

Introduction to IPFS

*Web Verifiability through Content
Addressing*



Guillaume Michel
@guissou

Interplanetary Shipyard



*Privacy x Verifiability
Conference - EPFL
7th March 2025*

Why verifiability matters



- ◆ Trust & Transparency
 - ◆ Security & Accountability
 - ◆ DApps Resilience
-
- ◆ Recent example: ByBit hack



*Could IPFS have prevented
the Bybit hack?*

Agenda



- 1.** What is IPFS?
- 2.** Content Addressing
- 3.** Content Routing
- 4.** IPFS on the Web
- 5.** Common Misconceptions

What is IPFS?



- ◆ IPFS = **I**nter**P**lanetary **F**ile**S**ystem
- ◆ Peer-to-peer hypermedia protocol for content addressing
- ◆ Aims to make the web faster, safer, and more open
- ◆ Does IPFS have a blockchain? No.
 - ▶ <https://doesipfshaveablockchain.com>



Location Addressing vs Content Addressing



Location Addressing	Content Addressing
<i>Where is the data</i>	<i>What is the data?</i>
Uniform Resource Locator (URL) <code>https://example.com/foo/bar.jpg</code>	Content Identifier (CID) <code>bafybeihfg3d7rdltd43u3tfvn...uc7y</code>
Server down? The data is unreachable	Anyone can serve the same data
Limited verifiability — trust the server to provide correct data	Full verifiability — if content changes, CID changes
Censor-prone — block IP or domain name	Hard to censor — all providers must be blocked

Content Identifiers (CIDs)



- ◆ Immutable, self certifying name
- ◆ Made of hash and meta data

CIDV1 (BASE32)

bafybeihfq3d7rdltd43u3tfvncx7n5loqofbsobojcadtmokrljfthuc7y

HUMAN READABLE CID

base32 - cidv1 - dag-pb - (sha2-256 : 256 : E536C7F88D731F374DCCB568AFF6F56E838A19382E488039B1CA8AD2599E82FE)

MULTIBASE - VERSION - MULTICODEC - MULTIHASH (NAME : SIZE : DIGEST IN HEX)

source: cid.ipfs.tech

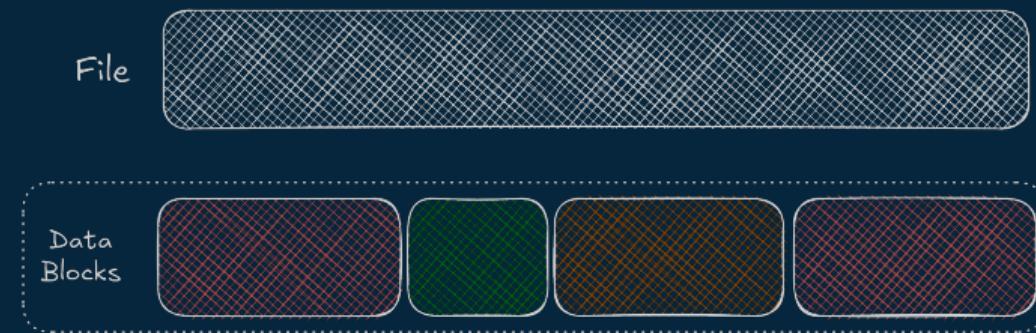
Merkle DAG



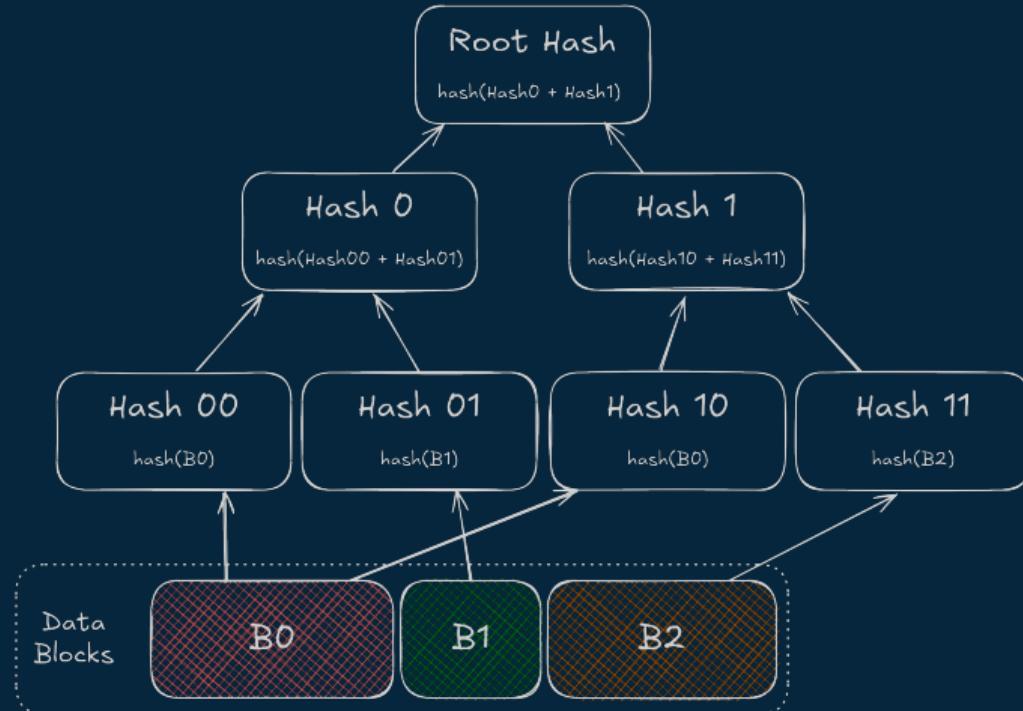
File



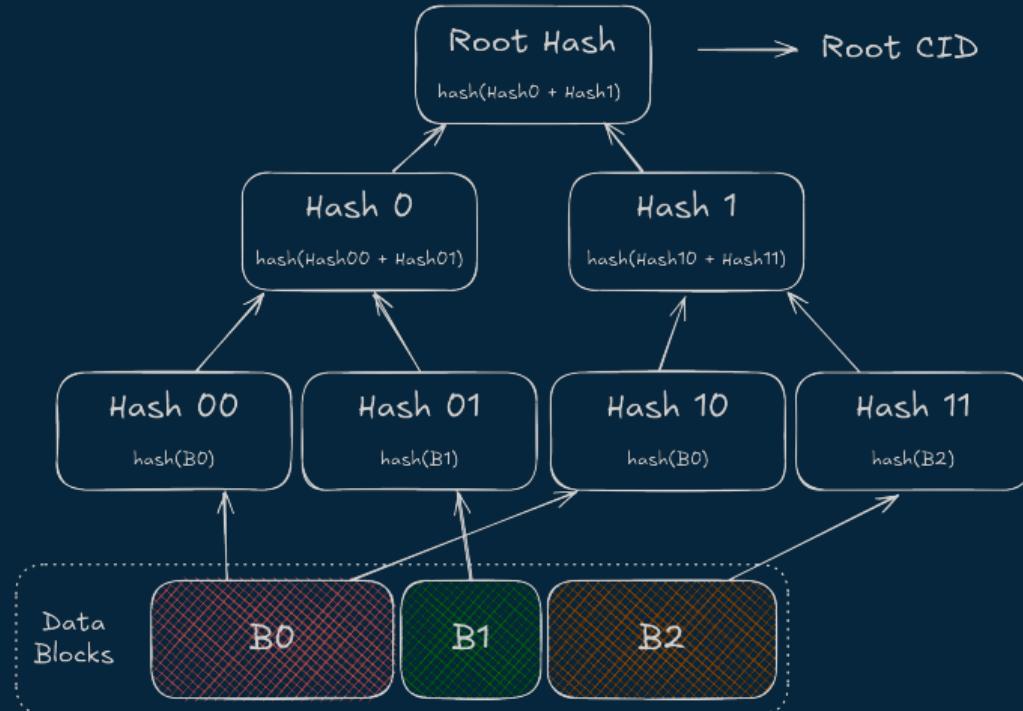
Merkle DAG



Merkle DAG



Merkle DAG



libp2p



- ◆ Open source peer-to-peer networking library
- ◆ Peer ID: unique peer identifier derived from public key (12D3KooWN1Mr...)
- ◆ Transports: tcp, quic, wss, webrtc, webtransport
- ◆ Encrypted by default
- ◆ Features: Holepunching, Relays, Kademlia DHT, Gossipsub, etc.
- ◆ Networking layer of IPFS, Ethereum, Polkadot, Filecoin, etc.



Content Routing



Alice

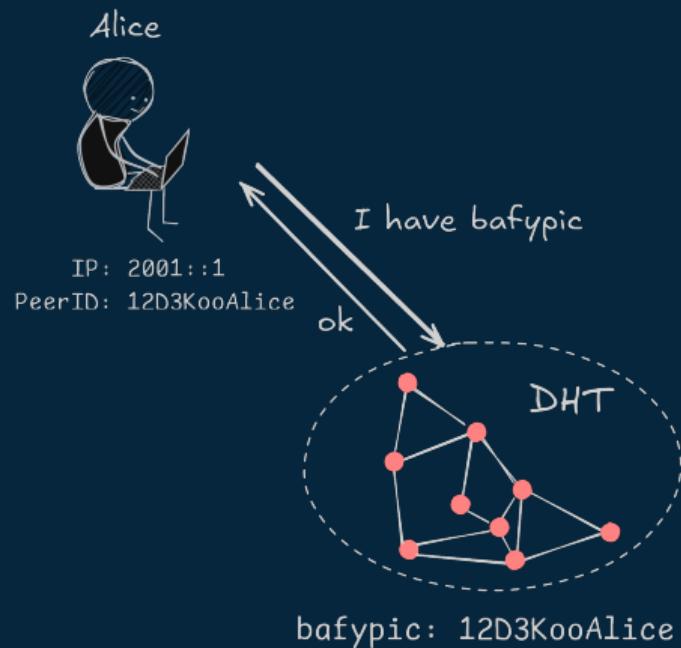


```
$ ipfs add  bafypic
```

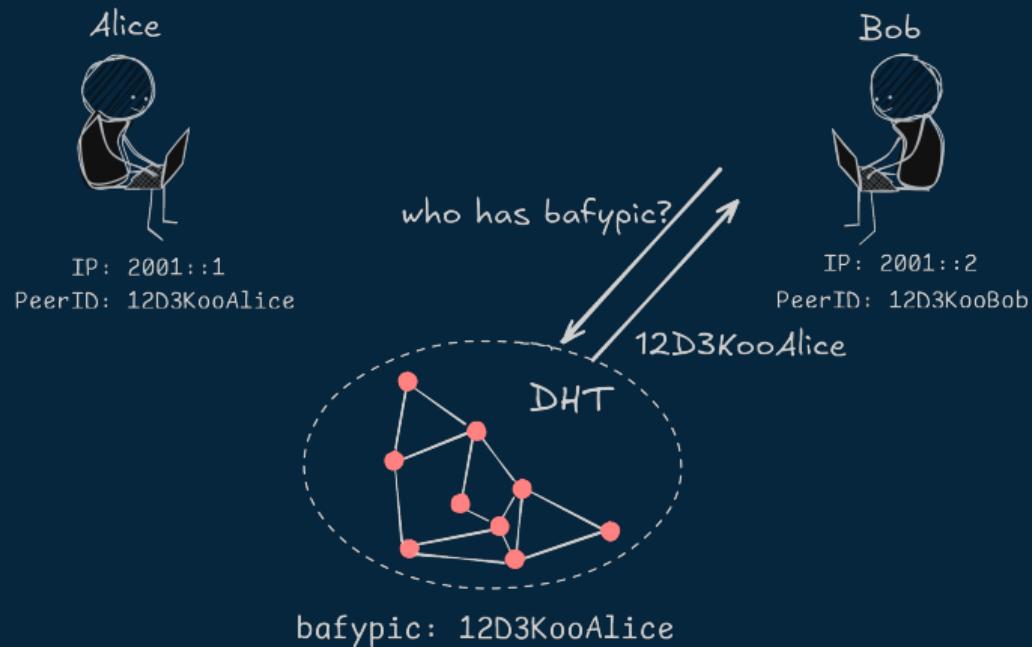
IP: 2001::1

PeerID: 12D3KooAlice

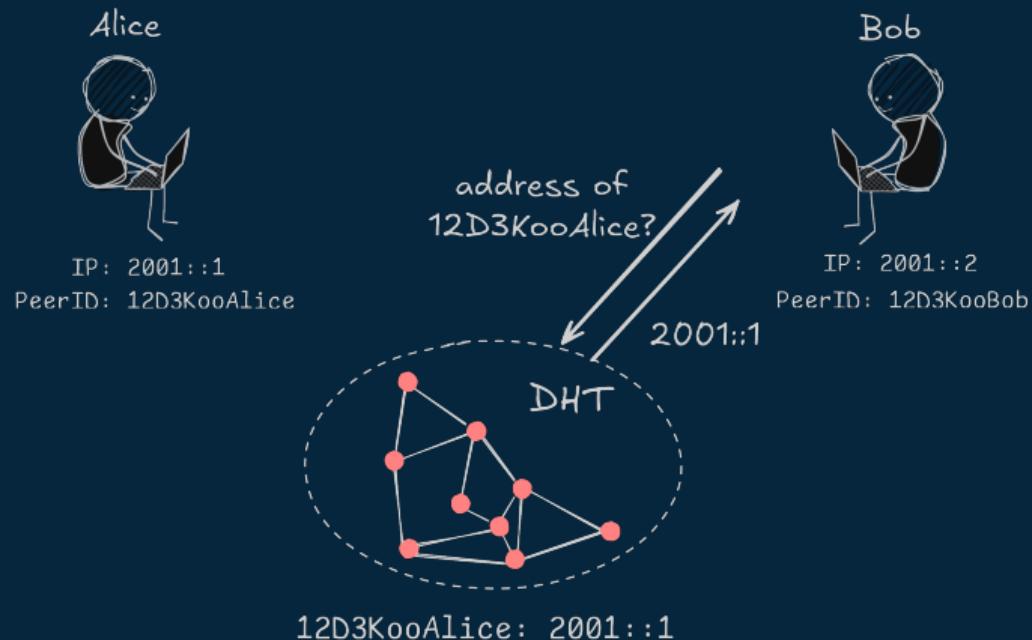
Content Routing



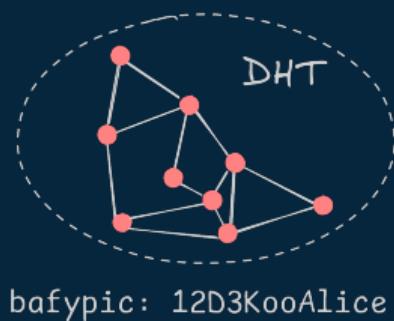
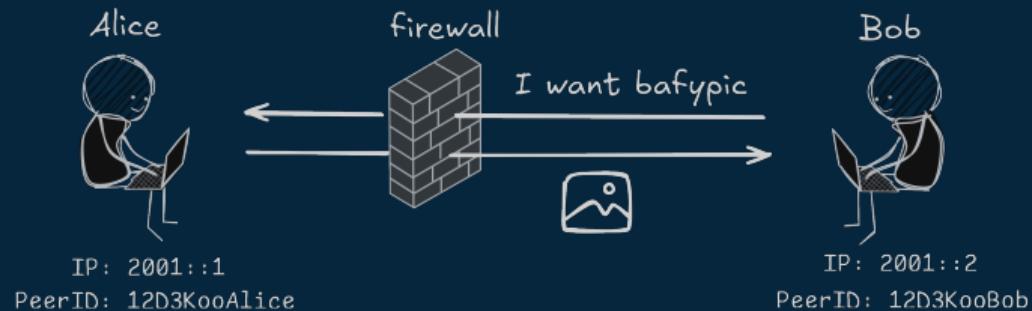
Content Routing



Content Routing



Content Routing





Content Routing

Alice



IP: 2001::1

PeerID: 12D3KooAlice

verify == bafypic?

Bob



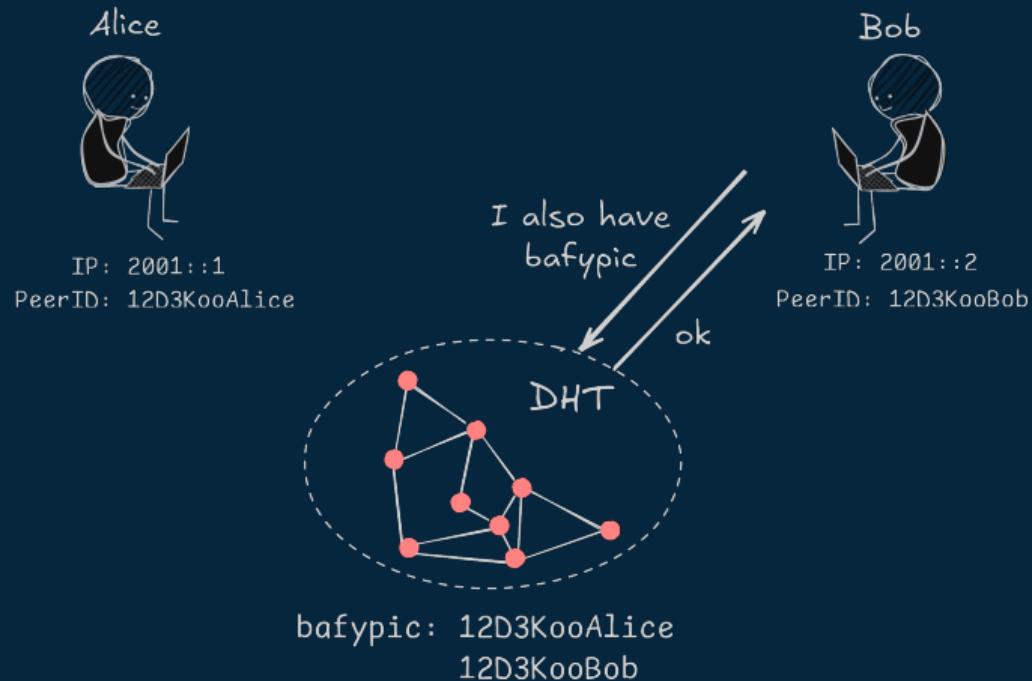
IP: 2001::2

PeerID: 12D3KooBob

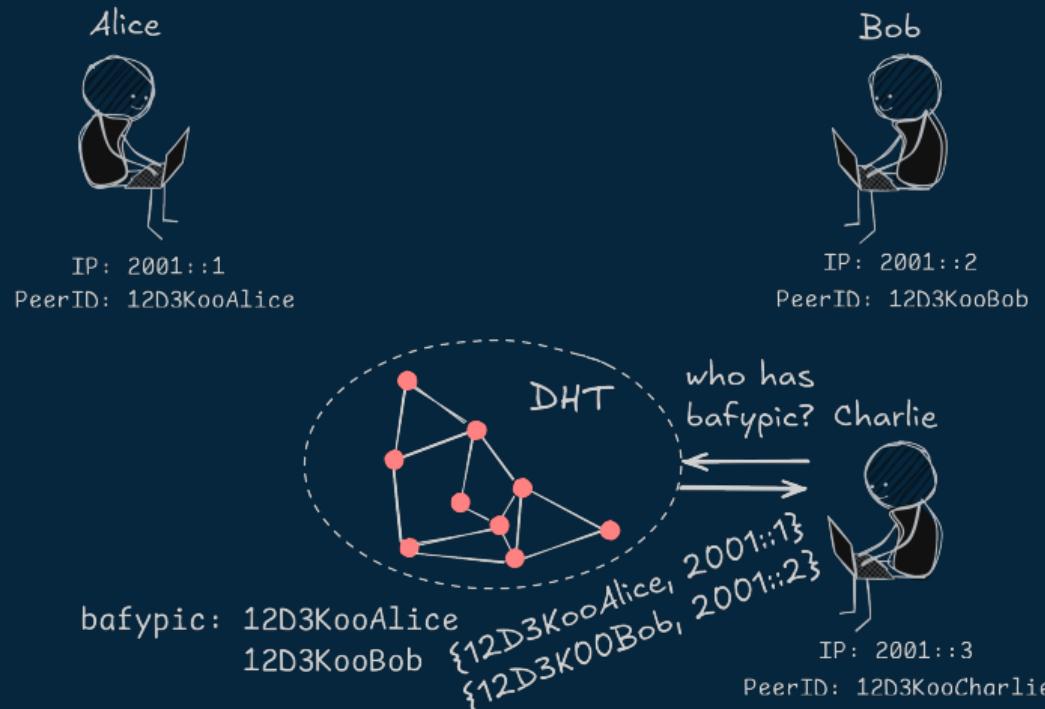


bafypic: 12D3KooAlice

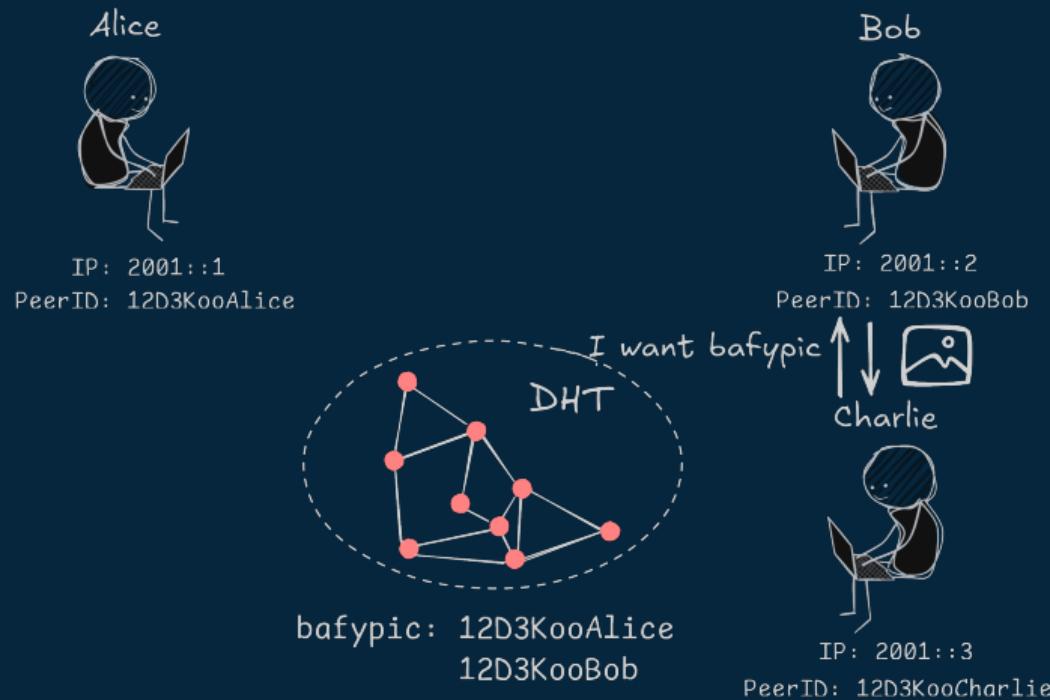
Content Routing



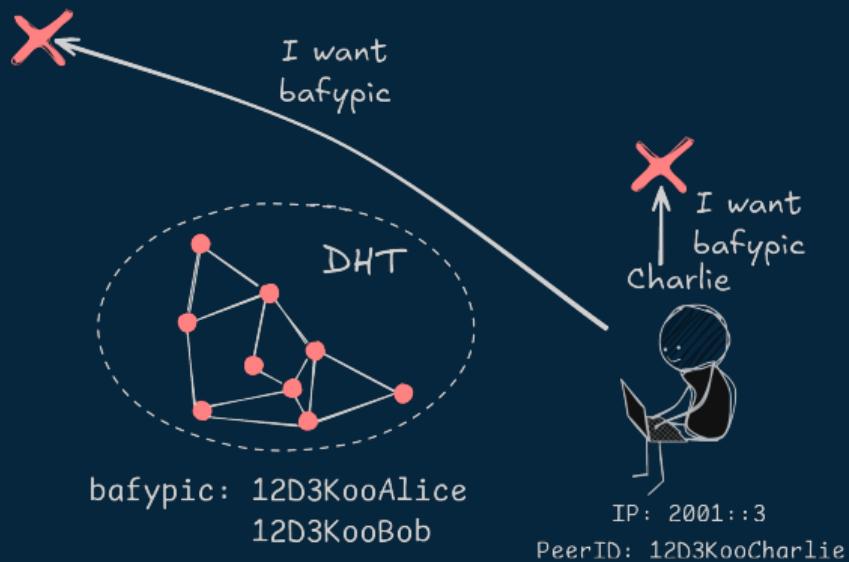
Content Routing



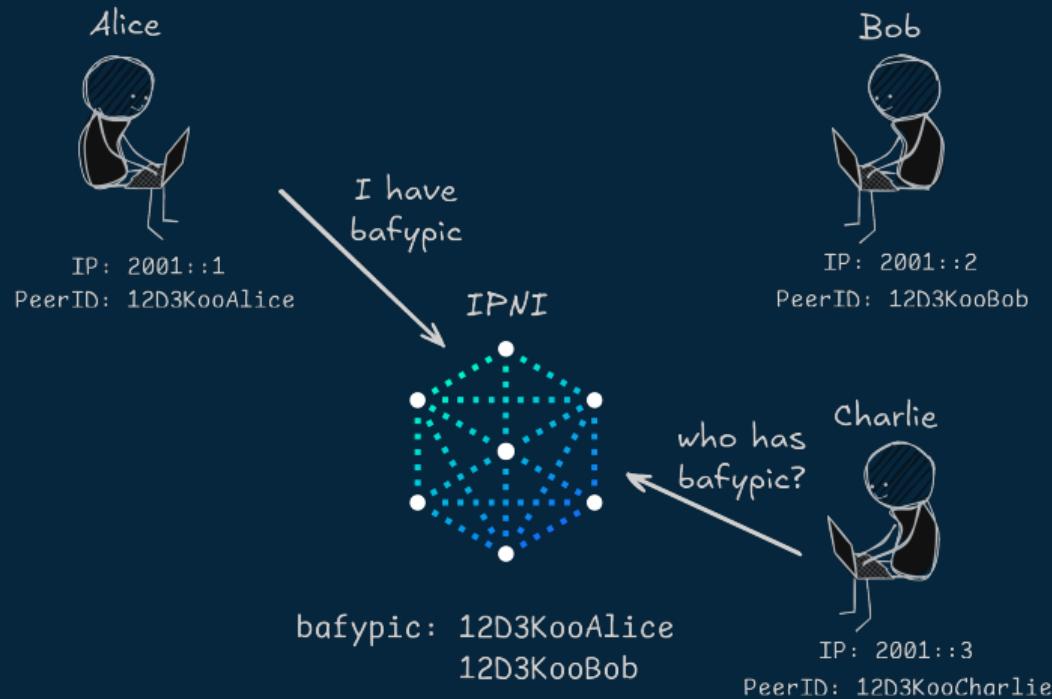
Content Routing



Content Routing



Content Routing

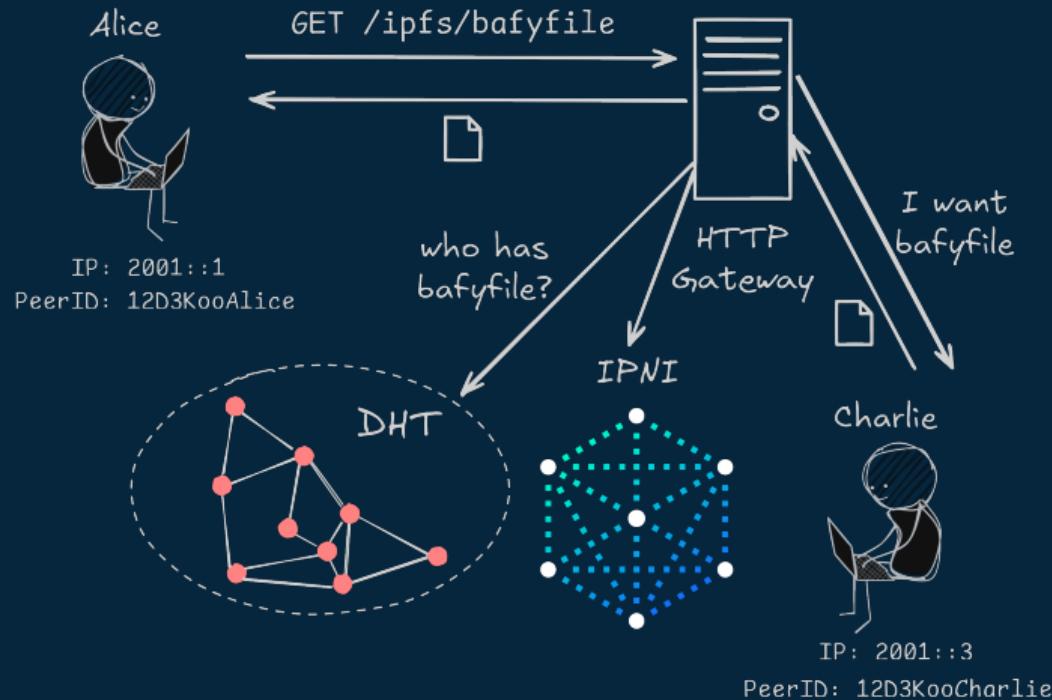


Bridging the Web: IPFS HTTP Gateways



- ◆ Browsers have limited connectivity:
 - ◆ wss
 - ◆ webrtc
 - ◆ webtransport (WIP)
- ◆ HTTP can be used for data retrieval!

Bridging the Web: IPFS HTTP Gateways



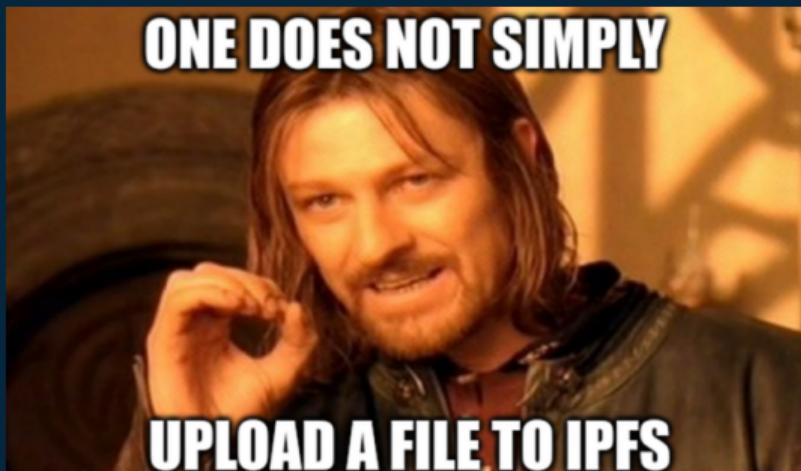
Bridging the Web: IPFS HTTP Gateways



- ◆ HTTP Gateways (ipfs.io, dweb.link)
 - ◆ GET /ipfs/{cid}
 - ◆ Gateway must be trusted
- ◆ Trustless gateways (inbrowser.dev)
 - ◆ Data verified in Service Worker
 - ◆ Uses Helia verified fetch
- ◆ Next Steps
 - ◆ Delegated Routing
 - ◆ Fetch from source

Common Misconceptions

1. Uploading files to IPFS
2. Permanent data storage
3. Data replication
4. IPFS fully replaces HTTP
5. Privacy



Shipyard



- ◆ Engineering collective stewarding IPFS & libp2p
 - ◆ We ❤️ Open Source
 - ◆ Graduated from Protocol Labs in 2024
- ◆ Championing IPFS & libp2p specs
- ◆ Developping & maintaining kubo, helia, go-libp2p, js-libp2p and associated tooling
- ◆ Operating IPFS Gateways (ipfs.io) and DHT Bootstrappers
- ◆ Open for collaborations and supporting your IPFS use cases
 - ◆ contact@ipshipyard.com



<https://ipshipyard.com>



Q&A



Guillaume Michel (@guissou)

- ◆ Github: **@guillaumemichel**
- ◆ Email: guillaume@ipshipyard.com
- ◆ Website: <https://ipshipyard.com>