

# Systeme II

## 4. Die Anwendungsschicht

Thomas Janson<sup>°</sup>, Kristof Van Laerhoven\*, Christian Ortolf<sup>°</sup>

Folien: Christian Schindelhauer<sup>°</sup>

Technische Fakultät

<sup>°</sup>: Rechnernetze und Telematik, \*: Eingebettete Systeme

Albert-Ludwigs-Universität Freiburg

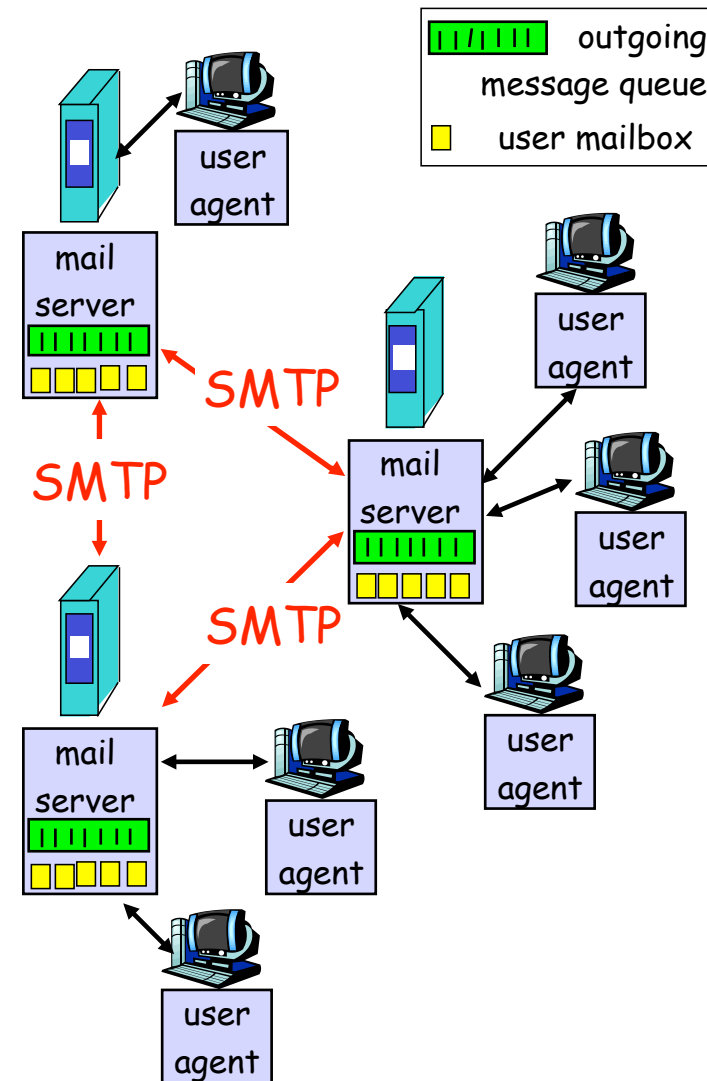
Version 13.05.2015

- In Deutschland wurde am 3. August 1984 die erste Internet-E-Mail an der Universität Karlsruhe empfangen
- die E-Mail wurde einen Tag zuvor an der US-amerikanischen Plattform CSNET aus Cambridge abgeschickt
- **Inhalt:** „Wilkommen in CSNET! Michael, This is your official welcome to CSNET.“

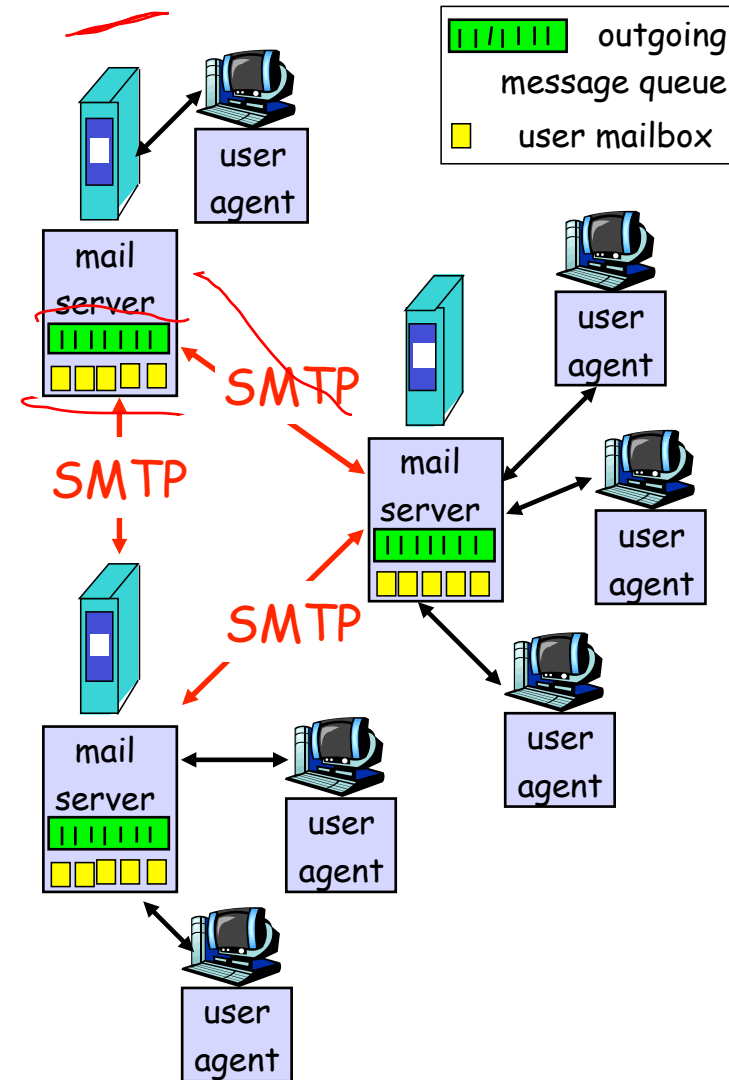


[Wikipedia]

- Hauptkomponenten
  - user agents
  - mail servers
  - simple mail transfer protocol: SMTP
- User Agent
  - Mail Client
  - Erstellen, ändern und lesen von E-Mail-Nachrichten
  - z.B. Eudora, Outlook, pine, Mozilla Thunderbird
  - ausgehende und ankommende Nachrichten werden auf dem Server gespeichert



- Mailbox speichert eingehende Nachrichten für den User ( ■■■■■ )
- Warteschlange (queue) für zu versendende Nachrichten ( ■■■■■ )
- SMTP-Protokoll zwischen Mail-Servern um E-Mail-Nachrichten zu verschicken

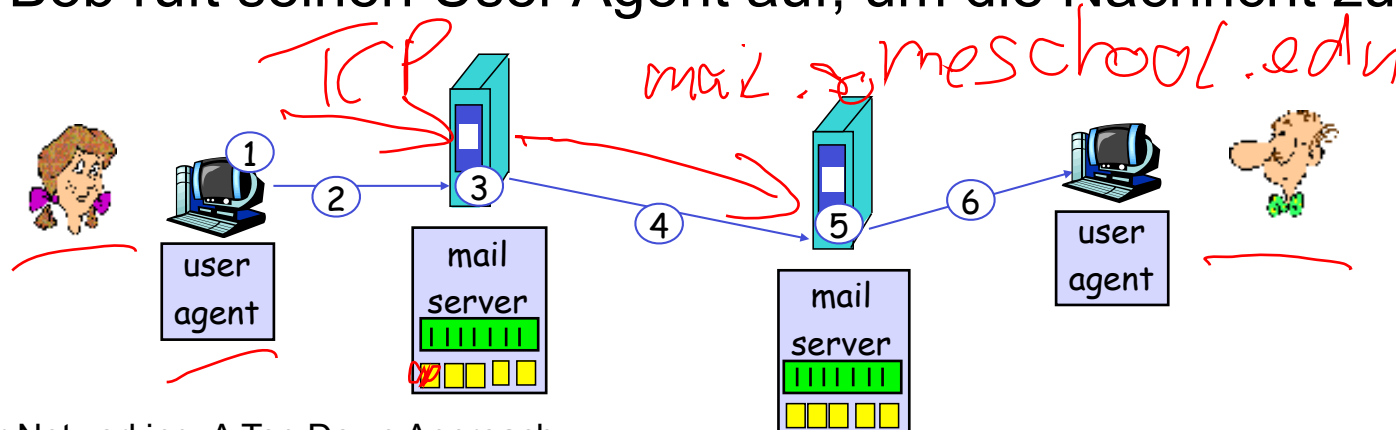


- Transport verwendet TCP um zuverlässig E-Mail-Nachrichten vom Client zum Mail-Server zu verschicken
  - Mail-Server hat Standard Port 25
  - mit zusätzlicher Sicherheit SSL/TLS Port 465
- Direkte Übertragung von Absender-Server zum Empfangs-Server
- 3 Phasen in der Übertragung
  1. Handshake
  2. Transfer der Nachricht
  3. Abschluss
- Befehle und Antwort
  - Befehle als ASCII text
  - Antwort: Status-Code und Kurzbeschreibung
- Nachrichten sind in 7-bit ASCII

!"#\$%&'()\*+,-./0123456789:;<=>@ABCDEFGHIJKLMNO  
PQRSTUVWXYZ[\]^\_`abcdefghijklmnopqrstuvwxy{|}~

# Beispiel: Alice sendet an Bob eine E-Mail Nachricht

- (1) Alice erzeugt an dem User Agent eine Nachricht mit Eintrag "to" bob@~~someschool.edu~~
- (2) Alice's User Agent sendet die Nachricht zu ihrem Mail-Server
  - Nachricht wird in der Nachrichtenwartenschlange platziert
- (3) Client-Seite des SMTP öffnet TCP-Verbindung mit Bobs Mail-Server
- (4) SMTP Client sendet Alice Nachricht über die TCP-Verbindung
- (5) Bobs Mail-Server schreibt die Nachricht in Bobs Mailbox
- (6) Bob ruft seinen User Agent auf, um die Nachricht zu lesen



# Beispiel SMTP Interaktion zwischen Client und Server

C: *TCP Verbindungsaufbau zu mail.hamburger.edu 25*

S: *220 hamburger.edu*

C: *HELO crepes.fr*

S: *250 Hello crepes.fr, pleased to meet you*

C: *MAIL FROM: <alice@crepes.fr>*

S: *250 alice@crepes.fr... Sender ok*

C: *RCPT TO: <bob@hamburger.edu>*

S: *250 bob@hamburger.edu ... Recipient ok*

C: *DATA*

S: *354 Enter mail, end with "." on a line by itself*

C: *Do you like ketchup?*

C: *How about pickles?*

C: *.*

S: *250 Message accepted for delivery*

C: *QUIT*

S: *221 hamburger.edu closing connection*

## ■ SMTP

- verwendet persistente Verbindungen
- verlangt Nachrichten (header & body) in 7-bit ASCII
- SMTP-Server verwenden „CRLF.CRLF“ um das Ende einer Nachricht zu beschreiben

## ■ Vergleich mit HTTP:

- Nachrichten in ASCII codiert, Nachricht mit Befehl und Antwort mit Status-Code
- HTTP: pull (Client fragt Information wie Web-Seite vom Server ab)
- SMTP: push (Client schickt E-Mail zum Server)

5 Byte

SPAM

## ■ HTTP

- jedes Objekt wird in eigener Nachricht verpackt

## ■ SMTP

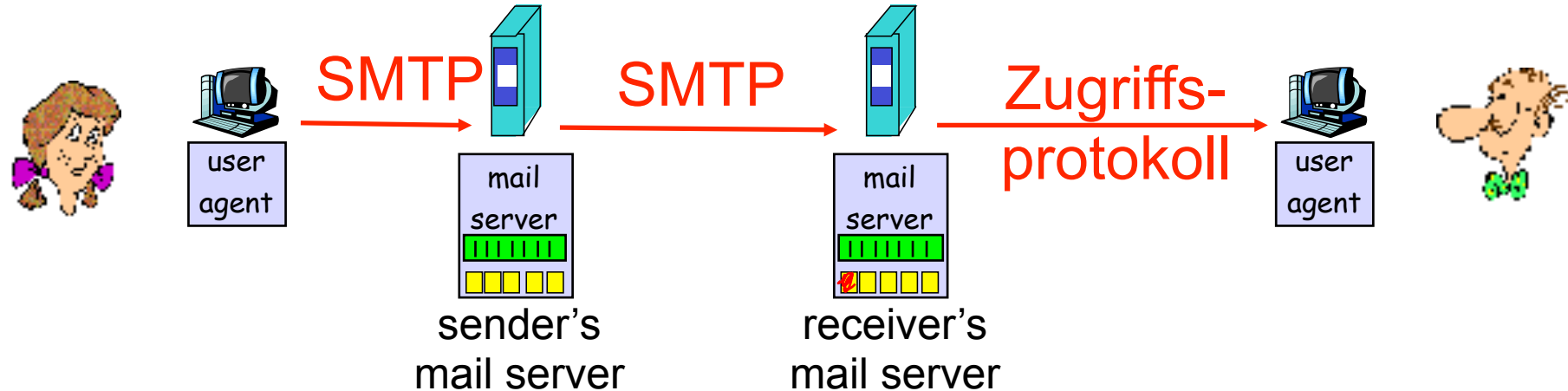
- verschiedene Objekte werden in einer Multipart-Nachricht verschickt

## ■ MIME (Multipurpose Internet Mail Extension):

- kodiert Binärdaten (Anhänge der E-Mail) für die Übertragung mit SMTP



# Mail-Zugriffsprotokolle



- SMTP: Auslieferung und Speicher zum Server des Empfängers
- Mail-Zugriffsprotokoll: E-Mail-Abruf vom Server
  - POP: Post Office Protocol [RFC 1939]
    - Authentifizierung (zwischen Agent und Server) und Download
  - IMAP: Internet Mail Access Protocol [RFC 1730]
    - mehr Features und komplexer
    - Bearbeitung von gespeicherten Nachrichten **auf** dem Server
  - HTTP im Web-Browser: gmail, Hotmail, Yahoo! Mail, web.de, etc.

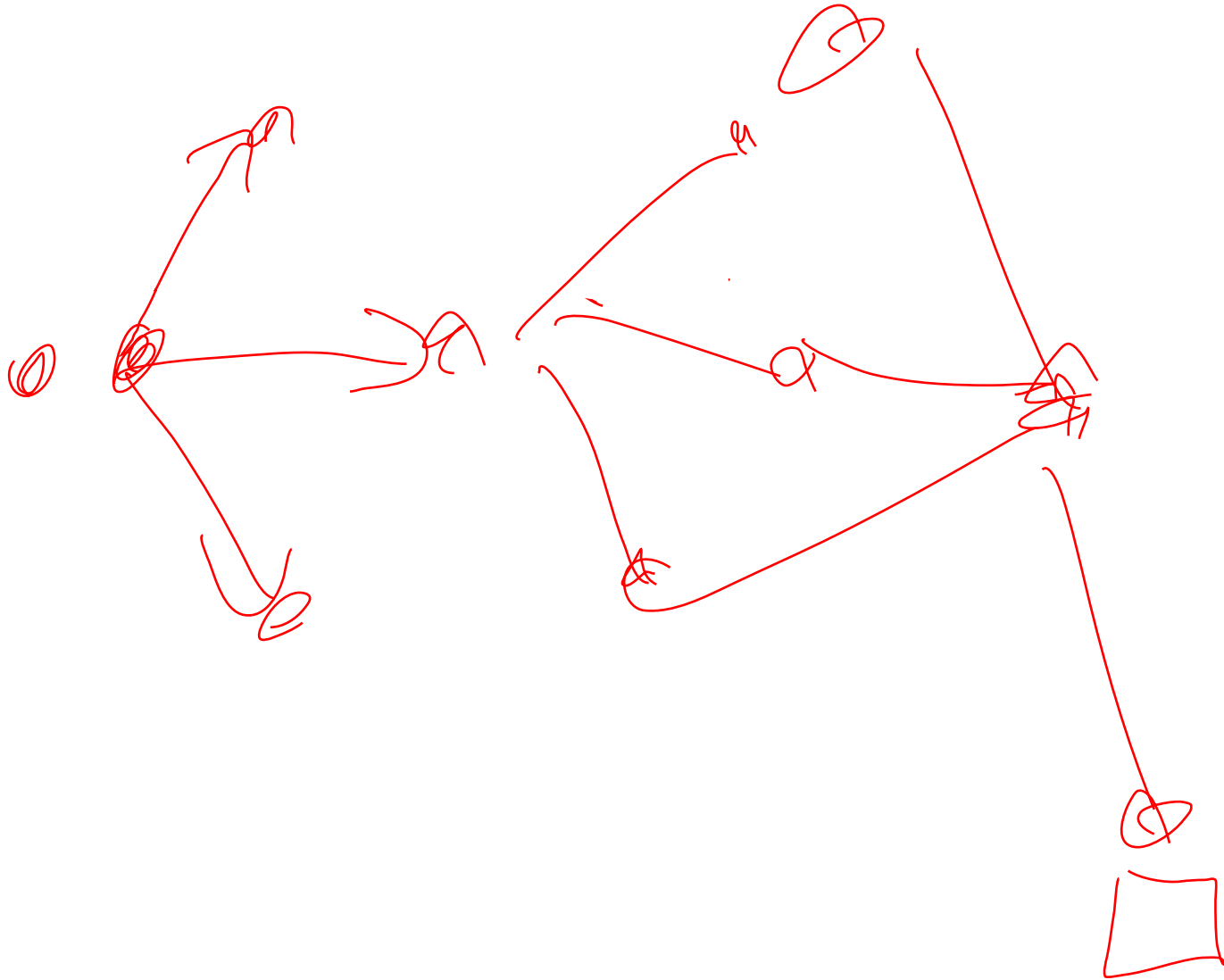
- POP3 (Post-Office-Protocol)
  - User kann im “download and delete” Modus E-Mails einmalig herunterladen
  - User kann E-Mails noch einmal lesen, wenn er den Client wechselt:
    - “Download-and-keep”: Kopien der Nachricht auf verschiedenen Clients
  - POP3 ist zustandslos (stateless) von einer Sitzung zur nächsten
- IMAP (Internet Message Access Protocol)
  - hält alle Nachrichten an einem Ort: dem Server
  - erlaubt dem User die Nachrichten in Ordnern zu organisieren
  - IMAP speichert den Benutzer-Status zwischen Sitzungen
  - Namen der Ordner und Zuordnung zwischen Nachrichten-ID und Ordnernamen

- Was ist P2P **NICHT**?
  - Ein Client-Server network
- Etymologie: peer
  - lateinisch: par = gleich
  - Ebenbürtiger
  - P2P, Peer-to-Peer: Beziehung zwischen gleichwertigen Partnern
- Definition
  - Ein Peer-to-Peer Netzwerk ist ein Kommunikationsnetzwerk im Internet
    - ohne zentrale Kontrolle
    - mit gleichwertigen, unzuverlässigen Partnern

- Napster 1999-2000 *Server*
  - Filesharing, nur rudimentäres P2P
- Gnutella 2000
  - erstes echtes P2P-Netzwerk
- Edonkey 2000
  - Mehr Filesharing als P2P
- FreeNet 2000
  - Anonymisiertes P2P-Netzwerk
  - Daten liegen nicht auf Quelle sondern verschlüsselt auf Fremdrechnern
  - meist zitierteste Arbeit in der Informatik von 2000 laut CiteSeer: „*A Distributed Anonymous Information Storage and Retrieval System*“ von Ian Clark
- FastTrack 2001
  - KaZaa, Morpheus, Grokster
- Bittorrent 2001
- Skype 2003



# Grantella



- Distributed Hash-Tables (DHT) (1997)
  - Ziel: Lastbalancierung für Web-Server
- CAN (2001)
  - DHT-Netzwerk-Struktur
- Chord (2001)
  - Erstes effiziente P2P-Netzwerk mit logarithmischer Suchzeit
- Pastry/Tapestry (2001), Kademlia (2002)
  - Effizientes verteiltes P2P-Netzwerk unter Verwendung des Plaxton-Routing
  - Kademlia wird in BitTorrent benutzt
- Und viele andere Ansätze
  - Viceroy, Distance-Halving, Koorde, Skip-Net, P-Grid, ...
- In den letzten fünf Jahren:
  - Network Coding für P2P
  - Spieltheorie in P2P
  - Anonymität, Sicherheit

# Distributed Hash-Table (DHT) $x \bmod 8$

- Hash-Tabellen

- nicht praktikabel in P2P

- Verteilte Hash-Tabellen

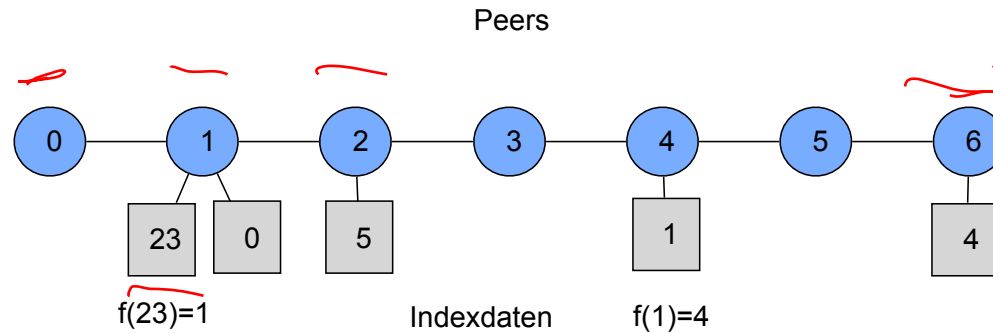
- *Consistent Hashing and Random Trees: Distributed Caching Protocols for Relieving Hot Spots on the World Wide Web*, Karger, Lehman, Leighton, Levine, Lewin, Panigrahy, STOC 1997

- Daten

- werden *gehasht* und nach Bereich den Peers zugeordnet

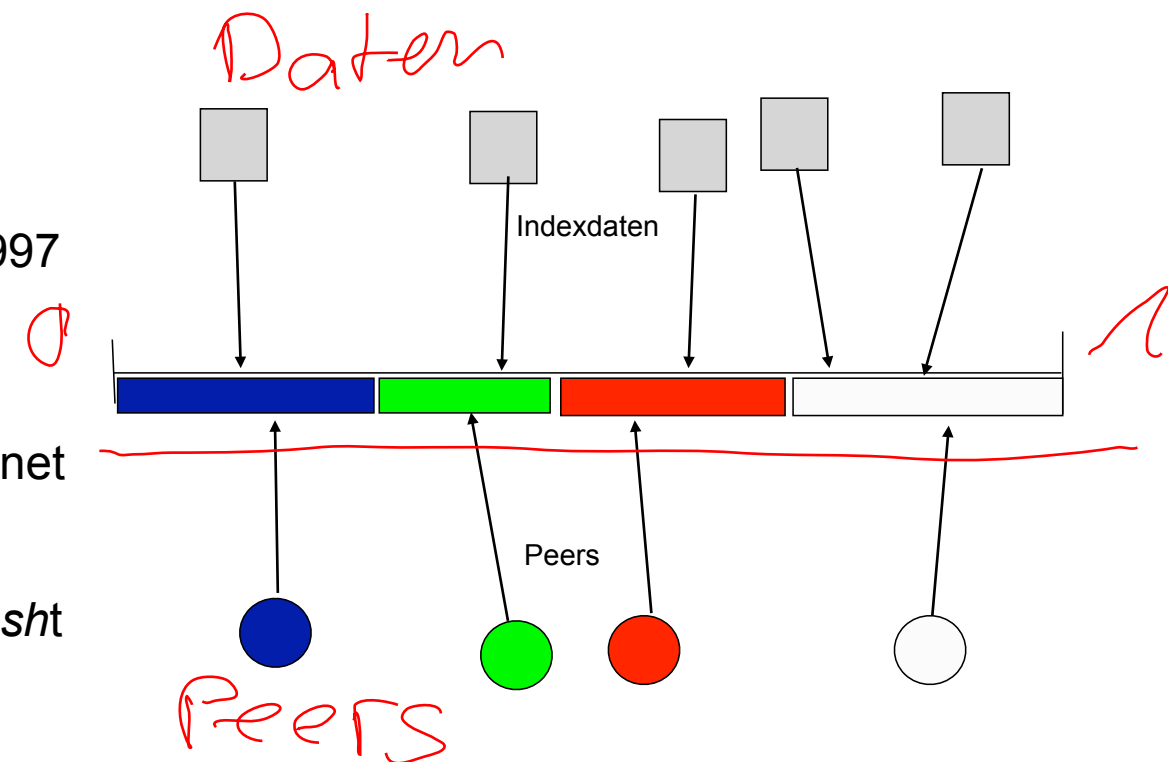
- Peers

- werden an eine Stelle *gehasht* und erhalten Bereiche des Wertebereichs der Hashfunktion zugeteilt

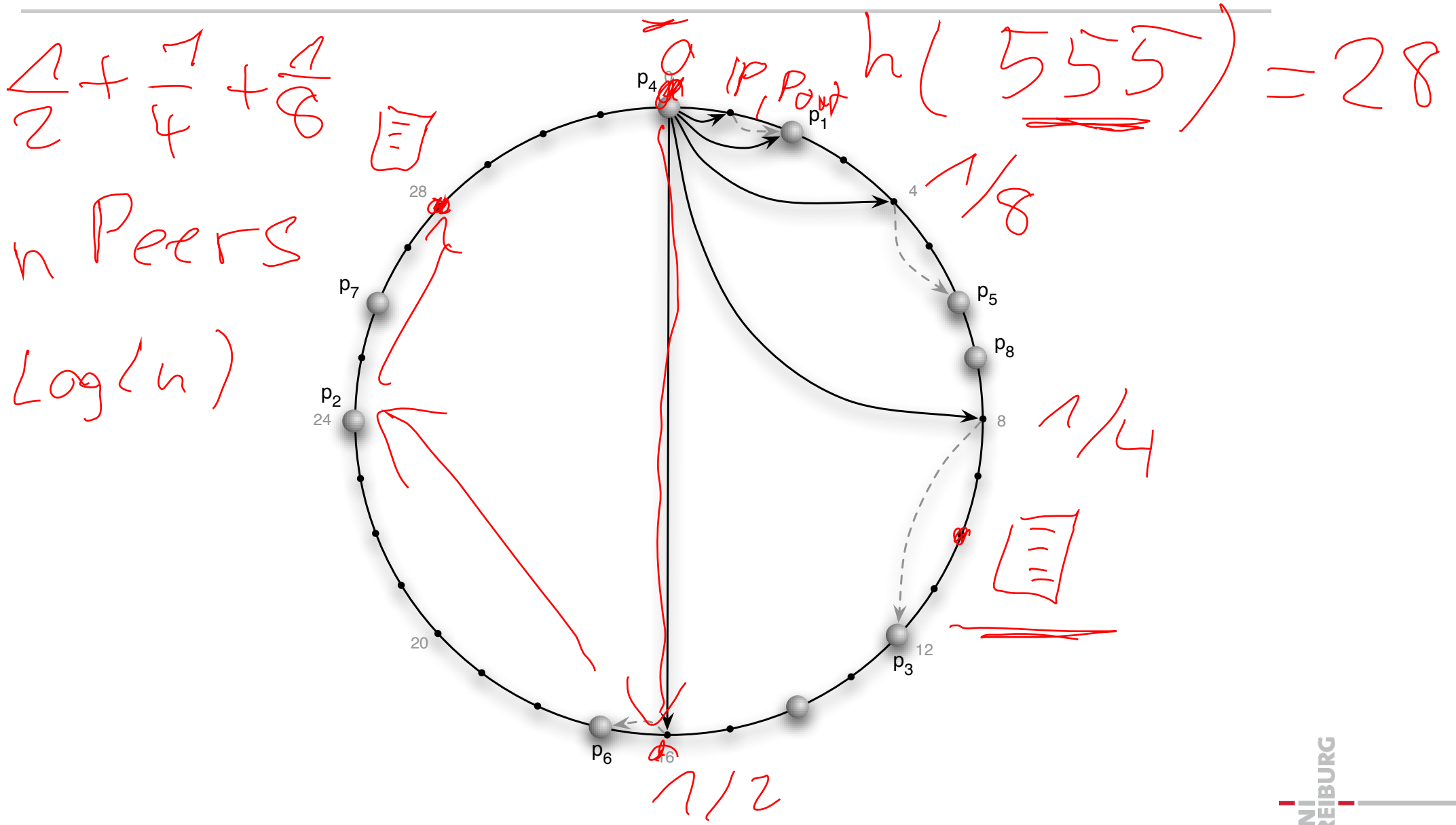


$x \bmod 7$

2 6

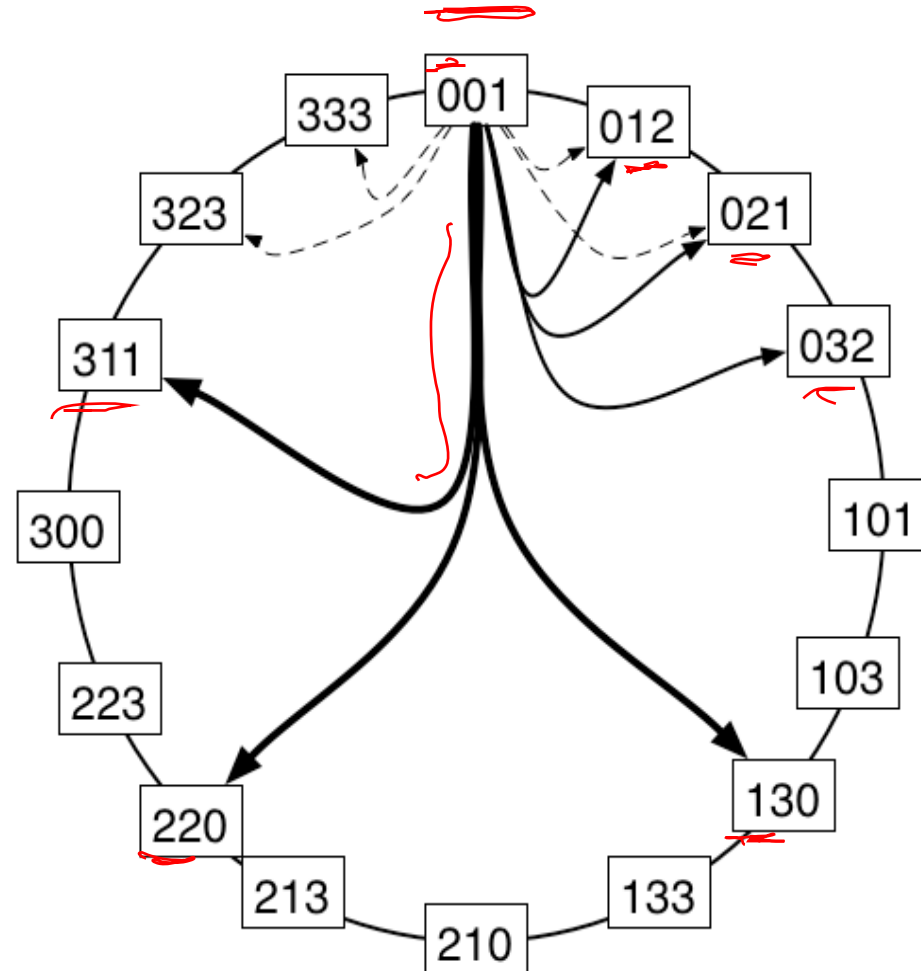


# Zeiger-Struktur in Chord





- Peter Druschel
  - jetzt Direktor des Max-Planck-Instituts für Informatik, Saarbrücken/Kaiserslautern
- Antony Rowstron
  - Microsoft Research, Cambridge, GB
- Pastry
  - *Scalable, decentralized object location and routing for large scale peer-to-peer-network*
  - Chord-ähnliches Netzwerk, welches das Routing von Plaxton, Rajamaran, Richa (1997) verwendet



## ■ Bram Cohen

- BitTorrent ist ein P2P-Netzwerk für den Download von Dateien
- Dateien werden in Blöcke aufgeteilt
- verwendet implizit Multicast-Bäume für die Verteilung von Blöcken

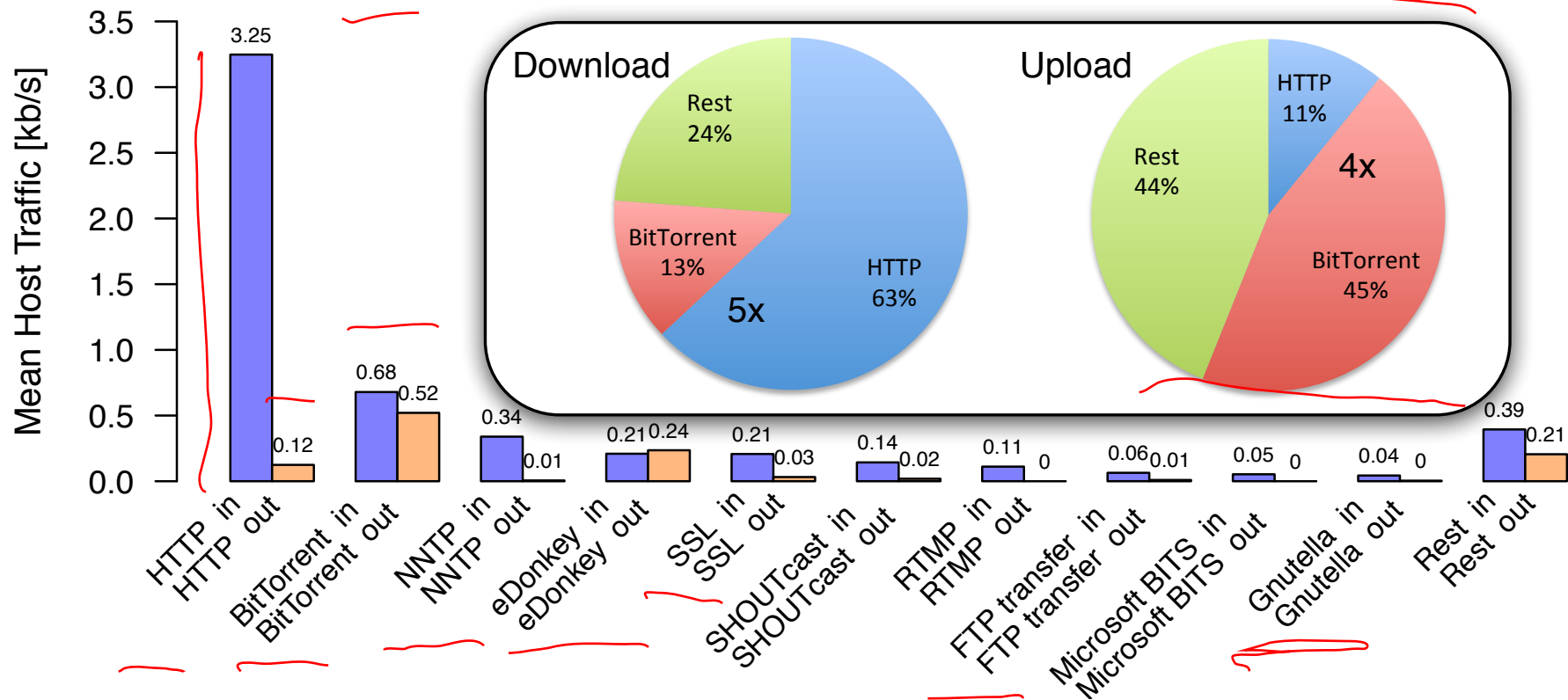
## ■ Ziele

- schneller Download einer Datei unter Verwendung des Uploads vieler Peers
  - Upload ist der Flaschenhals
  - z.B. wegen asymmetrischen Aufbau von ISDN oder DSL
- Fairness
  - seeders against leeches
- Gleichzeitige Verwendung vieler Peers

*Spieltheorie*

*Tit-for-tat*

- Nutzer Traffic eines Internet Service Provider im Raum Karlsruhe aus dem Jahre 2009



# Systeme II

## 4. Die Anwendungsschicht

Thomas Janson<sup>°</sup>, Kristof Van Laerhoven\*, Christian Ortolf<sup>°</sup>

Folien: Christian Schindelhauer<sup>°</sup>

Technische Fakultät

<sup>°</sup>: Rechnernetze und Telematik, \*: Eingebettete Systeme

Albert-Ludwigs-Universität Freiburg

Version 13.05.2015