CHEMNITZER LINUX TAGE 2022

Eine Einführung in P2P Netzwerke?

CONSULTANT

ELMAR DOTT

(M. Schulz) studierte an der HS Merseburg Diplominformatik und twittert regelmäßig über alle möglichen technischen Themen. Seine Schwerpunkte sind hauptsächlich Build und Konfiguration Management, Software Architekturen und Release Management.

Seit über fünfzehn Jahren realisiert er in internationalen Projekten für namhafte Unternehmen umfangreiche Webapplikationen. Er ist freier Consultant / Trainer. Sein Wissen teilt er mit anderen Technikbegeisterten auf Konferenzen, wenn er nicht gerade wieder einmal an einem neuen Fachbeitrag schreibt. https://elmar-dott.com

+ Consultant + Writer + Speaker + Trainer +

AGENDA

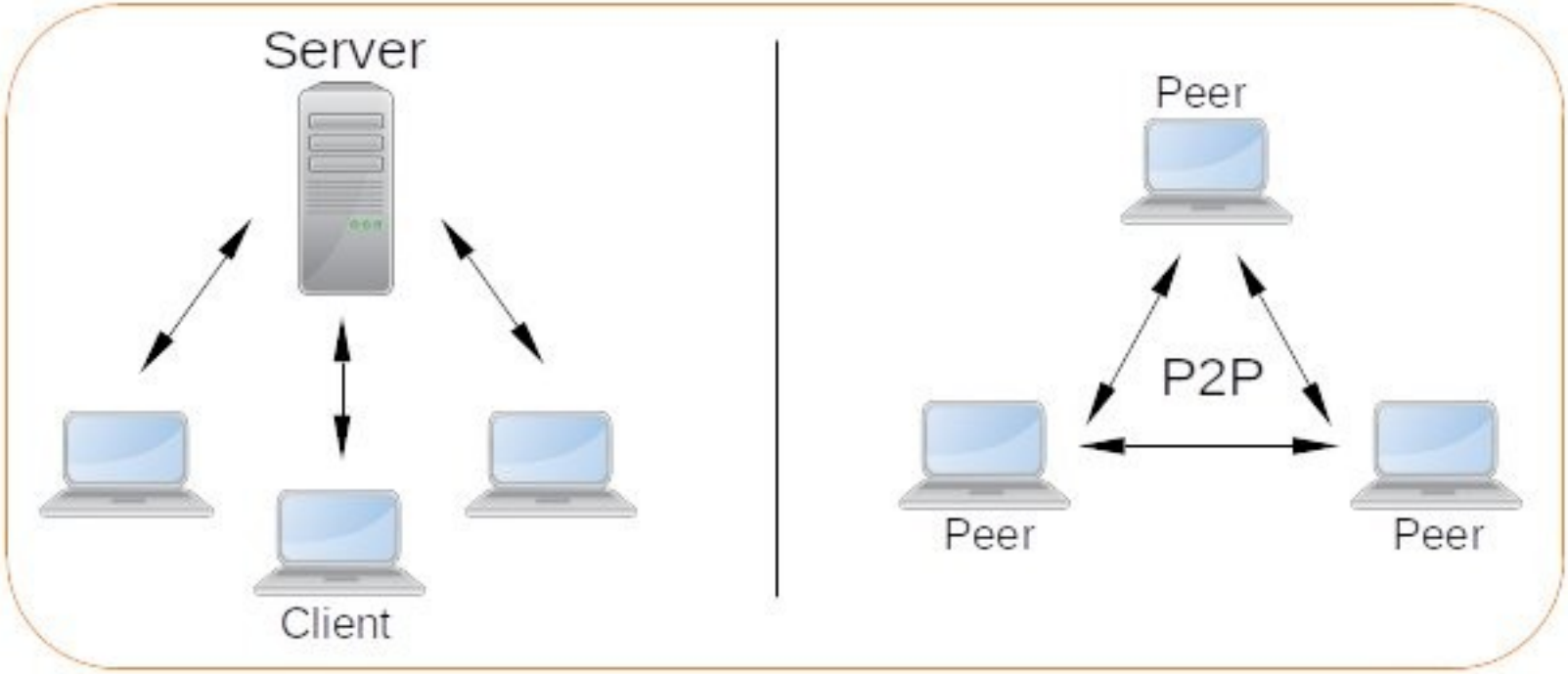
- Einführung in P2P
- Topologie
- Eigenschaften
- GUID
- Routing Tabelle
- Distanzfunktion

EINGENSCHAFTEN

- kommen ohne zentralen Server aus
- P2P sind verteilte, dezentralisierte Netzwerke
- Client = Server => (Node) **Peer**

| Das Netzwerk ist der Dienst!

Client-Server vs P2P



VOR & NACHTEILE

- ☑ günstig zu betreiben
- ☑ zuverlässig
- ☑ gut skalierbar
- ☑ geringe administrative Aufwände
- ☑ transparent

- ☒ Anfällig für Cyberangriff
- ☒ Backup ist schwer umzusetzen
- ☒ illegale Inhalte
- ☒ Performanz

NETZWERK TYPEN

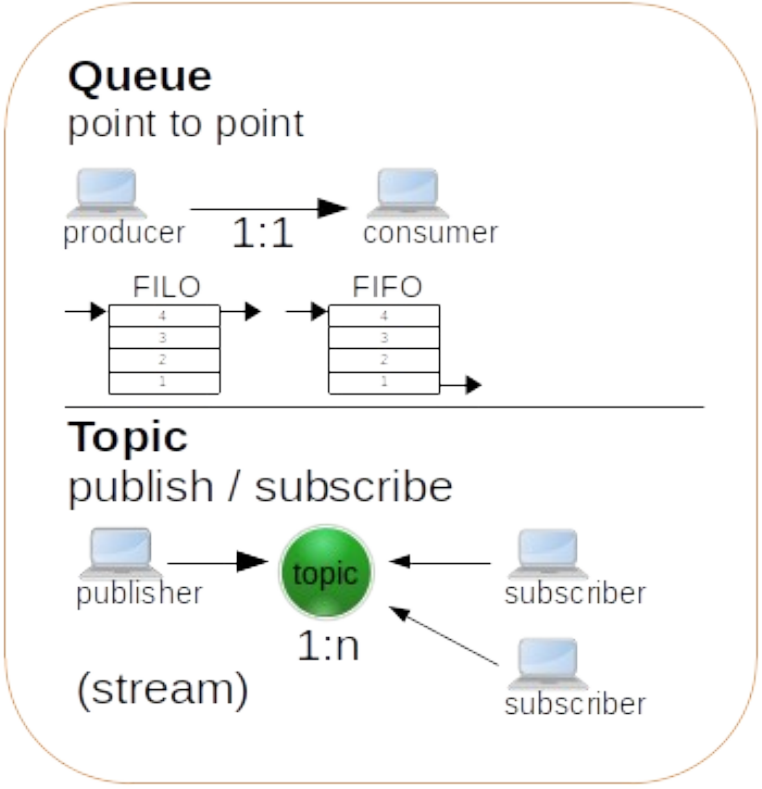
Unstrukturierte Netzwerke sind am leichtesten aufzubauen, können aber unter Umständen sehr Ressourcen fordernd (CPU, RAM, ..) sein.

Strukturierte Netzwerke erlauben es einzelnen Peers sehr effizient nach Objekten zu suchen. Je nach Anwendung haben sie eine spezifische Architektur um Aufgaben zu bewältigen. Sie können meist schlechter mit starken Abwanderungsraten der Peers umgehen.

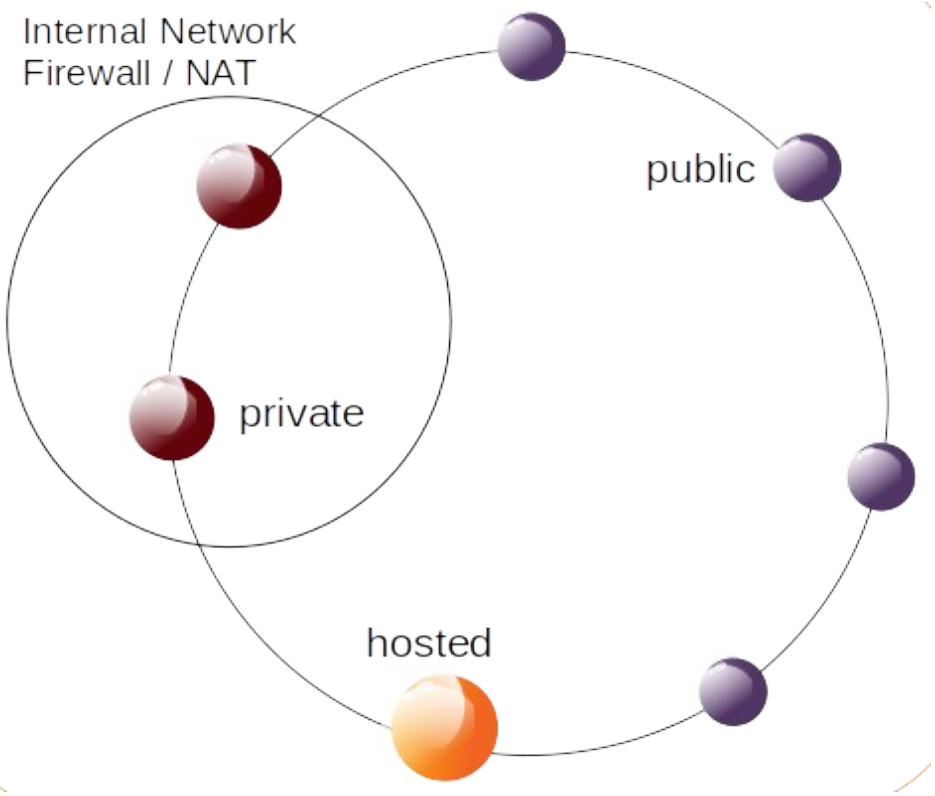
Gemischte Netzwerke vereinen die klassische Client-Server Architektur mit P2P Technologien. Sie zeichnen sich durch eine sehr gute Performance aus.

DIENSTE

- Messaging
- FileSharing (Search)
- Boardcasting
- Payments



TOPOLOGIE



HERAUSFORDERUNGEN

A) **Konnektivität** (Connectivity) – legt fest wie einzelne Teilnehmer sich in das Netzwerk verbinden können. Dazu gehört auch die Möglichkeit über Firewalls oder NAT hinweg zu kommunizieren.

B) **Adressierbarkeit** (Addressability) – stellt sicher das Peers im Netzwerk eineindeutig identifiziert werden können.

C) **Auffindbarkeit** (Routability) – ermittelt den Status eines Teilnehmers im Netzwerk und organisiert wie dieser erreicht werden kann.

Algorithmen:

- Chord
- Kademila
- Tapsty
- Pasty

GENERAL USER ID

GUID identifiziert einen Teilnehmer im Netzwerk eindeutig, denn die IP Adresse eines Peers kann sich ändern.

Die GUID kann eine ganzzahlige Nummer zwischen 1 bis N sein. Dabei stellt die Obergrenze von N die maximale Anzahl von Peers innerhalb eines Netzwerks dar.

Der GUID wird die IP Adresse und der zugehörige Port zugewiesen.

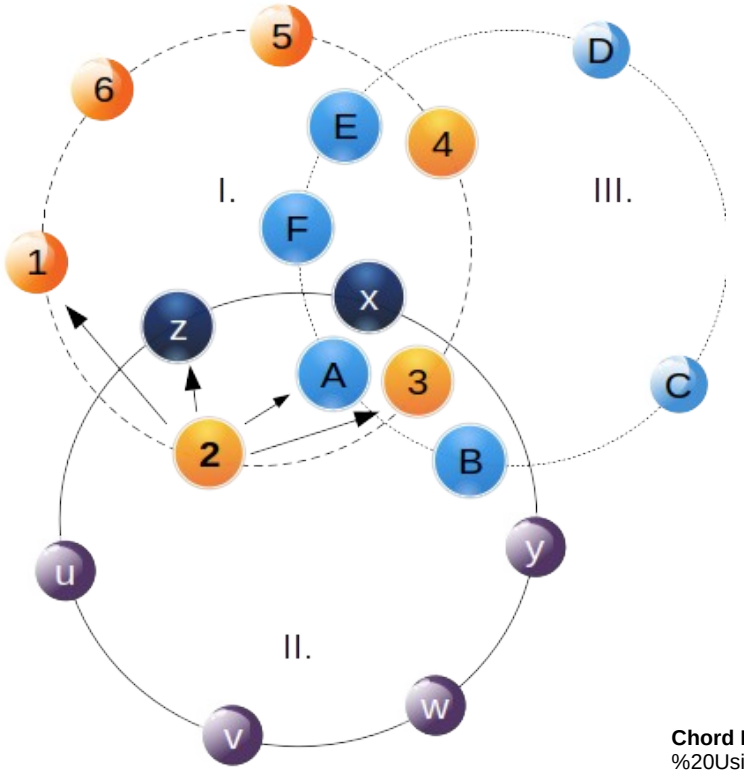
GUID: IP:Port > 0001:127.0.0.1:80

ROUTING TABLE

Ab einer ‚gewissen' Anzahl von Peers im Netzwerk wird es schwierig diese in einer einzigen **Routig Table** (hosts) zu verwalten. Daher bildet man Unternetzwerke, z. B. nach einer Distanzfunktion. Jeder Peer sucht nach den ihm nächstgelegene Peers um sie zu referenzieren.

```
CELL | GUID        | IP          | PORT    | DISTANCE
-----------------------------------------------------------------------------
0001 | 000.000.005 | 127.0.0.1   | 80      | 1
```

DISTANZFUNKTION

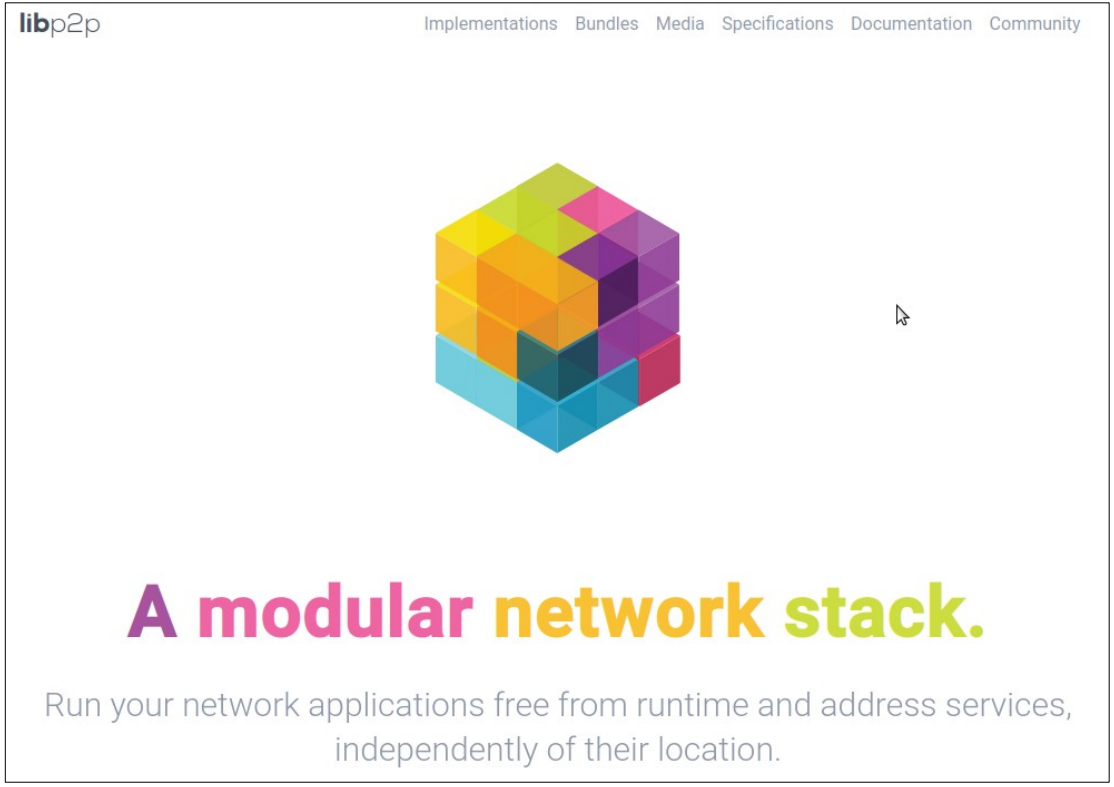


Peer 2 referenziert auf: {1, 3, z & A} weil sie zu ihm die geringste Distanz $D^1 = 1$ haben.

- $D^2 = \{6, 4\}$
- $D^3 = \{5\}$

Chord Distance Function: http://www.dblab.ntua.gr/~gtsat/collection/p2p%20proximity/Indexing%20the%20Distance%20Using%20Chord%20-%20novakzezula05delos.pdf

IMPLEMENTIERUNG



libp2p

- Open Source
- verfügbare Implementierungen:
  JavaScript, Go, Rust
- Java & Python sind angekündigt
- Kademila Algorithmus

Referenzen

[1] https://joinpeertube.org
[2] https://odysee.com
[3] https://developpaper.com/kademlia-protocol/
[4] https://en.wikipedia.org/wiki/Chord_(peer-to-peer)
[5] https://en.wikipedia.org/wiki/Tapestry_(DHT)
[6] https://libp2p.io
[7] https://jenkov.com/tutorials/p2p/index.html

CREDENTIALS


```
Homepage      : https://elmar-dott.com
GitHub        : https://github.com/ElmarDott
AnchorFM      : https://anchor.fm/elmar-dott
Twitter       : https://twitter.com/ElmarDott
Speaker Deck  : https://speakerdeck.com/elmardott

Lbry          : https://lbry.tv/@elmar.dott:8
BitChute      : https://www.bitchute.com/channel/3IyCzKdX8IpO/
```

Danke / thank you / Gracias

open source

ubuntu MATE

Libre Office
The Document Foundation

BITCHUTE

HECHO EN MÉXICO

creative commons