

Grundlagen Rechnernetze und Verteilte Systeme (GRNVS)

IN0010 – SoSe 2019

Prof. Dr.-Ing. Georg Carle

Dr.-Ing. Stephan Günther, Johannes Naab, Henning Stubbe

Lehrstuhl für Netzarchitekturen und Netzdienste
Fakultät für Informatik
Technische Universität München

Organisatorisches zur Vorlesung

Zusammenfassung der einzelnen Kapitel

Entstehung des Internets

Schichtenmodelle

Literaturangaben

Organisatorisches zur Vorlesung

Tutorübungen

Programmieraufgaben

Midterm und Quizze

Modulprüfung und Bonusregelung

Vorlesungsunterlagen

Zusammenfassung der einzelnen Kapitel

Entstehung des Internets

Schichtenmodelle

Literaturangaben

Organisatorisches zur Vorlesung

- Prof. Dr.-Ing. Georg Carle
- Email: carle@tum.de
- Raum: MI 03.05.054
- Sprechstunde: Nach Vereinbarung



Übungsleitung

- Dr.-Ing. Stephan Günther
- Raum: MI 03.05.061



- Henning Stubbe
- Raum: MI 03.05.057

- Johannes Naab
- Raum: MI 03.05.057



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

Vorlesungsbetrieb

- Termine
 - Mo 10:15–11:45 Uhr s.t., MW 2001
 - Di 10:15–11:45 Uhr s.t., MW 2001
- Zentralübung
 - Es ist keine regelmäßige Zentralübung geplant
 - Gelegentlich finden anstelle von Vorlesungen aber Fragestunden zur Klausurvorbereitung sowie Einführungen in die Programmieraufgaben statt

Vorlesungsbetrieb

- Termine
 - Mo 10:15–11:45 Uhr s.t., MW 2001
 - Di 10:15–11:45 Uhr s.t., MW 2001
- Zentralübung
 - Es ist keine regelmäßige Zentralübung geplant
 - Gelegentlich finden anstelle von Vorlesungen aber Fragestunden zur Klausurvorbereitung sowie Einführungen in die Programmieraufgaben statt

Übungsbetrieb

- Tutorübungen
 - Wöchentliche Übungsblätter
 - Bearbeitung während der Tutorübung in Kleingruppen
 - Keine Anwesenheitspflicht / Hausaufgaben
 - Bitte machen Sie sich vor der Übung mit dem Inhalt des jeweiligen Übungsblatts vertraut
- Termine und Anmeldung zu den Tutorübungen
 - Anmeldung über TUMonline
 - Anmeldung freigeschaltet ab **Freitag 26. April 19:00 bzw. 19:30 Uhr** (je nach Gruppe)
(genaue Uhrzeit finden Sie in TUMonline oder auf grnvs.net/tutorgruppen.txt)
 - Übungsbeginn in der 2. Vorlesungswoche (ab 29. April)

Vorlesungsbetrieb

- Termine
 - Mo 10:15–11:45 Uhr s.t., MW 2001
 - Di 10:15–11:45 Uhr s.t., MW 2001
- Zentralübung
 - Es ist keine regelmäßige Zentralübung geplant
 - Gelegentlich finden anstelle von Vorlesungen aber Fragestunden zur Klausurvorbereitung sowie Einführungen in die Programmieraufgaben statt

Übungsbetrieb

- Tutorübungen
 - Wöchentliche Übungsblätter
 - Bearbeitung während der Tutorübung in Kleingruppen
 - Keine Anwesenheitspflicht / Hausaufgaben
 - Bitte machen Sie sich vor der Übung mit dem Inhalt des jeweiligen Übungsblatts vertraut
- Termine und Anmeldung zu den Tutorübungen
 - Anmeldung über TUMonline
 - Anmeldung freigeschaltet ab Freitag 26. April 19:00 bzw. 19:30 Uhr (je nach Gruppe)
(genaue Uhrzeit finden Sie in TUMonline oder auf grnvs.net/tutorgruppen.txt)
 - Übungsbeginn in der 2. Vorlesungswoche (ab 29. April)

Schülerstudenten

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

Tutorübungen

Es werden insgesamt **45 Tutorgruppen** angeboten:

Tag	Start	Ende	# Gruppen
Montag	12:00	14:00	7
Montag	16:00	18:00	4
Dienstag	08:00	10:00	1
Dienstag	12:00	14:00	5
Dienstag	14:00	16:00	4
Dienstag	16:00	18:00	2
Mittwoch	08:00	10:00	1
Mittwoch	10:00	12:00	3
Mittwoch	12:00	14:00	3
Mittwoch	14:00	16:00	2
Mittwoch	16:00	18:00	1

Tag	Start	Ende	# Gruppen
Donnerstag	08:00	10:00	1
Donnerstag	10:00	12:00	4
Donnerstag	12:00	14:00	3
Donnerstag	14:00	16:00	1
Donnerstag	16:00	18:00	1
Freitag	10:00	12:00	1
Freitag	14:00	16:00	1

- Die jeweils aktuelle Liste mit Übungsterminen, Tutoren und ggf. Änderungen finden Sie auf grnvs.net/tutorgruppen.txt.
- Es steht aktuell noch nicht fest, welche Gruppe von welchem Tutor gehalten wird.
- Es werden **zusätzlich 6 Programmierübungen** (ohne Anmeldung) zur Unterstützung bei den Programmieraufgaben angeboten.

Bitte verteilen Sie sich auf gleichzeitig stattfindende Parallelgruppen!

Es gibt semesterbegleitend voraussichtlich 4 Programmieraufgaben:

- Einzelabgaben
- Anmeldung über Moodle (ab nächster Woche)
- Geplante Aufgaben:
 1. Git
 2. Statistiken über empfangene Rahmentypen
 3. Traceroute
 4. 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)

Programmieraufgaben

Tutorgruppen zu den Programmieraufgaben

Infolge des unterschiedlichen Vorwissens der einzelnen Vorlesungsteilnehmer bieten wir **zusätzlich** 6 spezielle Tutorgruppen an:

Tag	Raum	Start	Ende	
Montag	12:00	14:00	03.07.023	Philip H.
Montag	14:00	16:00	03.07.023	Adrian P.
Dienstag	12:00	14:00	03.07.023	Adrian P.
Dienstag	14:00	16:00	03.07.023	Christian S.
Mittwoch	12:00	14:00	03.07.023	Christian S.
Donnerstag	12:00	14:00	03.07.023	Philip H.

- 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

Ergebnisse werden nicht direkt anerkannt, aber **eigener** Code vom Vorjahr darf wieder abgeben werden sofern

- ein deutlicher Hinweis auf die Abgabe vom Vorjahr enthalten ist und
- der Code der Aufgabenstellung und den Abgeaberichtlinien entspricht.

Midterm

Wir bieten eine freiwillige Midterm-Klausur:

- schriftlich, 45 Minuten
- closed-book, ein nicht-programmierbarer Taschenrechner erlaubt
- **Fr 14.06.2019, 17:30 – 18:15 Uhr**
- Anmeldung¹ über TUMonline vom 24.05.2019 bis 07.06.2019
- Multiple Choice Multiple Answer möglich (max. eine Aufgabe)
- kein „Programmieren am Papier“

Quizze

Während der Vorlesung finden über das Semester verteilt unangekündigte Quizze zur Selbsteinschätzung statt:

- jeweils 10 – 15 Minuten
- schriftlich, Multiple Choice Multiple Answer
- digitale Auswertung und Online-Einsicht mit **TUMexam**²

¹ Bitte beachten Sie, dass sich der Anmeldezeitraum ggf. ändern kann. Informieren Sie sich bitte rechtzeitig in TUMonline über die aktuellen Fristen.

² TUMexam wird seit 2015 am Lehrstuhl für Netzarchitekturen und Netzdienste entwickelt. Mehr auf <https://tumexam.de>.

Endterm / Retake

- schriftlich, 90 Minuten, voraussichtlich 90 Punkte
- closed-book, ein nicht-programmierbarer Taschenrechner erlaubt
- Endterm: **Mo 29.07.2019, 10:30 – 12:00 Uhr**, Anmeldung¹ vom 27.05.2019 bis 30.06.2019
- Retake: **Di 08.10.2019, 13:30 – 15:00 Uhr**, Anmeldung¹ vom 09.09.2019 bis 23.09.2019
- Multiple Choice Multiple Answer möglich (max. 20 %)
- kein „Programmieren am Papier“

Bonusregelung

- In der Midterm und den Programmieraufgaben können **jeweils bis zu 10 Bonuspunkte** erreicht werden.
- Die **Gesamtzahl der anrechenbaren Bonuspunkte beträgt 15**, d. h. sofern Sie mehr als 15 Bonuspunkte erzielen, werden dennoch nur 15 angerechnet.
- Die Bonuspunkte werden auf das Ergebnis der Modulprüfung addiert, sofern diese **ohne Bonus mindestens mit der Note 4,0 bestanden** wurde.
- Der Bonus wird auch auf die Retake angerechnet.
- Die Quizze sind **nicht** Bestandteil der Bonusregelung.

¹ Bitte beachten Sie, dass sich der Anmeldezeitraum ggf. ändern kann. Informieren Sie sich bitte rechtzeitig in TUMonline über die aktuellen Fristen. Wenn Sie bereits zur Midterm angemeldet waren, werden Sie automatisch zur Endterm angemeldet. Bitte melden Sie sich **nicht** zur Midterm an, wenn Sie nur an der Endterm teilnehmen wollen. Sollten Sie zur Midterm teilnehmen wollen aber erst die Retake mitschreiben, melden Sie sich bitte von der Endterm ab und erst zur Retake an. Sie helfen uns damit bei der Planung zur Aufsicht und Anzahl der Hörsäle bzw. Standorte!

Wissenschaftliches Fehlverhalten, Unterschleif und Plagiate

- Programmieraufgaben erfolgen in Einzelarbeit
 - Diskussion von Konzepten und Ansätzen in Gruppe möglich
 - Coding selbst muss alleine erfolgen
- Es werden Plagiatschecks durchgeführt
- Eindeutige Hinweise auf Plagiate führen zum Ausschluss aller Beteiligter aus dem Bonussystem
 - alle Programmieraufgaben, die Midterm-Klausur und die Quizes werden mit 0 Punkten bewertet
- Kopien und Adaptionen externer Quellen wie z. B. StackOverflow sind eindeutig zu kennzeichnen
- Umbenennungen und Umsortierungen stellen **keine** signifikante Eigenleistung dar
- Täuschung und Ordnungsverstöße in einer Prüfung werden gemäß APSO gemeldet
 - Gesamter Kurs nicht bestanden
 - Eine solche Prüfung kann maximal einmal wiederholt werden (§24)
 - Täuschung kann den Ausschluss aus dem Studiengang nach sich ziehen (§22)

- Vorlesungsunterlagen und Übungsmaterialien werden über HTTP und Git bereitgestellt:

<https://grnvs.net>
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 über Moodle erforderlich (selbes Prozedere wie Anmeldung zu den Programmieraufgaben)
- Für den Zugriff auf Moodle ist die Vorlesungsanmeldung in TUMonline erforderlich
- Die Vorlesung wird dieses Jahr nicht aufgezeichnet
 - Vorlesungsaufzeichnung von 2013 auf media.net.in.tum.de
 - **Achtung:** Inhalte sind nicht deckungsgleich mit diesem Jahr

Hinweis: Aktuelle Informationen zur Vorlesung erhalten Sie ausschließlich über Moodle. Dies betrifft insbesondere Informationen zum Stand der Vorlesung oder Updates zu Klausuren.

Organisatorisches zur Vorlesung

Zusammenfassung der einzelnen Kapitel

Entstehung des Internets

Schichtenmodelle

Literaturangaben

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

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 Fehlerkennung

- 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

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

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)

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

Organisatorisches zur Vorlesung

Zusammenfassung der einzelnen Kapitel

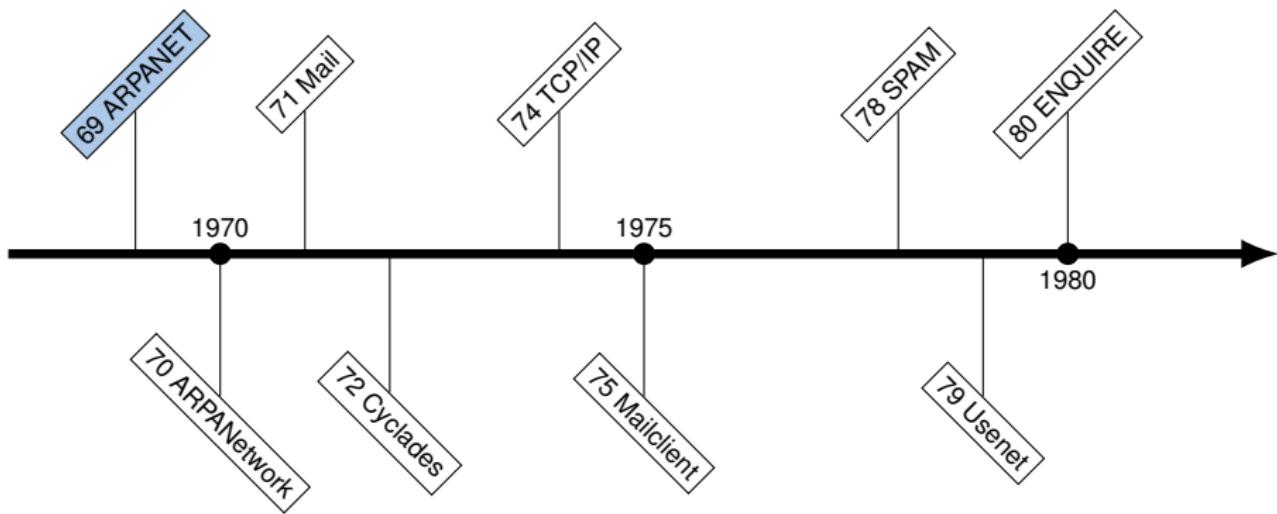
Entstehung des Internets

Schichtenmodelle

Literaturangaben

Entstehung des Internets

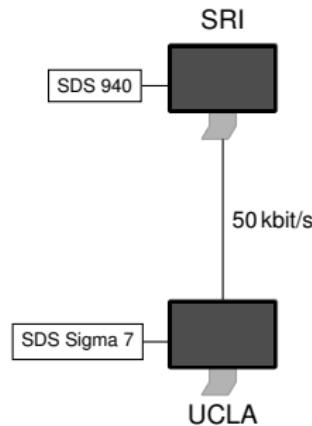
Übersicht bis 1980



Entstehung des Internets

ARPANET mit den ersten 4 Knoten

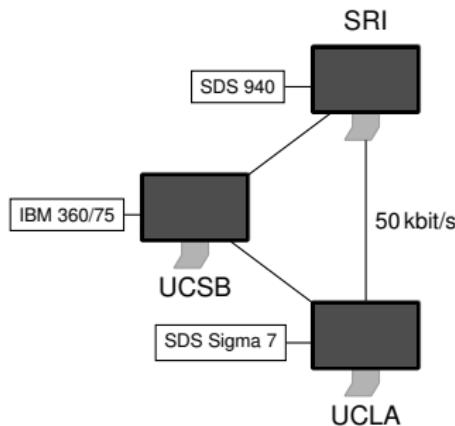
- University of California, Los Angeles (UCLA) 1.9.1969
- Stanford Research Institute (SRI) 1.10.1969



Entstehung des Internets

ARPANET mit den ersten 4 Knoten

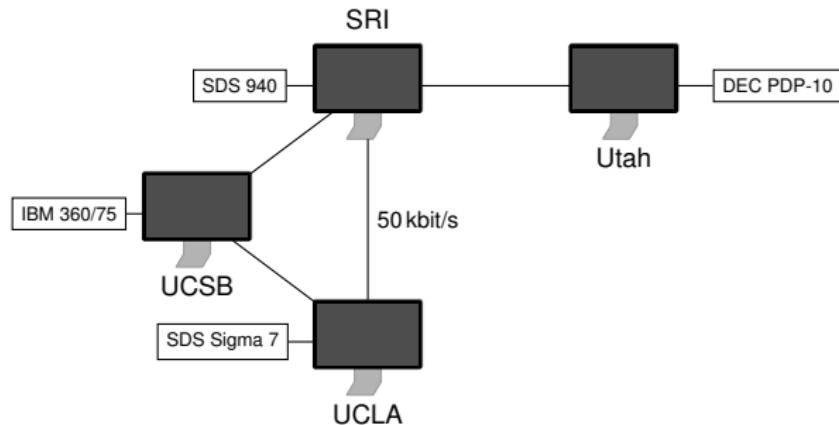
- University of California, Los Angeles (UCLA) 1.9.1969
- Stanford Research Institute (SRI) 1.10.1969
- UC Santa Barbara (UCSB) 1.11.1969



Entstehung des Internets

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

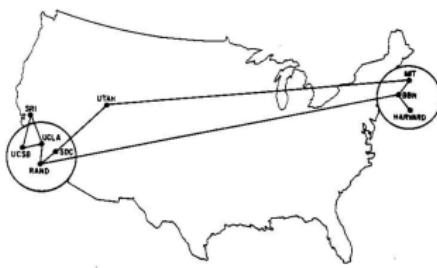


Entstehung des Internets

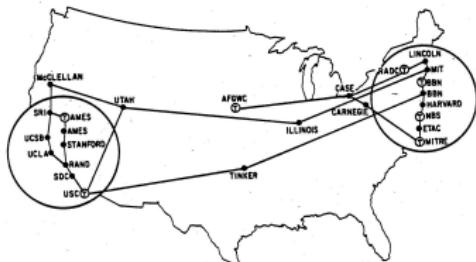
ARPANET von 1969 bis 1977



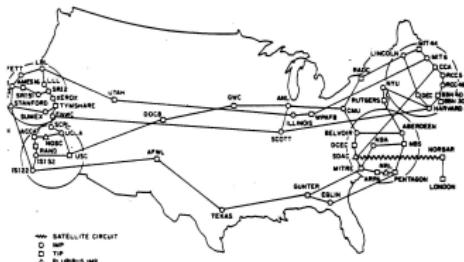
ARPANET 1969, 4 Knoten



ARPANET 1970, 9 Knoten



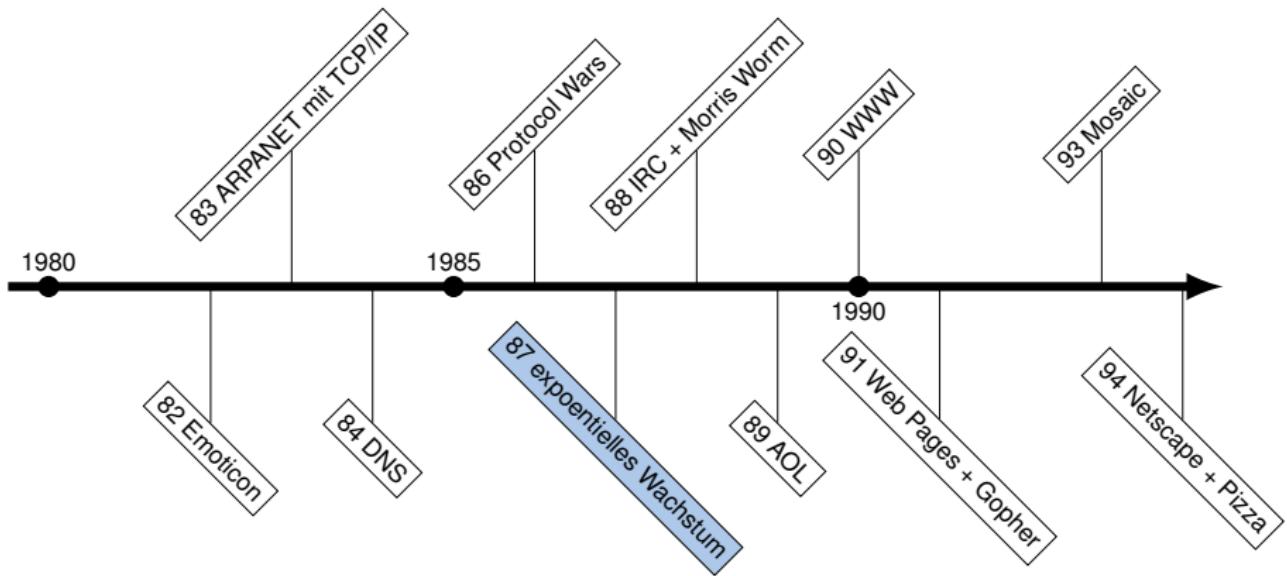
ARPANET 1972, 25 Knoten



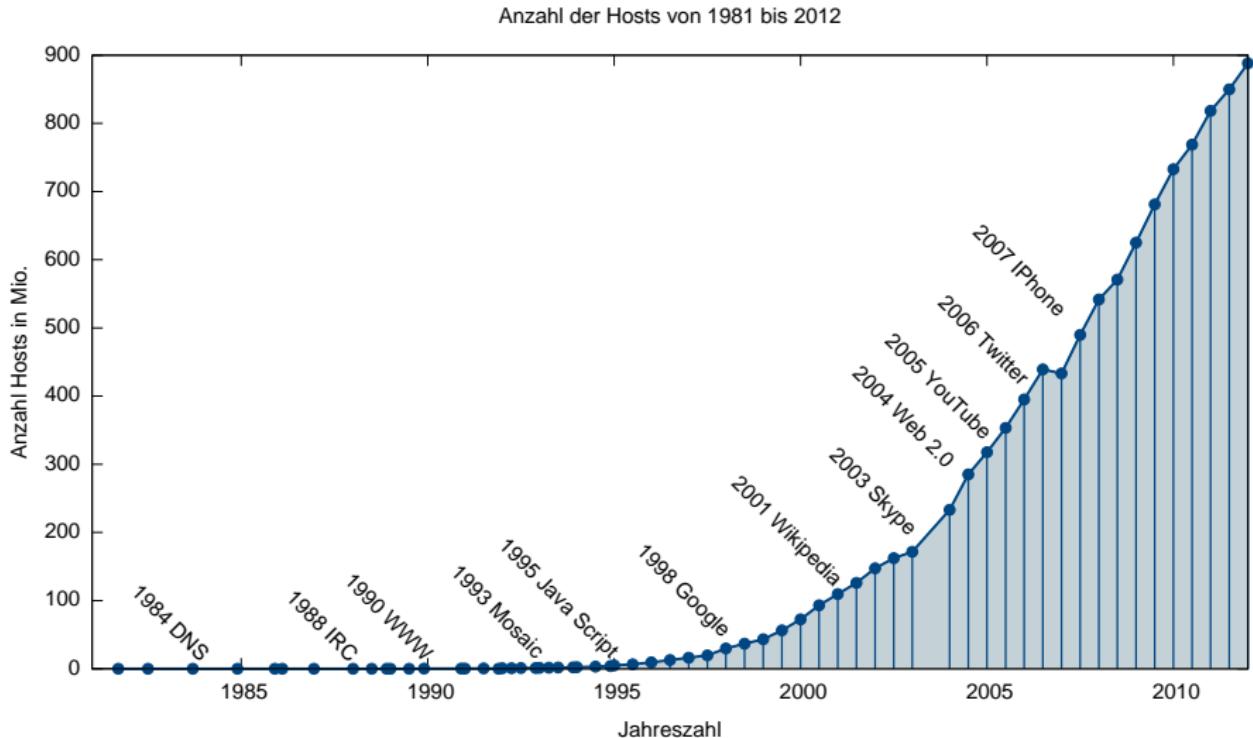
ARPANET 1977, 58 Knoten

Entstehung des Internets

Übersicht von 1980 bis 1994

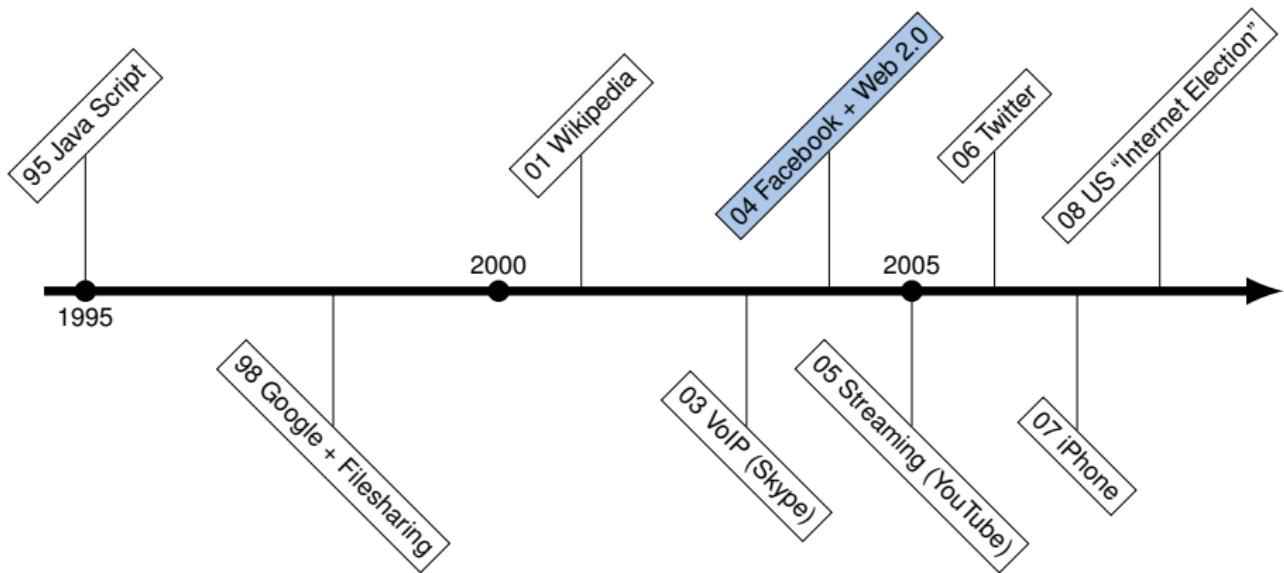


Entstehung des Internets



Entstehung des Internets

Übersicht ab 1994



Entstehung des Internets

Web 2.0 Meme Map, by Tim O'Reilly [2]

Strategic Positioning: The Web as platform

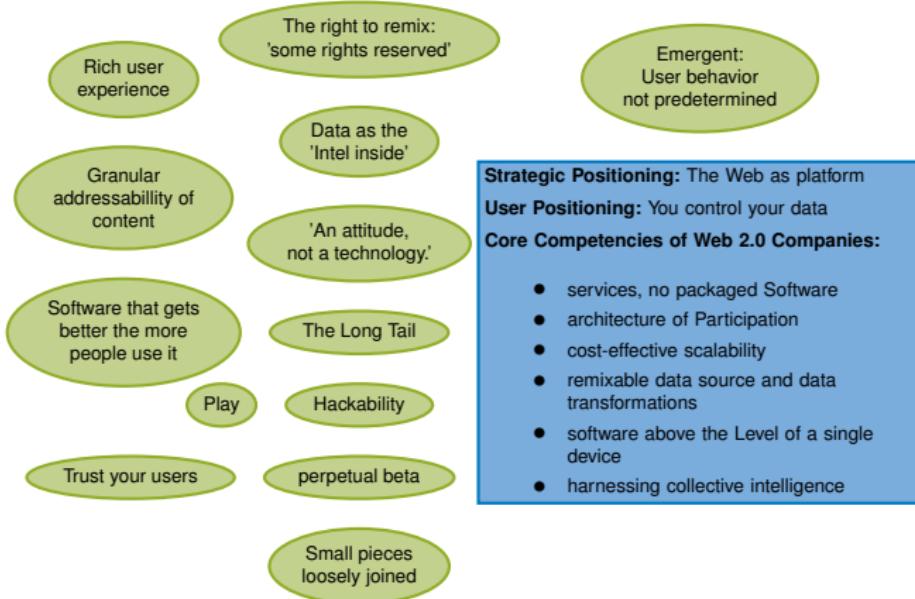
User Positioning: You control your data

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

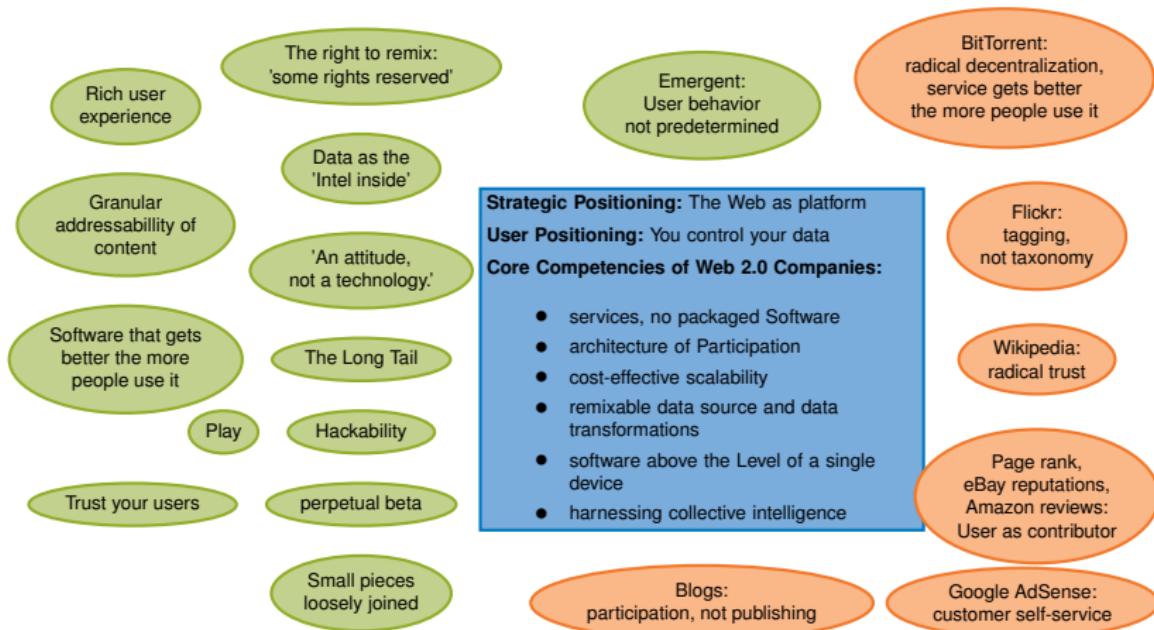
Entstehung des Internets

Web 2.0 Meme Map, by Tim O'Reilly [2]



Entstehung des Internets

Web 2.0 Meme Map, by Tim O'Reilly [2]



Entstehung des Internets

Das Internet heute



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

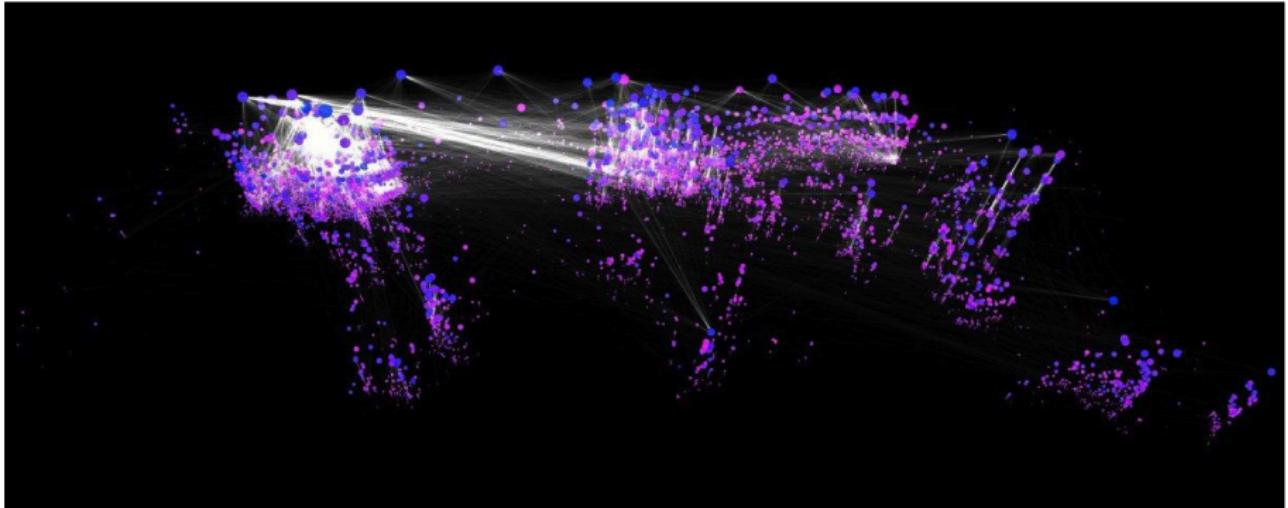


Abbildung 2: Lange existierende Autonome Systeme blau hervorgehoben [1]

Organisatorisches zur Vorlesung

Zusammenfassung der einzelnen Kapitel

Entstehung des Internets

Schichtenmodelle

Was sind Schichtenmodelle?

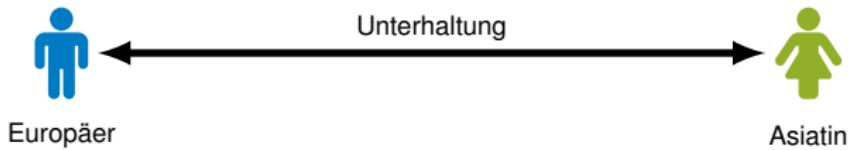
Wozu sind Schichtenmodelle gut?

Das ISO/OSI-Modell

Literaturangaben

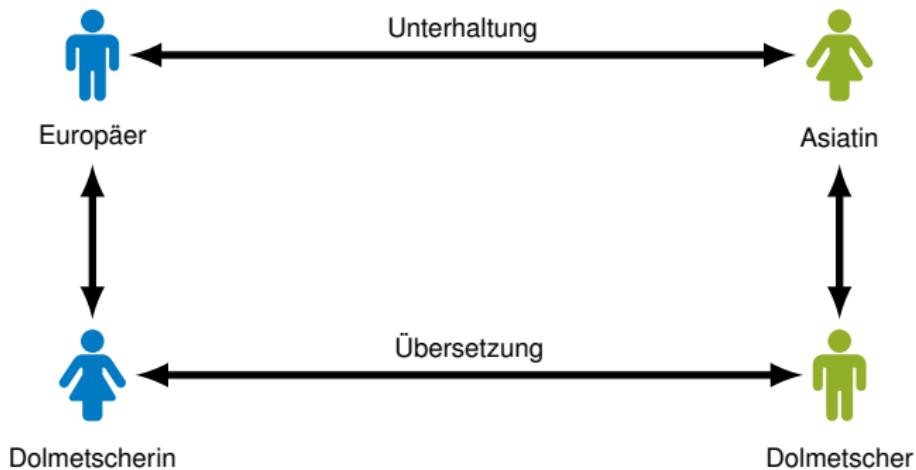
Was sind Schichtenmodelle?

Ein einfaches Beispiel



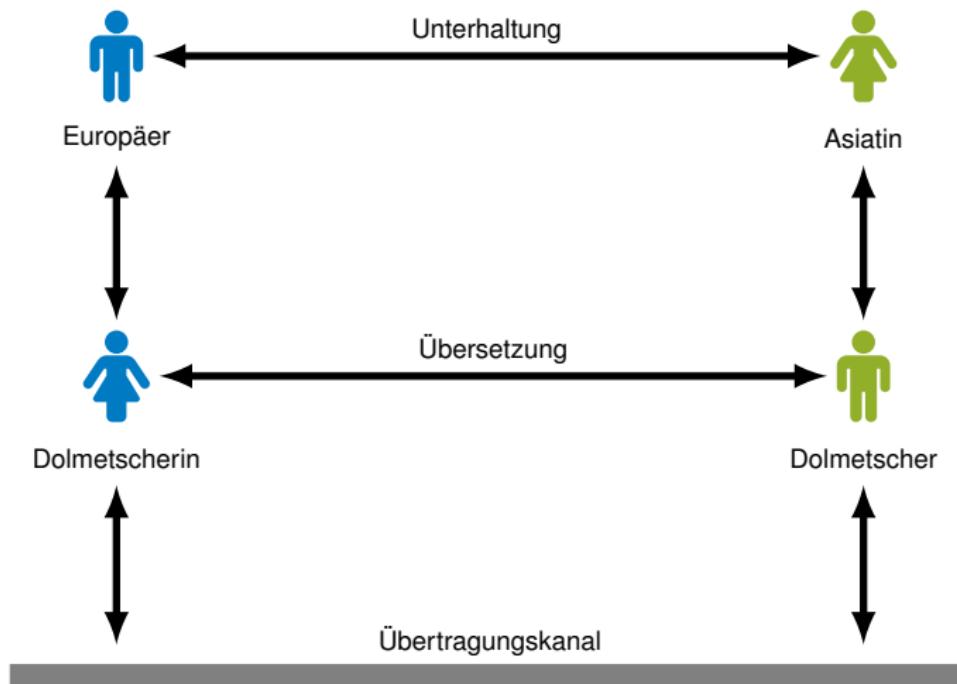
Was sind Schichtenmodelle?

Ein einfaches Beispiel



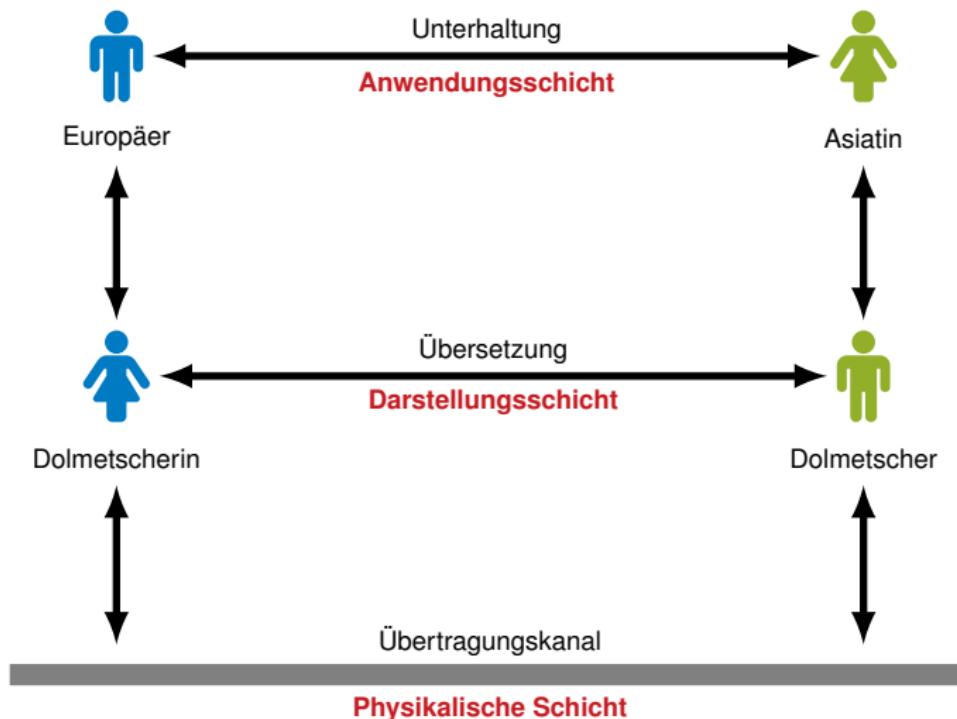
Was sind Schichtenmodelle?

Ein einfaches Beispiel



Was sind Schichtenmodelle?

Ein einfaches Beispiel



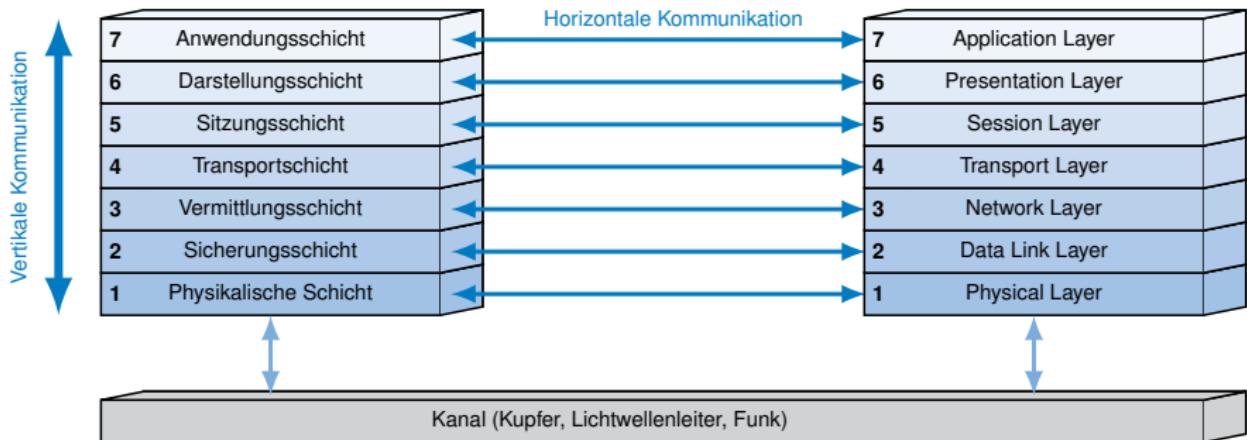
Wozu sind Schichtenmodelle gut?

- 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

- Entwickelt zwischen 1979 und 1983 von der International Organization for Standardization (ISO)
- OSI model = Open Systems Interconnection model
- Unterteilt die Architektur des Systems, das die Kommunikationsfunktionalität erbringt, in insgesamt 7 Schichten
- Jede Schicht erbringt bestimmte Dienste (z. B. Aufteilen einer Nachricht in kleinere Pakete)
- Keine Aussage, wie diese Dienste zu erbringen sind

Das ISO/OSI-Modell

Schematische Darstellung des OSI-Modells



Das ISO/OSI-Modell

Datenaustausch zwischen Schichten

$(N + 1)$ -Schicht

N -Schicht

$(N - 1)$ -Schicht

Das ISO/OSI-Modell

Datenaustausch zwischen Schichten

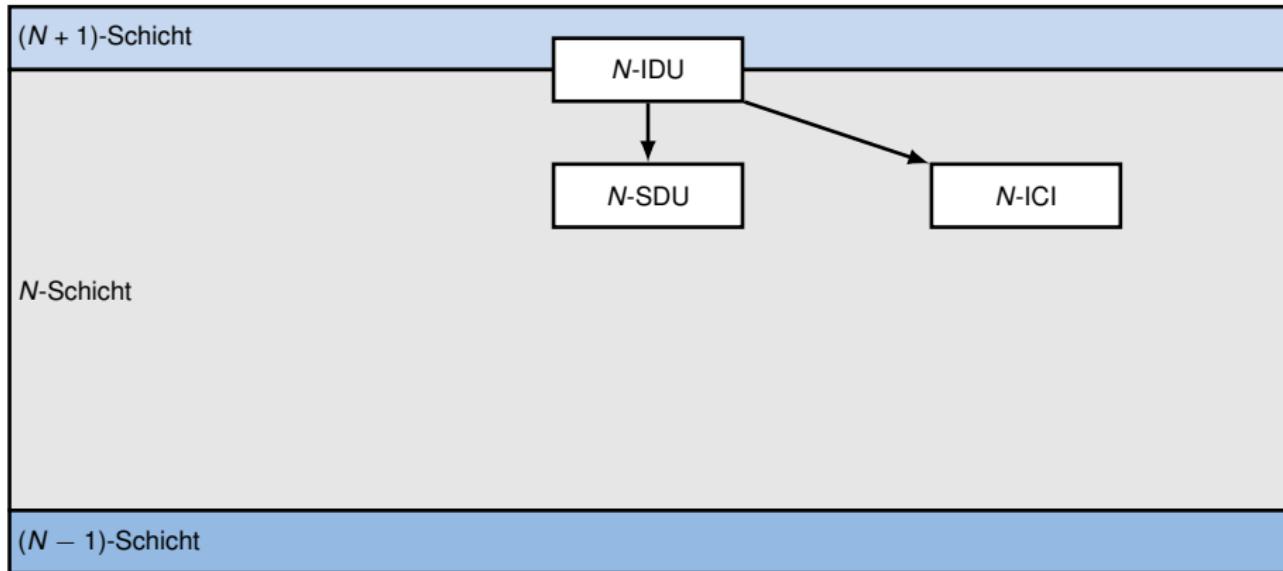


Die $(N + 1)$ -Schicht nimmt Dienste der N-Schicht in Anspruch:

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

Das ISO/OSI-Modell

Datenaustausch zwischen Schichten

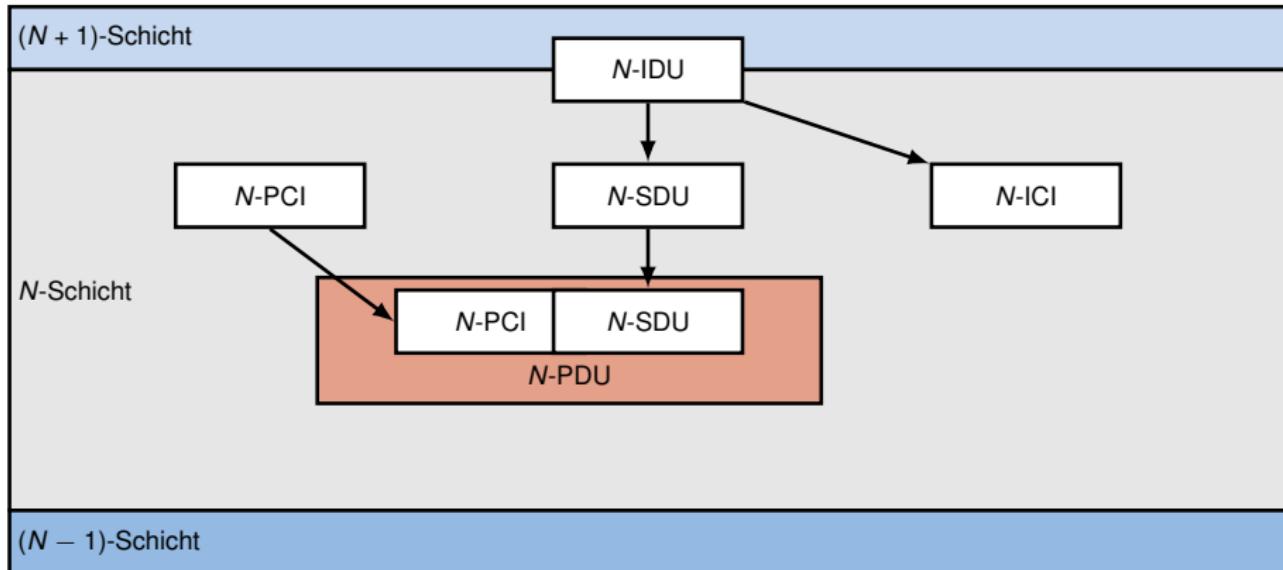


N-IDU enthält aus Sicht der N-Schicht

- Nutzdaten ([Service Data Unit \(SDU\)](#)) und
- Kontrollinformationen ([Interface Control Information \(ICI\)](#)), welche zum Erbringen des Dienstes notwendig sind (z. B. Länge der SDU oder Adressinformationen).

Das ISO/OSI-Modell

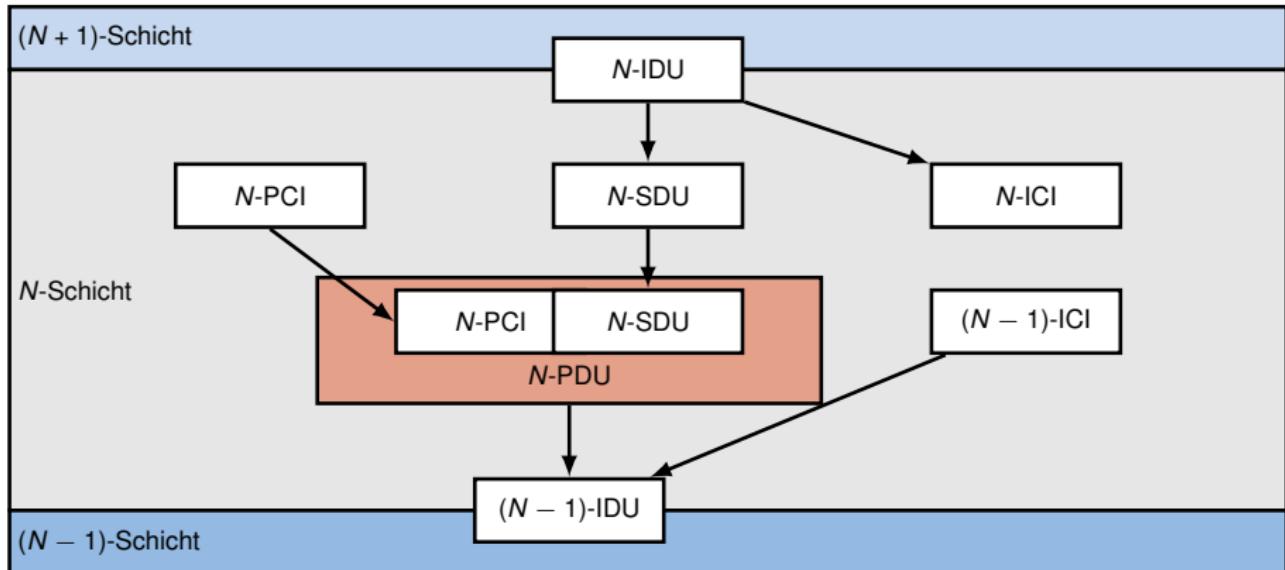
Datenaustausch zwischen Schichten



Die N-Schicht

- erbringt auf der N-SDU die angeforderten Dienste,
- fügt sogenannte **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, d. h. die SDU, 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

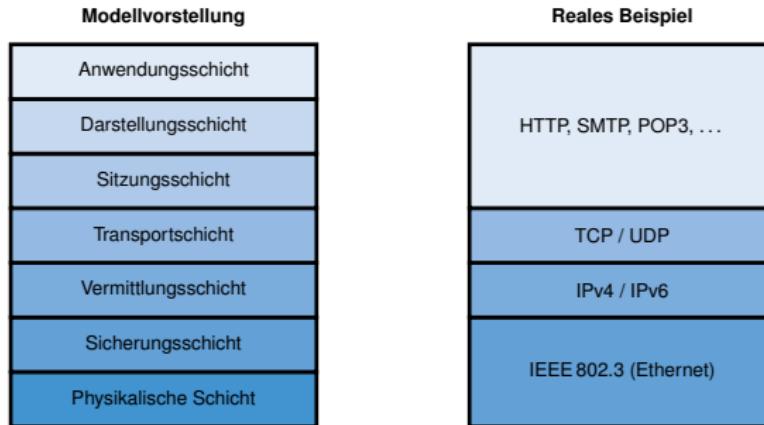
- **Segmenten** auf der Transportschicht,
- **Paketen** auf der Vermittlungsschicht bzw.
- **Rahmen** (engl. **Frames**) auf der Sicherungsschicht.

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

Das ISO/OSI-Modell

Schwächen des ISO/OSI-Modells

- Die Trennung der Schichten widerspricht manchmal anderen Interessen (z. B. der Effizienz)
- Einige Protokollmechanismen 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 [3].

Organisatorisches zur Vorlesung

Zusammenfassung der einzelnen Kapitel

Entstehung des Internets

Schichtenmodelle

Literaturangaben

- [1] C. Harrison.
World City-to-City Connections.
<http://www.chrisharrison.net/index.php/Visualizations/InternetMap>.
- [2] T. O'Reilly.
O'Reilly Network: What Is Web 2.0, Sept. 2005.
- [3] E. Stein.
Taschenbuch Rechnernetze und Internet, chapter Das OSI-Modell, pages 22–28.
Fachbuchverlag Leipzig, 2. edition, 2004.