

Mein Ziel

Ich will, dass Sie die Fähigkeiten erwerben, unter denen zu sein, die die Deployment Zeit um Größenordnungen verringern, ohne dabei die Kosten dafür zu zahlen, Torrents als Blackbox zu sehen.

BitTorrent Ein p2p-System zum Verteilen großer Datenmengen, bei dem die Verwaltung auf zentralisierten Trackern läuft

Anne Babenhäuserheide

Verteilte Systeme 1: peer-to-peer

Einstieg

Grundprobleme

Gnutella

Kademlia

BitTorrent Downloads

Freenet

Abschluss

Warum?

Projekte

- Notieren Sie bitte Ideen
- Modulplan: 39h Selbststudium
- Projekt interessanter und nützlicher als Klausur
- Projektideen sammeln.

Auf einem der aktuellen *NLnet-Projekte* aufbauen?

“Laptop multitasking hinders classroom learning for both users and nearby peers”
— Sana et al. (2013)

Arne Babenhauerheide

Verteilte Systeme 1: peer-to-peer

Einstieg Grundprobleme Gnuttella Kademilja BitTorrent Downloads Freenet Abschluss

Wiederholung

Wiederholung 2

- **Architektur:** Layered, Object, Resource, Event
- Schichten und Overlay Netze
- Prozesse sind isoliert, Threads teilen Speicher.
- **Middleware** als Schicht: Übernimmt Verteilung, gibt Garantien.
- **Messaging:** Request-Reply, Pub-Sub, Pipeline.
- **Overlay** metriken: Link Stress und Stretch

Was und warum?

- Was?
peer-to-peer (p2p) peers (gleichberechtigte Partner) arbeiten zusammen, um sich gegenseitig einen Dienst zu erbringen.
- Warum?
Sie haben ein unerwartet beliebtes Programm geschrieben. Jetzt wollen es 100 Millionen Leute herunterladen. Größe: 50GiB. Wie viel kostet die Verteilung?

Ziele für Grundprobleme

- Sie können die Grundprobleme beschreiben, die Peer-to-Peer-Netze lösen müssen:

- Kommunikation
- Störungsresistenz

Warum nicht p2p?

- Viele der einfachen Lösungen unmöglich, z.B. Geld auf Probleme werfen.

Arne Babenhausen



Einstieg: Wie finde ich meinen Platz im Netz?

- **Erste Adressen:** Wie finde ich Adressen anderer Knoten?
- **Wahl der Verbindungen:** Mit wem sollte ich mich verbinden?
- **Routing-Informationen:** Welche Daten brauchen die Knoten?



Suche: Wo gibt es, was ich brauche?

Zwei Konzepte:

- Pfade zu existierenden Daten finden: Gnutella
- Daten an die richtigen Orte bringen: Kademia, BitTorrent VHT, Freenet



Kommunikation

- Vier-Augen Gespräch (PM/DM/msg/Anruf/...)
- Unterhaltung in Gruppen (Chat, Forum, Videokonferenz, ...)
- Öffentliche Unterhaltung
- Von neuen Inhalten erfahren
- Informationen über Inhalte (Kommentare, Bewertung, ...)



Störungsresistenz: Wie skaliert Gewünschtes besser als Unerwünschtes?

Störung

"Disruption", alles, was die Qualität des Dienstes für die Nutzer verringert

Auf jeder Ebene nötig

- **Knotenauswahl:** Verbindung mit Angreifern
- **Suche:** Spam, Falschinformationen
- **Verbreitung:** Dateien korrumpieren
- **Kommunikation:** Spam, Belästigung und Zensur⁶

⁶ Das Web betrachtet Zensur als Störung und lenkt Anfragen darum herum.
The Internet treats censorship as a malfunction and routes around it. – John Perry Barlow



Weitere Eigenschaft: Grad der Verteilung

Serverkoordinierte Teilgruppen bis vollständig dezentrale Interaktion.



Ziele

Sie verstehen die grundlegende Funktionsweise von Gnutella als Beispiel einer effizienten, dezentralen Schlüsselwort-Suche.

Sie erkennen, wo die für Gnutella entwickelten Techniken sinnvoll genutzt werden können.

Sie wissen, welche Probleme ungelöst blieben.



Strukturiert vs. Unstrukturiert

Strukturiert

- **Erste Adressen:** Braucht Topologie⁴
- **Wahl der Verbindungen:** Nur bestimmte sinnvoll
- **Routing-Informationen:** Durch Auswahl der Partner (Peers)

Unstrukturiert

- **Erste Adressen:** Einfache Liste
- **Wahl der Verbindungen:** Beliebige Andere
- **Routing-Informationen:** Explizit austauschen

Kann ich alle direkt erreichen?

⁴Topologie: Struktur des Netzes.



Schlüssel zum Licht



Störungsresistenz: Wie skaliert Gewünschtes besser als Unerwünschtes?

Störung

"Disruption", alles, was den die Qualität des Dienstes für die Nutzenden verringert



Störquellen

Sammeln am Flipchart

⁷Werbung ist Spam durch die genutzte Plattform.



Zusammenfassung

- **Einstieg:** Erste Adressen und Routing-Info
- **Suche:** Schlüsselwort, Inhalt, Public Key
- **Störung:** Parasiten, Trolle, Spammer, Konkurrenten und Angreifer
- **Verbreitung:** Tracker, Download-Mesh, Fragmente mit Caching
- **Kommunikation:** Privatnachricht, Forum, Neuigkeiten, Kommentare



Inhalt

- **Nutzersicht:** Das war Gnutella
- **Einstieg:** GWebCaches
- **Suche:** Slow-Start + Keyword-Multicast
- **Verteilung:** Download-Mesh
- **Kommunikation:** Neues und Sammlung zeigen
- **Störungsresistenz:** Heuristik oder Inhalts-Matrizen



Suche: Wonach suchen?

- **Schlüsselwort:** Gnutella, Skype (vor MS)
- **Inhalts-Hash:** Kademia, BitTorrent VHT, Freenet
- **Öffentlicher Schlüssel:** Freenet

Glossar:



BitTorrent VHT Verteilte Hashtabelle, ein DHT

DHT Distributed Hashtable

Öffentlicher Schlüssel public key, das Gegenstück zum privaten Schlüssel in asymmetrischer Verschlüsselung.



Verbreitung: Wie vermeide ich Flaschenhälse?

- **Zentralisiert:** Streaming im Provider-Netz via Multicast
- **Swarming:** Nutzer übernehmen einen Teil der Verteilung
 - Koordiniert von zentraler Stelle: BitTorrent (Tracker)
 - Koordiniert durch die Nutzer: Gnutella (Download-Mesh)
 - Unabhängig verteilte Fragmente: Freenet⁵

Glossar:



Download-Mesh Name des Protokolls

Tracker Ein Server der den BitTorrent-Schwarm koordiniert

⁵Reduziert Swarming auf Download einzelner Dateien, braucht aber caching: Zeitlich begrenzte Zwischenspeicherung.



Störungsresistenz: Wie skaliert Gewünschtes besser als Unerwünschtes?

Störung

"Disruption", alles, was den die Qualität des Dienstes für die Nutzenden verringert

In Gruppen sammeln

- **Knotenauswahl:**
- **Suche:**
- **Verbreitung:**
- **Kommunikation:**



Störquellen

Sammeln am Flipchart

Quellen

- **Parasiten:** Bessere Leistung auf Kosten Anderer (leecher).
- **Trolle:** Kein Finanzinteresse, minimale Ressourcen, nutzen jegliche Lücke.
- **Spammer:** Erfolg durch Verbreitung eigener Inhalte.⁷
- **Konkurrenten:** Erfolg durch verringerte Qualität des Systems.
- **Angreifer:** Erfolg durch Schädigung von Nutzern.

⁷Werbung ist Spam durch die genutzte Plattform.



Gnutella

Erinnert mich bitte an die Pause

On March 14th, 2000, ... an early version ... with a note: "Justin and Tom work for Nullsoft, makers of Winamp and Shoutcast. See? AOL can bring you good things!" ... AOL ordered him to take the program down immediately ... calling Gnutella an "unauthorized freelance project." ... hackers had gone ... to reverse-engineer it ... into the hands of the open-source community ... — *The World's Most Dangerous Geek; Interviewed by David Kushner; RollingStone.com; January 13, 2004.*

Ursprung der ersten Tauschbörse⁸ Sie verschwand nach verlorenen Urheberrechtsklagen der Entwicklungsfirmer in der Obskurität und seine technischen Errungenschaften gerieten in Vergessenheit.

⁸Tauschbörse: Ein Dienst, in dem Nutzern und Nutzer Inhalte anbieten



Sicht der Nutzer/-innen

- 50 Millionen Knoten
- Globale Suche nach Dateinamen und ID3-Tags
- Filter für Creative-Commons-Lizenzen
- Suche nach den neusten Dateien (What's New?)
- Downloads von vielen Quellen ohne zentrale Koordination
- Audio-Streaming um 2004 („Dateivorschau“)
- LimeWire, Bearshare, Shareaza, Phex, gtk-gnutella, ...





Gnutella Routing Experiment

- Peers: Tisch + davor + dahinter
- Letzte 2 Hops
- Suche nach Namen
- Hash = 1. Buchstabe
- QRT¹⁹: Hash der Namen der Peers
- Intra-UP QRT: QRTs der Peers, zusammengefasst

Was müsst ihr vorher austauschen?

¹⁹QRT: Query Routing Table.



Zusammenfassung Gnutella

- Effiziente Suche nach Schlüsselworten
- TCP-basiertes Binärprotokoll, 50 mio Nutzer, 1kiB/s Leaf, 14kiB/s Ultrappeer
- Einstieg: WebCache-Server + Austausch QRT (wie Bloom-Filter)²⁰
- Suche: Slow-Start + QRT Routing
- Verteilung: Download-Mesh
- Störungsresistenz: Heuristik oder Objektbasiert

²⁰Set von schwachen Hashes der Suchwörter, Anzahl keys dynamisch skaliert und interpoliert



Projekideen

- Download-Mesh implementieren
 - Nur Range-Requests + magnet für Quellen
 - Quellen-Gossip via XAlt²¹
 - Mit Merkle-Tree oder hashliste für chunks und mit XNalt
- Suche über WebRTC in Javascript
 - flooding über vereinfachtes Binärprotokoll
 - QRP / QRT
 - Sharing als Upload in local storage
 - GGEP: Generic Gnutella Extension Protocol; Binarprotokoll für beliebige Daten.

²¹XAlt/XNalt: Header, der gute / kaputte Quellen beschreibt.



PAUSE

--- PAUSE ---

Kademlia

Lookup in einer Verteilten Hash-Tabelle (DHT) mit xor-Metrik.

- Nutzersicht
- Suche
- Einstieg (nutzt die Suche)

Ziele

Sie verstehen die grundlegende Funktionsweise von Kademlia als Beispiel einer effizienten, dezentralen Hash-Suche.

Sie erkennen, wo die in Kademlia entwickelten Techniken sinnvoll genutzt werden können.

Sicht der Nutzer/-innen

Werkzeuge

Ursprünglich Tauschbörsen: Kad in aMule, VHT in Torrent clients
Amazon Dynamo verwendet das sehr ähnliche Chord.

Anwendung

- Suche nach exakten Dateien
- Löst Magnet-links auf
- Server-Auswahl zum Schreiben; eventual consistency



Suche in Kademlia

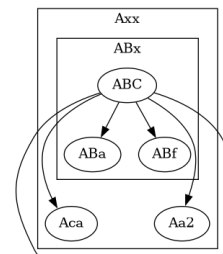
- Jeder Knoten hat eine zufällige ID
- Suche nach Hash → Distributed Hash Table
- Distanz zwischen Hash und ID via xor-Metrik²²
- Schritt für Schritt in O(log(N)) zum richtigen Server

Ähnlich: Chord, Pastry.

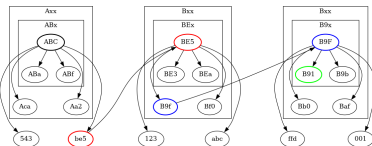
²²xor-Metrik: $4 \text{ xor } 2 \Rightarrow 100 \text{ xor } 010 \Rightarrow 110 \Rightarrow 6$.



Präfix-Buckets



Suche nach b91



Speichern

- Suche nach Knoten nahe Hash.
- STORE: Hash + Wert.

Einstieg in Kademlia

- Kontakt zu mindestens einem bestehenden Knoten.
- Suche nach eigener ID: FIND_NODE (nah = zuständig für ID)
- Erhält Adressen + IDs der am nächsten liegenden Knoten
- Angefragte Knoten behalten auch die eigene Adresse und ID.



Kademlia Routing-Experiment, Einstieg

IDs nach Sitzplatz:

11 - - - - - (Fenster)

10 - - - - -

01 - - - - -

00 - - - - -

000 001 010 011 100 101 110 111

(Tür) (Pu1t)

Präfix-Buckets (E=Eigener Bitwert):

1 4 4 2 1

EEEEx EEExx EEExx Exxxx xxxxx



Kademlia Routing-Experiment, Suche

- ID berechnen (vereinfacht: zufällig²³)
- Name in ID speichern.
- Andere Person: Name abfragen

²³shared state, global; in Realität stattdessen: Hash.



Projekideen

-
-
-
-
-



Zusammenfassung

- Distanz: key-hash XOR node-ID
- Suche: Nächstgelegenen bekannten Knoten nach besseren Knoten fragen
- Kennt mehr nahe als entfernte Knoten
- Speichern wie Suchen
- Einstieg:
 - Suche nach eigener ID
 - Erreichte Knoten nutzen Adresse und ID



BitTorrent

- Verbreitetste Lösung für Swarming
- BitTorrent, IPFS, Blizzard-Updater
- Upload für schnelleren Download
- Koordiniert durch Tracker
- Keine Suche



Ziele

- Sie kennen die grundlegende Funktionsweise von BitTorrent.
- Sie verstehen, wo BitTorrent durch teilweise Zentralisierung Komplexität vermeidet.
- Sie können erklären, warum Torrent für Twitter keine optimale Wahl war, trotzdem aber Faktor 100 schneller, als die vorherige Lösung.



³³ <https://gitlab.com/spritely/golem/blob/master/README.org>
³⁴ <https://fosdem.org/> — viele Vorträge zu decentralization, privacy

Zusammenfassung: Grundprobleme

- **Einstieg:** Wie finde ich meinen Platz im Netz?



- **Suche:** Wo gibt es, was ich brauche?



- **Störungsresistenz:** Wie skaliert Gewünschtes besser als Unerwünschtes?

- **Verbreitung:** Wie vermeide ich Flaschenhälse?



- **Kommunikation:** Wie fließen Informationen durchs Netz?



Anne Babenhauerheide
Verteilte Systeme 1: peer-to-peer
Literatur

Zusammenfassung: Implementierungen

	Einstieg	Suche
Gnutella	WebCache	Slow-Start + Keyword-Multicast
Kademlia	Suche nach eigener ID	xor-Hash-Hierarchie
BitTorrent	Tracker-URL	Kademlia / Tracker / Web
Freenet	Seed-Nodes suchen ID	Greedy Hash auf Small World
WebRTC	WebRTC Server	-
	Verteilung	Störung
Gnutella	Alt+NAlt, Range, Merkle-Tree	Heuristik/Credence
Kademlia	unterschiedlich	-
BitTorrent	Torrent	Wertung auf Tracker
Freenet	Chunk-Tree with Redundancy	Propagating Trust
WebRTC	-	-

Anne Babenhauerheide
Verteilte Systeme 1: peer-to-peer

Viel Erfolg beim Projekt!



Ich wünsche mir, dass einige von Ihnen in 5 Jahren zurückblicken und sagen:

Was ich in verteilte Systeme über p2p-Netze gelernt habe, war einer der Grundsteine meines Erfolges.

Anne Babenhauerheide
Verteilte Systeme 1: peer-to-peer

Verweise I

Mueller, P. A. and Oppenheimer, D. M. (2014). The pen is mightier than the keyboard: Advantages of longhand over laptop note taking. *Psychological Science*, 25(6):1159–1168. PMID: 24760141.

Sana, F., Weston, T., and Cepeda, N. J. (2013). Laptop multitasking hinders classroom learning for both users and nearby peers. *Computers & Education*, 62:24 – 31.

Bilder:
Merkle Tree Patent 1982
<https://patents.google.com/patent/search?family/022107098/publication/US43096687qpnL301US4309669>
Eingereicht 1979 als Methode Diffie-Authentication günstiger zu machen.

Anne Babenhauerheide
Verteilte Systeme 1: peer-to-peer