



Willkommen bei Verteilte Systeme!

Von Datenbanken
über Webdienste
bis zu p2p und Sensornetzen.

..

Heute: Peer-to-peer in der Praxis – wo und wie sich Verteilung lohnt(-e).

Wer nicht aus der Vergangenheit lernt, ist verdammt ihre Fehler wiederholen, mit weniger Zeit, denn „die Probleme sind ja schon gelöst.“

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Warum?

Darum ich

- Seit 2004 in p2p-Entwicklung
- Seit 2013 mit Kompetenz :-)
- Aktuell Release-Manager des Freenet/Hyphane Projektes

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Warum?

Vorträge

- 5 Minuten pro Person, Gruppen möglich.
- Weitere Ideen: Übersicht über einen FOSDEM 2023-Vortrag.

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Wiederholung

Fallacies of distributed Systems

- | | |
|--|--|
| <ul style="list-style-type: none"> ① The network is reliable ② The network is secure ③ The network is homogeneous ④ Topology does not change ⑤ Latency is zero ⑥ Bandwidth is infinite ⑦ Transport cost is zero ⑧ There is one administrator | <ul style="list-style-type: none"> ① Hard disks don't fail ② Power is stable ③ IPs are reachable ④ Constant factors are negligible ⑤ APIs stay compatible ⑥ Textfiles are simple |
|--|--|

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Ablauf

Ziele heute

- Sie kennen die zentralen Herausforderungen der Praxis:
 - Einstieg
 - Suche
 - Inhalte verbreiten
 - Kommunikation
- Sie können in Bezug auf zwei Herausforderungen die Eigenschaften von je zwei p2p-Systemen beschreiben, die sich in der echten Welt bewährt haben.
- Sie können einschätzen, ob ein bestimmtes Konzept diese Herausforderungen bestehen könnte.

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Ablauf

Grundprobleme in Peer-to-Peer-Netzen

- **Einstieg:** Wie finde ich meinen Platz im Netz?
- **Verbreitung:** Wie vermeide ich Flaschenhälse?
- **Kommunikation:** Wie fließen Informationen durchs Netz?
- **Störungsresistenz:** Wie skaliert Gewünschtes besser als Unwünschtes?

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Darum praktisch erprobte p2p-Netze verstehen

after a few days (and especially nights) of nervous full-site tinkering, it turned a 40 minute deploy process into one that lasted just 12 seconds!¹

■ Bittorrent-Deployment: <https://vimeo.com/11280885>

Spoiler: Cut-through routing.

¹<https://web.archive.org/web/20120807165933/http://engineering.twitter.com/2010/07/murder-fast-datacenter-code-deploys.html>

Heute: Peer-to-peer in der Praxis – wo und wie sich Verteilung lohnt(-e).

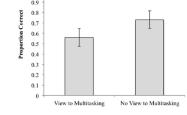
Wer nicht aus der Vergangenheit lernt, ist verdammt ihre Fehler wiederholen, mit weniger Zeit, denn „die Probleme sind ja schon gelöst.“

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Warum?

Laptops: Eigenverantwortlich



"even when laptops are used solely to take notes ... their use results in shallower processing."

— Mueller und Oppenheimer (2014)
Laptops

"Laptop multitasking hinders classroom learning for both users and nearby peers"

— Sana et al. (2013)

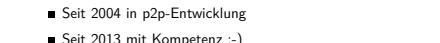
Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Warum?

Möglichst zu

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer

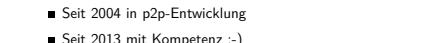


Wiederholung

Wiederholung 1

- Sammlung autonomer Knoten, die als ein kohärentes System erscheinen.
- **Ziele:** Ressourcen, Verteilungstransparenz, Skalierbarkeit
- **Skalierung:** Größe, Geographie, Administration
- Latenz, Partitionierung, Replikation, Caching
- **Fallacies!**
- Cluster, Grid, Cloud, Ubiquitous, Mobile, DIS, Sensornetze

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer

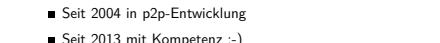


Warum?

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

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Wiederholung

Was und warum?

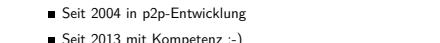
Was?

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?

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer

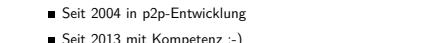


Wiederholung

Ablauf heute

- Grundprobleme
- Gnutella (das Erste verbreitete, komplett verteilte p2p-Netz)
-
- PAUSE 14:30 ---
- Kademia (das am weitesten verbreitete DHT)
- BitTorrent
- Freenet/Hyphane
- Weiteres (Aktuelles, WebRTC, ...)

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer

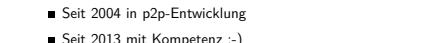


Ablauf

Ablauf heute

- 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?

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Ablauf

Ziele für Grundprobleme

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

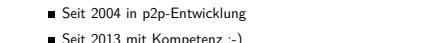
Fundamente

- Einstieg
- Suche
- Verbreitung

Aufbauend

- Kommunikation
- Störungsresistenz

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer

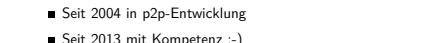


Ablauf

Welche p2p-Netze kennen Sie

- am FlipChart sammeln
-
-
-
-
-

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer

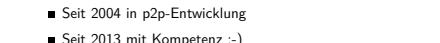


Ablauf

Grundprobleme in Peer-to-Peer-Netzen

- Sie können die Grundprobleme beschreiben, die Peer-to-Peer-Netze lösen müssen:
- **Einstieg:** Wie finde ich meinen Platz im Netz?
- **Suche:** Wo gibt es, was ich brauche?
- **Verbreitung:** Wie vermeide ich Flaschenhälse?
- **Kommunikation:** Wie fließen Informationen durchs Netz?
- **Störungsresistenz:** Wie skaliert Gewünschtes besser als Unwünschtes?

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer

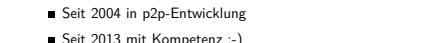


Ablauf

Warum p2p?

- Gestiegende Leistung von Servern. **dwd**² hielte größtenteils Stand (durch vereinfachte Seite³)
- Handys sind durch Batterie und Netz begrenzt → keine kontinuierliche Leistung. (Nachts möglich?)
- Viele der einfachen Lösungen unmöglich, z.B. Geld auf Probleme werfen

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



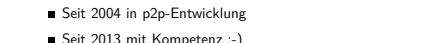
Ablauf

Warum nicht p2p?

²dwd: Deutscher Wetterdienst.

³gibt es eine einfache Lösung?

Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Ablauf



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?



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer

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.



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer

Schlüssel zum Licht



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 den die Qualität des Dienstes für die Nutzenden verringert

Störquellen

Sammeln am Flipchart

Störungen

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

In Gruppen sammeln

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

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.

Werbung ist Spam durch die genutzte Plattform.

Arne Babenhausrede

Verteilte Systeme 1: peer-to-peer

Einstieg: Wie finde ich meinen Platz im Netz?

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:

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.

Werbung ist Spam durch die genutzte Plattform.

Arne Babenhausrede

Verteilte Systeme 1: peer-to-peer

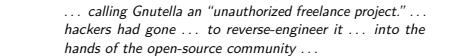
Einstieg: Wie finde ich meinen Platz im Netz?

Suche: Wonach suchen?

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

Glossar:

Search Engine Fin-Dienst in den Nutzern und Nutzer-Inhalten anhören



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, ...

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.

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



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer



Arne Babenhausrede
Verteilte Systeme 1: peer-to-peer





Sicht der Nutzer/-innen

- Informationen von Tracker-Seiten
- Download mit torrent-Datei oder Magnet-Link
- Unterstützt Ordner
- Heute: ipfs: Webseiten über BitTorrent
- NAT-Traversal und IP Verschleieren über Tor

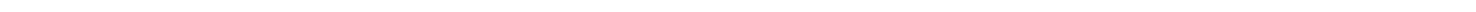
Arne Babenhausereide
Verteilte Systeme 1: peer-to-peer



Anreiz zum Hochladen

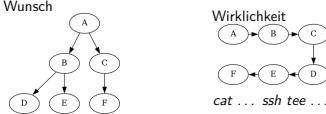
- Uploadverhältnis wird geprüft
- Freeloader²⁵ werden von anderen clients gedrosselt (choked: niedrigere Downloadrate)
- In Literatur ist der Anreiz im Vergleich zu anderen Themen stark vertreten, in der Praxis sind die angebundenen Foren wichtig

²⁵Freeloader: Leute, die nichts hochladen. Auch „Leech“. Gegenteil: „Seed“.



Torrent für Twitter-Deployment

- Kosten bei Twitter: Übertragung über viele Schritte
- Torrent überträgt in Fragmenten.



Cut-through routing / streaming.



Freenet/Hyphnet

Zensur-Resistente Kommunikation auf Freund-zu-Freund Netzwerk

Dezentrale Datenbank mit pubkey-Zugriff

- Ziele
- Verwendung
- Einstieg
- Small-World
- Suche
- Verteilung
- Mutability
- Kommunikation
- Schnittstellen



Einstieg in Freenet

- OpenNet:

 - Ähnlich Kademlia: Wähle bekannte Seednode²⁶, Seednode sucht nach ID → Referenzen
 - Unterschied zu Kademlia: Nicht nur IP, sondern Referenz mit Schlüssel

- Friend-to-Friend:

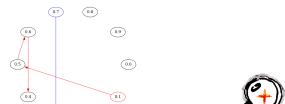
 - Feste Verbindungen
 - Knoten tauschen ihre IDs, um das soziale Small-World-Netzwerk zu rekonstruieren
⇒ Overlay-Kosten minimieren.

²⁶Seednode: Bekannter Knoten, der Verbindungen zu anderen vermittelt.



Freenet Suche

- Wie Kademlia, aber hop for hop weitergeleitet → keine Globale Erreichbarkeit oder Sichtbarkeit
- Suche nach public key möglich
- Keyspace: [0.0 : 0.1)



Arne Babenhausereide
Verteilte Systeme 1: peer-to-peer



Konzept von BitTorrent

Tracker: Webseite

- Koordiniert Schwärme
- Suche, Foren, Bewertung, Verifizierung, Gemeinschaft
- Statistiken: Seeder, Leacher
- Liefert keine Daten
- Aggregiert, wer wie viel hochlädt -> Anreiz



Weiteres

- VHT statt Tracker möglich (Kademlia)
- Freies Protokoll mit vielen Implementierungen
- Weiterentwicklung in der Community
- IPFS nutzt Torrents für dezentral gecachte Webseiten



Torrent-Datei

- Tracker URL(-s)
- Hashes für Chunks
- Namen der Datei(-en)
- Kann Ordner enthalten²⁴



24 http://www.bittorrent.org/beps/bep_0003.html



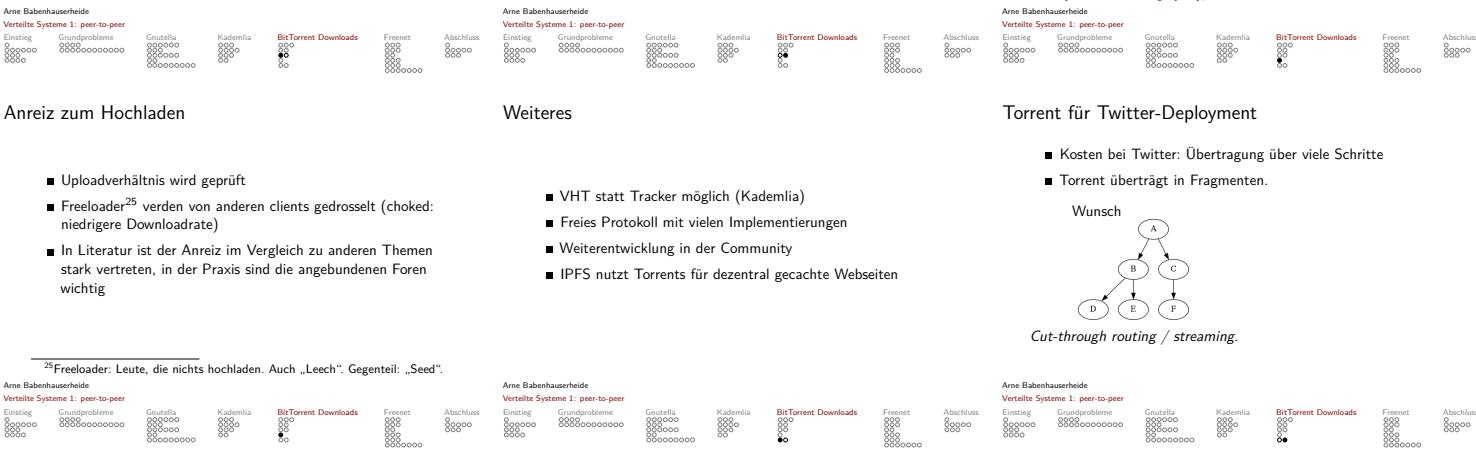
Torrent für Twitter-Deployment

- Kosten bei Twitter: Übertragung über viele Schritte
- Torrent überträgt in Fragmenten.

Wunsch



Cut-through routing / streaming.



Projektideen

Wirklichkeit





Freenet als Datenbank

- Suche nach Public Key + Pfad
- → persönlicher Keypair
- → tarballs für strukturierte Daten
- → pub-sub-Protokolle auf dezentraler Datenbank
- → Webseiten, Foren, Chat, ...

1 Minute Round-Trip-Time

Optimierung: Schlüssel Abonnieren, um 10k Schlüssel zu beobachten und Updates schnell zu sehen.

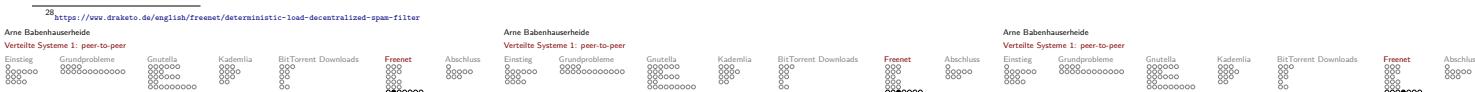


Spam-Abwehr

WoT (Web of Trust): Eine von zwei praktisch genutzten Möglichkeiten. Die andere ist FMS (Freenet Message System).

- ID = USK
- Trust -100 bis 100
- Rank: Distanz → capacity
- Score: Summe über alle Wertungen: trust * rank
- Skaliert bei 22 Nachrichten pro Tag und Person²⁸

²⁸ <https://www.draketo.de/english/freenet/deterministic-load-decentralized-spam-filter>



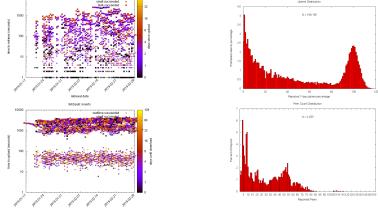
Schnittstelle: FCP

Async für Programme:

- Put/Putdir/Get
 - Password: KSK...
 - Inhalt: CHK.../datei.endung
 - Schlüssel: SSK@pubkey/ordner/datei
 - Updatable: USK@pubkey/ordner/version/datei
- Subscribe to key
- Plugins kontrollieren



Stats



Verschiedenes

- Queuing
- Magnet-Links
- WebRTC
- Verstreutes
- Aktuelles
- Zusammenfassung



WebRTC

- Läuft im Browser (Javascript)
- Liefert Audio, Video, ..., und Peer-Socket
- Erste Verbindung vom Server moderiert – erspart praktische Probleme
- p2p-Systeme, die nicht installiert werden müssen
- Beispiel: WebTorrent <https://webtorrent.io/>



Mutability: O(1) Zugriff auf neuste Version

- Nutzende: SSK@.../meine-seite-1/... → SSK@.../meine-seite-2/activelink.png
- Optimiert: USK@.../meine/seite/1
 - SSK@[key]/[sitename]-DATEHINT-[year]

HINT

46

2013-7-5

DATEHINT-[year], DATEHINT-[year]-WEEK-[week],
DATEHINT-[year]-[month]. DATEHINT-[year]-[month]-[day]

Swapping: Friend-to-Friend wird Small World

9	4	3
5	2	8
1	6	7

3	4	9
5	2	8
1	6	7

6	4	9
5	2	8
1	3	7

6	7	9
5	2	8
1	3	4

Capacity

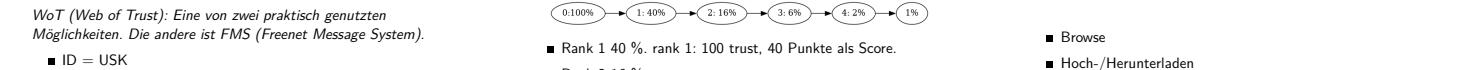
- 0:100% → 1: 40% → 2: 16% → 3: 6% → 4: 2% → 1%
- Rank 1 40 %. rank 1: 100 trust, 40 Punkte als Score.
 - Rank 2 16 %
 - Rank 3 6 %
 - Rank 4 2 %
 - Rank 5 und niedriger: 1 %

Integer-Mathematik: $2 * 6 / 100 = 0$.



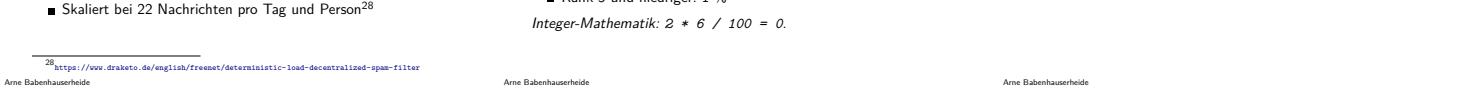
Schnittstelle: Web

- Browse
- Hoch-/Herunterladen
- Plugins
- Freund-zu-Freund-Nachrichten
- Lesezeichen mit Update-Info (5 min Latenz)



Latenz in der Praxis

- Bis zu 1kB, raw, realtime mode: <30s
 - Große Dateien, im Manifest: ~5 min
- Realtime**
- ```
PriorityClass . 2 ;; high
MaxRetries . 0 ;; default: 10
RealTimeFlag . true
ExtraInsertsSingleBlock . 0
ExtraInsertsSplitfileHeaderBlock . 0
```
- Bulk**
- ```
PriorityClass . 3 ;; medium
RealTimeFlag . false
DontCompress . false
```



Kommunikation über Freenet

- Einstieg: Seed-keys + Captcha²⁹-Queue: KSK-Prefix
- Suche: Nutzer spezifische Seiten mit Links, Update-Infos
- Verteilung: Gossip³⁰ keys, Dateien einfach hochladen
- Störungsresistenz: Web of Trust mit langsam steigender Sichtbarkeit

Autospawn node => Freenet als Backend, unsichtbar



²⁹CAPTCHA: Meist Bilder, auf denen Zeichen erkannt werden müssen, um automatische Systeme auszuschließen.
³⁰Gossip: Informationen während normaler Kommunikation von Knoten zu Knoten verteilen.



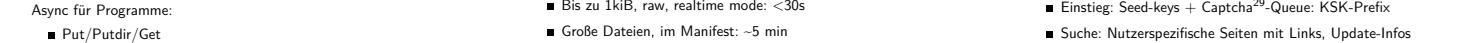
Projektideen

-
-
-
-
-



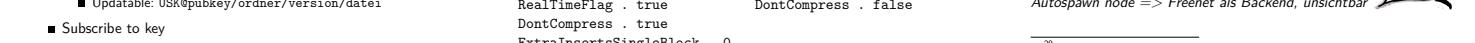
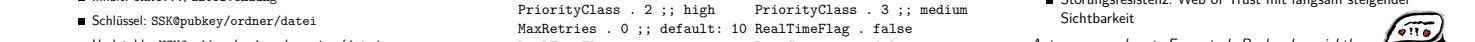
Zusammenfassung

- Einstieg: Suche nach meiner ID bei Seednode
- Suche: Greedy Hash auf Small World
- Verteilung: Chunk-Tree mit Redundancy
- Kommunikation:
 - Einstieg: Seed-keys + CAPTCHA-Queue
 - Suche: Index-Seiten, Update-Erkennung
 - Verteilung: Dateien hochladen, Webseiten
 - Propagating Trust mit langsam steigender Sichtbarkeit



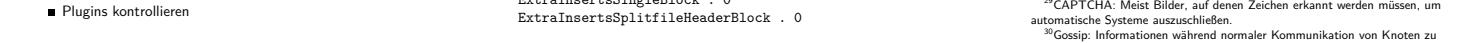
Queuing in p2p-Netzen

- Alle Upload-Warteschlangen sind immer voll
- Queuing-Strategien bestimmen, welche Dateien gut funktionieren
- FIFO Dateien: Große blockieren kleine (wie Alben in Musik-Spiellisten)
- FIFO Chunks: Overhead durch Ständigen Auf- und Abbau von Verbindungen
- HRRN: Große Dateien müssen warten, stört Preview
- Komplexer: Torrent-Superseed u.ä.



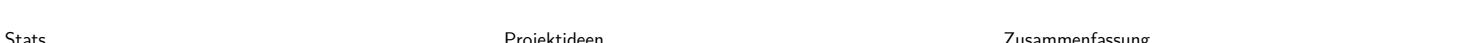
Magnet-Links

```
magnet:?xt=urn:bitprint:TIGER_TREE.SHA1
&xt=urn:btih:BITTORRENT_INFO_HASH
&xt=urn:sha1:HASH
&x1=LENGTH
&dn=NAME
&as=LINK_WITHOUT_HASH
&xas=LINK_WITH_HASH
&kt=SEARCH_STRING
Netzwerk-unabhängig, Link zu HTTP und p2p-Quellen, weitverbreitet
```



Verstreutes

- Optimierung für Netzbetreiber oft angedacht: Clients im gleichen (Sub-)Netz bevorzugen Bei Gnutella „p2p“. Bei Pastry (Windows) laut Ghosh umgesetzt.
- Beispiel für non-greedy routing³¹: Random Walk in ants (Programm). Nicht durchgesetzt.
- Geld auf Probleme werfen: MaidSafe hatte 2000\$ Hardwarekosten pro Monat. 2019 abgeschaltet.³² Freenet hat <20\$ pro Monat Kosten.



Aktuelles

Was gerade jetzt passiert:

- Spritely Golem: p2p distributable content for the fediverse³³
- Decentralized Internet and Privacy at FOSDEM³⁴
 - DAT, GNUnet, Fediverse, Tor, ...
- In Karlsruhe: 21. Gulaschprogrammernacht: <https://entropia.de/GPN21> 8. bis 11. Juni 2023



Greedy-Routing: Anfragen mit rein lokaler Information an den am besten passenden Knoten weiterleiten.

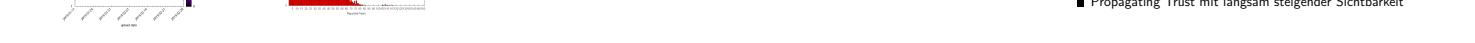
³¹Quelle: https://www.reddit.com/r/safenetwork/comments/erpvee/dumb_question_is_safe_live/



Aktuelles

³³<https://gitlab.com/spritely/golem/blob/master/README.org>

³⁴<https://foosm.org/> — viele Vorträge zu decentralization, privacy, ...



Aktuelles

Was gerade jetzt passiert:

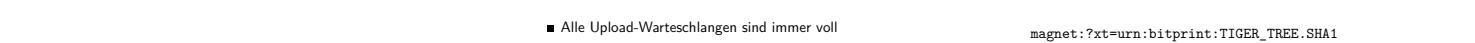
- Spritely Golem: p2p distributable content for the fediverse³³
- Decentralized Internet and Privacy at FOSDEM³⁴
 - DAT, GNUnet, Fediverse, Tor, ...
- In Karlsruhe: 21. Gulaschprogrammernacht: <https://entropia.de/GPN21> 8. bis 11. Juni 2023



Aktuelles

Was gerade jetzt passiert:

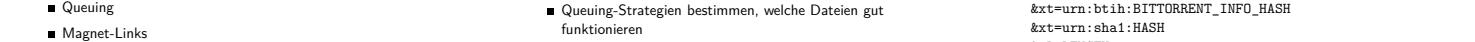
- Spritely Golem: p2p distributable content for the fediverse³³
- Decentralized Internet and Privacy at FOSDEM³⁴
 - DAT, GNUnet, Fediverse, Tor, ...
- In Karlsruhe: 21. Gulaschprogrammernacht: <https://entropia.de/GPN21> 8. bis 11. Juni 2023



Aktuelles

Was gerade jetzt passiert:

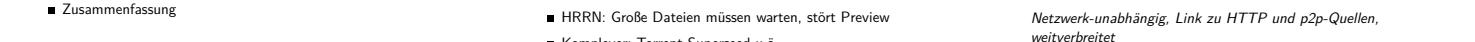
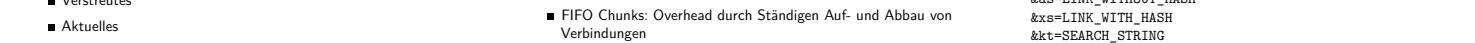
- Spritely Golem: p2p distributable content for the fediverse³³
- Decentralized Internet and Privacy at FOSDEM³⁴
 - DAT, GNUnet, Fediverse, Tor, ...
- In Karlsruhe: 21. Gulaschprogrammernacht: <https://entropia.de/GPN21> 8. bis 11. Juni 2023



Aktuelles

Was gerade jetzt passiert:

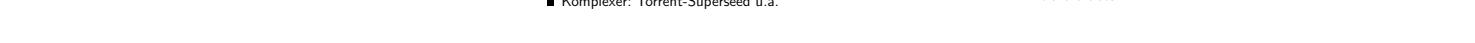
- Spritely Golem: p2p distributable content for the fediverse³³
- Decentralized Internet and Privacy at FOSDEM³⁴
 - DAT, GNUnet, Fediverse, Tor, ...
- In Karlsruhe: 21. Gulaschprogrammernacht: <https://entropia.de/GPN21> 8. bis 11. Juni 2023



Aktuelles

Was gerade jetzt passiert:

- Spritely Golem: p2p distributable content for the fediverse³³
- Decentralized Internet and Privacy at FOSDEM³⁴
 - DAT, GNUnet, Fediverse, Tor, ...
- In Karlsruhe: 21. Gulaschprogrammernacht: <https://entropia.de/GPN21> 8. bis 11. Juni 2023



Aktuelles

Was gerade jetzt passiert:

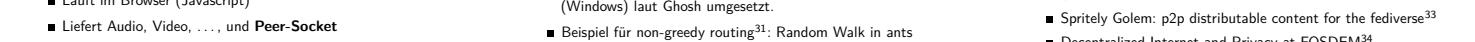
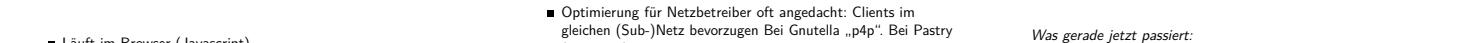
- Spritely Golem: p2p distributable content for the fediverse³³
- Decentralized Internet and Privacy at FOSDEM³⁴
 - DAT, GNUnet, Fediverse, Tor, ...
- In Karlsruhe: 21. Gulaschprogrammernacht: <https://entropia.de/GPN21> 8. bis 11. Juni 2023



Aktuelles

Was gerade jetzt passiert:

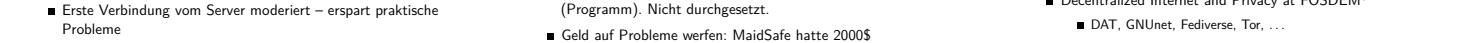
- Spritely Golem: p2p distributable content for the fediverse³³
- Decentralized Internet and Privacy at FOSDEM³⁴
 - DAT, GNUnet, Fediverse, Tor, ...
- In Karlsruhe: 21. Gulaschprogrammernacht: <https://entropia.de/GPN21> 8. bis 11. Juni 2023



Aktuelles

Was gerade jetzt passiert:

- Spritely Golem: p2p distributable content for the fediverse³³
- Decentralized Internet and Privacy at FOSDEM³⁴
 - DAT, GNUnet, Fediverse, Tor, ...
- In Karlsruhe: 21. Gulaschprogrammernacht: <https://entropia.de/GPN21> 8. bis 11. Juni 2023





Zusammenfassung: Grundprobleme

- **Einstieg:** Wie finde ich meinen Platz im Netz?
- **Verbreitung:** Wie vermeide ich Flaschenhälse?
- **Kommunikation:** Wie fließen Informationen durchs Netz?
- **Störungsresistenz:** Wie skaliert Gewünschtes besser als Unerwünschtes?

Arne Babenhausenheide
Verteilte Systeme 1: peer-to-peer
Literatur

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.espacenet.com/patent/search/family/022107098/publication/US4309669A?q=pn%39014309669>
Eingerichtet 1979 als Methode Diffie-Authentication günstiger zu machen.

Arne Babenhausenheide
Verteilte Systeme 1: peer-to-peer

Zusammenfassung: Implementierungen

	Einstieg	Suche
Gnutella	WebCache	Slow-Start + Keyword-Multicast
Kademia	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 unterschiedlich	Heuristik/Credence
Kademia	-	-
BitTorrent	Torrent	Wertung auf Tracker
Freenet	Chunk-Tree with Redundancy	Propagating Trust
WebRTC	-	-

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.



Arne Babenhausenheide
Verteilte Systeme 1: peer-to-peer

Arne Babenhausenheide
Verteilte Systeme 1: peer-to-peer