Peer-to-Peer-Netzwerke

Albert-Ludwigs-Universität Freiburg
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Christian Schindelhauer

Sommersemester 2006

23. Vorlesung

26.07.2006

schindel@informatik.uni-freiburg.de

Inhalte

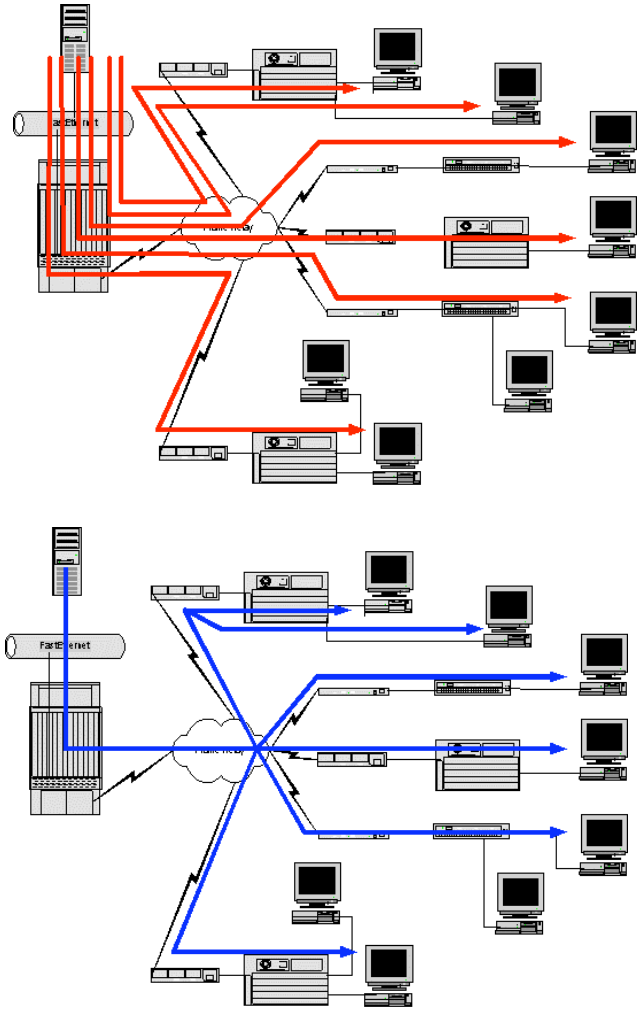
Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Kurze Geschichte der Peer-to-Peer-Netzwerke**
- **Das Internet: Unter dem Overlay**
- **Die ersten Peer-to-Peer-Netzwerke**
  - Napster
  - Gnutella
- **CAN**
- **Chord**
- **Pastry und Tapestry**
- **Gradoptimierte Netzwerke**
  - Viceroy
  - Distance-Halving
  - Koorde
- **Netzwerke mit geordneter Speicherung**
  - P-Grid
  - Skip-Net und Skip-Graphs
- **Selbstorganisation**
  - Pareto-Netzwerke
  - Zufallsnetzwerke
  - Topologie-Management
- **Sicherheit in Peer-to-Peer-Netzwerken**
- **Anonymität**
- **Datenzugriff: Der schnellere Download**
- **Peer-to-Peer-Netzwerke in der Praxis**
  - eDonkey
  - FastTrack
  - Bittorrent
- **Ausblick**
  - Juristische Situation
  - Anwendungen
  - Offene Fragen

IP Multicast

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Motivation**
  - Übertragung eines Stroms an viele Empfänger
- **Unicast**
  - Strom muss mehrfach einzeln übertragen werden
  - Bottleneck am Sender
- **Multicast**
  - Strom wird über die Router vervielfältigt
  - Kein Bottleneck mehr



Bilder von Peter J. Welcher
www.netcraftsmen.net/.../ papers/multicast01.html

Funktionsprinzip

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **IPv4 Multicast-Adressen**
  - in der Klasse D (außerhalb des CIDR - Classless Interdomain Routings)
  - 224.0.0.0 - 239.255.255.255
- **Hosts melden sich per IGMP bei der Adresse an**
  - IGMP = Internet Group Management Protocol
  - Nach der Anmeldung wird der Multicast-Tree aktualisiert
- **Source sendet an die Multicast-Adresse**
  - Router duplizieren die Nachrichten an den Routern
  - und verteilen sie in die Bäume
- **Angemeldete Hosts erhalten diese Nachrichten**
  - bis zu einem Time-Out
  - oder bis sie sich abmelden
- **Achtung:**
  - Kein TCP, nur UDP
  - Viele Router lehnen die Beförderung von Multicast-Nachrichten ab
    - Lösung: Tunneln

Routing Protokolle

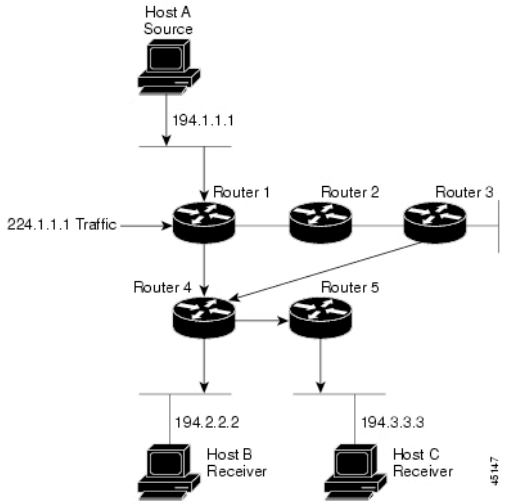
Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Distance Vector Multicast Routing Protocol (DVMRP)**
  - jahrelang eingesetzt in MBONE (insbesondere in Freiburg)
  - Eigene Routing-Tabelle für Multicast
- **Protocol Independent Multicast (PIM)**
  - im Sparse Mode (PIM-SM)
  - aktueller Standard
  - beschneidet den Multicast Baum
  - benutzt Unicast-Routing-Tabellen
  - ist damit weitestgehend Protokoll unabhängig
- **Voraussetzung PIM-SM:**
  - benötigt Rendevous-Point (RP) in ein-Hop-Entfernung
  - RP muss PIM-SM unterstützen
  - oder Tunneling zu einem Proxy in der Nähe eines RP

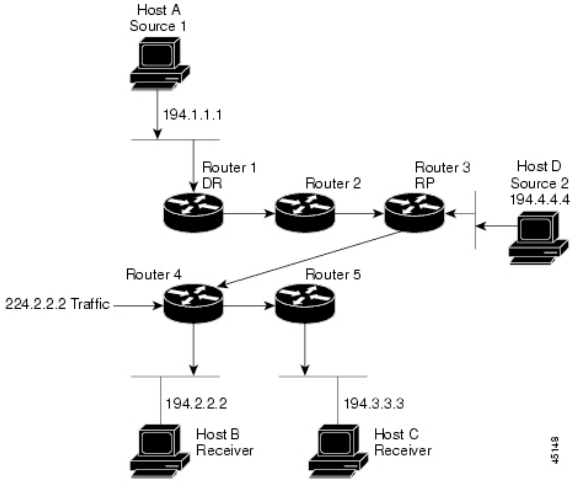
PIM-SM Baumkonstrukte

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Host A Shortest-Path-Tree**



- **Shared Distribution Tree**

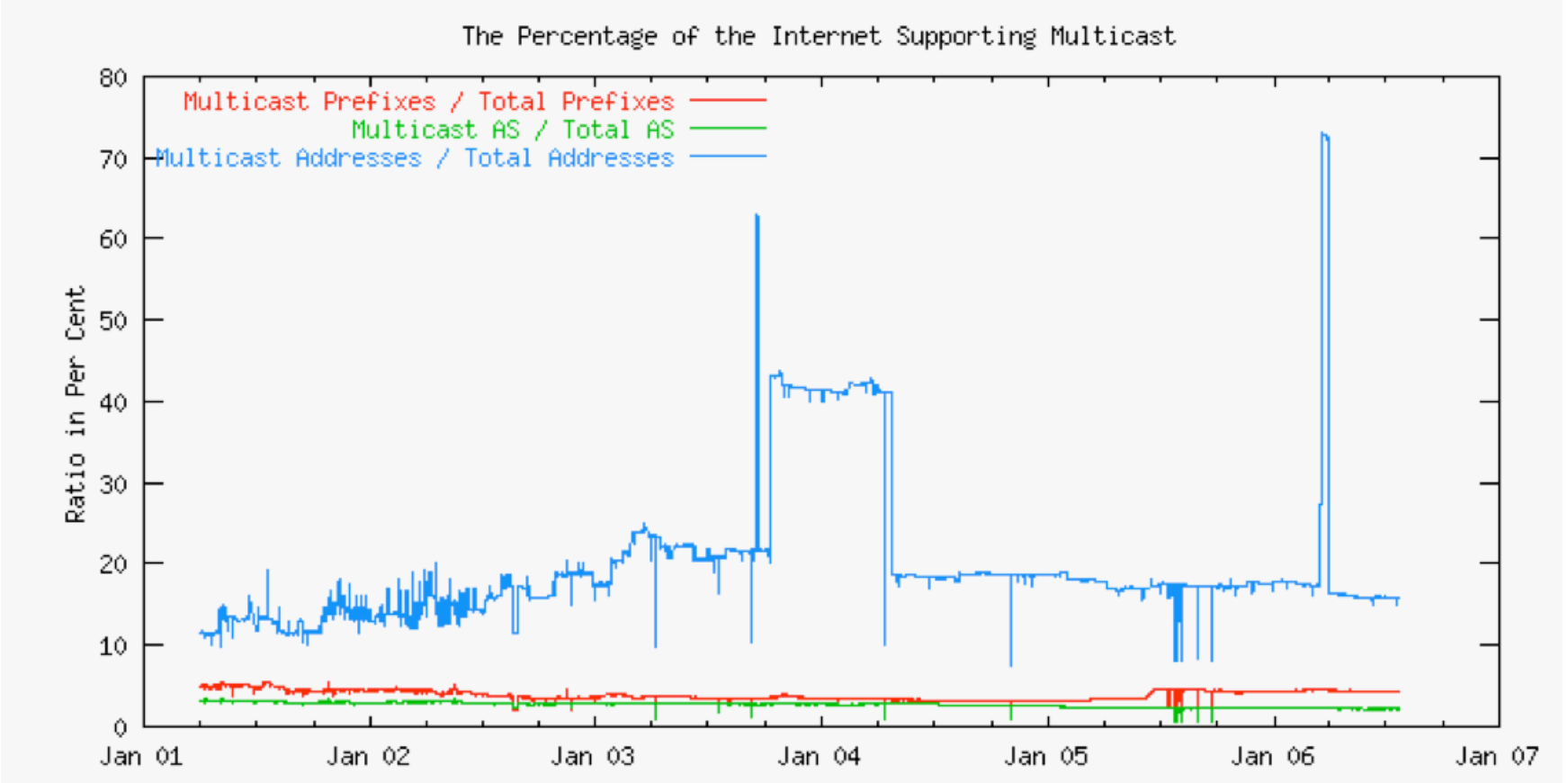


From Cisco: http://www.cisco.com/en/US/products/hw/switches/ps646/products_configuration_guide_chapter09186a008014f350.html

IP Multicast wird kaum unterstützt

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **IP Multicast ist die beste der hier vorgestellten Lösungen**
- **Aber: Status Mitte 2006**
  - http://www.multicasttech.com/status/



Warum so wenig IP Multicast?

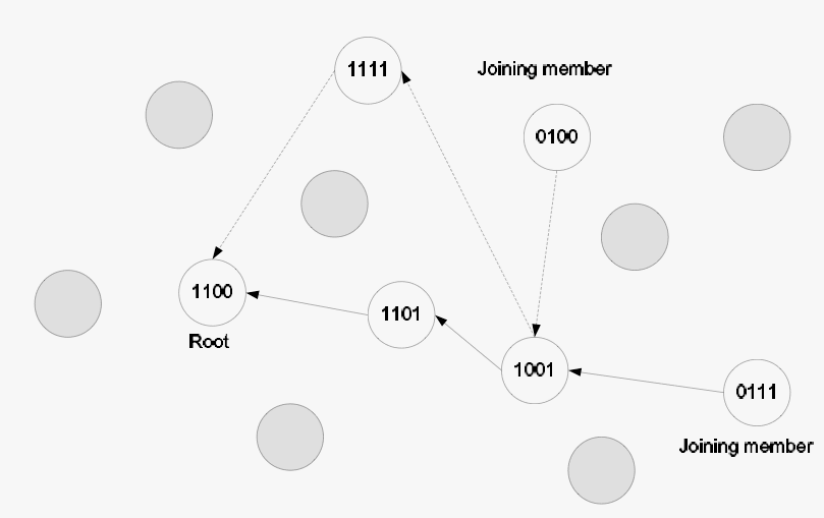
Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Trotz erfolgreichen Einsatz**
  - in Video-Übertragung von IETF-Meetings
  - MBONE (Multicast Backbone)
- **gibt es wenig ISP welche IP Multicast in den Routern unterstützen**

- **Zusätzlicher Wartungsaufwand**
  - Schwierig zu konfigurieren
  - Verschiedene Protokolle
- **Gefahr von Denial-of-Service-Attacken**
  - Implikationen größer als bei Unicast
- **Transport-Protokoll**
  - Nur UDP einsetzbar
  - Zuverlässige Protokolle
    - Vorwärtsfehlerkorrektur
    - Oder propertiäre Protokolle in den Routern (z.B. CISCO)
- **Marktsituation**
  - Endkunden fragen kaum Multicast nach (benutzen lieber P2P-Netzwerke)
  - Wegen einzelner Dateien und weniger Abnehmer erscheint ein Multicast wenig erstrebenswert (Adressenknappheit!)

Scribe

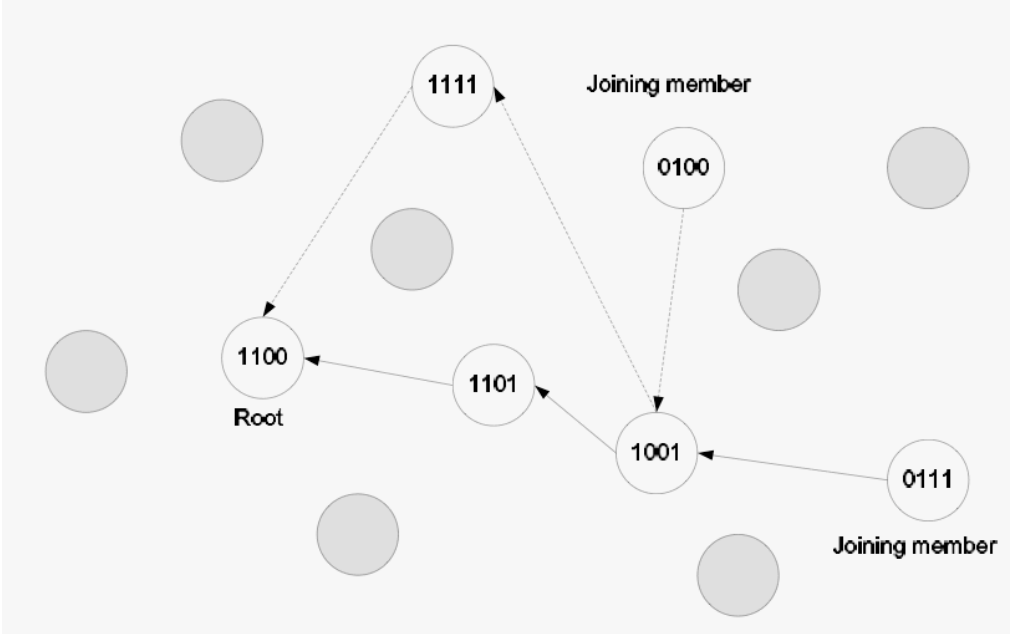
- **Multicast-Baum im Overlay-Netzwerk**
- **Scribe [2001] basiert auf Pastry**
  - Castro, Druschel, Kermarrec, Rowstron
- **Vergleichbare Ansätze**
  - CAN Multicast [2001] basiert auf CAN
  - Bayeux [2001] basiert auf Tapestry
- **Andere Ansätze**
  - Overcast [´00] und Narada [´00]
  - bauen auch Multicast-Tree auf Unicast-Verbindungen
  - skalieren nicht



Funktionsweise Scribe

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Create**
  - GroupID wird einem Peer gemäß nächsten Pastry-Index zugewiesen
- **Join**
  - Interessierter Peer macht Look-up zur Group-ID
  - Sobald ein Peer im Multicast-Baum gefunden worden ist, wird neuer Teilpfad eingefügt
- **Download**
  - Nachrichten werden baumförmig verteilt
  - Knoten duplizieren Teile



Scribe: Optimierung

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

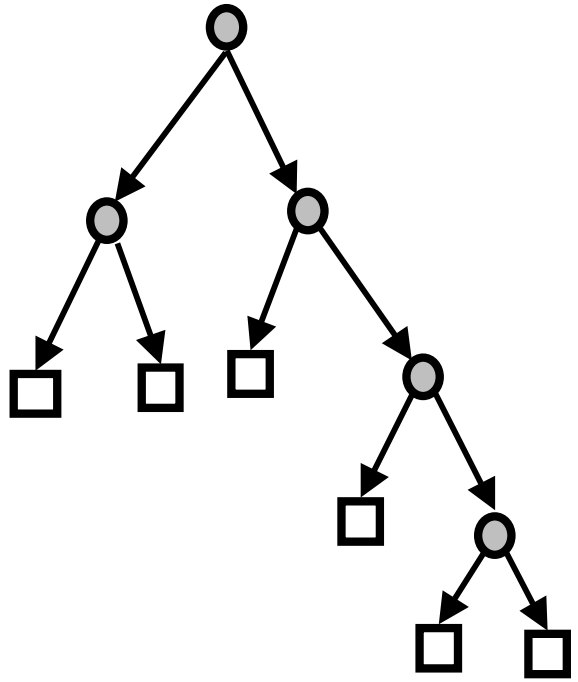
- **Bottleneck-Remover**
  - Wenn ein Knoten überlastet ist
  - dann wird die Gruppe ausgewählt mit der größten Last
  - In dieser Gruppe wird das Kind im Teilbaum ausgewählt, der in der Netzwerk-Metrik am weitesten ist
  - Dieser Knoten misst den Delay zwischen sich und den anderen Kindern
  - Dann hängt sich dieser Knoten unter den nächsten Bruder

Split-Stream Motivation

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Multicast-Bäume benachteiligen gewisse Knoten**
- **Lemma**
  - In jedem Binärbaum ist die Anzahl der Blätter = Anzahl interner Knoten +1
- **Schlussfolgerung**
  - Fast die Hälfte aller Knoten verteilen Daten
  - Während die andere Hälfte profitiert
  - Als interner Knoten hat man den doppelten Upload (verglichen mit dem Durchschnitt)
- **Lösung: größerer Grad?**
- **Lemma**
  - In jedem Baum mit Grad d gilt für die Anzahl interner Knoten k und die Blätter b:

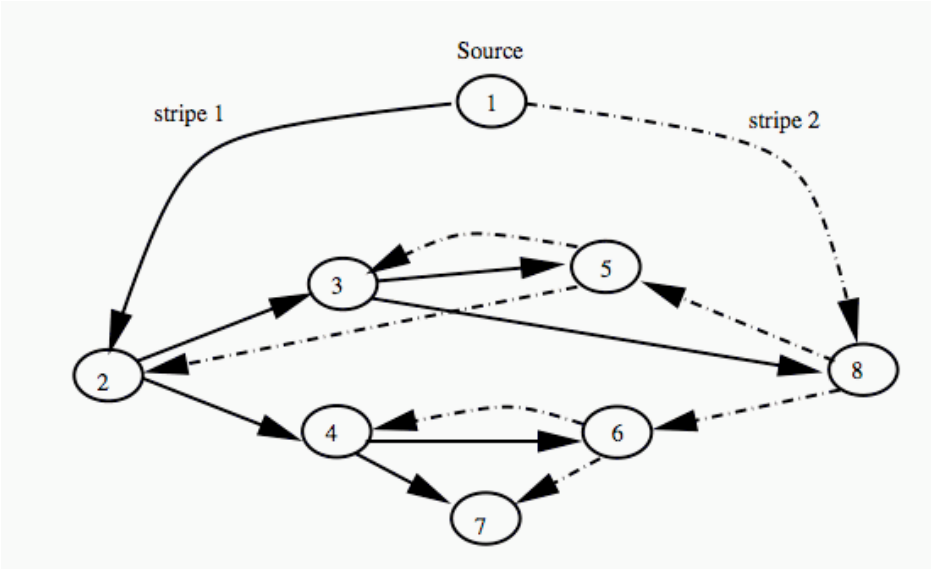
    $(d-1)\,k = b - 1$
- **Schlussfolgerung**
  - Damit müssen noch weniger Peers mehr Upload verrichten als im Binärbaum



Split-Stream Lösung

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Castro, Druschel, Kermarrec, Nandi, Rowstron, Singh 2001**
- **Idee**
  - Teile die Datei der Größe B in k kleinere Teile
  - Verwende anderen Multicast-Baum für jeden der Teile
  - Dadurch wird jeder Peer mal als Blatt oder als Verteil-Knoten fungieren
    - außer der Quelle
- **Der Upload jedes Knotens ist dann (im Idealfall) höchstens der Download**



Bittorrent

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Bram Cohen**
- **Bittorrent ist ein reales (sehr erfolgreiches) Peer-to-Peer-Netzwerke**
  - konzentriert sich auf Download
  - verwendet (implizit) Multicast-Trees für die Verteilung der Daten
- **Beschreibung ist Peer-orientiert und nicht Daten-orientiert**

- **Ziele:**
  - Möglichst effizienter Download einer Datei unter Zuhilfenahme der Upload-Fähigkeit der Peers
  - Möglichst effiziente Ausnutzung des Upload von Peers
    - In der Praxis ist der Upload der Bottleneck
    - z.B. wegen der asymmetrischen Protokoll-Gestaldung von ISDN, Bitübertragungsschicht von DSL
  - Fairness zwischen den Peers
    - Seeders versus Leechers
  - Verwendung verschiedener Quellen

Bittorrent
Koordination und Datei

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Zentrale Koordination**
  - durch sogenannten Tracker-Host
  - Für jede Datei gibt der Tracker eine Menge von Zufallspeers aus der Menge der downloadenden Peers
  - Zusätzlich: Ausgabe des Hash-Codes und anderer Kontroll-Information
  - Tracker-Hosts haben keine Dateien
    - Trotzdem kann das Anlegen einer Tracker-Datei auf einem Tracker-Host rechtliche Probleme ergeben (Urheberschutzsgesetz)
- **Datei**
  - ist in kleinere Dateien zerlegt (in Tracker-Datei festgelegt)
  - Jeder teilnehmende Host kann heruntergeladenen Teil weiterverbreiten, sobald er vollständig erhalten wurde
  - Damit ist Bittorrent die Umsetzung eines Split-Stream-ähnlichen Protokolls
- **Interaktion zwischen Peers**
  - Zwei Peers tauschen die Information über ihre vorhandenen Teile aus
  - Gemäß der Politik von Bittorrent werden dann noch nicht vorhandene Teile von dem einen Peer zum anderen übertragen

Bittorrent
Auswahl der Teile

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Problem**
  - Das Coupon-Collector-Problem kann gewisse Teile einer Datei rar machen
    - siehe Übung
- **Maßnahmen**
  - Rarest First
    - Jeder Peer versucht die seltensten Teile zu laden
      - Häufigkeit erschließt sich aus der Kommunikation mit anderen Peers
    - Ist die Quelle nicht mehr vorhanden, so erhöht dies die Wahrscheinlichkeit, dass alle vollenden können
  - Random First (Ausnahme für neue Peers)
    - Startet ein Peer, dann lädt er ein zufälliges Teil
    - Dadurch wird die Nachfrage bei den Peers mit seltenen Teilen verringert
  - Endgame Mode
    - Sind fast alle Teile geladen, dann fragt der Peer bei allen verbundenen Peers nach allen fehlenden Dateien
    - Dadurch kann vermieden werden, dass ein langsamer Peer mit einem Teil den Download am Ende noch herauszögert

Bittorrent Politik

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Ziel**
  - Selbstorganisierendes System
  - Gute (Uploader/Seeder) werden belohnt
  - Böse (Downloader/Leecher) werden bestraft
- **Belohnung**
  - Gute Download-Bandweite
  - Rücknahme einer Drosselung (un-choke)
- **Bestrafung**
  - Drosselung der Bandweite (choke)
- **Bewertung**
  - Jeder Peers bewertet selbst sein Umfeld aufgrund der vergangenen Erfahrungen

Bittorrent Drosseln

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

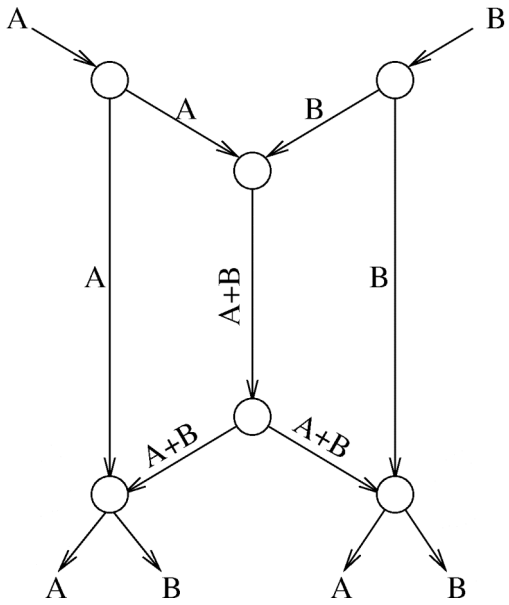
- **Jeder Peer hat eine Drossel-Liste**
  - Die Anfragen der gedrosselten Peers nach Teilen werden zeitweise nicht bedient
  - Dieser Zustand kann aber wieder rückgängig gemacht werden
- **Aufnahme in die Drossel-Liste**
  - Jeder Peer hat eine feste Mindestanzahl von gedrosselten Peers (z.B. 4)
  - Die Peers welche den schlechtesten Download bieten werden in diese Liste aufgenommen
    - und ersetzen andere Peers
- **Optimistisches Entdrosseln (Optimistic Unchoking)**
  - Aus der Liste der Drossel-Kandidaten wird ein Kandidat willkürlich herausgenommen
    - Dadurch kann man verhindern, dass alle Peers einen Peer nicht bedienen, nur weil er eine schlechte Verbindung hat

Netzwerk-Kodierung

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network Information Flow", *(IEEE Transactions on Information Theory*, IT-46, pp. 1204-1216, 2000)**

- **Beispiel:**
  - Die Bits A und B sollen übertragen werden
  - Über jede Leitung darf nur ein Bit übertragen werden
  - Werden nur die Bits unverändert übertragen, so
    - kann entweder nur links oder nur rechts A und B erhalten werden
  - Durch Verwendung des Xor A+B kann in beiden das Ergebnis bekommen werden

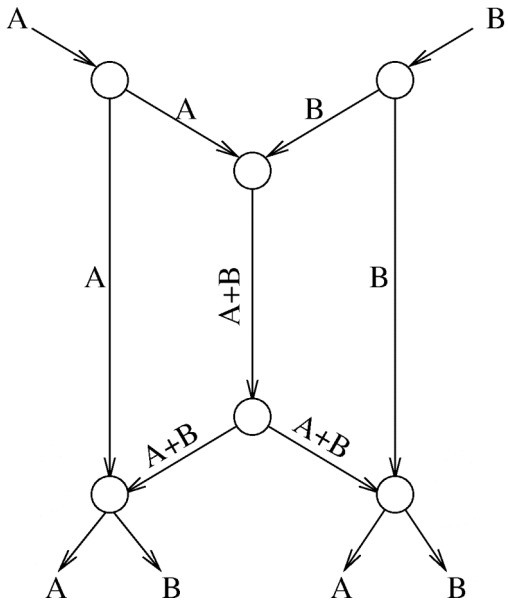


Netzwerk-Kodierung

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network Information Flow", *(IEEE Transactions on Information Theory*, IT-46, pp. 1204-1216, 2000)**

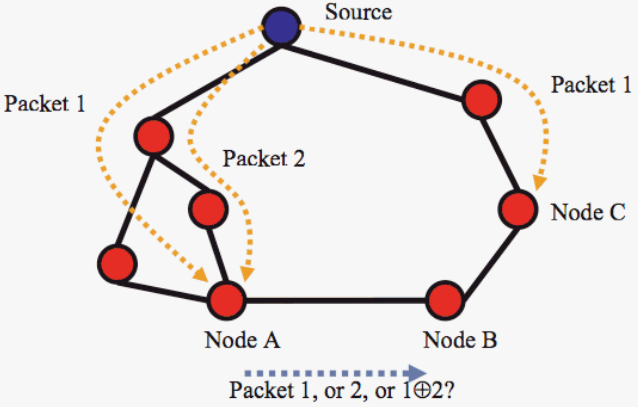
- **Theorem [Ahlswede et al.]**
  - Es gibt einen Netzwerk-Code für jeden Graphen, so dass man so viel Information von den Quellen erhalten kann,
  - wie das zugehörige Fluss-Problem erlaubt.

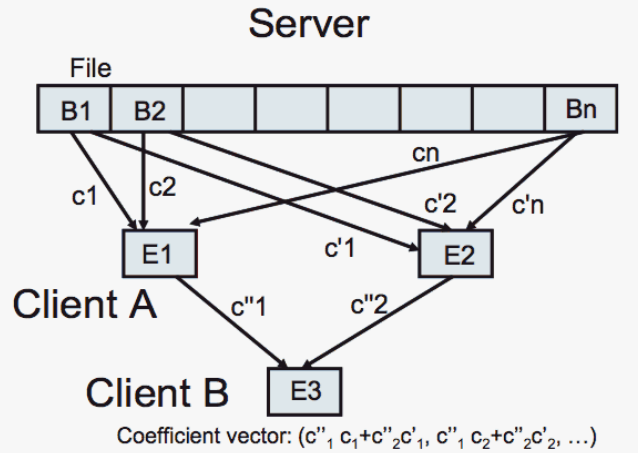


Praktische Netzwerk-Kodierung

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Christos Gkantsidis, Pablo Rodriguez Rodriguez, 2005**
- **Ziel**
  - Überwendung des Coupon-Collector-Problems bei der Partitionierung von Dateien
    - Eine Datei aus m Teilen kann von den Vorgängern erhalten werden, wenn die Summe ihrer Teile mindestens m ist
  - Optimale Übertragung von Dateien hinsichtlich der verfügbaren Bandbreite
- **Methode**
  - Verwende als Codes Linear-Kombinationen der Teile einer Datei
    - Entstehender Code beinhaltet den Vektor
  - Bei der Verteilung werden Linear-Kombination rekombiniert zu neuen Teilen
  - Beim Empfänger werden die Linear-Kombinationen gesammelt
  - und mittels Matrix-Invertierung die Original-Datei rekonstruiert





Kodierung und Dekodierung

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- Datei: $x_1, x_2, \ldots, x_m$
- Codes: $y_1, y_2, \ldots, y_m$
- Mit zufälligen Variablen r_{ij}

$$(r_{i1} r_{i2} \ldots r_{im}) \cdot \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} = y_i$$

- Also

$$\begin{pmatrix} r_{11} & \cdots & r_{1m} \\ \vdots & \ddots & \vdots \\ r_{m1} & \cdots & r_{mm} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} = \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix}$$

- Falls die Matrix (r_{ij}) invertierbar ist, erhält man

$$\begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} = \begin{pmatrix} r_{11} & \cdots & r_{1m} \\ \vdots & \ddots & \vdots \\ r_{m1} & \cdots & r_{mm} \end{pmatrix}^{-1} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix}$$

Invertierbarkeit einer Zufallsmatrix

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Theorem**
  - Werden die Zahlen uniform aus dem (endlichen) Körper mit Kardinalität b gewählt, so ist die m x m- Zufallsmatrix mit mindestens der Wahrscheinlichkeit

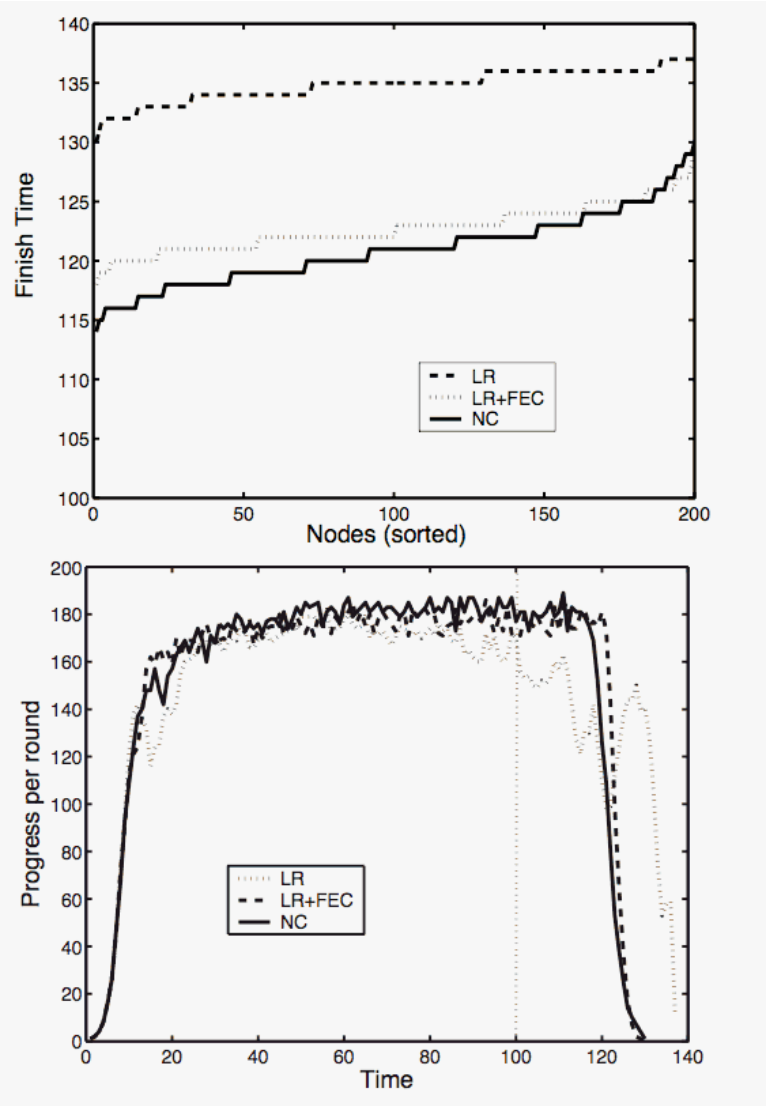
$$1 - \sum_{i=1}^{m} \frac{1}{b^i}$$

    invertierbar.
- **Idee: Wähle endlichen Körper F[2^8]**
  - Rechnung mit Bytes effizient möglich
  - Dann ist die Erfolgswahrscheinlichkeit mindestens über 99%
  - Im Fehlerfall wird durch Übertragung eines zusätzlichen Blocks die Erfolgswahrscheinlichkeit wieder größer als 99%

Geschwindigkeit von Network-Coding

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Vergleich**
  - Network-Coding (NC) versus
  - Local-Rarest (LR) und
  - Local-Rarest+Forward-Error-Correction (LR+FEC)



Probleme mit Network-Coding

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Zusatzaufwand: Speichern der Variablen**
  - Pro Block 1 Variablen-Vektor
  - Z.B. 4 GByte-Datei hat mit 100 kByte-Block hat Variablenvektor von
    - 4 GByte/100 kByte = 40 kByte
    - Damit entsteht ein Overhead von rund 40% pro Datei
  - Besser: 4 GByte und 1 MByte-Block
    - ergibt 4kByte Overhead von 0,4%
- **Dekodierungsaufwand**
  - Invertierung einer m x m- Matrix benötigt Zeit $O(m^3)$ und Speicher $O(m^2)$
  - Damit ist der Speicherverbrauch bei
    - 4 kByte-Variablen-Vektor = 16 MByte
    - 40 kByte-Variablen-Vektor = 1,6 GByte
- **Schreib-/Lese-Zugriffe**
  - Um m Blöcke zu kodieren muss jeder Teil der Datei m-mal gelesen werden
  - Zur Dekodierung muss jeder Code-Teil ebenfalls m-mal gelesen werden
  - Abnutzung der Speichermedien (Festplatten)
  - Zeitverbrauch für Lese/Schreib-Operationen (Disk-Cache wird nicht ausgenutzt)

Ende der 23. Vorlesung

Albert-Ludwigs-Universität Freiburg
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Peer-to-Peer-Netzwerke

Christian Schindelhauer
schindel@informatik.uni-freiburg.de