



# Decentralized Storage with IPFS

How does it work under the hood?

**FOSDEM - February 2023**

**Dennis Trautwein**

**Research Engineer**

**Protocol Labs**





Who am I?

# Dennis Trautwein



- Research Engineer **@Protocol Labs**
- Industrial Ph.D. candidate **@University of Göttingen**



[@dennis-tra](#) on **GitHub**

[@dtrautwein\\_eu](#) on **Twitter**

<https://dtrautwein.eu> on the **Web**

[dennis@protocol.ai](mailto:dennis@protocol.ai) via **Email**



# Today's Agenda

- What is IPFS?
- Importing Content
- Connecting to the Network
- Content Routing
- Call Outs



# WHAT IS IPFS?



What is IPFS?

# In Words



stands for the InterPlanetary File System

IPFS is a **decentralized storage and delivery network** which builds on *P2P networking* and *content-based addressing*.

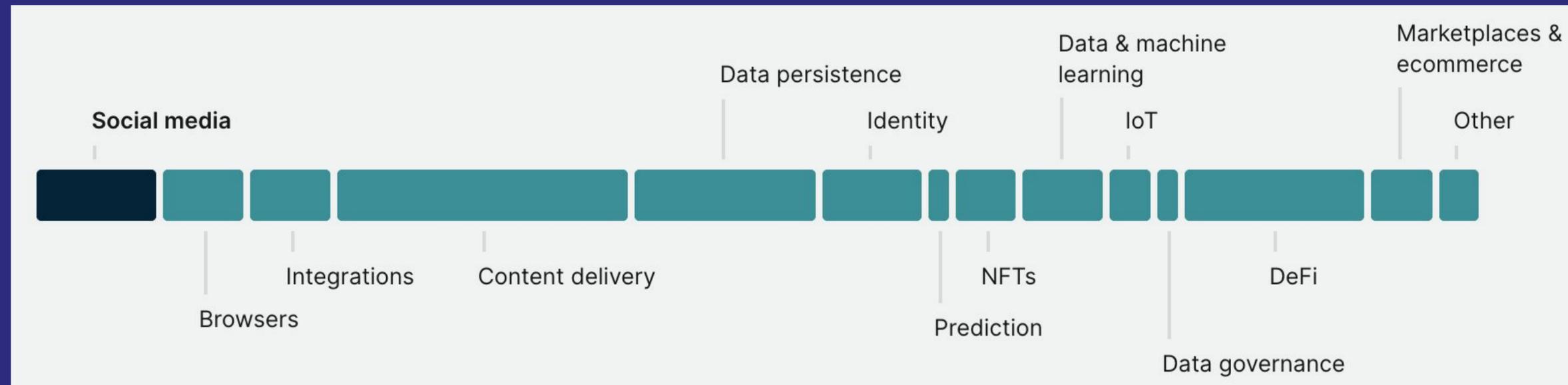
IPFS is **not** a blockchain.



## What is IPFS?

# In Numbers

- Operational since 2015
- # of Requests: >> 1B requests (weekly)
- Volume of Traffic: hundreds of TBs
- Unique weekly users: tens of Ms



*Disclaimer: These are estimates from our vantage points. IPFS is a decentralized network. No one has a full view of the network. Real numbers are likely to be much higher than those.*



What is IPFS?

# Value Proposition

- Decouples content from hosts
- Permanent, verifiable links
- Censorship resistance
- Alleviate backbone addiction
  - Efficient bandwidth use
  - Offline friendly
  - Emerging networks





What is IPFS?

# Value Proposition

## Location Addressing fails on us

- URL points to a single copy
- No way to know where other copies are
- Not possible to validate integrity
  - e.g., DNS poisoning, change copy
- No Request Aggregation
- Emerging Networks
- Offline Use
- Censorship
- Breaking Links



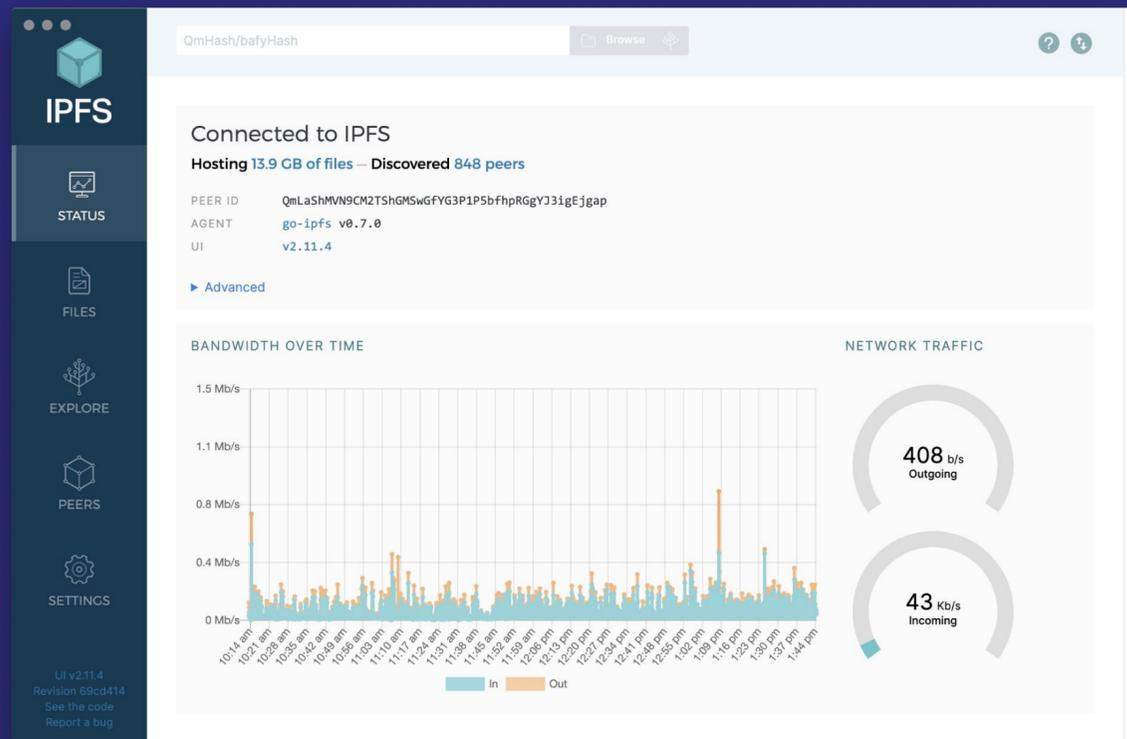
# What is IPFS? Installation

## COMMAND LINE

### IPFS Desktop

### IPFS Companion

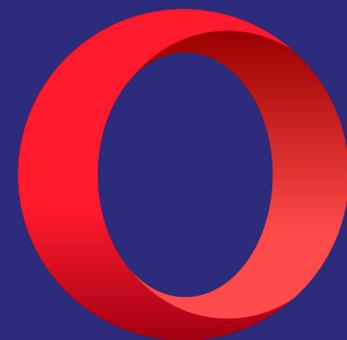
### Brave/Opera



The screenshot shows the IPFS Companion web interface at ipfs.io. The main heading is "IPFS can help here and now" with the subtext "No matter what you do with the web, IPFS helps make it better today." Below this are six use case cards:

- Archivists**: IPFS provides deduplication, high performance, and clustered persistence — empowering you to store the world's information for future generations.
- Service providers**: Providing large amounts of data to users? IPFS offers secure, peer-to-peer content delivery — an approach that could save you millions in bandwidth costs.
- Developing world**: High-latency networks are a big barrier for those with poor internet infrastructure. IPFS
- Blockchains**: With IPFS, you can address large amounts of data and put immutable, permanent links in
- Content creators**: IPFS brings the freedom and independent spirit of the web in full force — and can help

On the right side, there is a settings sidebar for the IPFS 0.7.0 node, including options for "Enable for ipfs.io", "Copy Shareable Link", "Copy Snapshot Link", "Copy IPNS Path", "Copy IPFS Path", "Copy CID", and "Pin IPFS Resource".



# **IMPORTING CONTENT**



# Importing Content

## ipfs init

```
$ ipfs init
generating ED25519 keypair...done
peer identity: 12D3KooWB7JBgNBj3SaNujhY14zfGcxxWphFJPv8YDUkygtX9ETR
initializing IPFS node at /home/<user>/.ipfs
to get started, enter:

  ipfs cat /ipfs/QmQPeNsJPYVWPFDVHb77w8G42Fvo15z4bG2X8D2GhfbSXc/readme
```

- Generates PeerID
- Initializes IPFS Repository
- Just a Local Operation



# Importing Content

## ipfs add FILE



```
$ ipfs add kubo_v0.15.0_linux-amd64.tar.gz  
added QmdVRTMRe9HiWBiFbyHveM4mKpKqvifo9CBEuTSGNuBgKZ kubo_v0.15.0_linux-amd64.tar.gz  
31.85 MiB / 31.85 MiB [=====] 100.00%
```





## Importing Content

# Content Identifier

```
$ ipfs add kubo_v0.15.0_linux-amd64.tar.gz
added QmdVRTMRe9HiWBiFbyHveM4mKpKqvifo9CBEuTSGNuBqKZ kubo_v0.15.0_linux-amd64.tar.gz
31.85 MiB / 31.85 MiB [=====] 100.00%
```

### Content Identifier (CID)

- Most Fundamental Ingredient
- Hash with Metadata
- Self-Describing
- Self-Certifying
- Immutable



Multiformats

`<base>base(<cid-version><multicodec><multihash>)`



# Importing Content CID Inspector

<https://cid.ipfs.tech>

The screenshot shows a web browser window with the URL `cid.ipfs.tech`. The page header features the IPFS logo on the left and "CID INSPECTOR" with a GitHub icon on the right. Below the header, there is a "CID" label and three links: "Docs", "Spec", and "Tutorial". A text input field contains the CID `QmUvSqPqYsjeab2JgsNc4PjbAGnCzfn5xid6piJgYYzehH`. Below the input field, the "HUMAN READABLE CID" section displays `base58btc - cidv0 - dag-pb - (sha2-256 : 256 : 61CE7154D3342FF0924F5F916739B87D2BAC626DB98EF01CF697BAC721B9AECC)` and a smaller line of text: `MULTIBASE - VERSION - MULTICODEC - MULTIHASH (NAME : SIZE : DIGEST IN HEX)`. The "MULTIBASE" section shows `PREFIX: implicit` and `NAME: base58btc`. The "MULTICODEC" section shows `CODE: 0x70`, `NAME: dag-pb`, and `DESCRIPTION:`.



# Importing Content

## Chunking

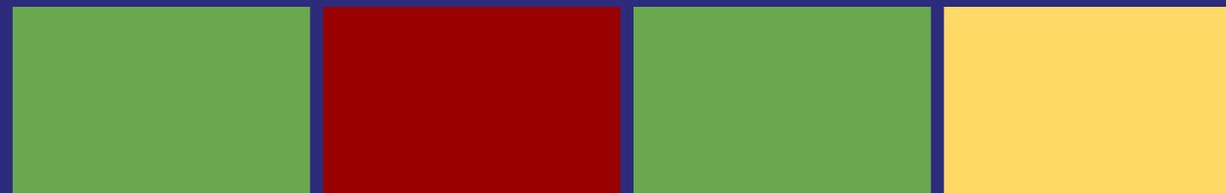


```
$ ipfs add kubo_v0.15.0_linux-amd64.tar.gz  
added QmdVRTMRe9HiWBiFbyHveM4mKpKqvifo9CBEuTSGNuBqKZ kubo_v0.15.0_linux-amd64.tar.gz  
31.85 MiB / 31.85 MiB [=====] 100.00%
```

File



Chunks



Deduplication



- Piecewise Transfer
- Deduplication
- Random Access

(Each Chunk Hashed)



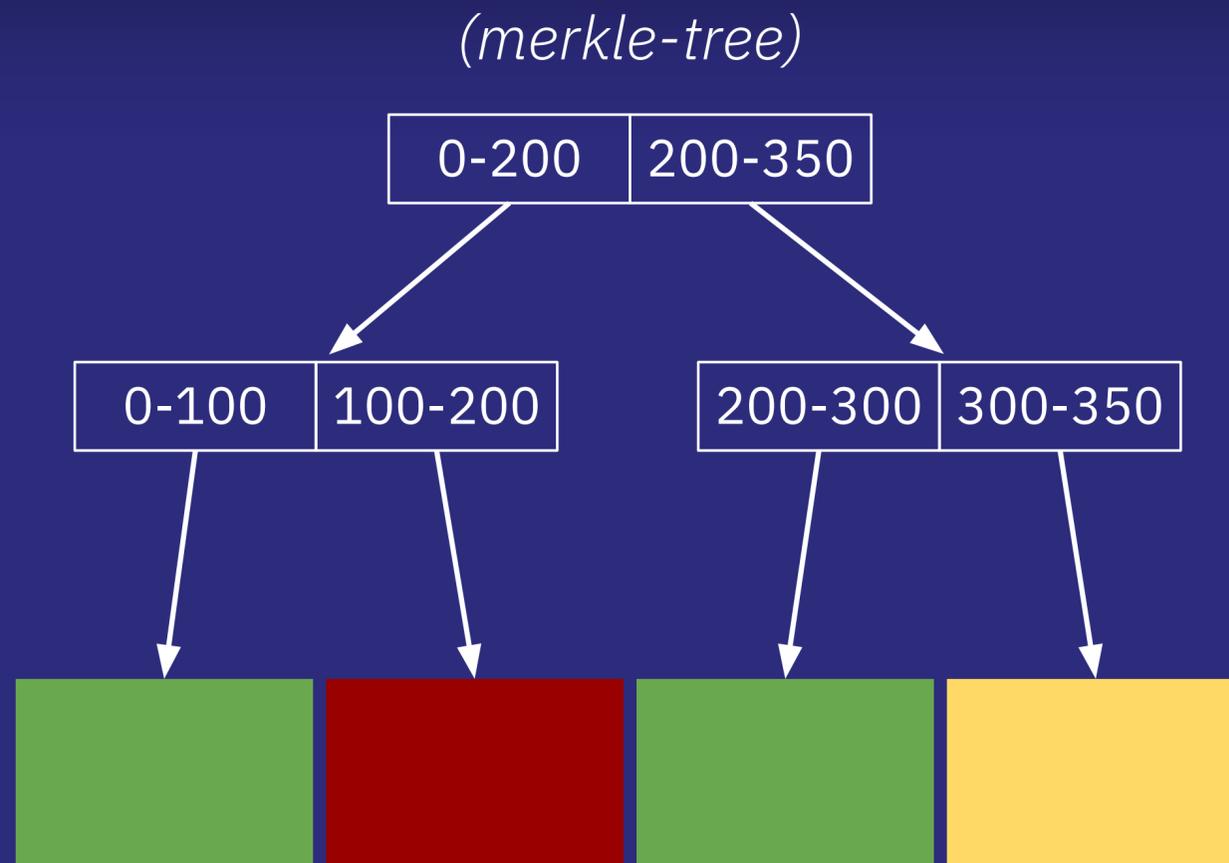
# Importing Content UnixFS



```
$ ipfs add kubo_v0.15.0_linux-amd64.tar.gz  
added QmdVRTMRe9HiWBiFbyHveM4mKpKqvifo9CBEuTSGNuBqKZ kubo_v0.15.0_linux-amd64.tar.gz  
31.85 MiB / 31.85 MiB [=====] 100.00%
```

UnixFS

Chunks





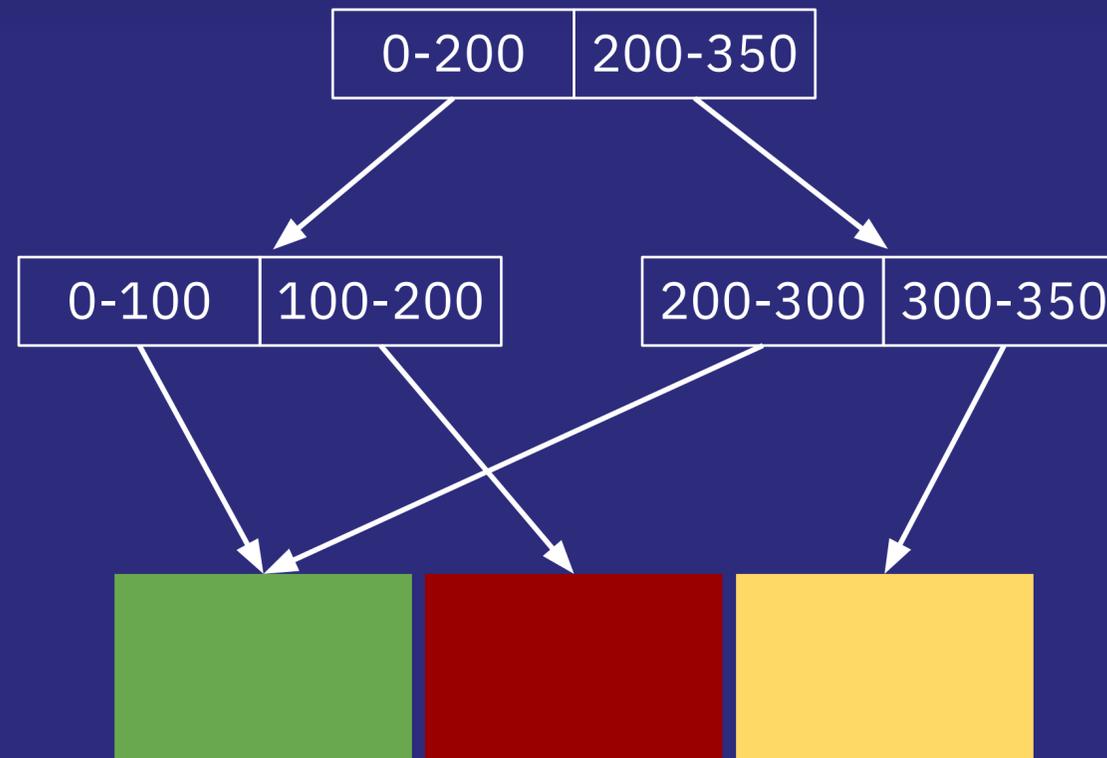
# Importing Content UnixFS



```
$ ipfs add kubo_v0.15.0_linux-amd64.tar.gz  
added QmdVRTMRe9HiWBiFbyHveM4mKpKqvifo9CBEuTSGNuBqKZ kubo_v0.15.0_linux-amd64.tar.gz  
31.85 MiB / 31.85 MiB [=====] 100.00%
```

UnixFS

(merkle-tree-dag)



Chunks



IPLD



# Importing Content IPLD Explorer

<https://explore.ipfs.io>

The screenshot shows the IPLD Explorer interface for a specific IPFS object. The browser address bar shows `explore.ipfs.io`. The interface includes a navigation bar with 'IPFS', 'CID', 'CAR', and a search input field containing 'QmHash'. The main content area displays the following information:

- Object:** dag-pb UnixFS ObjectInfo.publicGateway
- CID:** QmdVRTMRe9HiWBiFbyHveM4mKpKqvi fo9CBEuTSGNuBgKZ
- SIZE:** 32 MB
- LINKS:** 128
- DATA:** `Object {type: "file", data: undefined, blockSizes: Array[128]}`

A table lists the links and their corresponding CIDs:

PATH	CID
0 Links/0	QmPftnxWN2ZSSBZ9za2TM7RKhWKA3G7QT6m9ve2GFkMsMq
1 Links/1	QmeTTityebgpv8NDRq9sxtjab2FFcsuorrAXoPeFksPnD
2 Links/2	Qmekzn6bQNsL6KjD4rYfyH7opJzMvdsVgA3VMiMn76oZG9
3 Links/3	QmWGrzdtVBDPpDgRGBHwtTuJWz9zEeTu3pVB7r8cDwPvYm
4 Links/4	QmVRmajabDMnJFYwqmPv6StD2tNpeH4i3wUXrAkopcXmJd
5 Links/5	QmZ4qCgiwaeJ5Z9K6Bgpmeqt2PYmmRWzP2czewLC1NiCve
6 Links/6	QmVeJ4hmNrZQb1rCiKysyqBrzrexwPgrWqWFrBKHlQUpTe
7 Links/7	QmeeuLZQJdVHsHtui7BFAXw1mK6VAWdp3hu9FhXSTfThaZ
8 Links/8	QmRrYJgjsiAD7yJ2kqXEGMNWQsZAZDxfclN91m1hZdzwLH

On the right side, the 'CID INFO' section shows the base58btc representation: `base58btc - cidv0 - dag-pb - sha2-256~256~E11E8B...`. The 'MULTIHASH' section displays the hex digest: `0x1220E11E8B5E61AF0B3249E41381B96709AE BAF25DCA710A849FE1D6D072794F71E0`, with a legend indicating `0x12 = sha2-256` and `0x20 = 256 bits`. Below this, a red-bordered box contains a triangular graph visualization representing the object's link structure.

The background is a dark blue gradient with a pattern of light blue hexagons. Some hexagons are filled with a lighter blue, while others are just outlines. There are several small, glowing light blue dots scattered across the background.

# **CONNECTING TO THE NETWORK**



# Connecting to the Network

## Daemon

- Long running network-connected IPFS node
- Connects to Bootstrap Peers
- Learns about other Peers
- Announces itself to the Network



```
$ ipfs daemon
Initializing daemon...
Kubo version: 0.15.0
Repo version: 12
System version: amd64/linux
Golang version: go1.18.5
Swarm listening on /ip4/xxx.xxx.xxx.xxx/tcp/4001
Swarm listening on /ip4/xxx.xxx.xxx.xxx/udp/4001/quic
Swarm listening on /ip4/127.0.0.1/tcp/4001
Swarm listening on /ip4/127.0.0.1/udp/4001/quic
Swarm listening on /ip6/xxxx:xxxx:xxxx:xxxx::1/tcp/4001
Swarm listening on /ip6/xxxx:xxxx:xxxx:xxxx::1/udp/4001/quic
Swarm listening on /ip6/::1/tcp/4001
Swarm listening on /ip6/::1/udp/4001/quic
Swarm listening on /p2p-circuit
Swarm announcing /ip4/xxx.xxx.xxx.xxx/tcp/4001
Swarm announcing /ip4/xxx.xxx.xxx.xxx/udp/4001/quic
Swarm announcing /ip4/127.0.0.1/tcp/4001
Swarm announcing /ip4/127.0.0.1/udp/4001/quic
Swarm announcing /ip6/xxxx:xxxx:xxxx:xxxx::1/tcp/4001
Swarm announcing /ip6/xxxx:xxxx:xxxx:xxxx::1/udp/4001/quic
Swarm announcing /ip6/::1/tcp/4001
Swarm announcing /ip6/::1/udp/4001/quic
API server listening on /ip4/127.0.0.1/tcp/5001
WebUI: http://127.0.0.1:5001/webui
Gateway (readonly) server listening on /ip4/127.0.0.1/tcp/8080
Daemon is ready
```



# Connecting to the Network

## Challenge I

- **Problem:** How do we find content hosts for a given CID?
  - *Solution:* Keep a mapping table!

Key	Value
CID_1	PeerID_X
CID_2	PeerID_Y
...	...





# Connecting to the Network

## Challenge II

- **Problem:** The mapping table gets too big!
  - *Solution:* Split and distribute the table to each participating peer

PeerID\_X

Key	Value
CID_1	PeerID_Y
...	...

PeerID\_Y

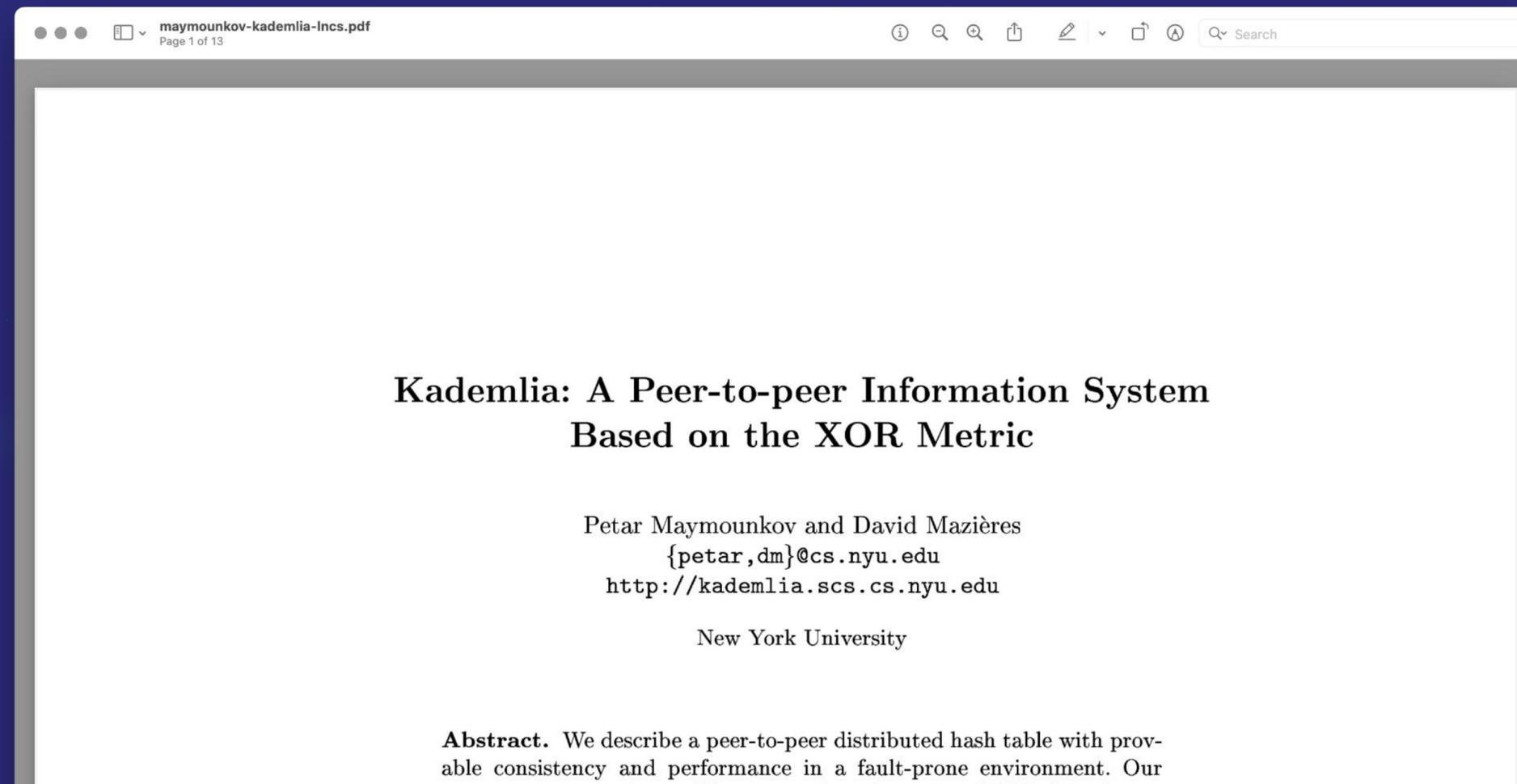
Key	Value
CID_2	PeerID_X
...	...



# Connecting to the Network

## Challenge III

- **Problem:** How do we know who has which piece of that table?
  - *Solution:* Deterministic distribution based on the Kademlia DHT.



# CONTENT ROUTING

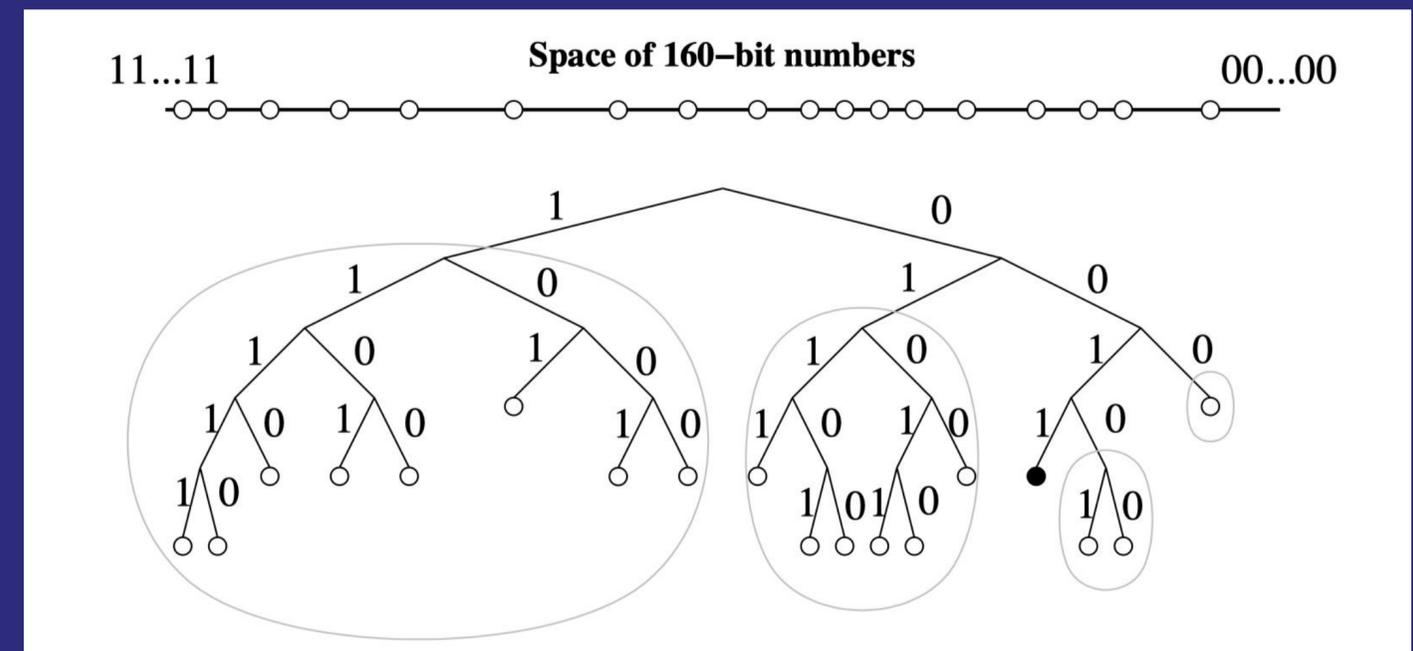


## Connecting to the Network

# Distributed Hash Table

- IPFS uses adaptation of the Kademlia DHT
  - 256 bit key space (SHA256)
- Distributed system that maps keys to values
  - 2-column table
  - **Provider Records:** CID -> PeerID
  - **Peer Records:** PeerID -> Network Addresses
- Two key features:
  - XOR Distance Metric: XOR
    - notion of closeness (not geographically!)
  - Tree-based routing
- $O(\log N)$  lookups

Key	Value
CID_1	PeerID_X
CID_2	PeerID_Y
PeerID_X	Network Addr.
...	...





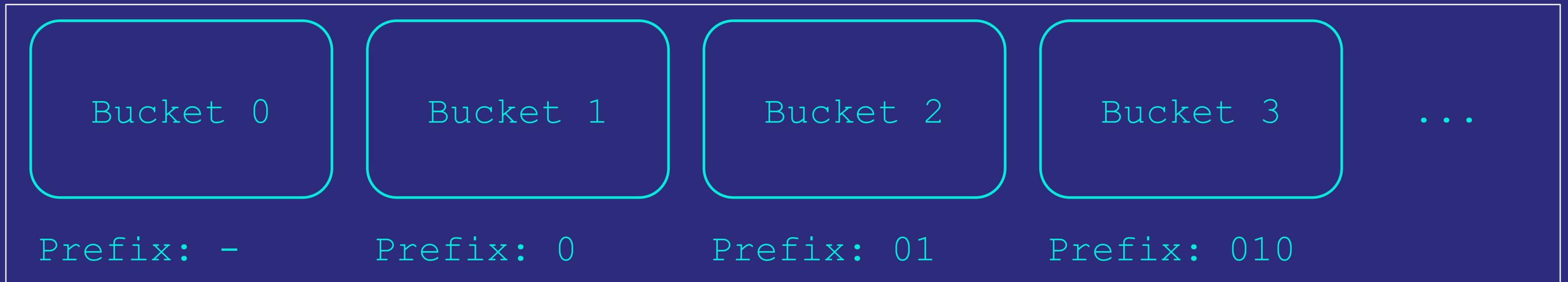
# Connecting to the Network

## Bootstrapping

- Calculate SHA256 of own PeerID
- Ask bootstrap nodes if they know peers whose SHA256(PeerID) start with:
  - 1... (no common prefix)
  - 00... (one common prefix)
  - 011... (two common prefixes)

**010**01110010000010111...

Routing Table





# Connecting to the Network

## Retrieving Content

```
$ ipfs get QmUvSqPqYsjeab2JgsNc4PjbAGnCzfn5xid6piJgYYzehH
Saving file(s) to QmUvSqPqYsjeab2JgsNc4PjbAGnCzfn5xid6piJgYYzehH
2.10 MiB / 2.10 MiB [=====] 100.00% 0s
```

- Calculate SHA256 of CID
- Locate appropriate bucket
- Get list of peers in that bucket
- Start parallel request for that CID
  - If peer know that CID: Returns Provider Record (CID -> PeerID mapping)
  - If peer doesn't know that CID: Returns list of closer peers

01111011110001010111...

Routing Table





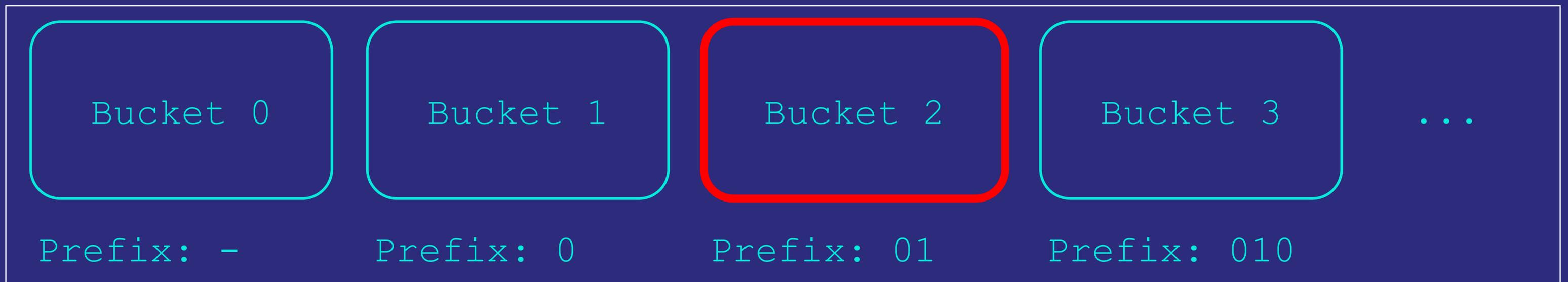
# Connecting to the Network

## Publishing Content

- Calculate SHA256 of CID
- Locate appropriate bucket
- Get list of peers in that bucket
- Start parallel request for closer peer to that CID
- Terminate when the closest known three peers have been successfully queried
- Store Provider Record with the 20 closest peers

**011**11011110001010111...

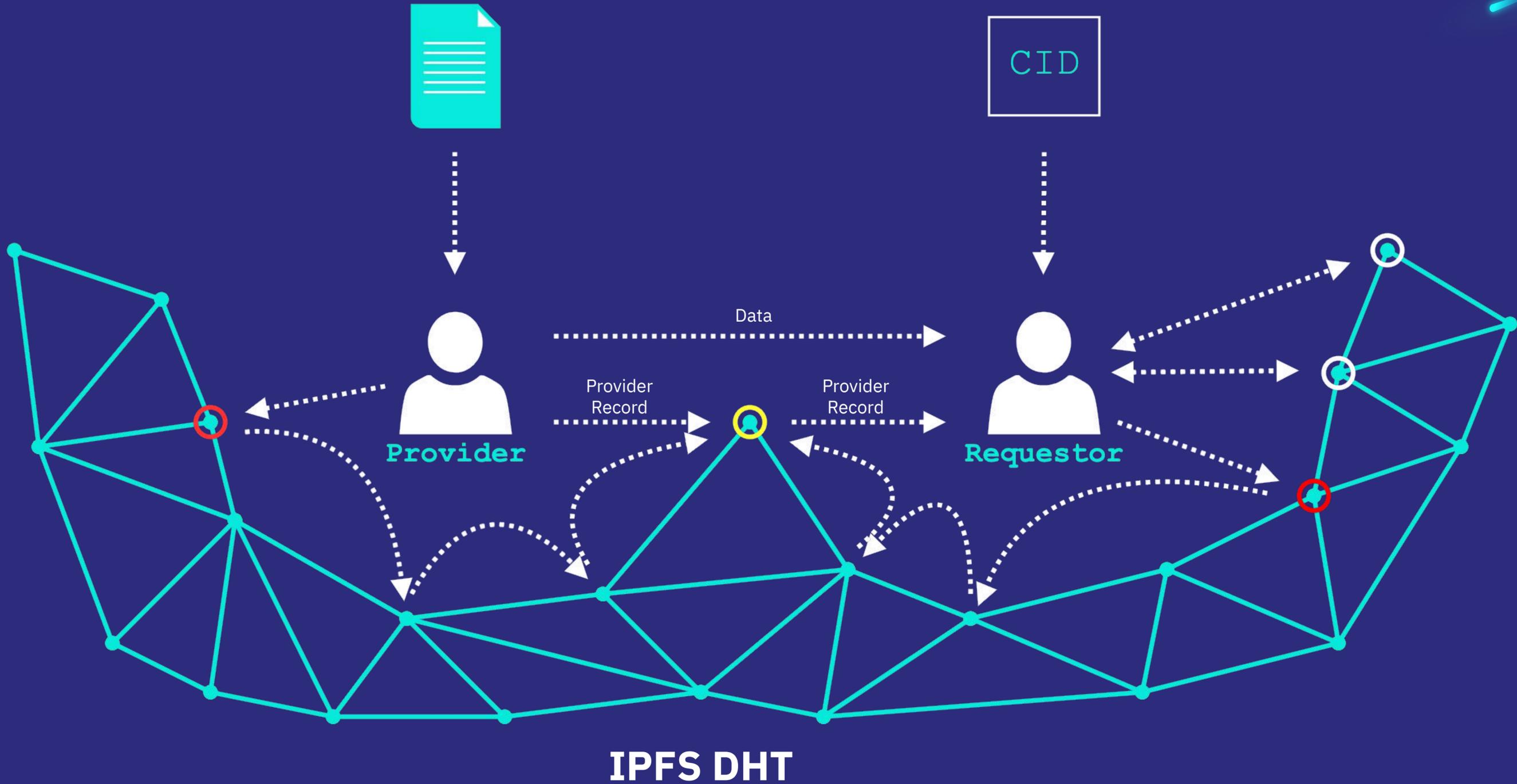
Routing Table

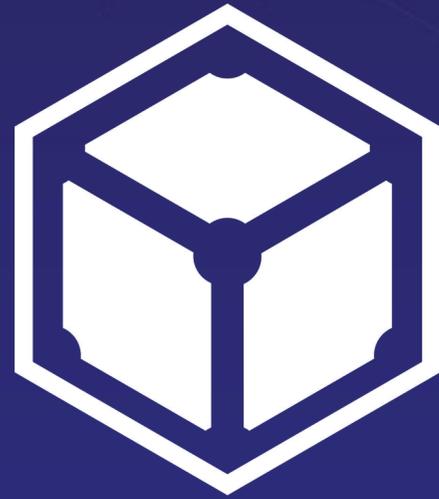




Summary

# Content Lifecycle





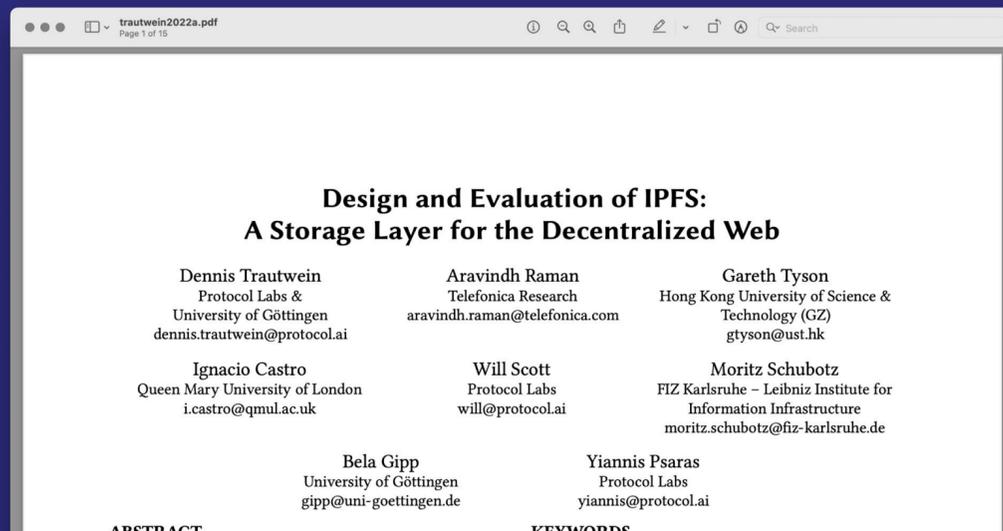
# CALL OUTS

# Future/Open Projects Get Involved!



## ACM SigCOMM '22

QmUvSqPqYsjeab2JgsNc4PjbAGnCzfn5xid6piJgYYzehH



Status: **Open**

Amount: 🌐 500 FIL

Funder: Protocol Labs

Next Deadline: 5/30/2022 - 10:59 PM

Ecosystem: 🌐 IPFS

Discussion Link: [Link](#)

Application Required: Yes

App Deadline: 5/30/2022 - 10:59 PM

Connect to apply!

**Connect Wallet**

### Grant

#### Effectiveness of Bitswap Discovery Process

Bitswap is involved in IPFS's content discovery and precedes the DHT walk. This step adds a delay of 1sec before content is resolved through the DHT. We want to find out whether this delay is worth by investigating Bitswap's success rate.

Author

@yiannisbot

#### Problem Description

Bitswap is involved in the content discovery process and precedes the DHT walk. Nodes ask all of their connected peers for the CID they're interested in, wait for 1sec to receive responses and in case of a negative result resort to the DHT.

- **Funding!**
  - Several grants open at: <https://app.radius.space/>
  - Get your application in!
- **All the action is public!**
  - Check the Network Measurements GH repo: <https://github.com/protocol/network-measurements>
  - More than 15 Requests for Measurements (RFMs)
  - Extra ideas very welcome!



# THANK YOU!



@dennis-tra on **GitHub**

@dtrautwein\_eu on **Twitter**

<https://dtrautwein.eu> on the **Web**

dennis@protocol.ai via **Email**



Mutability

# IPNS

- InterPlanetary Name System (IPNS)
- Content addressing in IPFS is immutable by nature
- E.g., publishing a website requires mutable pointer