

Beyond Music File Sharing: A Technical Introduction to P2P Networks

Christian Cikryt

Fachbereich Informatik, Freie Universität Berlin

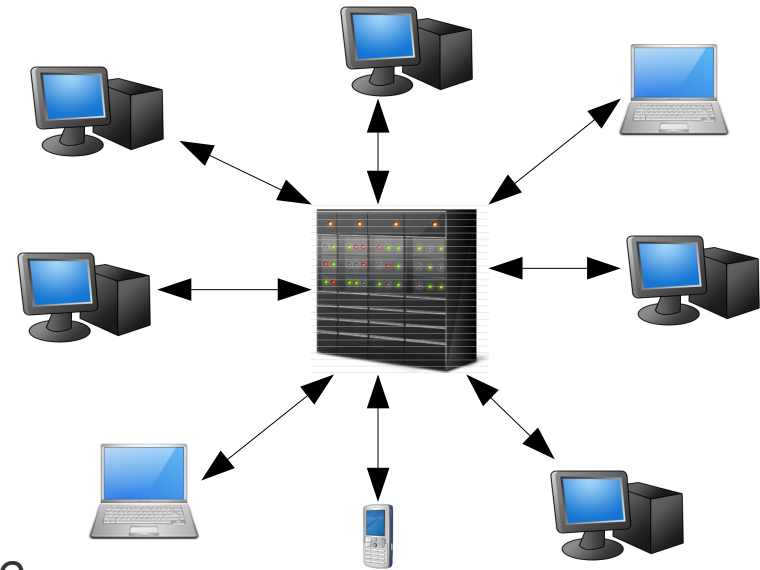
29. Januar 2010

Gliederung

1. Motivation
2. Überblick über P2P Netzwerke
3. Das BitTorrent-Protokoll
4. Zusammenfassung und Ausblick

Motivation

- Probleme klassischer Client-Server-Architekturen:
 - Abhängigkeit vom Serverbetreiber
 - Flaschenhals und Single Point of Failure
 - Skalierbarkeit und Kosten
- Wunsch der Nutzer, eigene Inhalte unmittelbar anzubieten
- Verbreitung von Breitbandanschlüssen

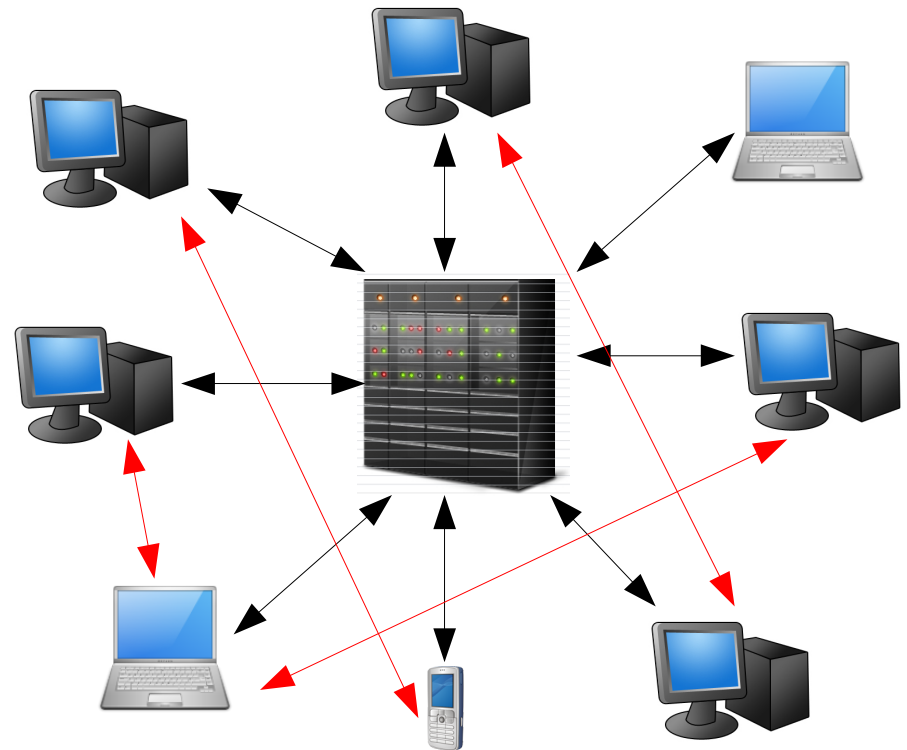


Überblick über P2P Netzwerke

- P2P Netzwerke: verteilte Architektur, in der Teilnehmer einander Ressourcen bereitstellen
- entstehende Probleme:
 - Auffinden von Ressourcen
 - Verschiebung der Skalierungsproblematik auf die Koordination
- Trennung von Koordination und Datentransfer:
 - Overlay-Netzwerk
- unstrukturierte Netzwerke:
 - keine Verknüpfung von Daten zu Location Identifier
- strukturierte Netzwerke:
 - Abbildung von Inhalten und Anbieterinformationen in den selben Adressraum

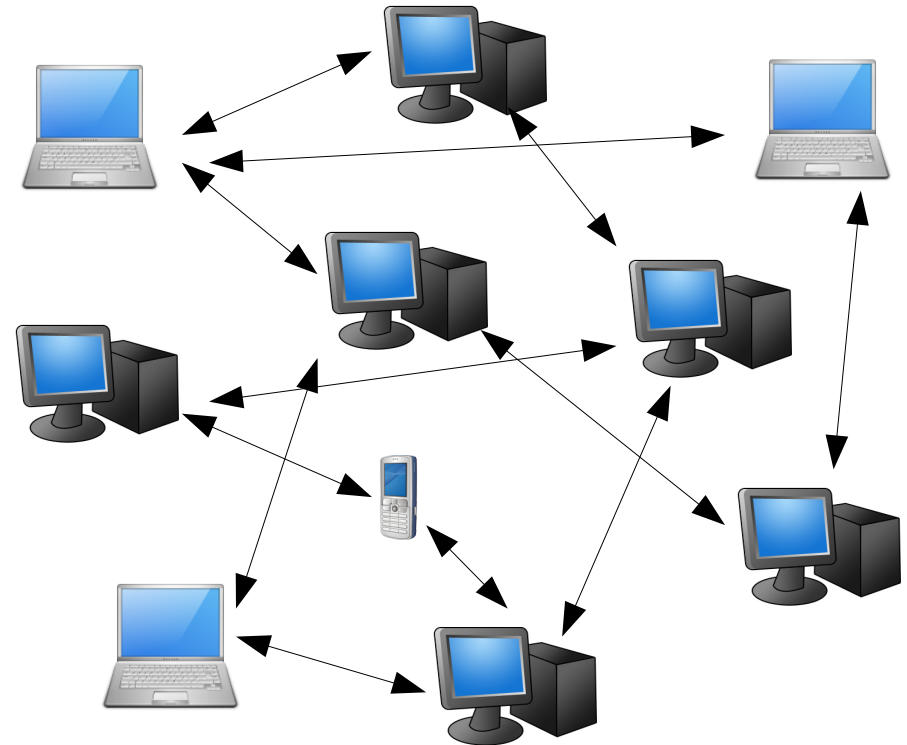
Unstrukturierte Netzwerke: Zentralisiert

- **zentraler** Lookup-Server
- **dezentraler** Datentransfer
- Vorteil:
 - direkte Beantwortung von Anfragen durch den Server
 - wenig Overhead
- Nachteile:
 - skaliert nicht
 - Single Point of Failure
 - Abhängigkeit vom Betreiber



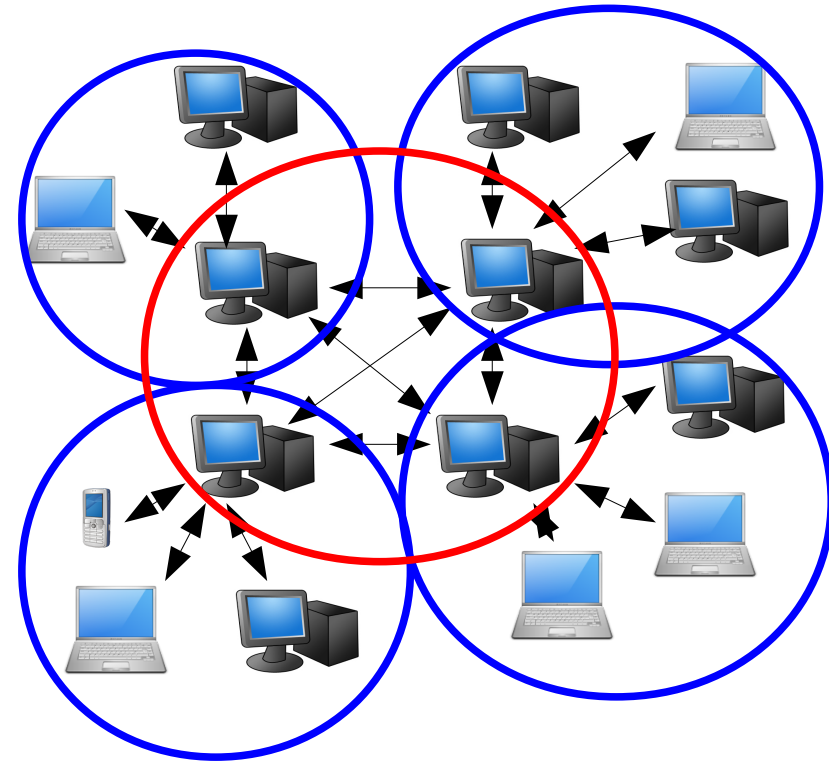
Unstrukturierte Netzwerke: Reine P2P Netze

- keine zentralen Elemente
- Nachrichten werden geflutet
- Vorteile:
 - selbst organisierend
 - prinzipiell skalierbar
- Nachteile:
 - hoher Overhead
 - begrenzte Auffindbarkeit von Ressourcen

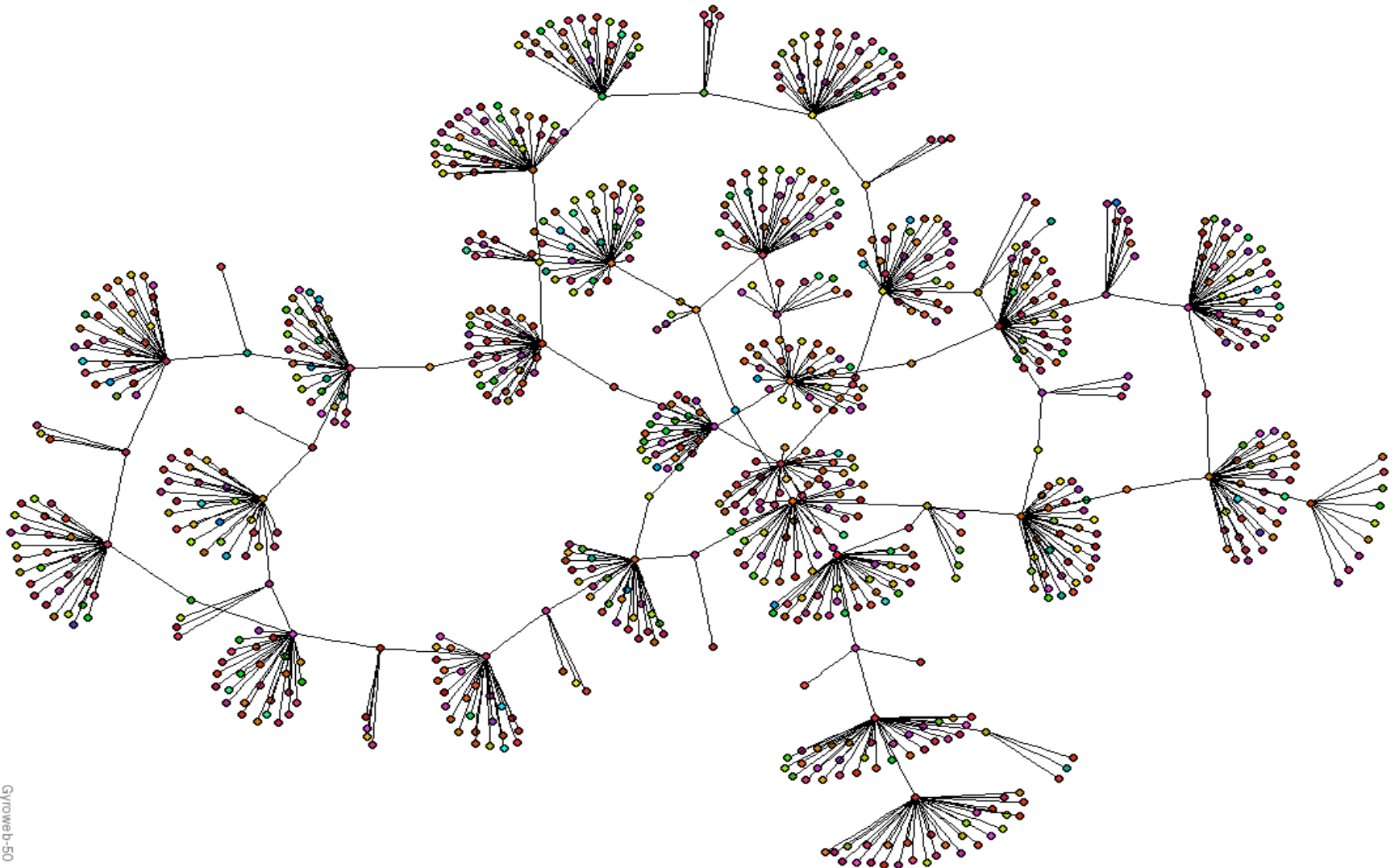


Unstrukturierte Netzwerke: Hybrid

- Hierarchische Struktur:
 - Leaf: direkt mit Ultrapeer verbunden
 - Ultrapeers: bilden reines P2P Netz untereinander
- Vorteile:
 - selbst organisierend und skalierbar
 - geringerer Overhead als reine P2P Systeme
- Nachteile:
 - Kosten für Ultrapeer-Wahl
 - begrenzte Auffindbarkeit von Inhalten



Simulation eines Gnutella 0.6 Netzwerks

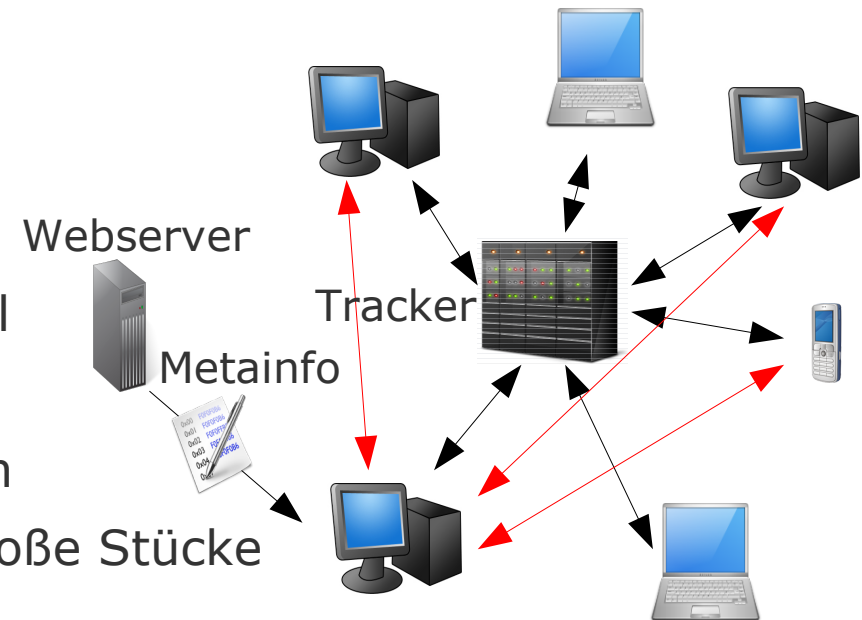


Gyroweb-50

Quelle: Le-Blond and L. Viennot, Mai 2004

BitTorrent

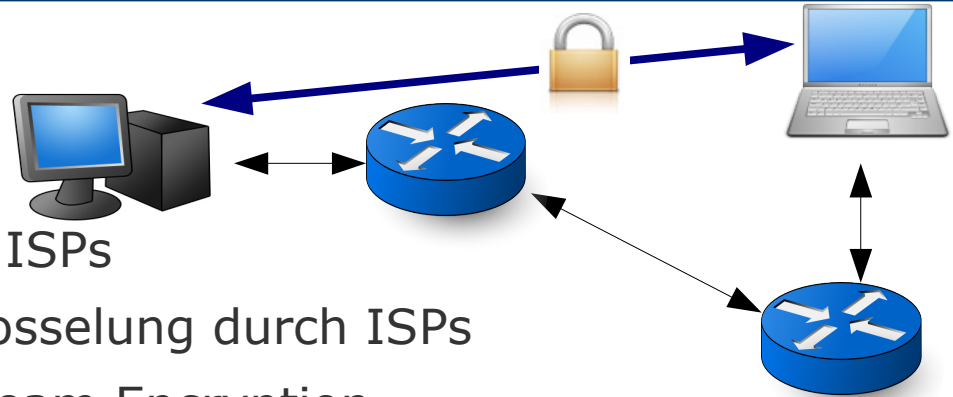
- derzeit meistbenutztes P2P Protokoll
- gleichmäßige Lastverteilung
- Upload von unvollständigen Inhalten
- Unterteilung der Inhalte in gleich große Stücke
- Tracker: Koordination der Peers
- Peer Wire Protocol zwischen Clients:
 - TCP basiert
 - Zustände: (not) choking, (not) interested



BitTorrent - Free-riding

- Free-riding: Downloaden ohne selbst Daten bereitzustellen
- Probleme:
 - Uploaddrosselung
 - Beendigung nach erfolgreichem Download
- Lösungen:
 - Tit-for-Tat Verfahren
 - Optimistic Unchoking
 - Speicherung von Übertragungsvolumen
- Grundvoraussetzung für P2P Systeme: Bereitschaft der Nutzer, aktiv teilzunehmen

BitTorrent - Bandbreitendrosselung



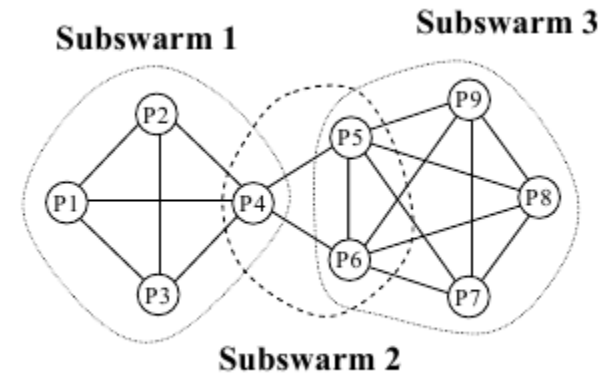
- hohes Traffic-Aufkommen stört ISPs
- Deep Packet Inspection und Drosselung durch ISPs
- Gegenmaßnahme: Message Stream Encryption
 - verschleiert Inhalt und Anwendungsprotokoll
 - Effizienz vor Sicherheit
 - gewisses Maß an Authentifikation
 - Hash der Torrent-Info-Datei als Schlüssel
- Effektivität der Verschlüsselung:
 - überwindbar für einzelne Verbindungen
 - wirksam gegen kollektive Drosselung

BitTorrent - Ausfallsicherheit und Verfügbarkeit

Problem: Tracker als Single Point of Failure und dedizierter Falschenhals

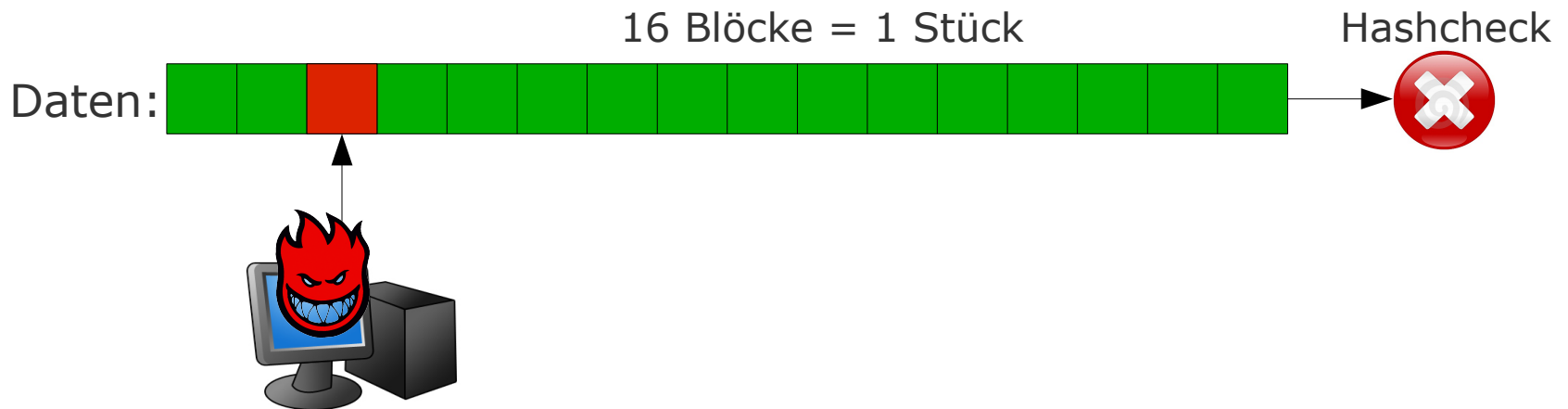
Lösungen:

- mehrere Tracker:
 - Lastverteilung und Redundanz
 - höhere Verfügbarkeit
 - losere Kopplung der Peers
- Peer Exchange (PEX)
- Verteilte Hash-Tabellen:
 - Verteilung der Trackerfunktionen unter den Peers
 - sehr hohe Ausfallsicherheit
- Varianten kombinierbar



BitTorrent - Angriffe auf Downloader

- durch Content-Industrie beauftragte Störung von P2P Netzen
- Fake-block Angriff:



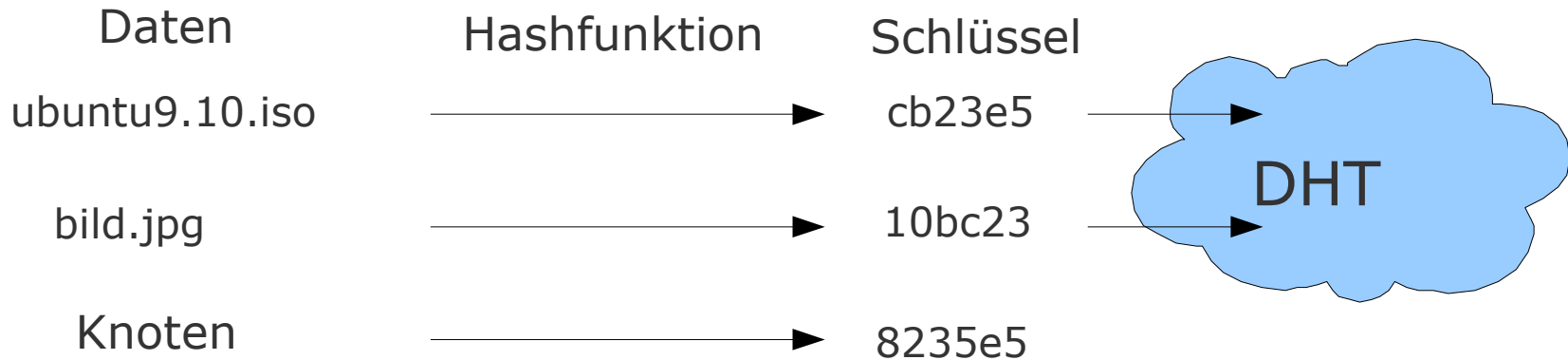
- Uncooperative-Peer Angriff:
 - nur Angebot aber niemals Senden von Daten
 - chatty Peer Angriff als Variation
- Effektivität: Downloadverzögerung um maximal 50%
- Gegenmaßnahme: IP Blacklisting

Zusammenfassung und Ausblick

- verschiedene Lösungsansätze zur Koordination der Peers:
 - effiziente, zentralisierte Systeme mit Single Point of Failure
 - robuste, dezentralisierte Netzwerke mit hohem Overhead
- BitTorrent: robustes Protokoll für lastverteiltes Filesharing
- steigende Popularität von Filehostern
 - asynchroner Down- und Upload
 - Abhängigkeit und Kosten
- neue Einsatzgebiete, z.B. Nano Data Centers

Strukturierte P2P Netzwerke

- weder zentrale Elemente noch Fluten von Nachrichten
- Knoten und seine Daten werden verknüpft
- Daten und Netzwerkknoten in den selben Wertebereich abgebildet



Strukturierte P2P Netzwerke

- Aufteilung des Schlüsselraum
- Routingtabelle mit Nachbarnknoten
- Vorteile:
 - skalierbar, selbst organisierend
 - Nachrichtenzustellung in $O(\log n)$
- Nachteil:
 - Strukturaufbau und -erhalt aufwendig

