains

# Zusammenfassung der einzelnen Kapitel

## Kapitel 3: Vermittlungsschicht

### 1 Vermittlungsarten

- ▶ Leitungsvermittlung
- ▶ Nachrichtenvermittlung
- ▶ Paketvermittlung

### 2 Adressierung im Internet

- ▶ Internet Protocol (IP)
- ▶ Adressauflösung (ARP)
- ▶ Internet Control Message Protocol (ICMP)
- ▶ Adressklassen (Classful Routing)
- ▶ Subnetting (Classless Routing)

### 3 Routing

- ▶ Statisches Routing
- ▶ Longest Prefix Matching
- ▶ Dynamisches Routing
- ▶ Algorithmen von Bellman-Ford und Dijkstra
- ▶ Routingprotokolle (Distance Vector und Link State)
- ▶ Autonome Systeme

### 4 Nachfolge von IP(v4): IPv6

# Zusammenfassung der einzelnen Kapitel

## Kapitel 4: Transportschicht

- 1 Aufgaben der Transportschicht
- 2 Multiplexing durch Portnummern
- 3 Verbindungslose Übertragung: UDP
  - ▶ Case-Study: UDP
  - ▶ Code-Study: SOCK\_DGRAM (C)
- 4 Verbindungsorientierte Übertragung
  - ▶ Sliding-Window-Protokolle (Go-Back-N und Selective Repeat)
  - ▶ Case-Study: TCP (Fluss- und Staukontrolle)
  - ▶ Code-Study: SOCK\_STREAM (C)
- 5 Network Address Translation (NAT)

# Zusammenfassung der einzelnen Kapitel

## Kapitel 5: Die Schichten 5 – 7

### 1 Sitzungsschicht

- ▶ Dienste
- ▶ Funktionseinheiten
- ▶ Synchronisation
- ▶ Quality of Service
- ▶ Performance Parameter

### 2 Exkurs: Kryptographie

- ▶ Symmetrische und asymmetrische Verfahren
- ▶ RC4 und DH76
- ▶ Zertifikate
- ▶ Transport Layer Security (TLS)

### 3 Darstellungsschicht

- ▶ Datenkompression (Huffman Code)
- ▶ Einheitliche Syntax (ASN.1, BER)

### 4 Anwendungsschicht

- ▶ Namensauflösung im Internet (DNS)
- ▶ Demos: DNS, HTTP, (Telnet)

# Zusammenfassung der einzelnen Kapitel

## Kapitel 6: Verteilte Systeme

### 1 Homogene, skalierbare Paradigmen

- ▶ Message Passing Interface (MPI)
- ▶ MapReduce
- ▶ Pipes, netcat, DUP

### 2 Remote Procedure Call

- ▶ Funktionsaufrufe und Parameterkodierung
- ▶ Stubs, IDL, Binding
- ▶ Java RMI
- ▶ RPC/RMI

### 3 Shared Memory

- ▶ NUMA
- ▶ Virtueller Speicher
- ▶ Auslagerung
- ▶ Distributed Shared Memory
- ▶ Konsistenz in parallelen Programmen

### 4 Einbettung in Programmiersprachen

- ▶ Erlang
- ▶ Actor Model

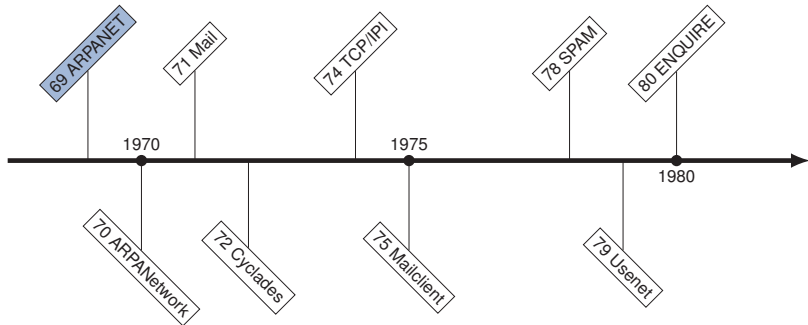
## Zurück zu Kapitel 0:

- 1 Die Vorlesung im Überblick
  - Organisatorisches zur Vorlesung
  - Zusammenfassung der einzelnen Kapitel
- 2 Geschichte des Internets
  - Von der Entstehung bis zum heutigen Internet
  - Bedeutung des Internets für die Gesellschaft
- 3 Schichtenmodelle
  - Was sind Schichtenmodelle?
  - Wozu sind Schichtenmodelle gut?
  - Das ISO/OSI-Modell



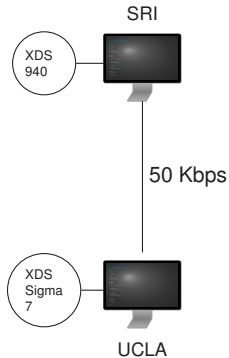
## Von der Entstehung bis zum heutigen Internet

### Geschichte des Internets: Übersicht bis 1980



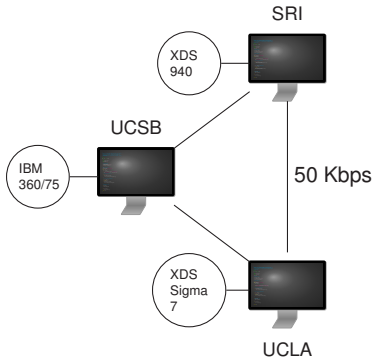
## ARPANET mit den ersten 4 Knoten

- ▶ University of California, Los Angeles (UCLA) 1.9.1969
- ▶ Stanford Research Institute (SRI) 1.10.1969



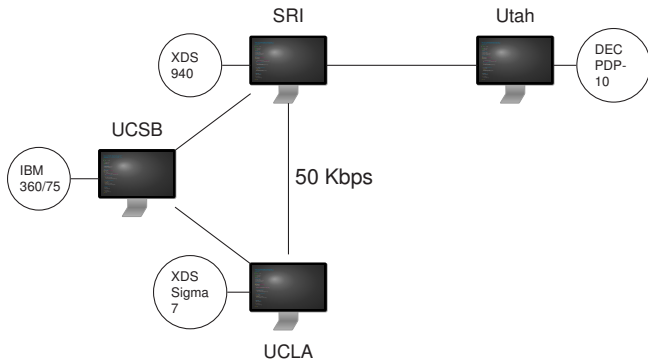
## ARPANET mit den ersten 4 Knoten

- ▶ University of California, Los Angeles (UCLA) 1.9.1969
- ▶ Stanford Research Institute (SRI) 1.10.1969
- ▶ UC Santa Barbara (UCSB) 1.11.1969

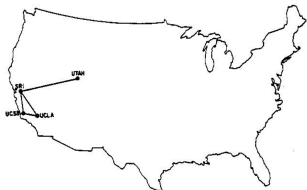


## ARPANET mit den ersten 4 Knoten

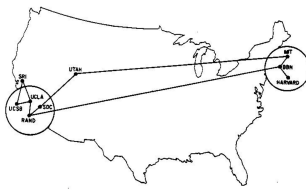
- ▶ University of California, Los Angeles (UCLA) 1.9.1969
- ▶ Stanford Research Institute (SRI) 1.10.1969
- ▶ UC Santa Barbara (UCSB) 1.11.1969
- ▶ Universität von Utah 12.1969



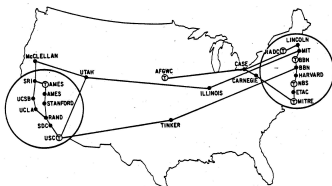
## ARPANET von 1969 bis 1977



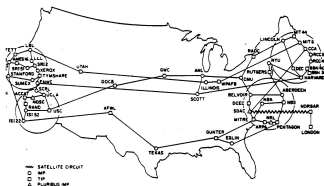
ARPANET 1969, 4 Knoten



ARPANET 1970, 9 Knoten

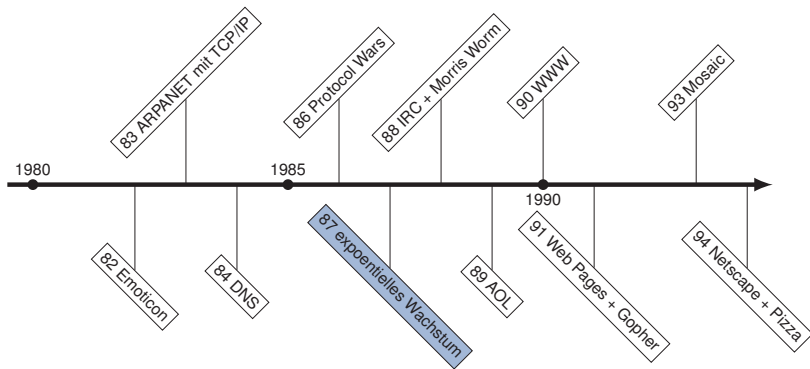


ARPANET 1972, 25 Knoten

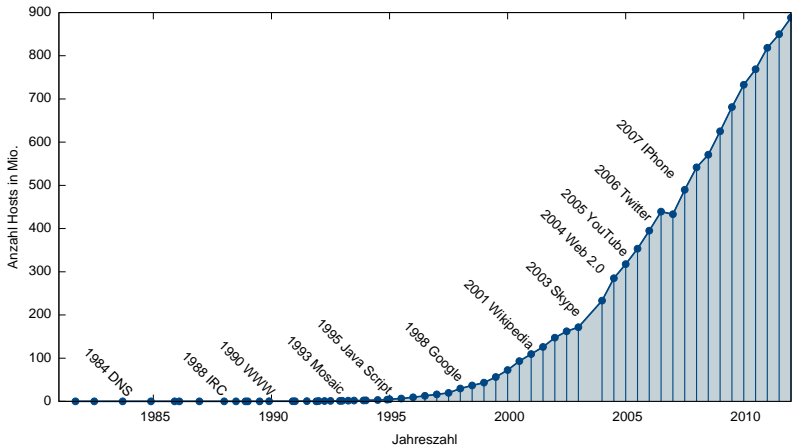


ARPANET 1977, 58 Knoten

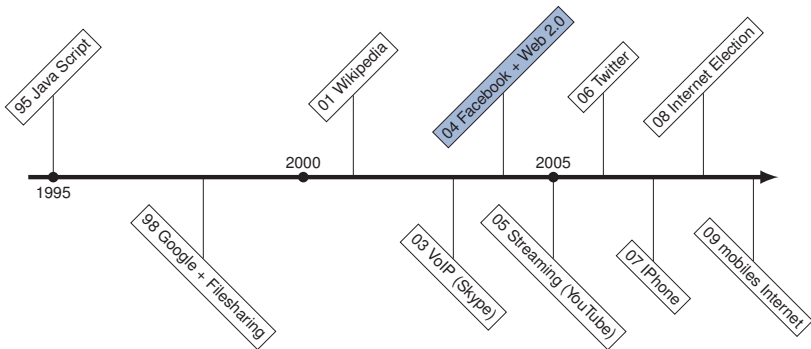
## Geschichte des Internets: Übersicht von 1980 bis 1994



Anzahl der Hosts von 1981 bis 2012



## Geschichte des Internets: Übersicht ab 1994





## Web 2.0 Meme Map

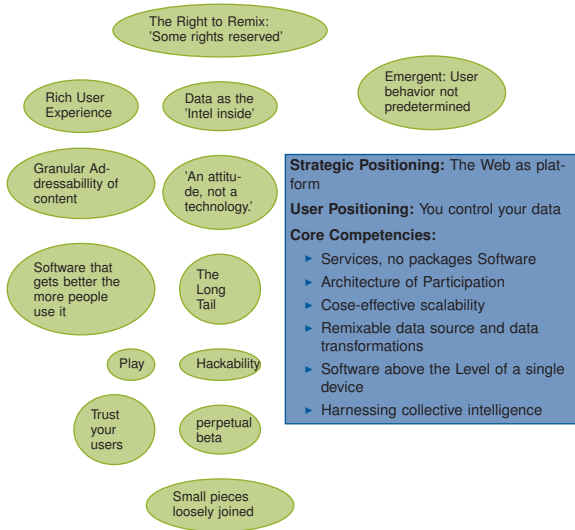
**Strategic Positioning:** The Web as platform

**User Positioning:** You control your data

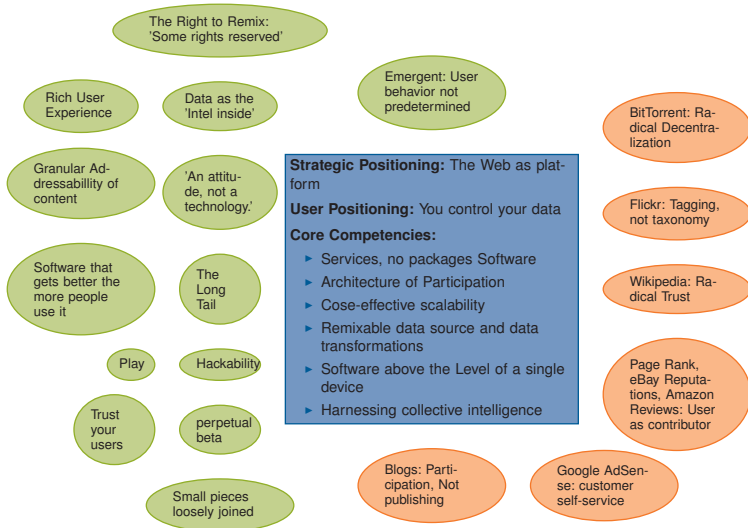
**Core Competencies:**

- ▶ Services, no packages Software
- ▶ Architecture of Participation
- ▶ Cost-effective scalability
- ▶ Remixable data source and data transformations
- ▶ Software above the Level of a single device
- ▶ Harnessing collective intelligence

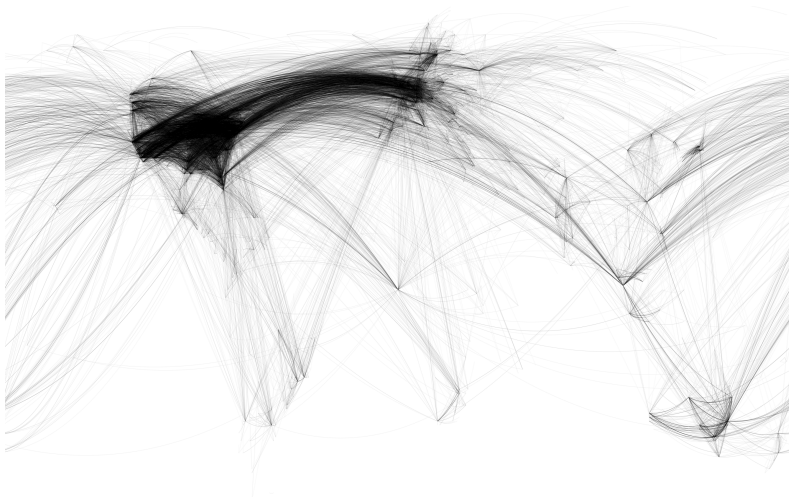
## Web 2.0 Meme Map



## Web 2.0 Meme Map

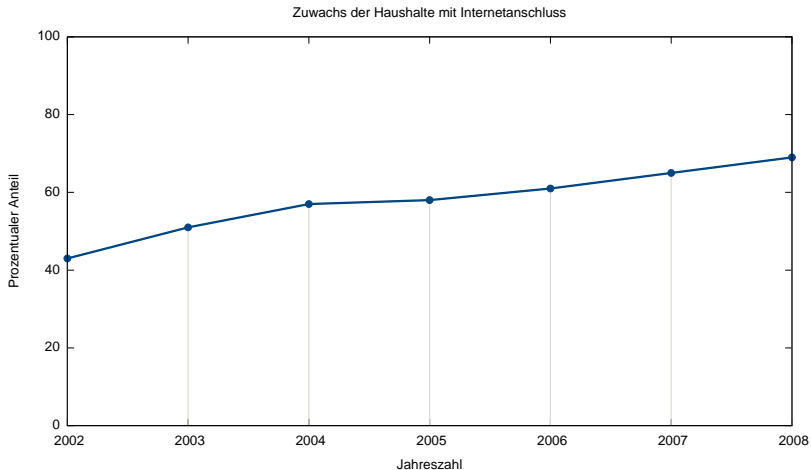


## Das Internet Heute

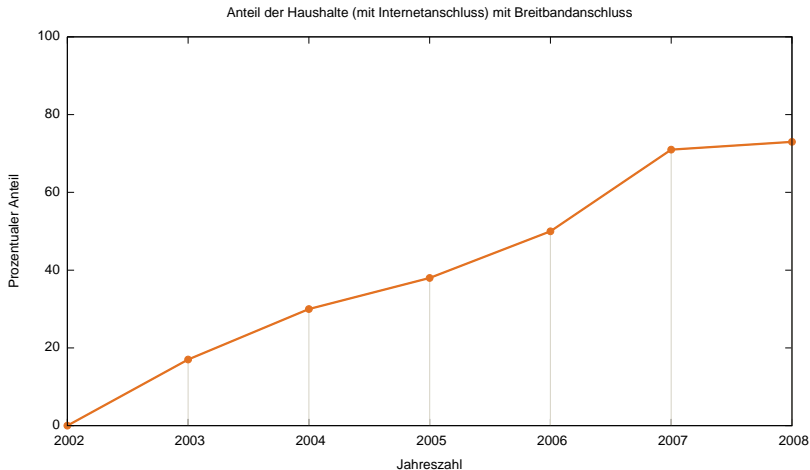


Internet 2007 [1]

## Bedeutung des Internets für die Gesellschaft

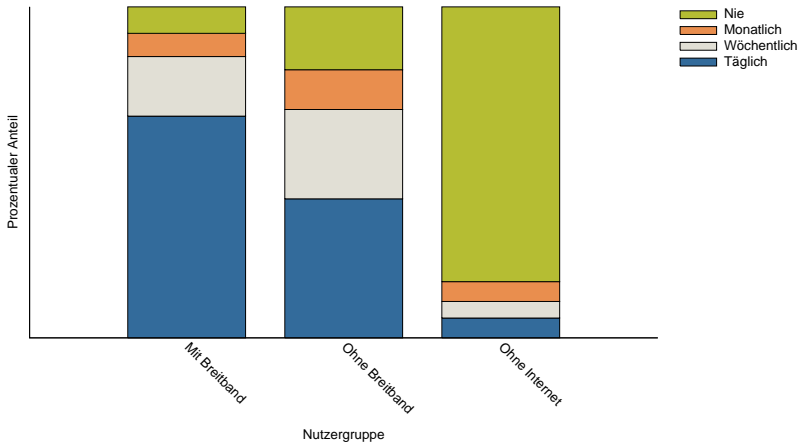


## Bedeutung des Internets für die Gesellschaft

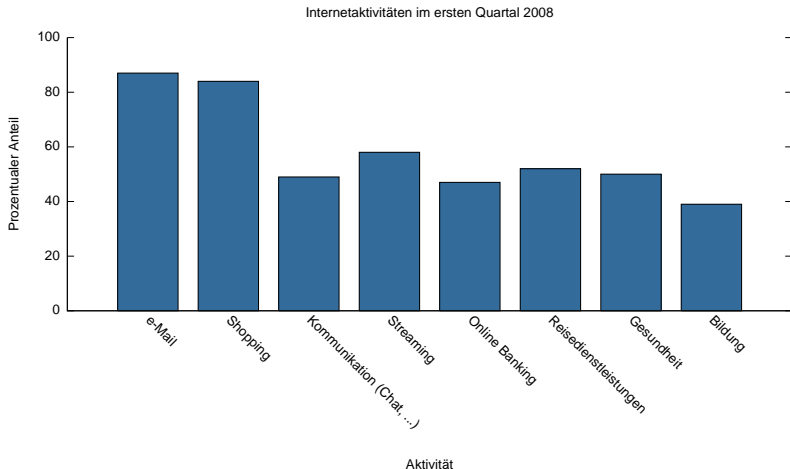


## Bedeutung des Internets für die Gesellschaft

Häufigkeit der Internetnutzung durch Personen im ersten Quartal 2008



## Bedeutung des Internets für die Gesellschaft



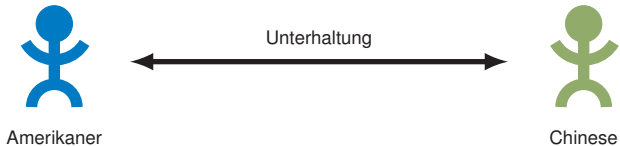


# Inhalt

- 1 Die Vorlesung im Überblick
  - Organisatorisches zur Vorlesung
  - Zusammenfassung der einzelnen Kapitel
- 2 Geschichte des Internets
  - Von der Entstehung bis zum heutigen Internet
  - Bedeutung des Internets für die Gesellschaft
- 3 Schichtenmodelle
  - Was sind Schichtenmodelle?
  - Wozu sind Schichtenmodelle gut?
  - Das ISO/OSI-Modell

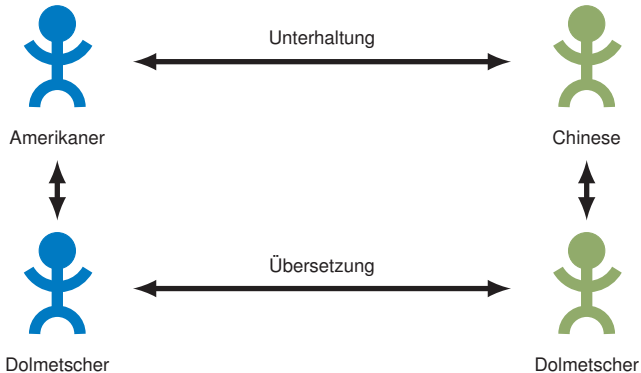
## Was sind Schichtenmodelle?

Ein einfaches Beispiel:



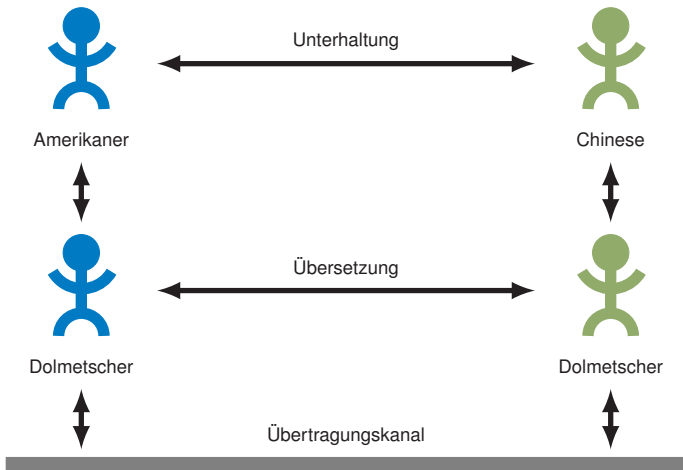
## Was sind Schichtenmodelle?

Ein einfaches Beispiel:



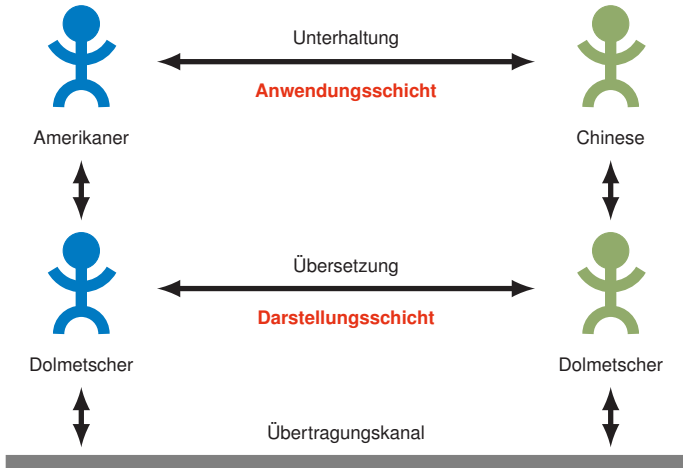
## Was sind Schichtenmodelle?

Ein einfaches Beispiel:



## Was sind Schichtenmodelle?

Ein einfaches Beispiel:



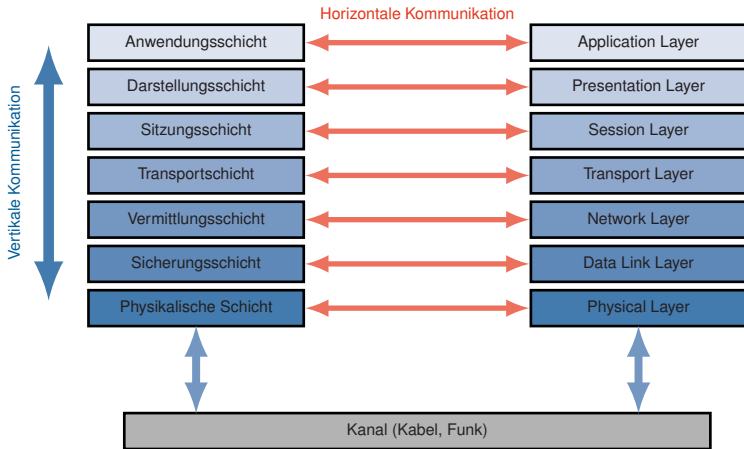
## Wozu sind Schichtenmodelle gut?

- ▶ Unterteilung des komplexen Kommunikationsvorgangs
  - ▶ Niedrigere Schichten **bieten** höheren Schichten **Dienste an**
  - ▶ Höhere Schichten **nehmen Dienste** der jeweils niedrigeren Schicht **in Anspruch**
- ▶ Abstraktion von der Implementierung einer Schicht
  - ▶ Festlegung, **welche** Dienste angeboten werden, aber **nicht wie** sie erfüllt werden
  - ▶ Austauschbarkeit einzelner Implementierungen
- ▶ Anwendbar auf beliebige Kommunikationsvorgänge

## Das ISO/OSI-Modell

- ▶ Entwickelt zwischen 1979 und 1983 von der *International Organization for Standardization (ISO)*
- ▶ OSI = *Open Systems Interconnect*
- ▶ Unterteilt den Kommunikationsvorgang in **7 Schichten**
- ▶ Jede Schicht erbringt bestimmte Dienste (z. B. Aufteilen einer Nachricht in kleinere Pakete)
- ▶ Keine Aussage, wie diese Dienste zu erbringen sind

## Schematische Darstellung des OSI-Modells:





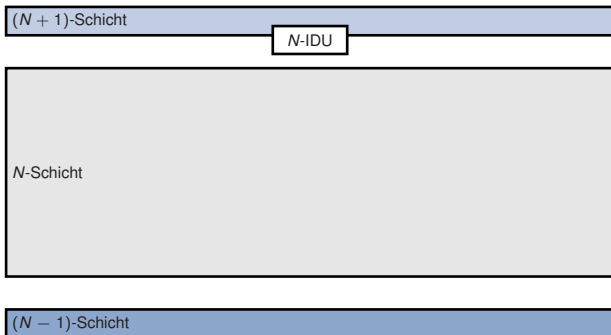
## Datenaustausch zwischen Schichten

$(N + 1)$ -Schicht

$N$ -Schicht

$(N - 1)$ -Schicht

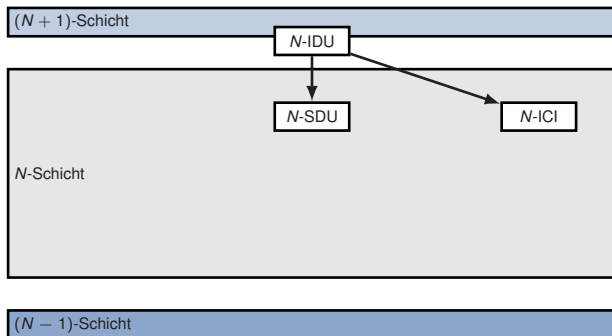
## Datenaustausch zwischen Schichten



$(N + 1)$ -Schicht nimmt Dienste der  $N$ -Schicht in Anspruch:

- ▶  $N$ -Schicht erhält eine **Interface Data Unit (IDU)** von der  $(N + 1)$ -Schicht.

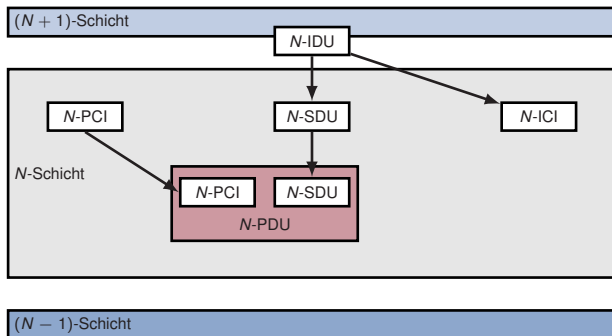
## Datenaustausch zwischen Schichten



$N$ -IDU enthält aus Sicht der  $N$ -Schicht

- ▶ Nutzdaten (**Service Data Unit (SDU)**) und
- ▶ Kontrollinformationen (**Interface Control Information (ICI)**), welche zum Erbringen des Dienstes notwendig sind (z. B. Adressinformationen).

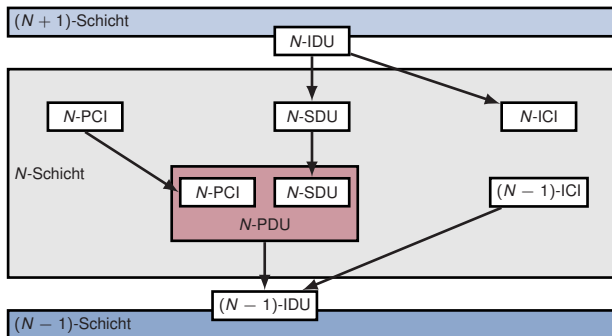
## Datenaustausch zwischen Schichten



### *N*-Schicht

- ▶ erbringt auf der *N*-SDU die angeforderten Dienste,
- ▶ fügt sog. **Protocol Control Information (PCI)** für die *N*-Schicht der gegenseite hinzu und
- ▶ erzeugt so aus PCI und SDU die **Protocol Data Unit (PDU)**.

## Datenaustausch zwischen Schichten



$N$ -Schicht fordert nun Dienste der  $(N - 1)$ -Schicht an indem

- ▶  $(N - 1)$ -ICI erzeugt und
- ▶ zusammen mit der  $N$ -PDU

als  $(N - 1)$ -IDU der nächst niedrigeren Schicht übergeben werden.

Üblich ist der Begriff **Protocol Data Unit (PDU)**, welcher auf der  $N$ -Schicht

- ▶ die (ggf. bearbeiteten) Nutzdaten der  $(N - 1)$ -Schicht sowie
- ▶ protokollspezifische Informationen (PCI) der  $N$ -Schicht

bezeichnet. Die PCI wird dabei häufig in Form eines **Headers** den Nutzdaten vorangestellt.

PDU's einiger Schichten haben eigene Bezeichnungen. Man spricht von

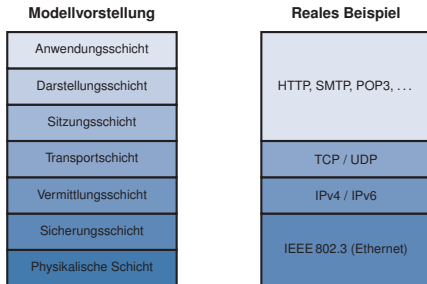
- ▶ **Segmenten** auf der Transportschicht,
- ▶ **Paketen** auf der Vermittlungsschicht bzw.
- ▶ **Rahmen** (engl. **Frames**) auf der Sicherungsschicht.

Die Unterscheidung ermöglicht es, implizit die gerade betrachtete Schicht anzugeben.

Insbesondere die Unterscheidung zwischen Rahmen und Paketen ist in der Literatur häufig fließend.

## Schwächen des ISO/OSI-Modells

- ▶ Die Trennung der Schichten widerspricht manchmal anderen Interessen (z. B. der Effizienz)
- ▶ Einige Protokolle sind daher nicht klar einer bestimmten Schicht zuzuordnen oder arbeiten sogar auf mehreren Schichten ([Cross Layer](#))
- ▶ Die Zuordnung von Protokollen auf einzelne Schichten hängt häufig von der Betrachtungsweise ab



Eine kurze Übersicht zum ISO/OSI-Modell finden Sie in [2].

## Literaturhinweise und Quellenangaben

- [1] Harrison, C.: World City-to-City Connections.  
<http://www.chrisharrison.net/index.php/Visualizations/InternetMap>.
- [2] Stein, E.: Taschenbuch Rechnernetze und Internet, Kapitel Das OSI-Modell, Seiten 22–28.  
Fachbuchverlag Leipzig, 2. Auflage, 2004.  
Auszug s. Moodle/SVN.