

Peer-to-Peer- Netzwerke



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

Christian Schindelhauer

Sommersemester 2006

8. Vorlesung

18.05.2006

schindel@informatik.uni-freiburg.de



Inhalte

- **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 Suchbäumen**
 - Skipnet und Skip-Graphs
 - P-Grid

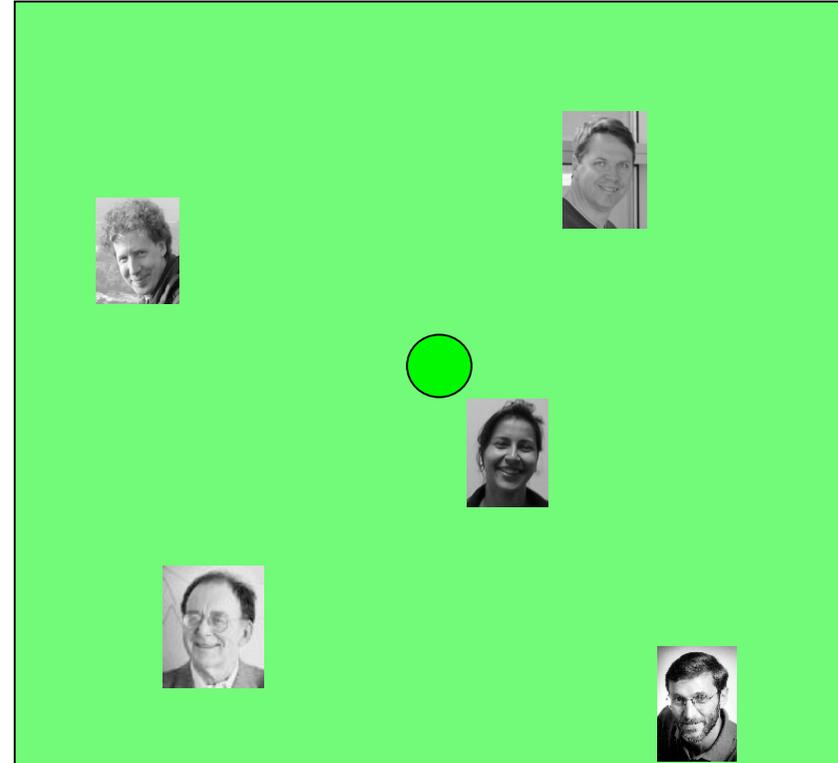
- **Selbstorganisation**
 - Pareto-Netzwerke
 - Zufallsnetzwerke
 - Selbstorganisation
 - Metrikbasierte Netzwerke Sicherheit in Peer-to-Peer-Netzwerken
- **Anonymität**
- **Datenzugriff: Der schnellere Download**
- **Peer-to-Peer-Netzwerke in der Praxis**
 - eDonkey
 - FastTrack
 - Bittorrent
- **Peer-to-Peer-Verkehr**
- **Juristische Situation**



Content Addressable Network (CAN)

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

- Dateien werden in durch (zweiwertige)-Hash-Funktion in das Quadrat abgebildet
- Am Anfang ist ein leeres Quadrat mit nur einem Peer als Besitzer





Content Addressable Network (CAN)

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

- Dateien werden in durch (zweiwertige)-Hash-Funktion in das Quadrat abgebildet
- Am Anfang ist ein leeres Quadrat mit nur einem Peer als Besitzer
- Der Besitzer einer Fläche speichert alle Einträge in der Fläche
- Ein Peer wählt einen zufälligen Punkt in der Ebene

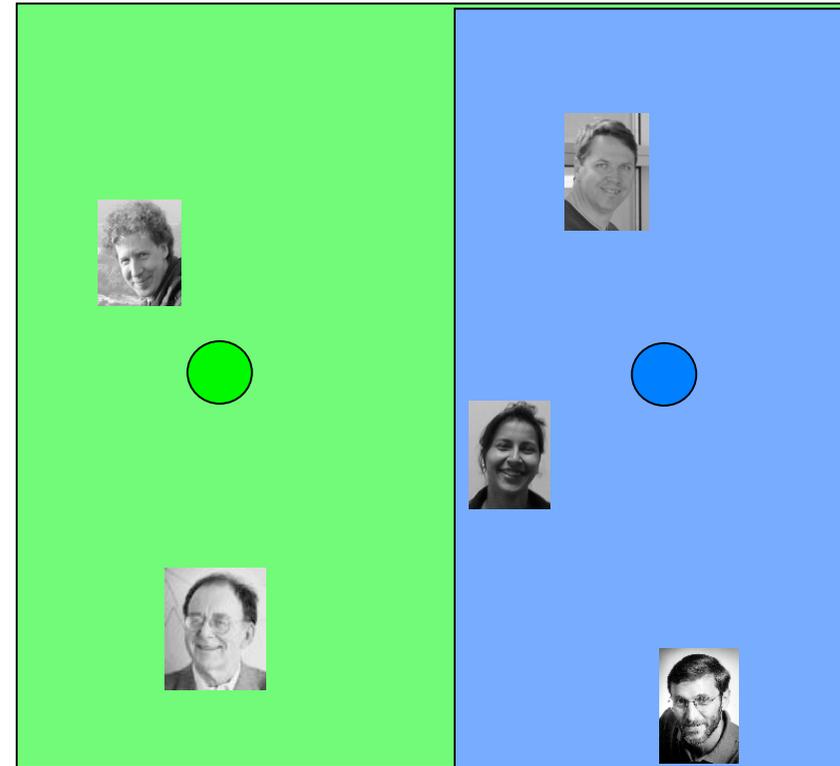




Content Addressable Network (CAN)

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

- Dateien werden in durch (zweiwertige)-Hash-Funktion in das Quadrat abgebildet
- Am Anfang ist ein leeres Quadrat mit nur einem Peer als Besitzer
- Der Besitzer einer Fläche speichert alle Einträge in der Fläche
- Ein Peer wählt einen zufälligen Punkt in der Ebene
 - Der Besitzer des entsprechenden Quadrats teilt seine Fläche und
 - übergibt die Hälfte dem neuen Peer

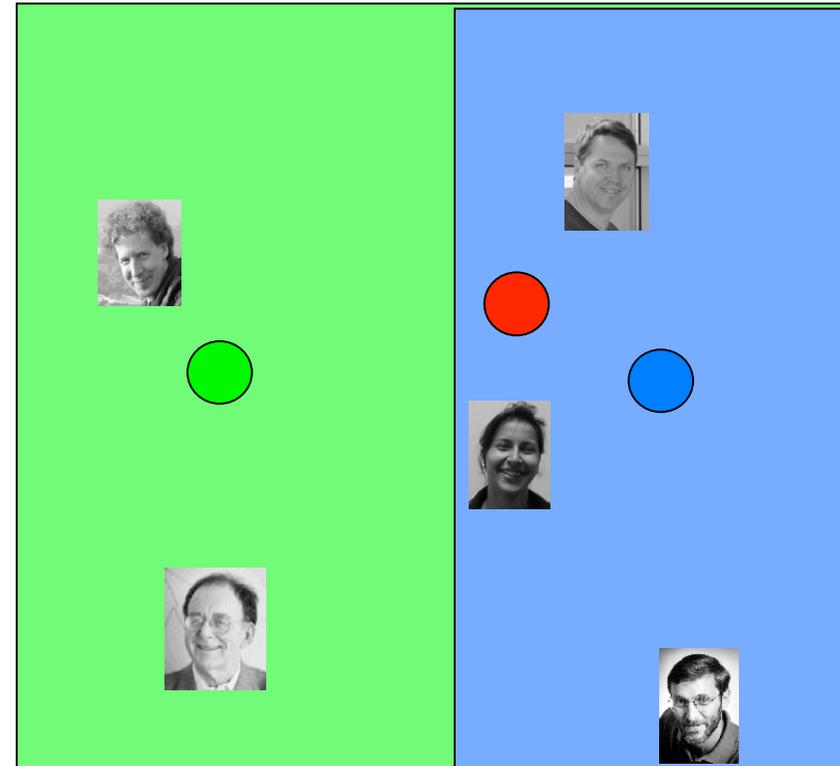




Content Addressable Network (CAN)

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

- Dateien werden in durch (zweiwertige)-Hash-Funktion in das Quadrat abgebildet
- Am Anfang ist ein leeres Quadrat mit nur einem Peer als Besitzer
- Der Besitzer einer Fläche speichert alle Einträge in der Fläche
- Ein Peer wählt einen zufälligen Punkt in der Ebene
 - Der Besitzer des entsprechenden Quadrats teilt seine Fläche und
 - übergibt die Hälfte dem neuen Peer

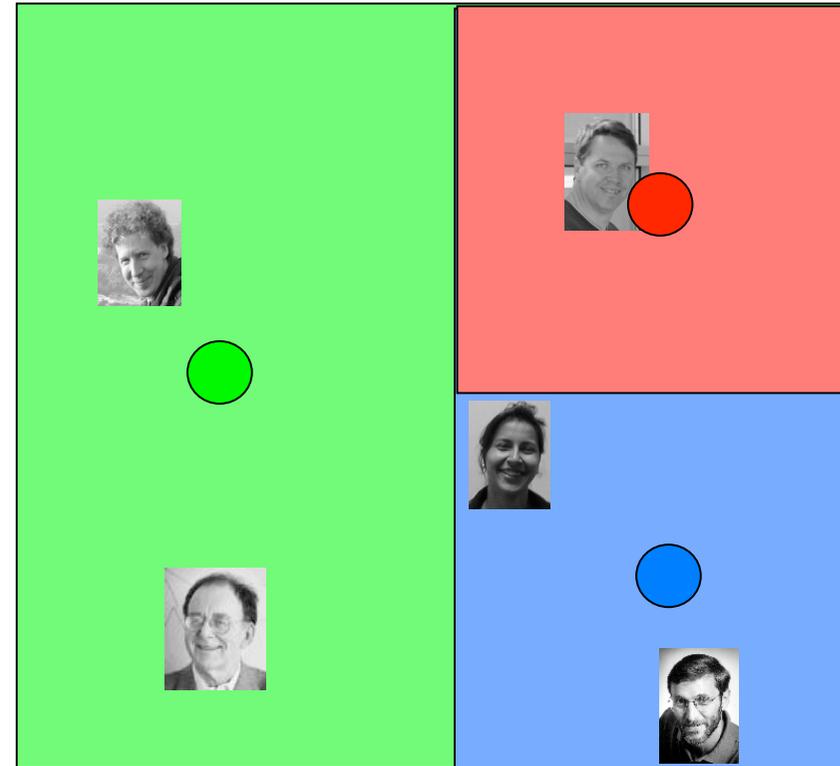




Content Addressable Network (CAN)

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

- Dateien werden in durch (zweiwertige)-Hash-Funktion in das Quadrat abgebildet
- Am Anfang ist ein leeres Quadrat mit nur einem Peer als Besitzer
- Der Besitzer einer Fläche speichert alle Einträge in der Fläche
- Ein Peer wählt einen zufälligen Punkt in der Ebene
 - Der Besitzer des entsprechenden Quadrats teilt seine Fläche und
 - übergibt die Hälfte dem neuen Peer

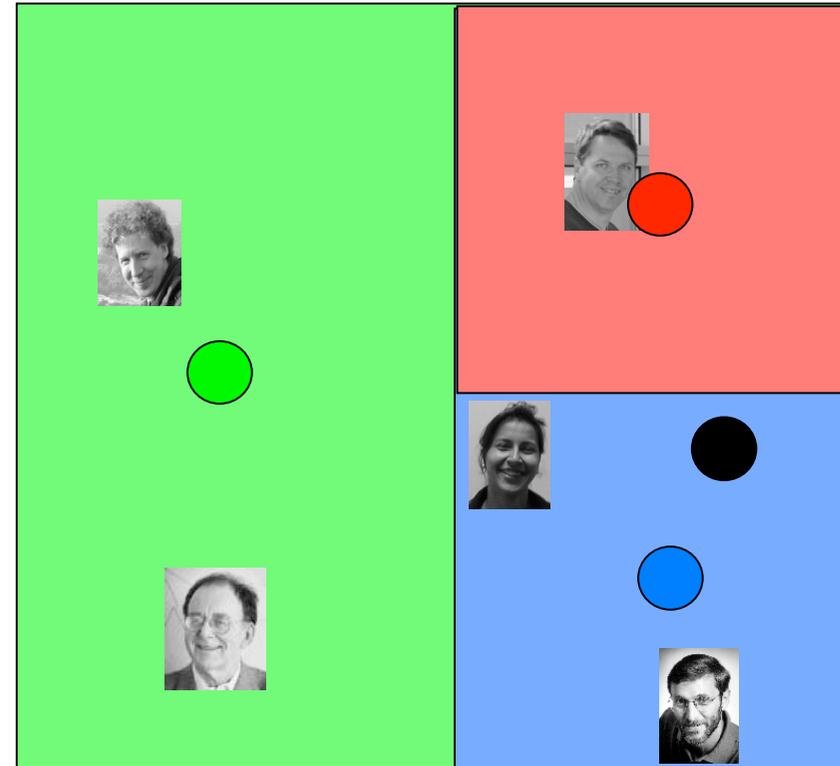




Content Addressable Network (CAN)

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

- Dateien werden in durch (zweiwertige)-Hash-Funktion in das Quadrat abgebildet
- Am Anfang ist ein leeres Quadrat mit nur einem Peer als Besitzer
- Der Besitzer einer Fläche speichert alle Einträge in der Fläche
- Ein Peer wählt einen zufälligen Punkt in der Ebene
 - Der Besitzer des entsprechenden Quadrats teilt seine Fläche und
 - übergibt die Hälfte dem neuen Peer

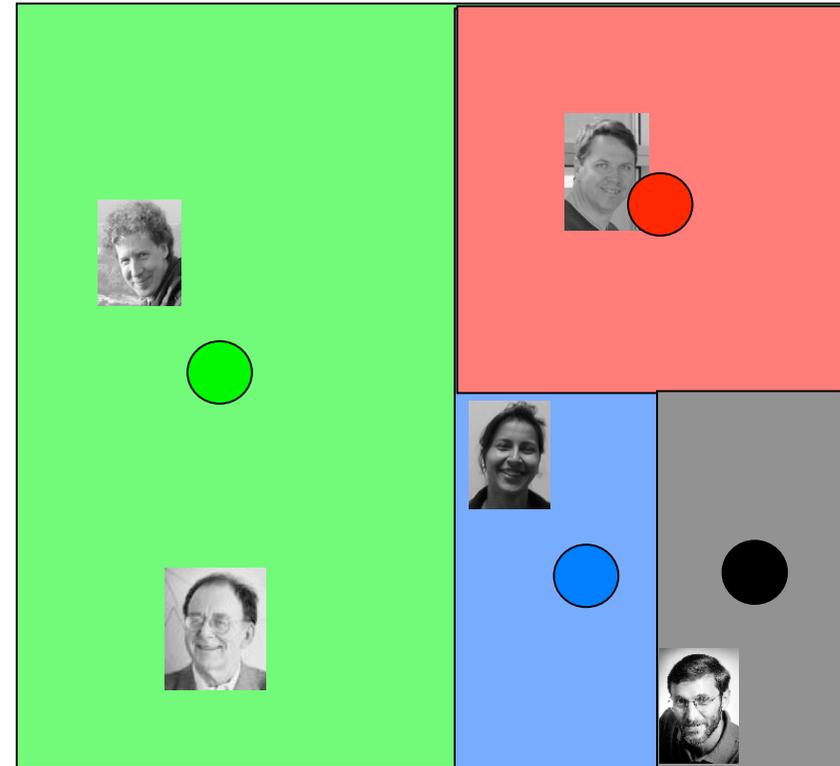




Content Addressable Network (CAN)

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

- Dateien werden in durch (zweiwertige)-Hash-Funktion in das Quadrat abgebildet
- Am Anfang ist ein leeres Quadrat mit nur einem Peer als Besitzer
- Der Besitzer einer Fläche speichert alle Einträge in der Fläche
- Ein Peer wählt einen zufälligen Punkt in der Ebene
 - Der Besitzer des entsprechenden Quadrats teilt seine Fläche und
 - übergibt die Hälfte dem neuen Peer

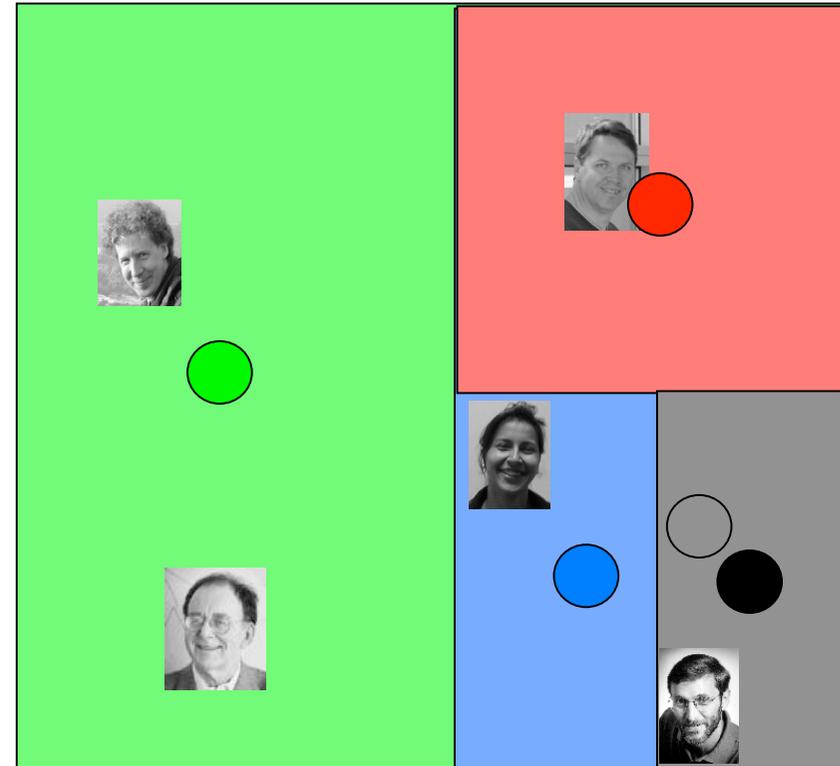




Content Addressable Network (CAN)

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

- Dateien werden in durch (zweiwertige)-Hash-Funktion in das Quadrat abgebildet
- Am Anfang ist ein leeres Quadrat mit nur einem Peer als Besitzer
- Der Besitzer einer Fläche speichert alle Einträge in der Fläche
- Ein Peer wählt einen zufälligen Punkt in der Ebene
 - Der Besitzer des entsprechenden Quadrats teilt seine Fläche und
 - übergibt die Hälfte dem neuen Peer

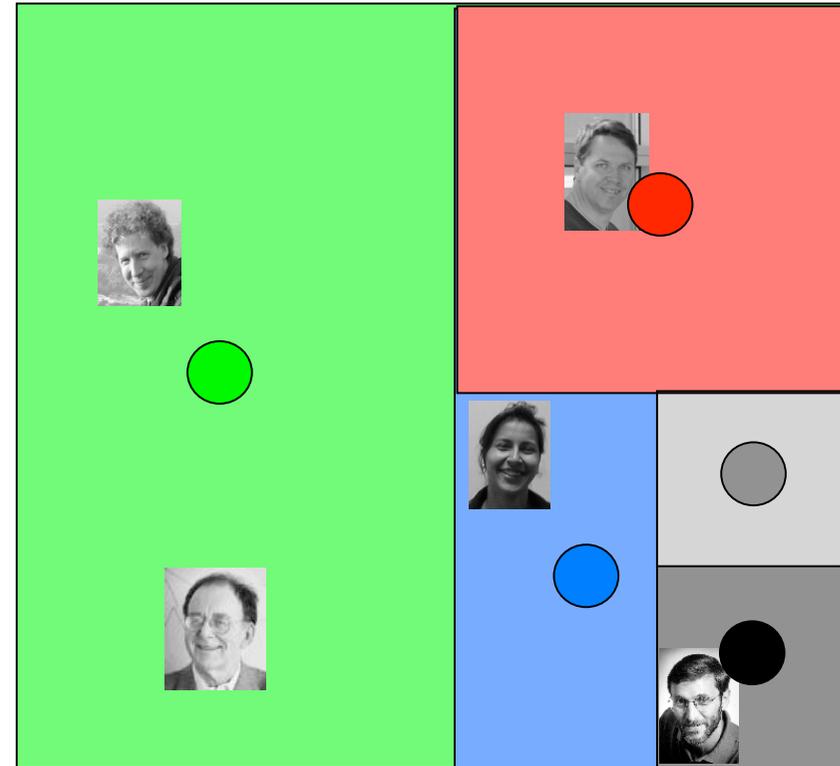




Content Addressable Network (CAN)

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

- Dateien werden in durch (zweiwertige)-Hash-Funktion in das Quadrat abgebildet
- Am Anfang ist ein leeres Quadrat mit nur einem Peer als Besitzer
- Der Besitzer einer Fläche speichert alle Einträge in der Fläche
- Ein Peer wählt einen zufälligen Punkt in der Ebene
 - Der Besitzer des entsprechenden Quadrats teilt seine Fläche und
 - übergibt die Hälfte dem neuen Peer





Wie gleichmäßig werden die Daten verteilt?

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

➤ Lemma

- Mit Wahrscheinlichkeit $(\log n) n^{-c}$ wird ein Rechteck der Größe $2c(\ln n)/n$ nicht geteilt.

➤ Wenn m Elemente insgesamt gespeichert werden,

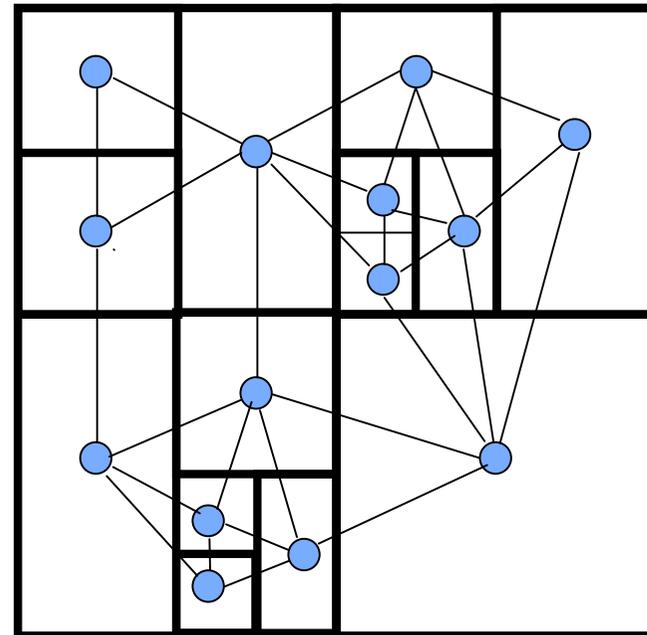
- so erhält jeder Peer also maximal $2c(\ln n) m/n$ Elemente,
- während der Durchschnitt m/n Elemente speichert

➤ Also speichert jeder Peer höchstens $2c(\ln n)$ mal mehr als der Durchschnittspeer mit hoher Wahrscheinlichkeit.



Lookup in CAN

- Zuerst wird Ort des Indexes durch Berechnung der Hash-Funktion bestimmt
- Zwischen den Besitzer benachbarter Rechtecke bestehen Kanten
- Anfrage wird in Richtung des Index weitergeleitet
- **d Dimension des Quadrats**
 - 1: Linie
 - 2: Quadrat
 - 3: Würfel
 - 4: ...
- Erwartete Anzahl Hops in d Dimensionen:
 $n^{1/d}$
- Durchschnittlicher Grad eines Knotens:
 $O(d)$





Einfügen in CAN = Random Tree

➤ Random Tree

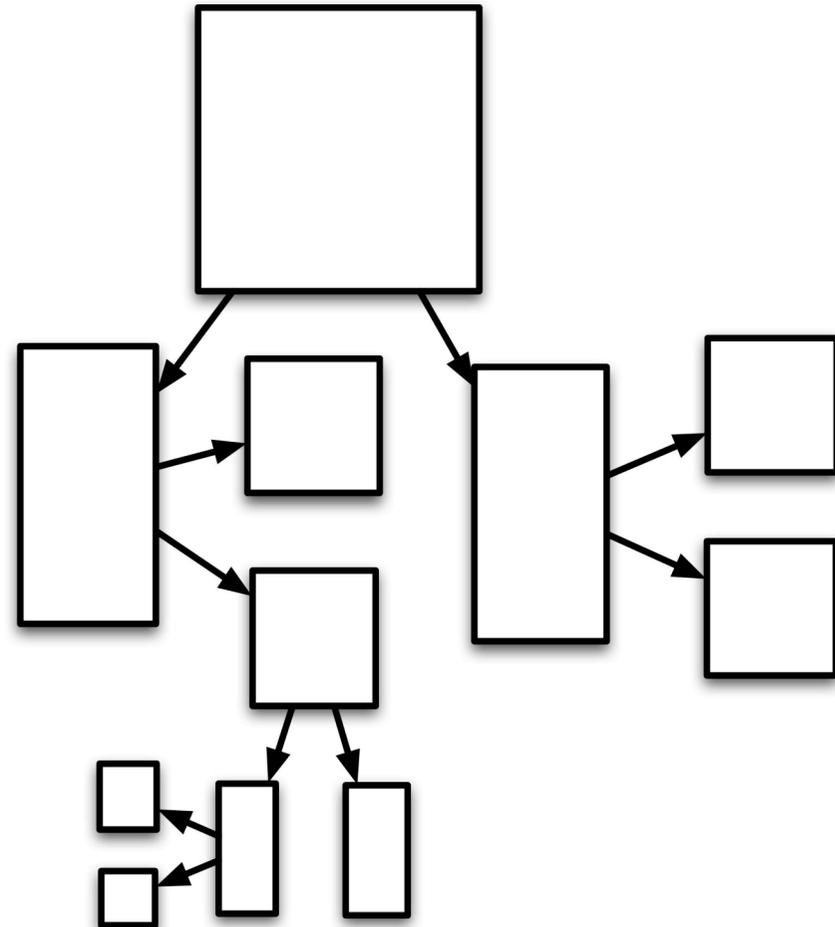
- Neue Blätter werden zufällig eingefügt
- Falls Wurzel interner Knoten, gehe zufällig in linken oder rechten Teilbaum
- Falls Wurzel ist Blatt, füge zwei Blatt an diese Wurzel an

➤ Tiefe:

- im Erwartungswert $2 \log n + O(1)$
- Tiefe $O(\log n)$ mit hoher Wahrscheinlichkeit, d.h. $1 - n^{-c}$

➤ Beobachtung

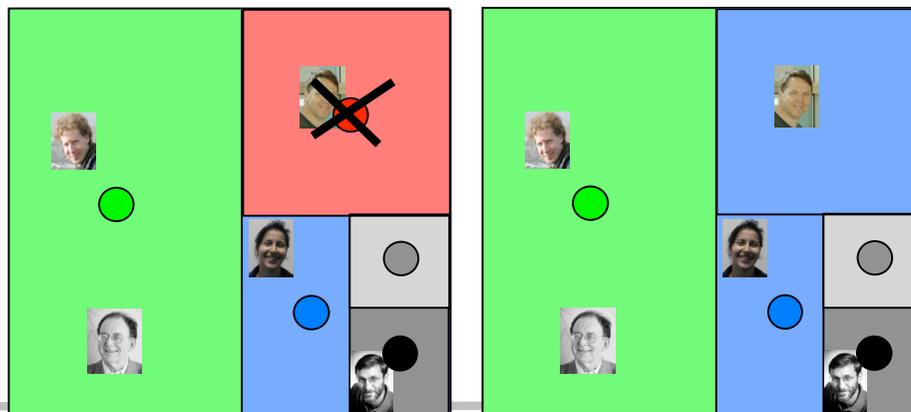
- CAN fügt neue Peers ein wie neue Blätter beim Random Tree eingefügt werden



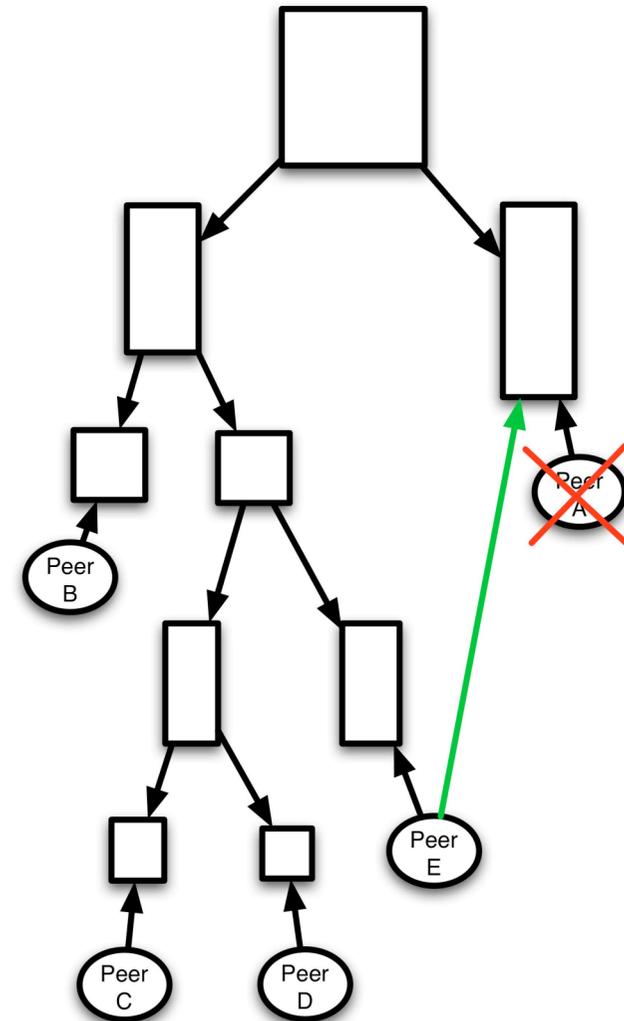


Entfernen von Peers in CAN

- **Verschwundet ein Peer,**
 - meldet er das nicht vorher an
 - Daher Nachbarn testen regelmäßig Anwesenheit
 - übernimmt der erste Nachbar der das merkt das Gebiet des verschwundenen Peers
- **Peers können mehrere Gebiete verwalten**
- **Häufiges Einfügen und Entfernen führt zur Kleinstaaterei (Fragmentierung)**



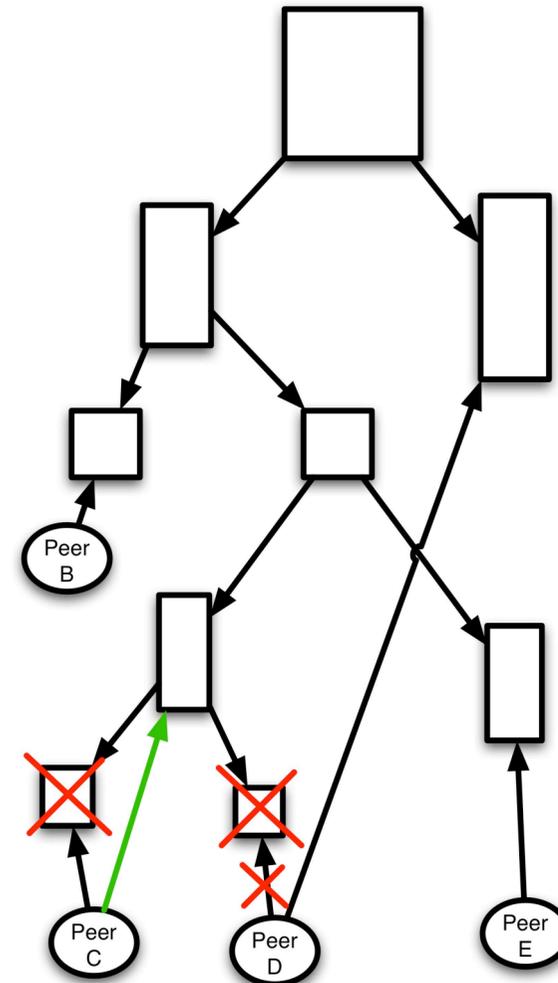
Peer-to-Peer-Netzwerke





Defragmentierung - der einfache Fall

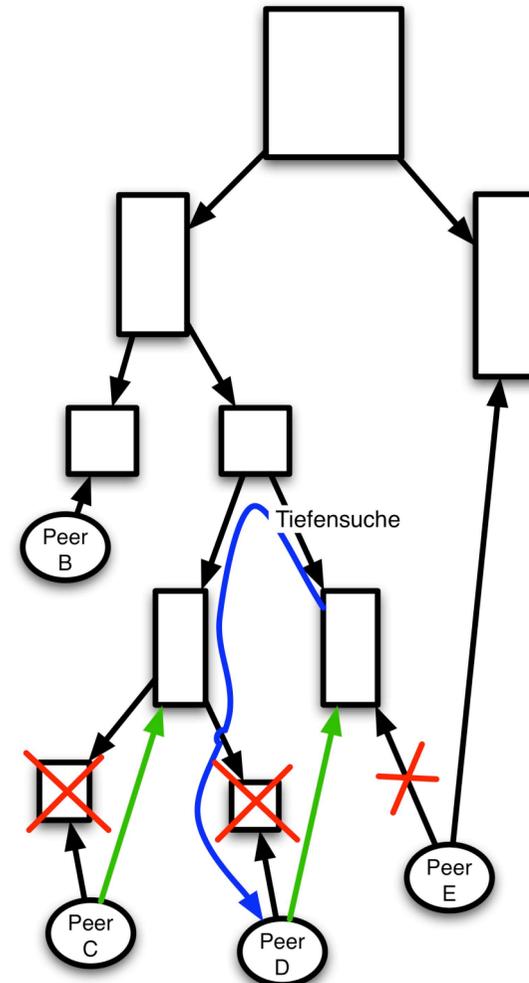
- Um die Fragmentierung zu beseitigen, wird von Zeit zu Zeit eine Zonenneuzuweisung durchgeführt
- Für jeden Peer, der mindestens zwei Zonen hat,
- Lösche kleinste Zone des Peers und finde Ersatzpeer für dieses Gebiet
- **1. Fall:**
Nachbarzone im Baum ist ungeteilt
 - Dann sind beide Peers Blätter im CAN-Baum
 - Übertrage Zone dem Nachbarknoten





Defragmentierung - der schwierige Fall

- Für jeden Peer, der mindestens zwei Zonen hat,
 - Lösche kleinste Zone des Peers und finde Ersatzpeer für dieses Gebiet
- 2. Fall:
Nachbarzone im Baum ist weiter unterteilt
 - Führe Tiefensuche in Nachbarbaum durch, bis zwei benachbarte Blätter gefunden worden sind
 - Übertrage einem Blatt (Peer) die Zonen beider Blätter und
 - wähle das andere Blatt (Peer) als Ersatzpeer





Systemverbesserungen für CAN

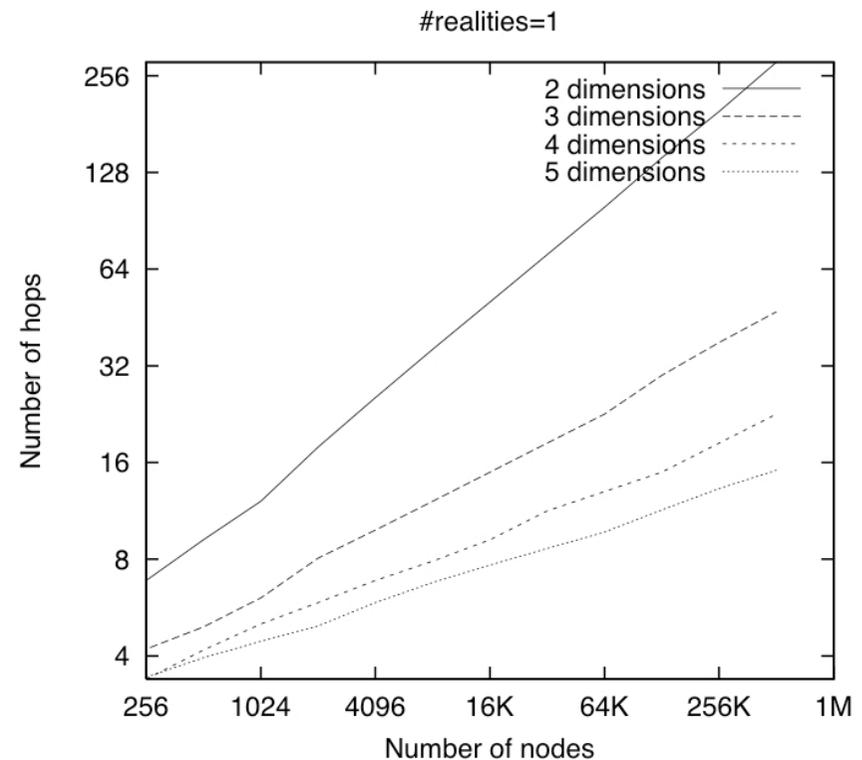
Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelbauer

1. **Mehrdimensionale Räume**
2. **Verschiedene Realitäten**
3. **Abstandsmetrik für Routing**
4. **Überladen der Zonen**
5. **Mehrfaches Hashing**
6. **Topologie-angepasste Netzwerkkonstruktion**
7. **Gleichmäßigere Partitionierung**
8. **Caching, Replikation und Hot-Spot-Management**



Mehrdimensionale Räume

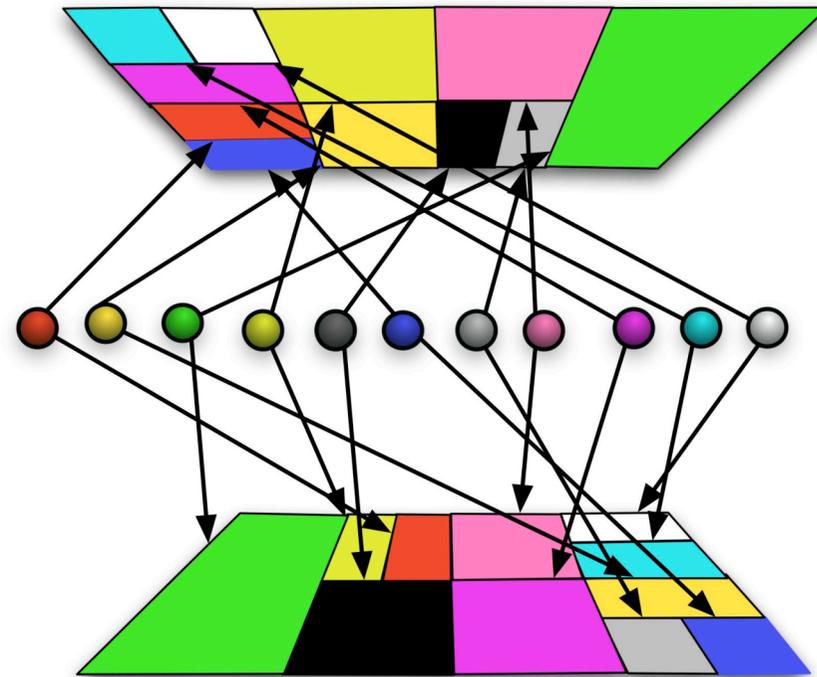
- **D-dimensionaler Raum (statt 2-D)**
 - 1: Linie
 - 2: Quadrat
 - 3: Würfel
 - ...
- **Die erwartete Pfadlänge bei d Dimensionen ist $O(n^{1/d})$**
- **Erwartete Anzahl von Nachbarn $O(d)$**





Mehrere Realitäten

- **Simultan werden r CAN-Netzwerke aufgebaut**
- **Jedes CAN-Netzwerk wird Realität genannt**
- **Auf der Suche nach einem Feld**
 - springt man zwischen den Realitäten
 - wählt man die Realität, in welcher der Abstand zum Ziel am geringsten ist
- **Vorteile**
 - Hohe Robustheit

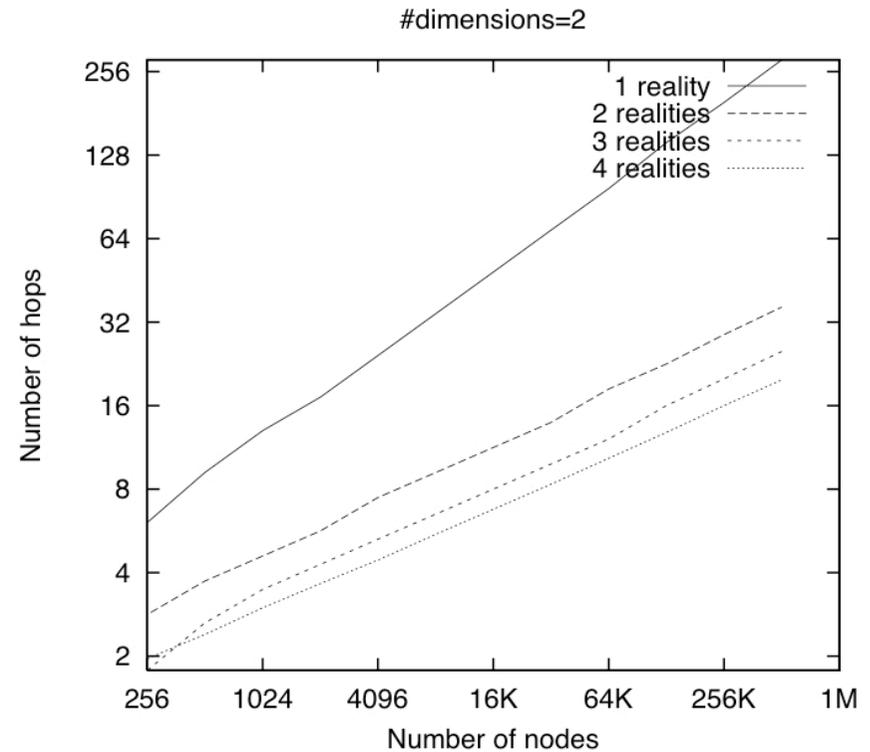




Mehrere Realitäten

➤ Vorteile

- Hohe Robustheit
- Kürzere Wege

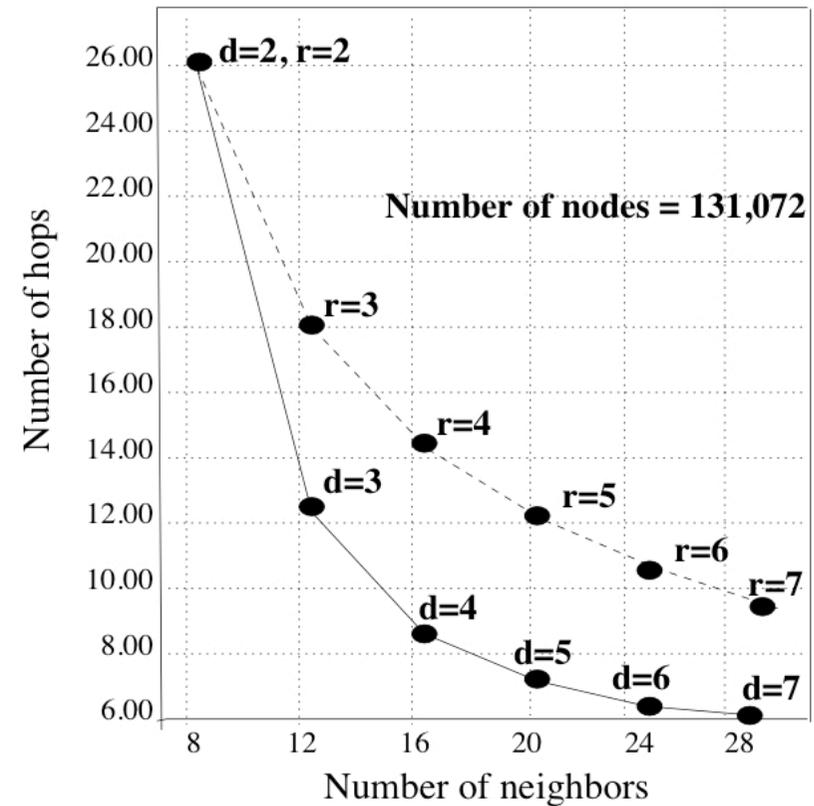




Realitäten versus Dimensionen

- Dimensionen verkürzen die Wege besser
- Realitäten erzeugen robustere Netzwerke

- ————— increasing dimensions, #realities=2
- - - - - - increasing realities, #dimensions=2



Ende der

8. Vorlesung



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

Peer-to-Peer-Netzwerke
Christian Schindelhauer
schindel@informatik.uni-freiburg.de