

Grundlagen Rechnernetze und Verteilte Systeme (GRNVS)

IN0010 – SoSe 2016

Prof. Dr.-Ing. Georg Carle

Johannes Naab, Stephan Günther, Maurice Leclaire

Lehrstuhl für Netzarchitekturen und Netzdienste
Fakultät für Informatik
Technische Universität München

Die Vorlesung im Überblick

Organisatorisches zur Vorlesung

Zusammenfassung der einzelnen Kapitel

Geschichte des Internets

Von der Entstehung bis zum heutigen Internet

Schichtenmodelle

Was sind Schichtenmodelle?

Wozu sind Schichtenmodelle gut?

Das ISO/OSI-Modell

Literaturangaben

Die Vorlesung im Überblick

Organisatorisches zur Vorlesung

Zusammenfassung der einzelnen Kapitel

Geschichte des Internets

Von der Entstehung bis zum heutigen Internet

Schichtenmodelle

Was sind Schichtenmodelle?

Wozu sind Schichtenmodelle gut?

Das ISO/OSI-Modell

Literaturangaben

- Prof. Dr.-Ing. Georg Carle
- Email: carle@tum.de
- Raum: MI 03.05.054
- Sprechstunde: Mo 18-18:30 Uhr oder nach Vereinbarung



Übungsleitung



- Johannes Naab
- Raum: MI 03.05.057



- Maurice Leclaire
- Raum: MI 03.05.061



- Stephan Günther
- Raum: MI 03.05.061

Fragen an die Übungsleitung: grnvs@net.in.tum.de

Vorlesungsbetrieb

- **Termine**
 - Mo 14:15 – 15:45, MW 2001
 - Di 10:15 – 11:45, MW 2001
- **Zentralübung**
 - Nach Ankündigung findet anstelle einer Vorlesung eine Zentralübung statt
 - Inhalte sind Programmieraufgaben sowie Klausurvorbereitung

Vorlesungsbetrieb

- Termine
 - Mo 14:15 – 15:45, MW 2001
 - Di 10:15 – 11:45, MW 2001
- 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 **Dienstag, 12. April 20:00 bzw. 20:30 Uhr**
 - Übungsbeginn in der 2. Vorlesungswoche (ab 18. April)

Vorlesungsbetrieb

- Termine
 - Mo 14:15 – 15:45, MW 2001
 - Di 10:15 – 11:45, MW 2001
- 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 [Dienstag, 12. April 20:00 bzw. 20:30 Uhr](#)
 - Übungsbeginn in der 2. Vorlesungswoche (ab 18. April)

Schülerstudenten

- Schülerstudenten melden sich bitte bei der Übungsleitung (grnvs@net.in.tum.de)

Organisatorisches zur Vorlesung

Vorläufige Gruppen

| Gruppe | Tag | Start | Ende | Raum |
|--------|----------|-------|-------|-----------|
| Mo-T1 | Montag | 12:00 | 14:00 | 01.06.011 |
| Mo-T2 | Montag | 12:00 | 14:00 | 03.11.018 |
| Mo-T3 | Montag | 12:00 | 14:00 | 00.08.036 |
| Mo-T4 | Montag | 12:00 | 14:00 | 01.09.014 |
| Mo-T5 | Montag | 12:00 | 14:00 | 03.09.012 |
| Mo-T6 | Montag | 16:00 | 18:00 | 02.07.023 |
| Mo-T7 | Montag | 16:00 | 18:00 | 00.08.059 |
| Mo-T8 | Montag | 16:00 | 18:00 | 00.13.054 |
| Mo-T9 | Montag | 16:00 | 18:00 | 00.08.036 |
| Di-T10 | Dienstag | 08:00 | 10:00 | 00.08.036 |
| Di-T11 | Dienstag | 12:00 | 14:00 | 00.08.036 |
| Di-T12 | Dienstag | 12:00 | 14:00 | 01.07.014 |
| Di-T13 | Dienstag | 12:00 | 14:00 | 00.08.053 |
| Di-T14 | Dienstag | 14:00 | 16:00 | 00.08.059 |
| Di-T15 | Dienstag | 16:00 | 18:00 | 00.08.036 |
| Di-T16 | Dienstag | 16:00 | 18:00 | 00.08.059 |
| Mi-T17 | Mittwoch | 08:00 | 10:00 | 03.07.023 |
| Mi-T18 | Mittwoch | 10:00 | 12:00 | 03.07.023 |
| Mi-T19 | Mittwoch | 10:00 | 12:00 | 01.06.020 |

| Gruppe | Tag | Start | Ende | Raum |
|--------|------------|-------|-------|-----------|
| Mi-T20 | Mittwoch | 11:00 | 13:00 | 01.13.010 |
| Mi-T21 | Mittwoch | 12:00 | 14:00 | 00.08.038 |
| Mi-T22 | Mittwoch | 12:00 | 14:00 | 00.08.053 |
| Mi-T23 | Mittwoch | 12:00 | 14:00 | 00.08.055 |
| Mi-T24 | Mittwoch | 14:00 | 16:00 | 00.08.036 |
| Mi-T25 | Mittwoch | 14:00 | 16:00 | 00.08.053 |
| Mi-T26 | Mittwoch | 16:00 | 18:00 | 00.08.036 |
| Do-T27 | Donnerstag | 08:00 | 10:00 | 03.07.023 |
| Do-T28 | Donnerstag | 10:00 | 12:00 | 00.08.036 |
| Do-T29 | Donnerstag | 10:00 | 12:00 | 00.08.059 |
| Do-T30 | Donnerstag | 12:00 | 14:00 | 01.10.011 |
| Do-T31 | Donnerstag | 12:00 | 14:00 | 00.13.036 |
| Do-T32 | Donnerstag | 12:00 | 14:00 | 00.08.059 |
| Do-T33 | Donnerstag | 12:00 | 14:00 | 00.08.036 |
| Do-T34 | Donnerstag | 12:30 | 14:30 | 00.08.055 |
| Do-T35 | Donnerstag | 14:00 | 16:00 | 00.08.036 |
| Do-T36 | Donnerstag | 16:00 | 18:00 | 00.13.054 |
| Fr-T37 | Freitag | 14:00 | 16:00 | 00.08.038 |
| Fr-T38 | Freitag | 16:00 | 18:00 | 00.08.059 |

- Es steht aktuell noch nicht fest welche Gruppe von welchem Tutor gehalten wird.

Programmieraufgaben

Es gibt semesterbegleitend voraussichtlich 5 Programmieraufgaben:

- Einzelabgaben
- Anmeldung über Moodle (ab nächste Woche)
- Geplante Aufgaben:
 1. Git & FizzBuzz
 2. Wake-on-LAN über RAW-Sockets
 3. Neighbor Discovery über RAW-Sockets
 4. Router
 5. TCP Sockets
- Rahmenprogramme werden in C und Java zur Verfügung gestellt
- Abgabe in anderen Programmiersprachen ist erlaubt
- Abgabe der Programmieraufgaben über Git
- Automatische Tests der Abgaben (→ Details folgen)

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 **zusätzlich** 4 spezielle Tutorgruppen an:

| Tag | Raum | Start | Ende | |
|------------|-------|-------|-----------|-----------------|
| Montag | 12:00 | 14:00 | 00.13.054 | Lukas Erlacher |
| Dienstag | 12:00 | 14:00 | 03.07.023 | Lukas Erlacher |
| Mittwoch | 12:00 | 14:00 | 03.07.023 | Markus Ongyerth |
| Donnerstag | 12:00 | 14:00 | 03.07.023 | Markus Ongyerth |

- Keine Anmeldung notwendig
- **Konkrete Fragen / Probleme zu den Programmieraufgaben**
- **Kein** regulärer Tutorbetrieb (keine Tutoraufgaben)
- Die Übungen finden **nur dann** statt, wenn auch gerade eine Programmieraufgabe zu bearbeiten ist

Anerkennung der Programmieraufgaben aus Vorjahren

Infolge der Änderungen am Bonusverfahren und der damit einhergehenden Einzelabgaben der Programmieraufgaben können Lösungen vergangener Jahre **nicht** anerkannt werden.

Modulprüfung

- schriftlich, 90 Minuten, voraussichtlich 85 Punkte
- closed-book
- Endterm am 26.07.2016 von 10:30 – 12:00 Uhr
- Anmeldung über TUMOnline: von 16.05. bis 30.06.2016, entfällt bei Anmeldung zur Midterm
- Wiederholung am Ende der Vorlesungsfreien Zeit

Bonusregelung

- Programmieraufgaben
 - Abhängig von der jeweiligen Aufgabe können bis zu 3 Punkte pro Aufgabe erzielt werden.
 - Die Gesamtanzahl der mittels Programmieraufgaben erzielbaren Punkte beträgt 10.
- Midterm-Prüfung
 - 45 Minuten
 - 10 Bonus-Punkte
 - closed-book
 - voraussichtlich am Freitag, 10.06.2016 von 18:30 – 19:15 Uhr
 - Anmeldung über TUMOnline: von 02.05. bis 03.06.2016
- Die Bonuspunkte werden auf das Ergebnis der Endterm addiert, sofern diese **ohne Bonus mindestens mit der Note 4,0 bestanden** wurde.
- Die Gesamtzahl der anrechenbaren Bonuspunkte beträgt 15, d. h. werden sowohl in den Programmieraufgaben als auch in der Midterm jeweils volle Punktzahl erzielt, so werden dennoch maximal 15 Punkte angerechnet.
- Der Bonus wird auch auf die Wiederholung angerechnet.

Vorlesungs- und Übungsmaterial

- Vorlesungsunterlagen und Übungsmaterialien werden über HTTP und Git bereitgestellt:

`http://grnvs.net.in.tum.de/
grnvs@git.net.in.tum.de:material`
- Alle Unterlagen, die einem gesonderten Urheberrecht unterliegen (beispielsweise Buchscans), sind nur nach Authentifizierung über git zu erreichen.
- Sollten Sie nicht an der TUM studieren (beispielsweise LMU- oder Schülerstudenten), wenden Sie sich bitte an die Übungsleitung grnvs@net.in.tum.de
- Für den Zugriff auf die Git Repositories ist eine SSH-Key Abgabe im Moodle erforderlich. Details hierzu nächste Woche bei der Vorstellung der ersten Programmieraufgabe.
- Für den Zugriff auf Moodle ist die Vorlesungsanmeldung in TUMOnline erforderlich

Hinweis:

Aktuelle Informationen zur Vorlesung erhalten Sie ausschließlich über das Moodle, dies beinhaltet Informationen zum Stand der Vorlesung oder Updates zu Klausuren.

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

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)
- Fallstudie: IEEE 802.11a/b/g/n (Wireless LAN)

4. Verbindungen auf Schicht 1 und 2

- Hubs, Bridges und Switches
- Collision und Broadcast Domains

Kapitel 3: Vermittlungsschicht

1. Vermittlungsarten

- Leitungsvermittlung
- Nachrichtenvermittlung
- Paketvermittlung

2. Adressierung im Internet

- Internet Protocol version 4 (IPv4)
 - Adressauflösung (ARP)
 - Internet Control Message Protocol (ICMP)
 - Adressklassen (für Classful Routing)
 - Subnetting und Präfixe (für Classless Routing)
- Internet Protocol version 6 (IPv6)
 - Stateless Address Autoconfiguration (SLAAC)
 - Internet Control Message Protocol v6 (ICMPv6)
 - Neighbor Discovery Protocol (NDP)

3. Routing

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

Kapitel 4: Transportschicht

1. Aufgaben der Transportschicht
2. Multiplexing durch Port-Nummern
3. Verbindungslose Übertragung: UDP
 - Case-Study: UDP
 - Code-Study: SOCK_DGRAM (C)
4. Verbindungsorientierte Übertragung: TCP
 - 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)

Kapitel 5: Die Schichten 5 – 7

1. Schichten

- Vor- und Nachteile verschiedener Schichtenmodelle

2. Sitzungsschicht

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

3. Darstellungsschicht

- Datenkompression (Huffman Code)

4. Anwendungsschicht

- Namensauflösung im Internet (DNS)
- HTTP
- SMTP

Die Vorlesung im Überblick

Organisatorisches zur Vorlesung

Zusammenfassung der einzelnen Kapitel

Geschichte des Internets

Von der Entstehung bis zum heutigen Internet

Schichtenmodelle

Was sind Schichtenmodelle?

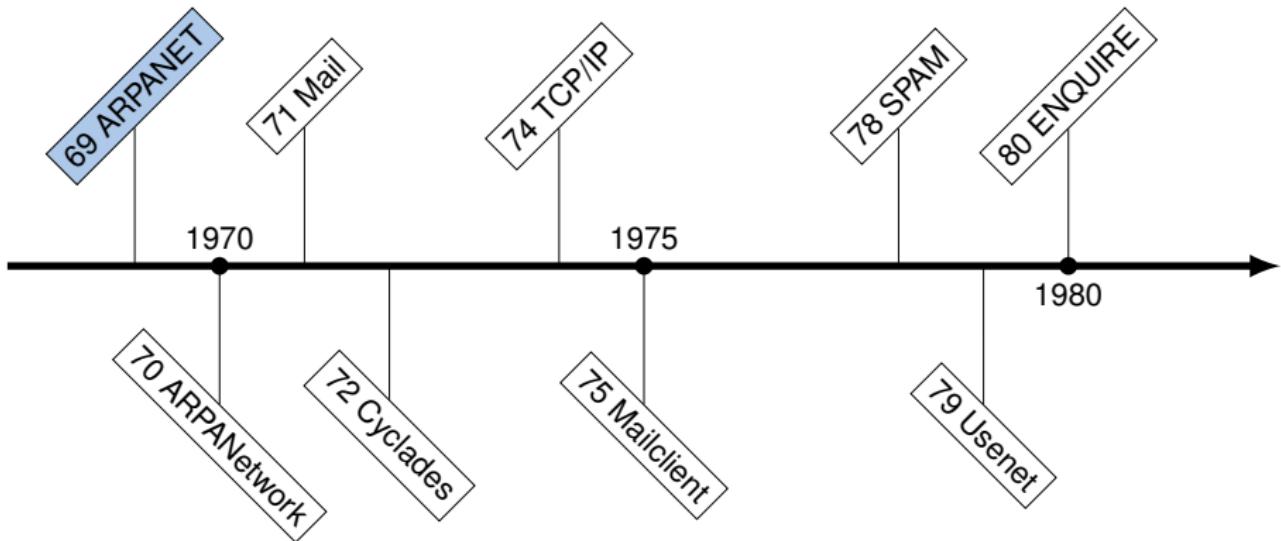
Wozu sind Schichtenmodelle gut?

Das ISO/OSI-Modell

Literaturangaben

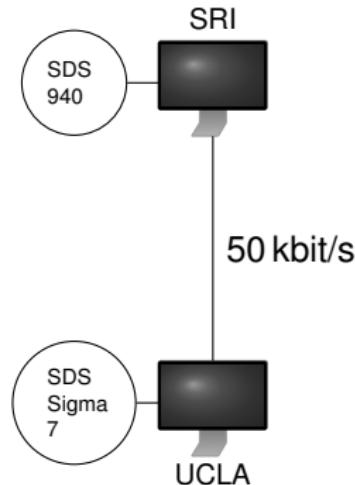
Von der Entstehung bis zum heutigen Internet

Geschichte des Internets: Übersicht bis 1980



ARPANET mit den ersten 4 Knoten

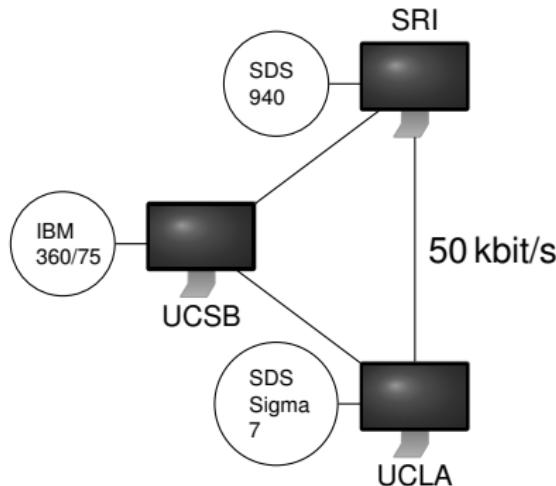
- University of California, Los Angeles (UCLA) 1.9.1969
- Stanford Research Institute (SRI) 1.10.1969



Von der Entstehung bis zum heutigen Internet

ARPANET mit den ersten 4 Knoten

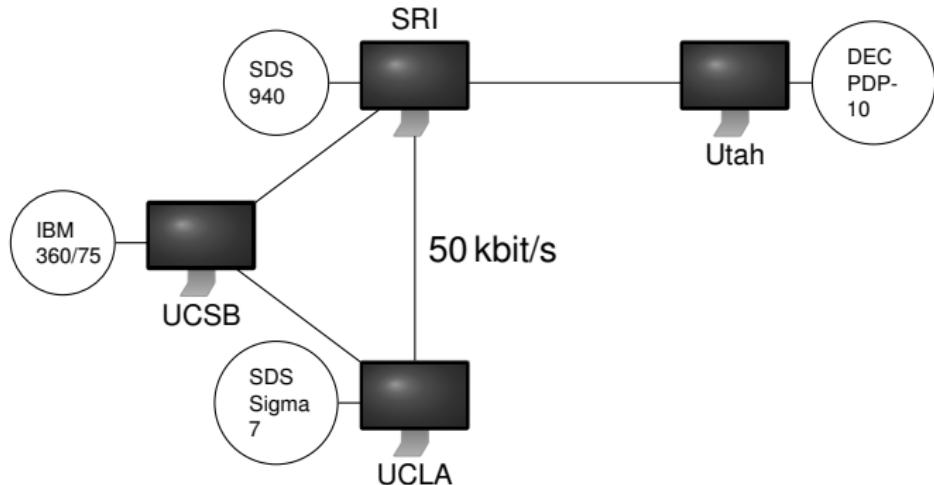
- University of California, Los Angeles (UCLA) 1.9.1969
- Stanford Research Institute (SRI) 1.10.1969
- UC Santa Barbara (UCSB) 1.11.1969



Von der Entstehung bis zum heutigen Internet

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
- University of Utah 12.1969

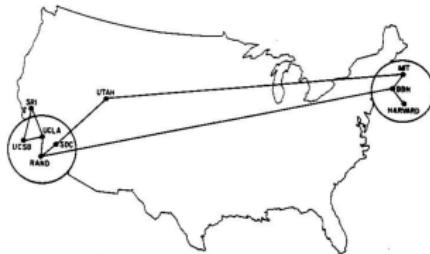


Von der Entstehung bis zum heutigen Internet

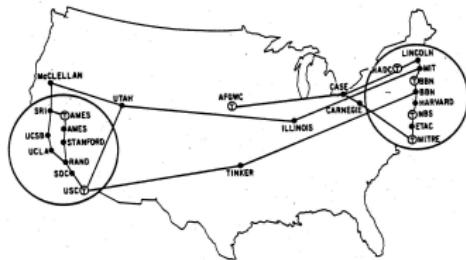
ARPANET von 1969 bis 1977



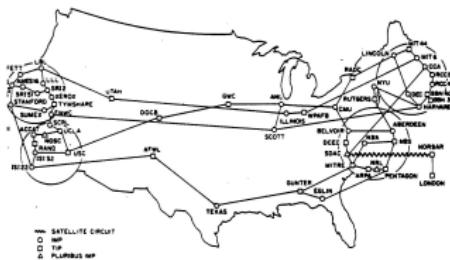
ARPANET 1969, 4 Knoten



ARPANET 1970, 9 Knoten

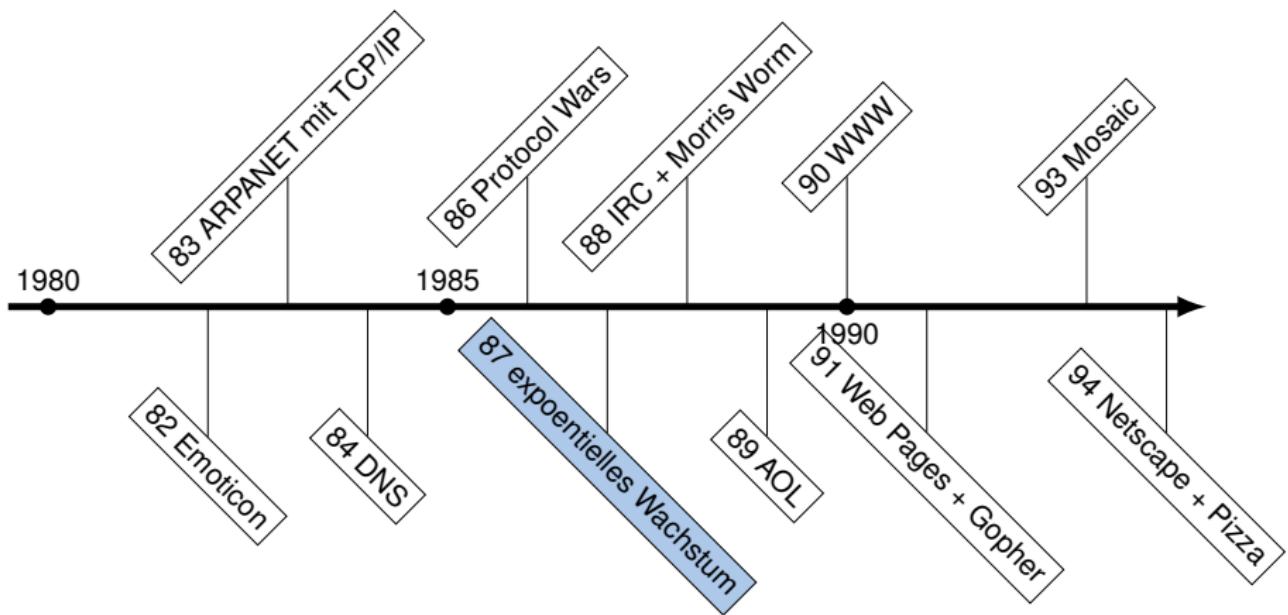


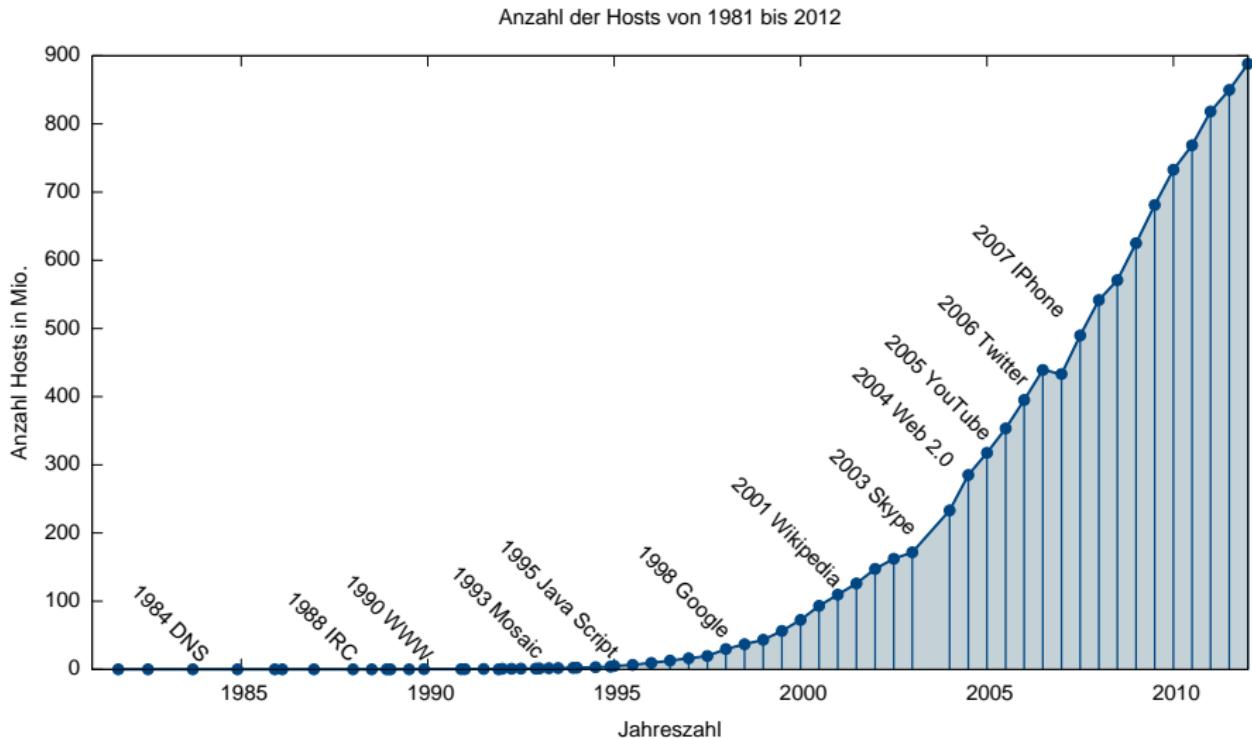
ARPANET 1972, 25 Knoten



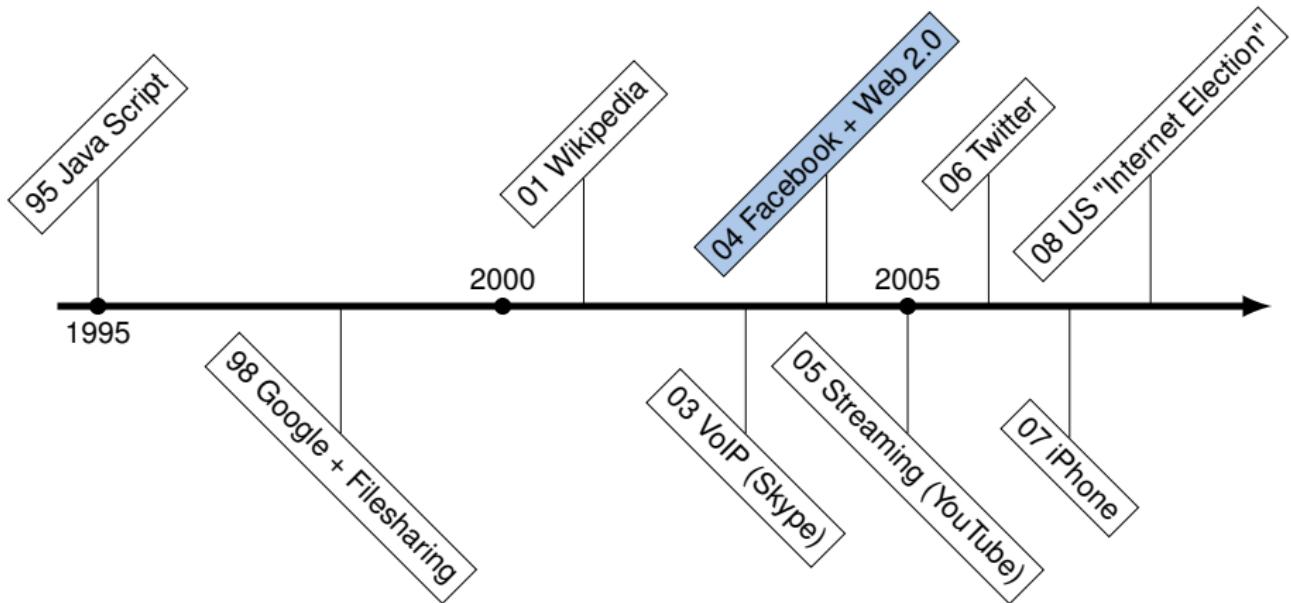
ARPANET 1977, 58 Knoten

Geschichte des Internets: Übersicht von 1980 bis 1994





Geschichte des Internets: Übersicht ab 1994



Von der Entstehung bis zum heutigen Internet

Web 2.0 Meme Map, by Tim O'Reilly [?]

Strategic Positioning: The Web as platform

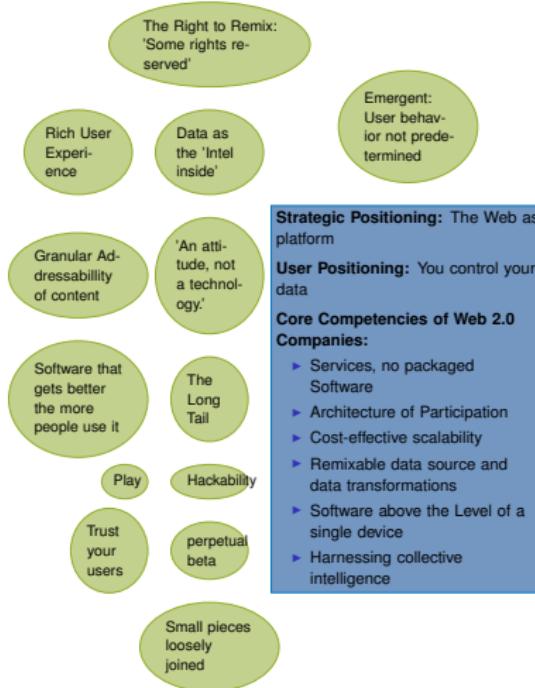
User Positioning: You control your data

Core Competencies of Web 2.0 Companies:

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

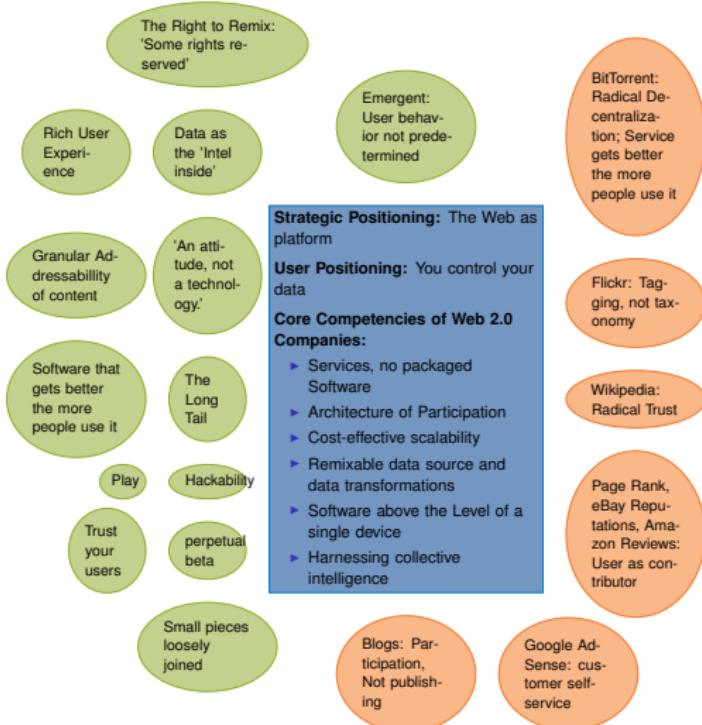
Von der Entstehung bis zum heutigen Internet

Web 2.0 Meme Map, by Tim O'Reilly [?]



Von der Entstehung bis zum heutigen Internet

Web 2.0 Meme Map, by Tim O'Reilly [?]



Das Internet heute



Abbildung: Verbindungen zwischen Autonomen System im Internet (2007) [1]

Das Internet heute

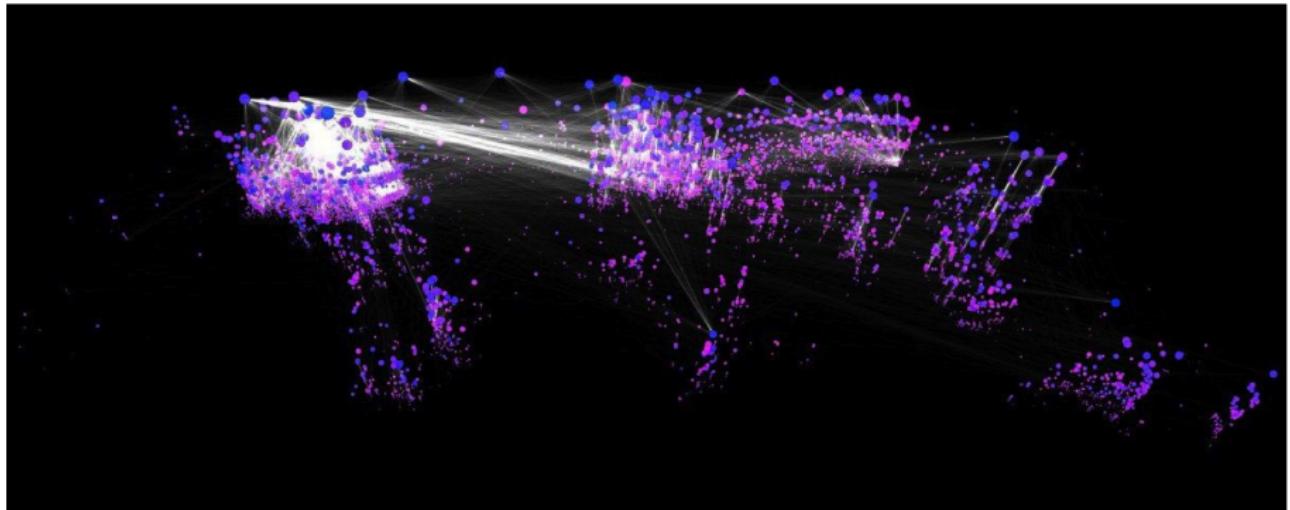


Abbildung: Lange existierende Autonome Systeme blau hervorgehoben [1]

Die Vorlesung im Überblick

Organisatorisches zur Vorlesung

Zusammenfassung der einzelnen Kapitel

Geschichte des Internets

Von der Entstehung bis zum heutigen Internet

Schichtenmodelle

Was sind Schichtenmodelle?

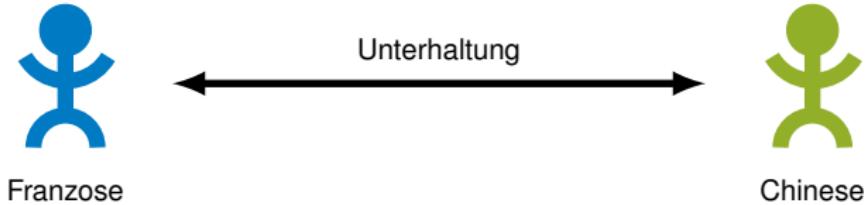
Wozu sind Schichtenmodelle gut?

Das ISO/OSI-Modell

Literaturangaben

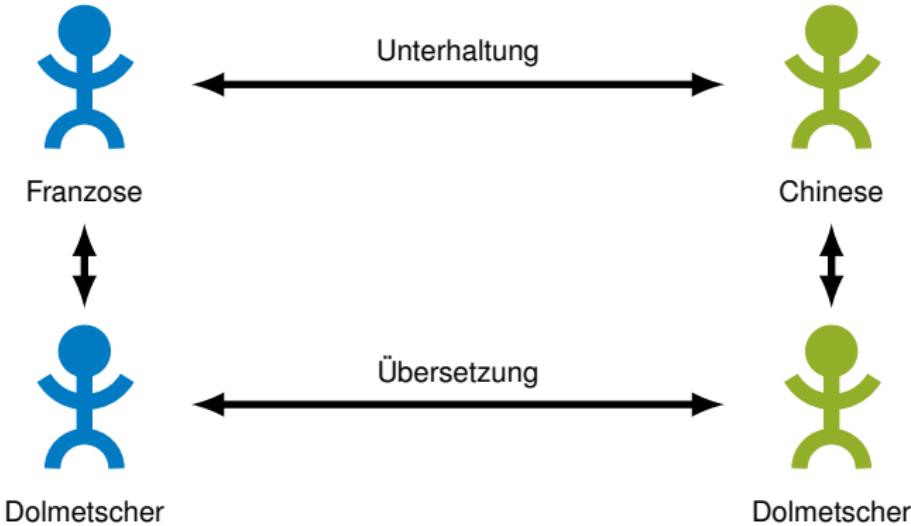
Was sind Schichtenmodelle?

Ein einfaches Beispiel:



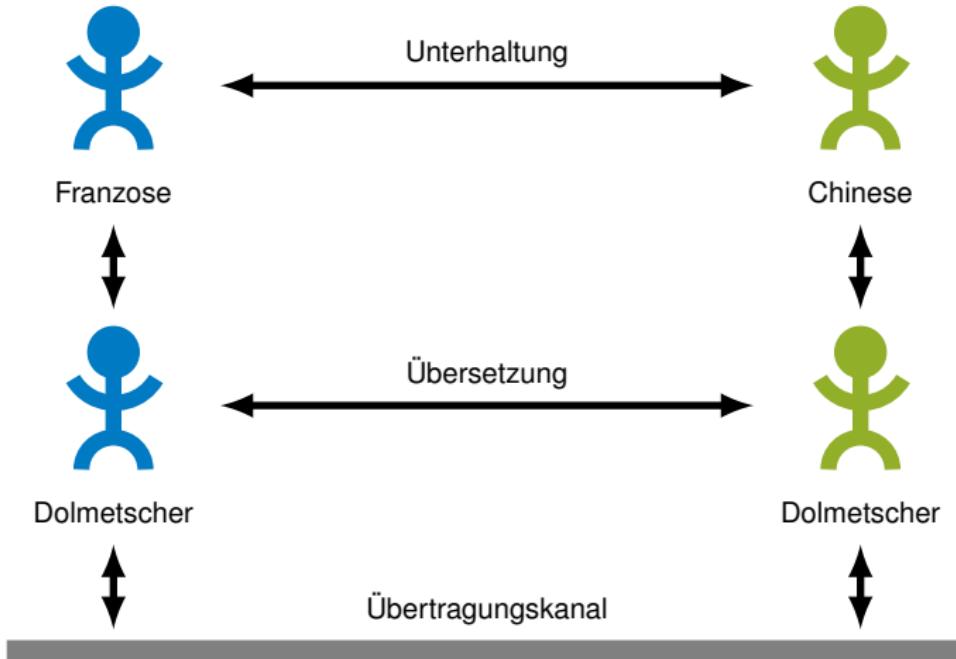
Was sind Schichtenmodelle?

Ein einfaches Beispiel:



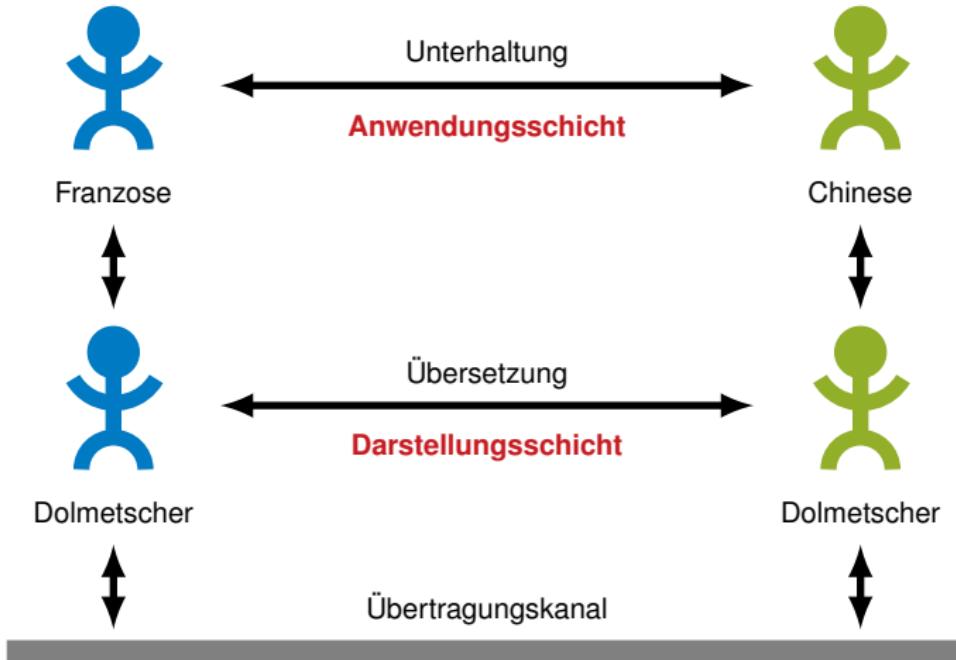
Was sind Schichtenmodelle?

Ein einfaches Beispiel:



Was sind Schichtenmodelle?

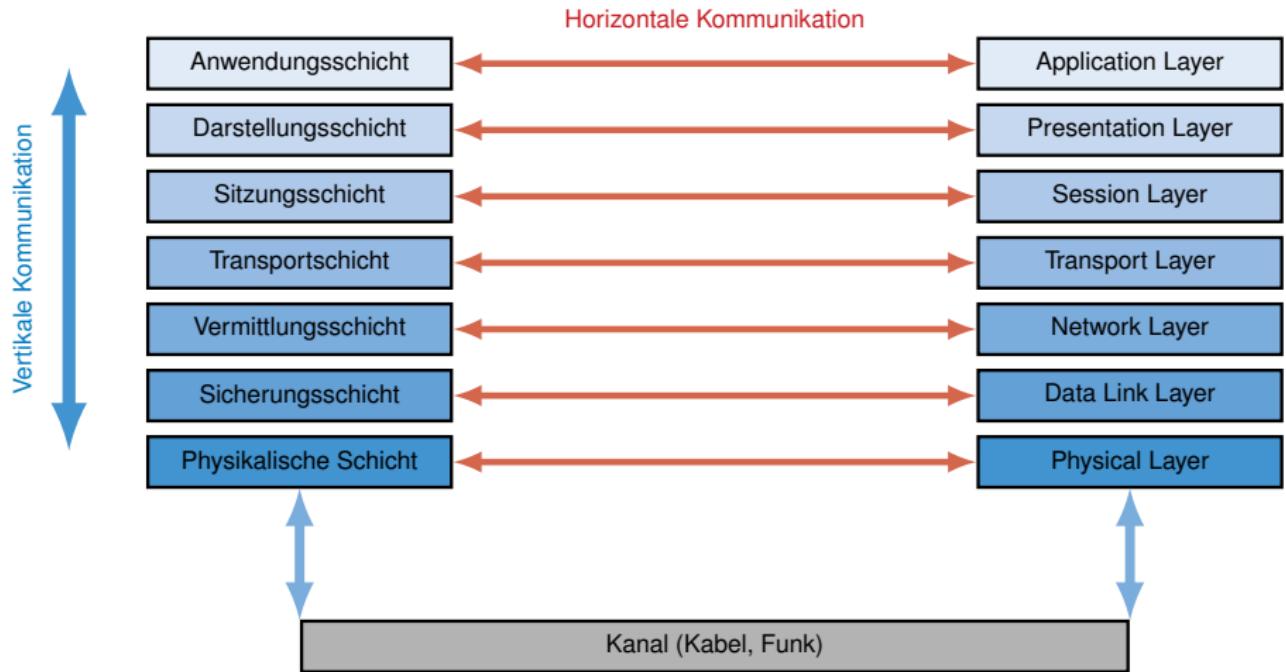
Ein einfaches Beispiel:



- 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

- Entwickelt zwischen 1979 und 1983 von der *International Organization for Standardization (ISO)*
- OSI model = *Open Systems Interconnection model*
- 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:



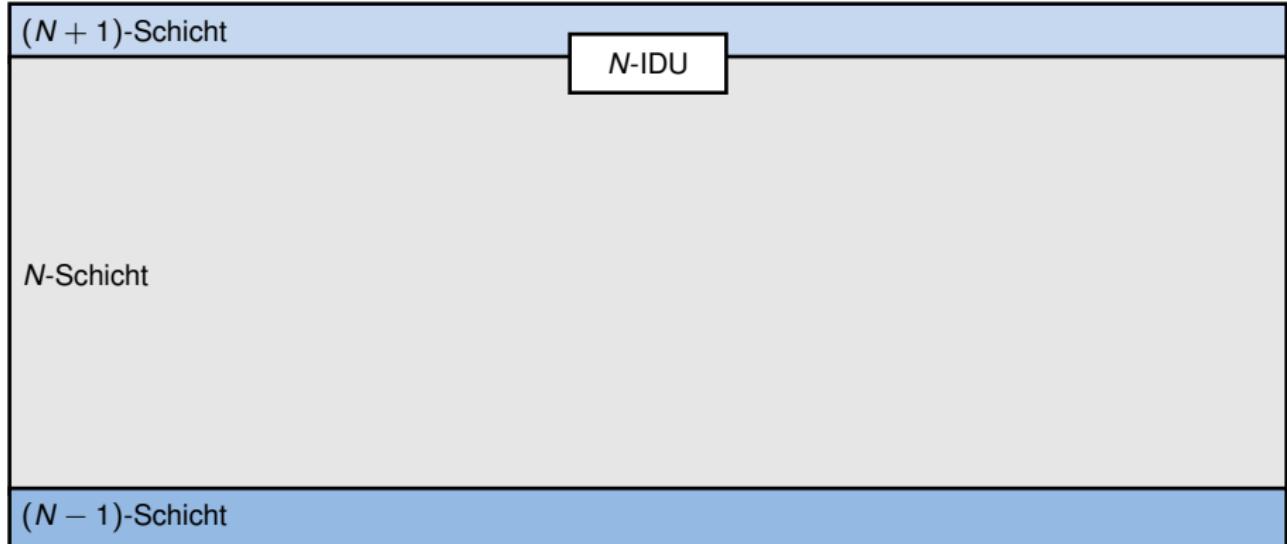
Datenaustausch zwischen Schichten

$(N + 1)$ -Schicht

N -Schicht

$(N - 1)$ -Schicht

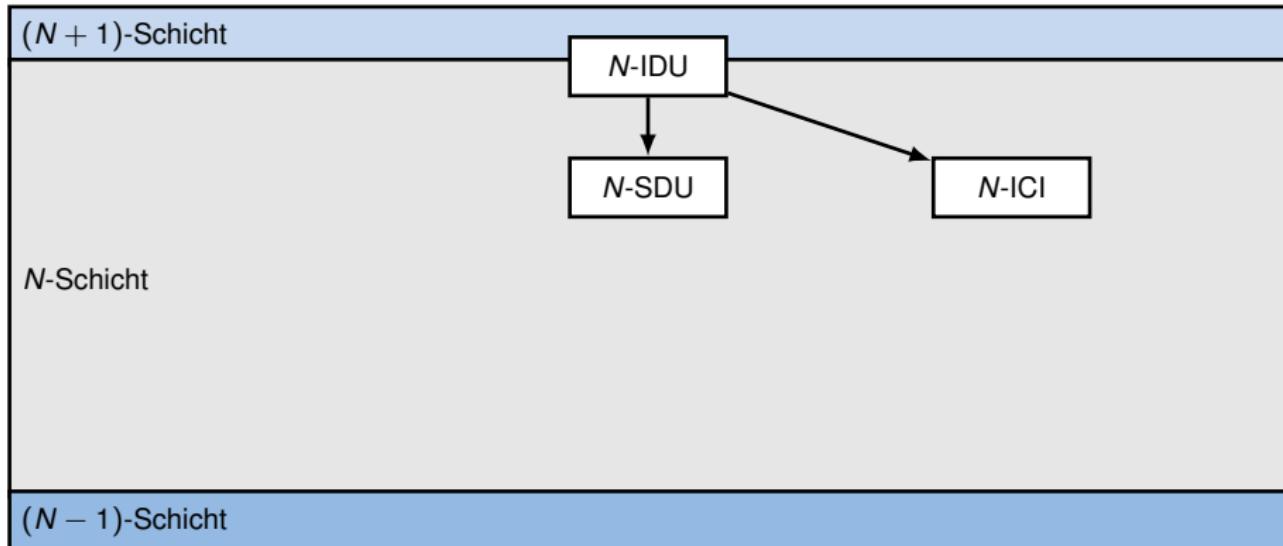
Datenaustausch zwischen Schichten



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

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

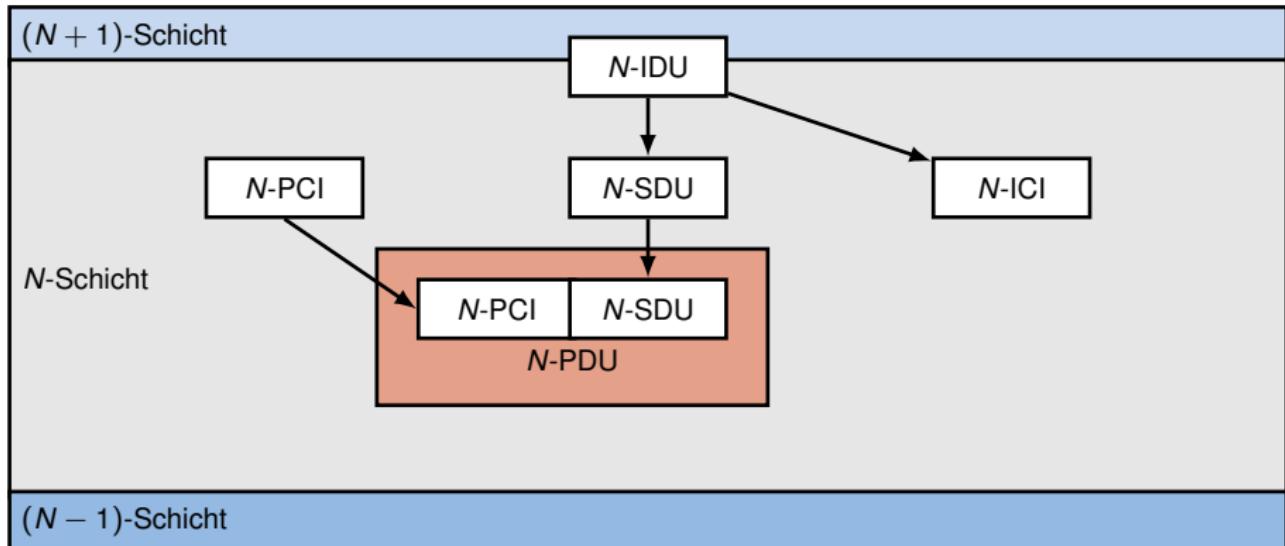
Datenaustausch zwischen Schichten



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).

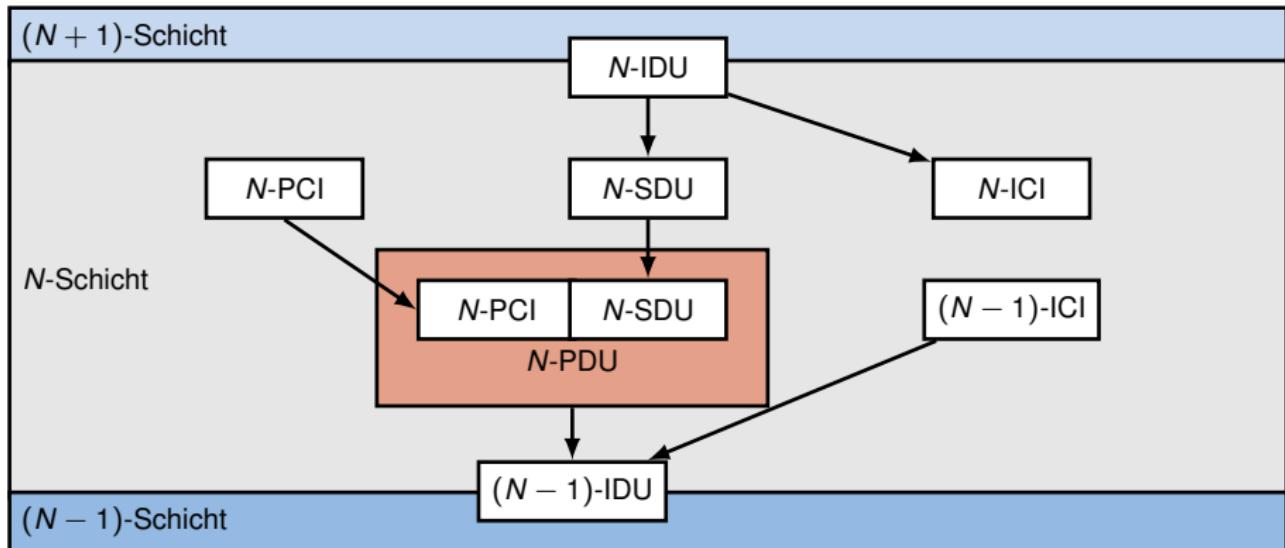
Datenaustausch zwischen Schichten



Die 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



Die N -Schicht nutzt den Dienst der $(N-1)$ -Schicht.

- Sie erzeugt eine $(N-1)$ -ICI, und
- übergibt diese zusammen mit der N -PDU als $(N-1)$ -IDU der nächst niedrigeren Schicht

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

- die (ggf. bearbeiteten) Nutzdaten der $(N - 1)$ -Schicht sowie
- Protokollsteuerungsinformationen (Protocol Control Information - PCI) der N -Schicht

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

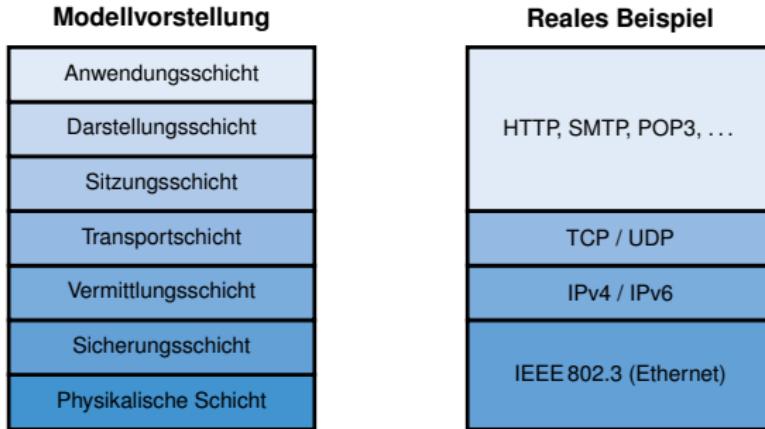
PDUs einiger Schichten haben eigene Bezeichnungen. Man spricht von

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

Diese Unterscheidungen ermöglichen es, implizit die gerade betrachtete Schicht anzugeben. Die Verwendung der Begriffe in der Literatur ist allerdings nicht immer einheitlich.

Schwächen des ISO/OSI-Modells

- Die Trennung der Schichten widerspricht manchmal anderen Interessen (z. B. der Effizienz)
- Einige Protokolle sind nicht klar einer bestimmten Schicht zuzuordnen, bzw. arbeiten sogar auf mehreren Schichten (**Cross Layer**)
- Die Zuordnung von Protokollen auf einzelne Schichten kann vom konkreten Einsatz der Protokolle abhängen



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

Die Vorlesung im Überblick

Organisatorisches zur Vorlesung

Zusammenfassung der einzelnen Kapitel

Geschichte des Internets

Von der Entstehung bis zum heutigen Internet

Schichtenmodelle

Was sind Schichtenmodelle?

Wozu sind Schichtenmodelle gut?

Das ISO/OSI-Modell

Literaturangaben

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