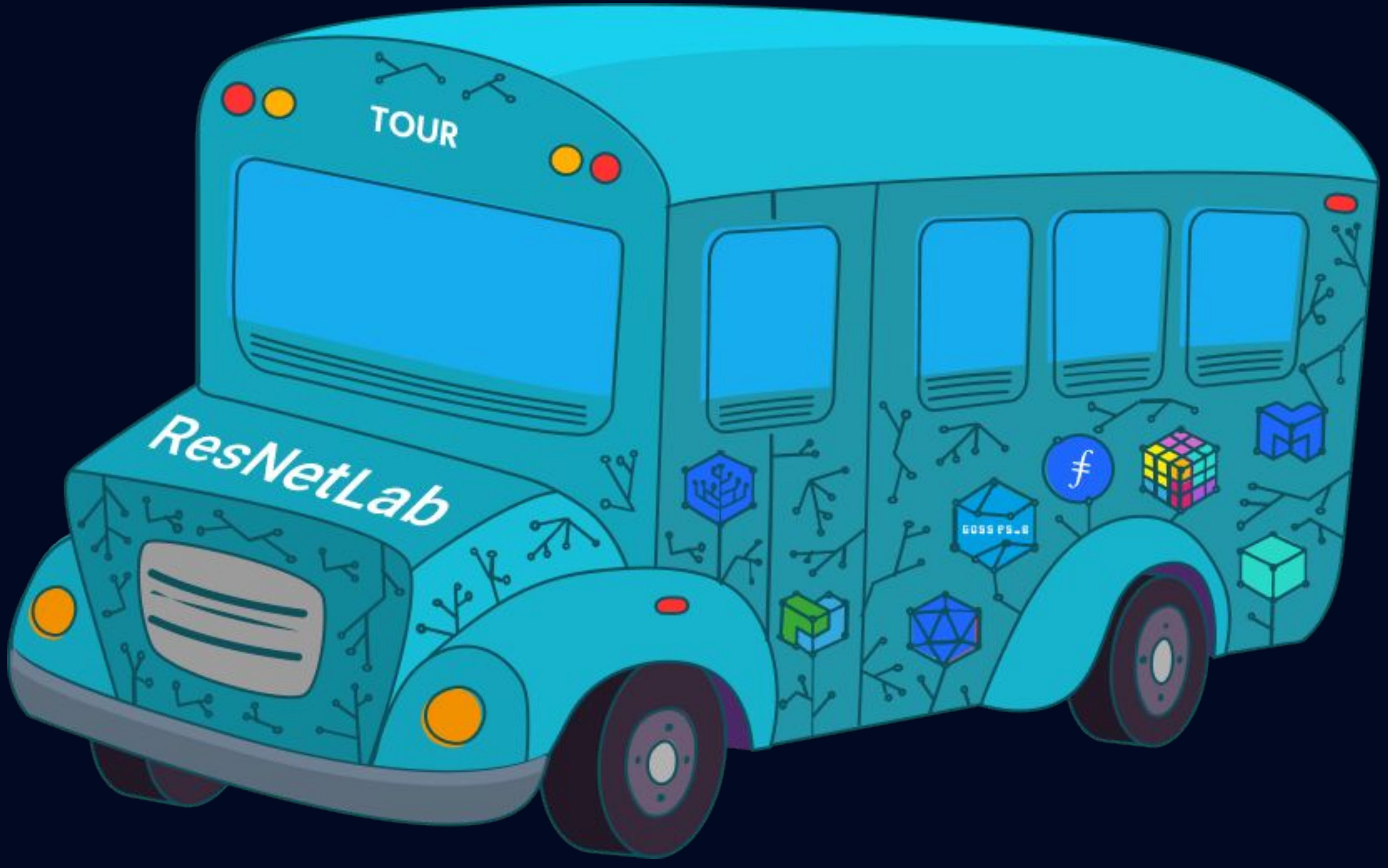
ResNetLab

The Interplanetary File System
A fresh look at distributed storage and delivery

An introduction to IPFS

Dr Yiannis Psaras
Research Scientist
Protocol Labs



ResNetLab on Tour

You can find a series of video tutorials on IPFS, libp2p and Filecoin at:
https://research.protocol.ai/tutorials/resnetlab-on-tour/

Who am I

Now: Research Scientist @ Protocol Labs

Before: Senior Lecturer @ University College London (UCL)

Interests: Networks, Security, Internet Architecture, Decentralised Internet Services, Content Addressable Networks, Edge Computing

Protocol Labs

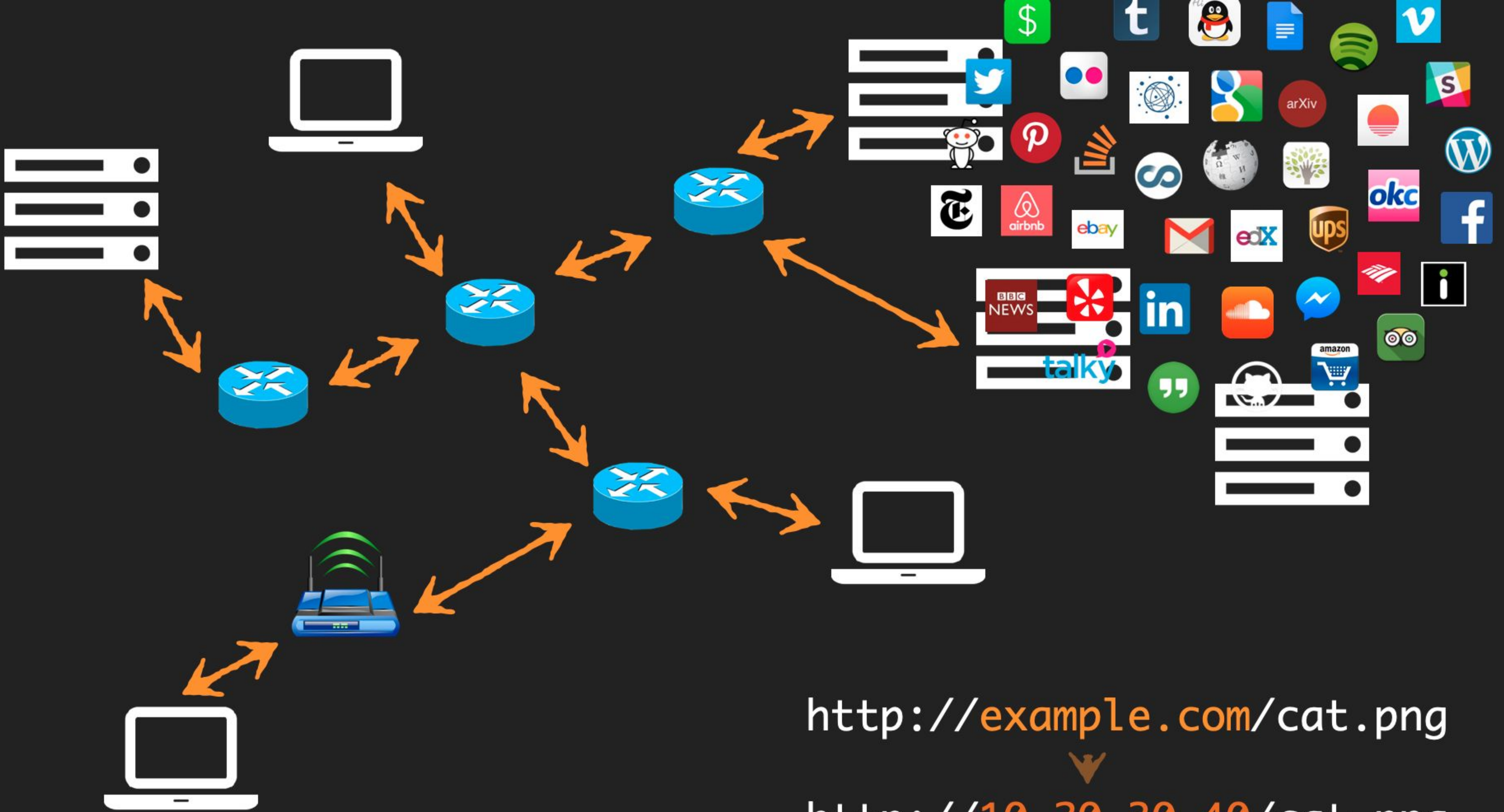
ResNetLab

Agenda

- Web 3.0 & the Decentralized Cloud
- Content Addressing
- Content Routing
- Context Exchange

in IPFS

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

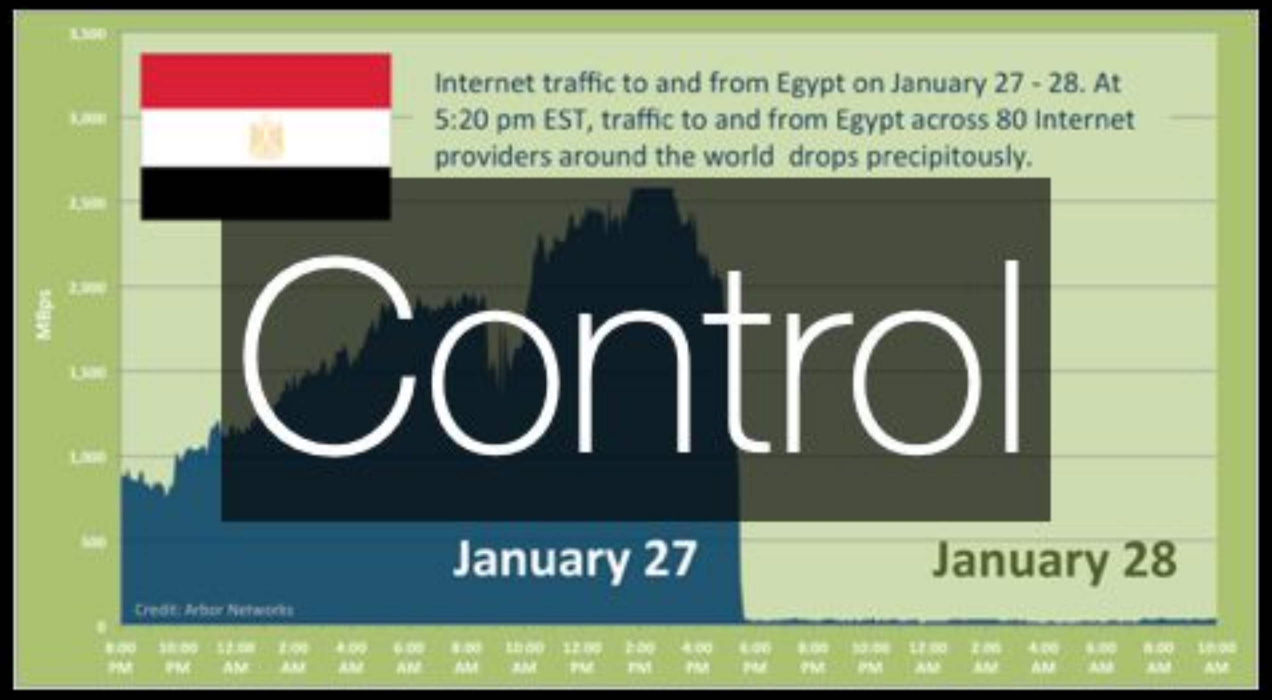


http://example.com/cat.png

⬇

http://10.20.30.40/cat.png

Disconnected


Control

Internet traffic to and from Egypt on January 27 - 28. At 5:20 pm EST, traffic to and from Egypt across 80 Internet providers around the world drops precipitously.

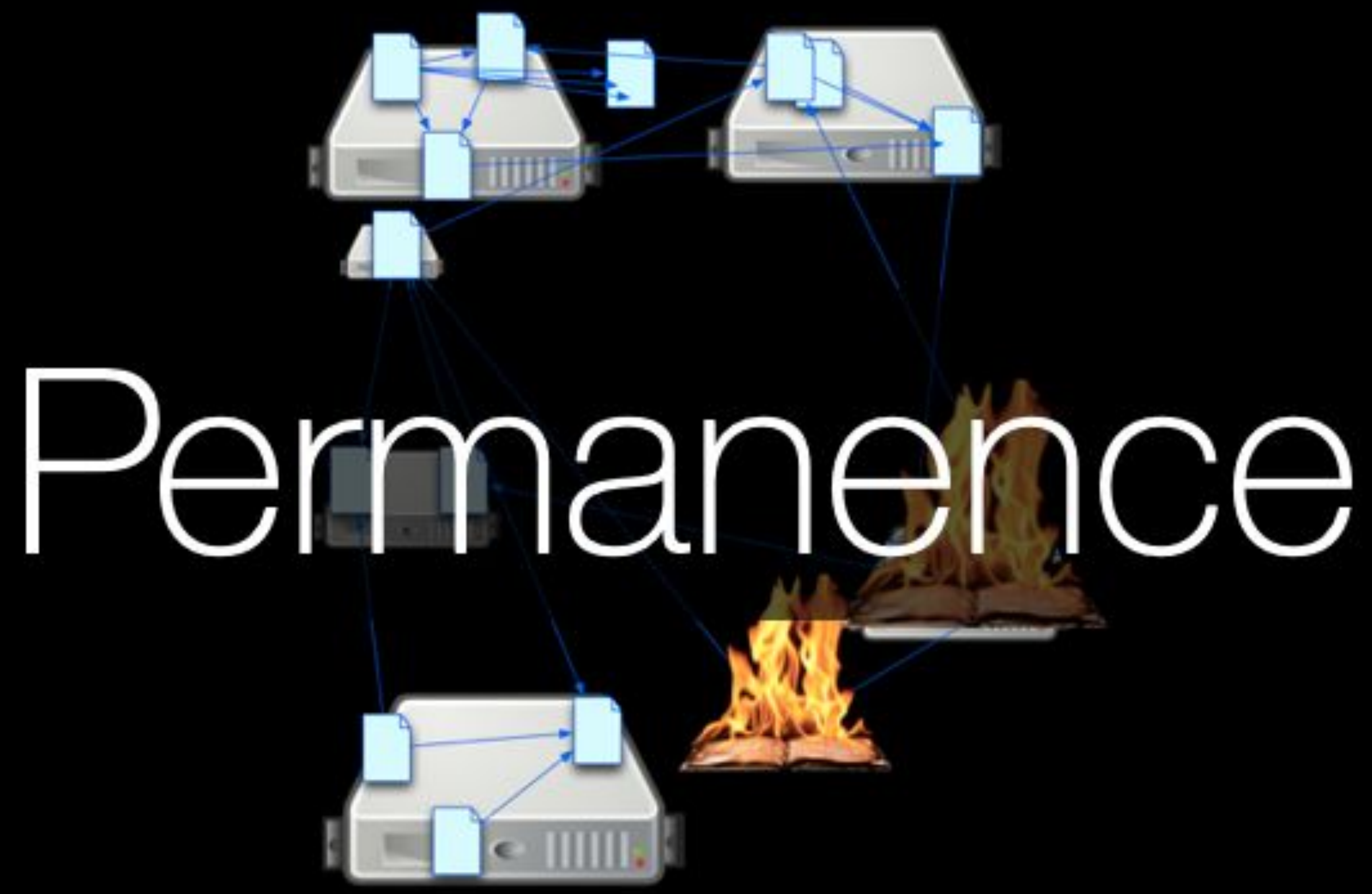
Credit: Arbor Networks

200 MB x 30 x 8 = 48 GB

Bandwidth

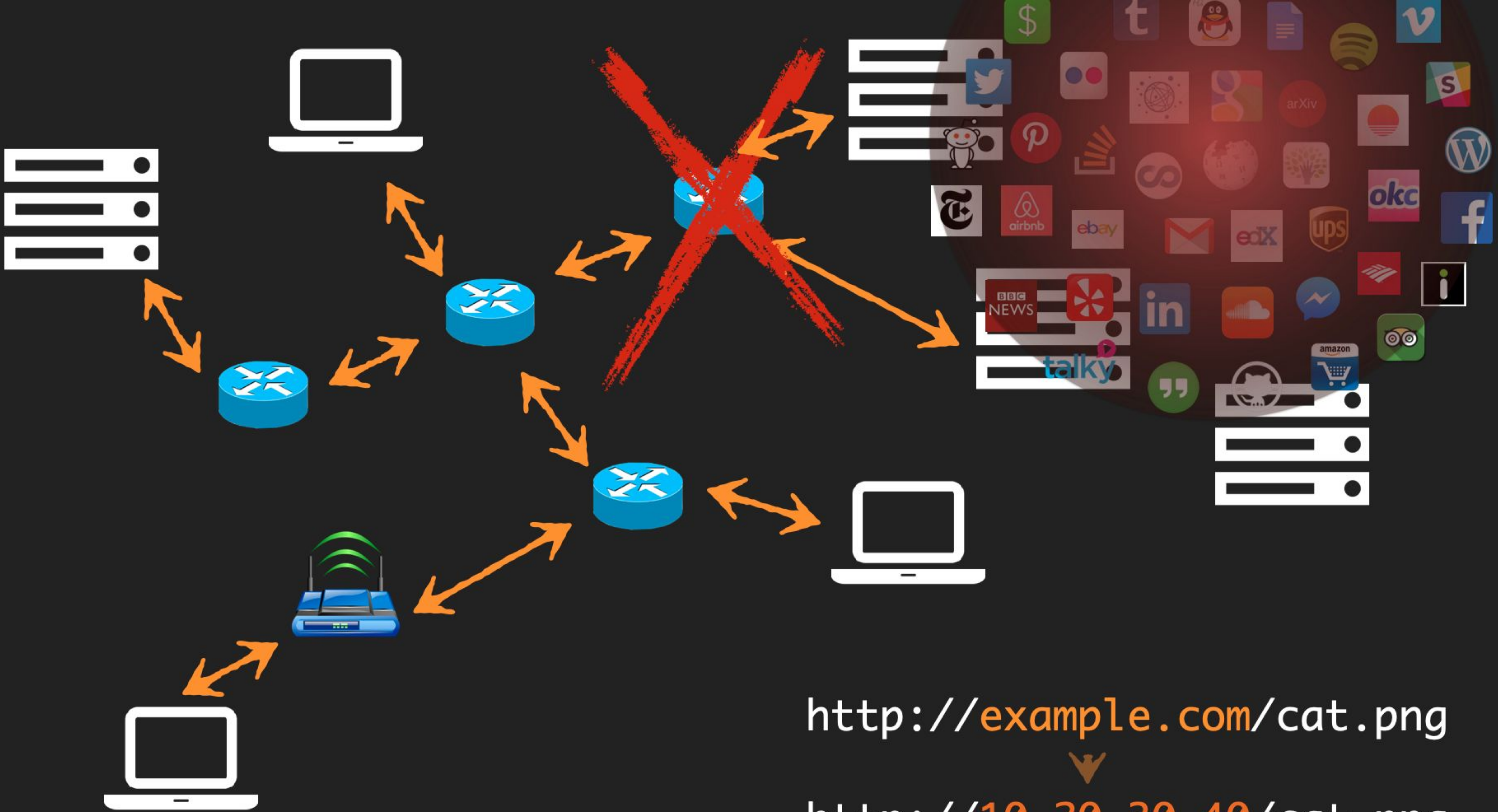
IoT

Offline


Permanence

Security

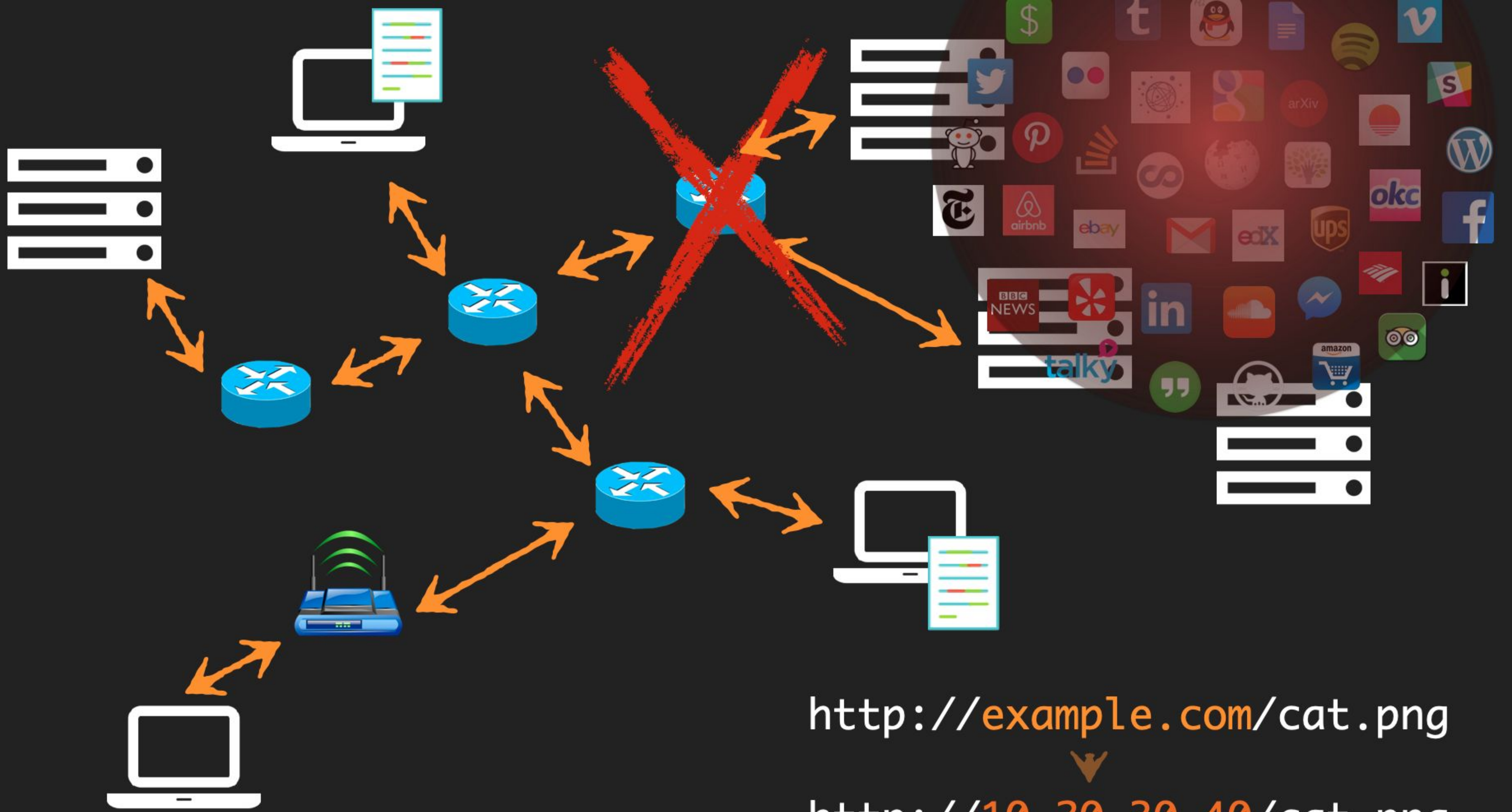
AUTHENTICATED
& ENCRYPTED
AT REST



http://example.com/cat.png

↓

http://10.20.30.40/cat.png



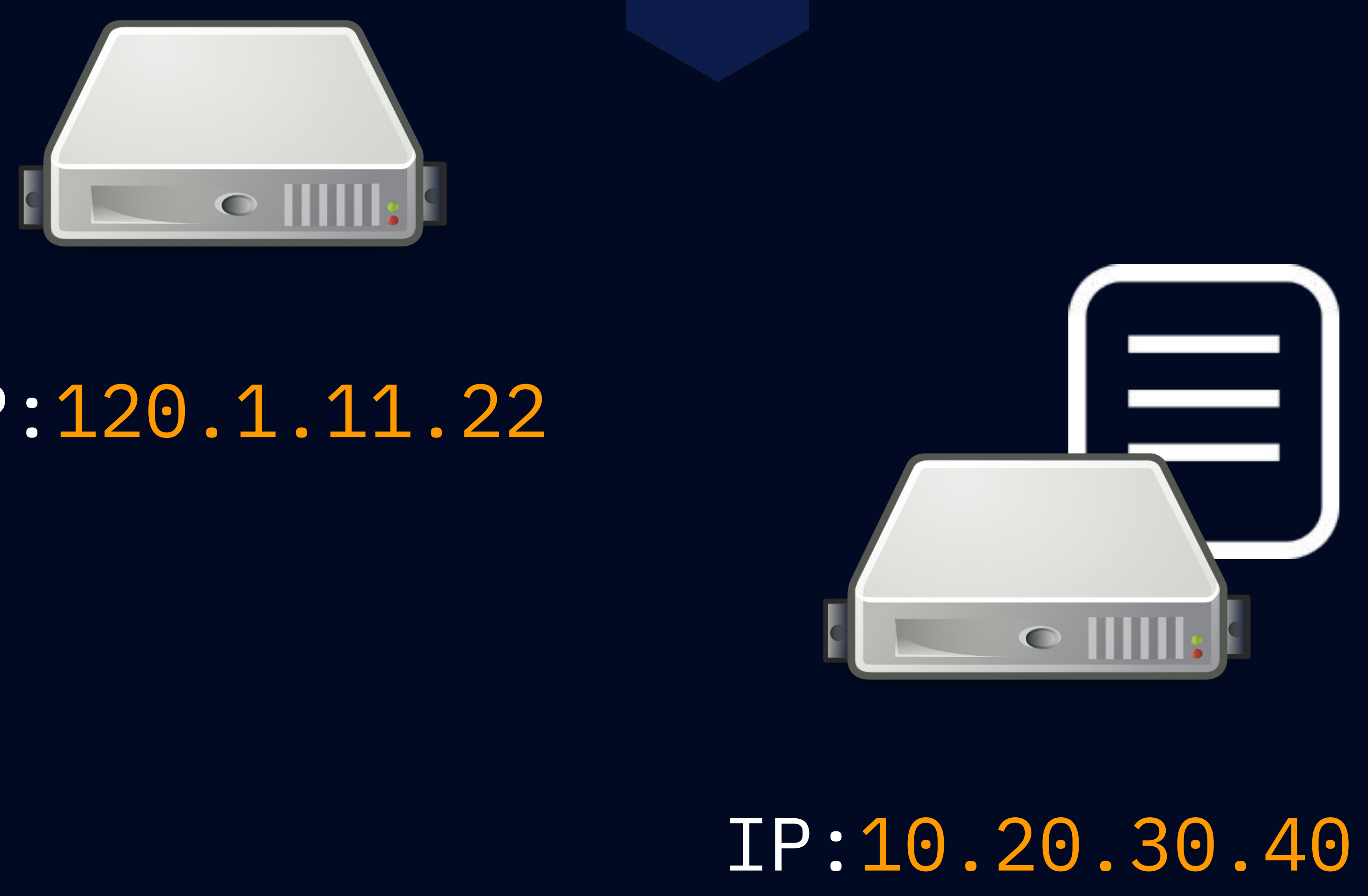
http://example.com/cat.png

↓

http://10.20.30.40/cat.png

200 MB x 30 x 8 = 48 GB

You Tube
0:26 / 4:12
HD



http://example.com/cat.png

↓

http://10.20.30.40/cat.png

location

/ipfs/QmW98pJrc6FZ6

content

↓

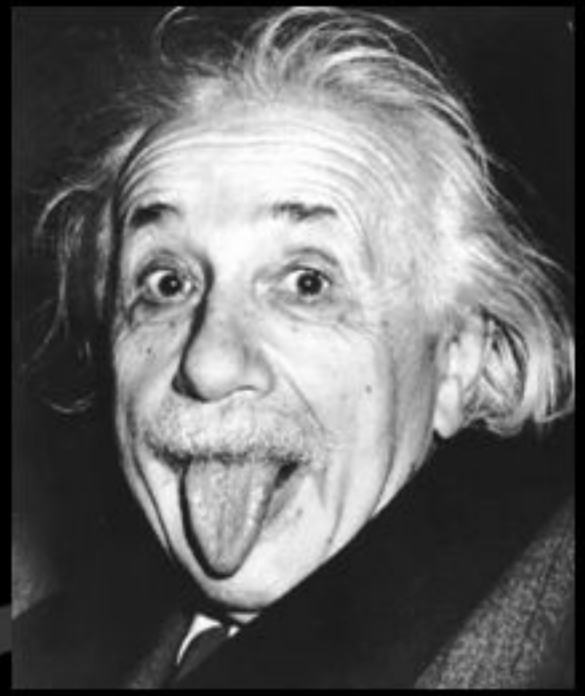
ipfs://QmW98pJrc6FZ6

ipfs

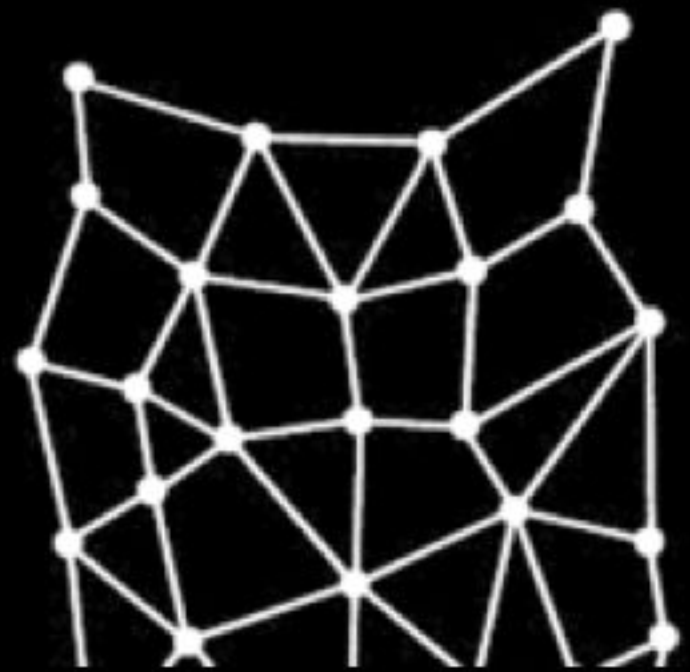
The Distributed Web

A protocol to upgrade the Web

Offline



Smarter



Distributed

Permanent

Safer

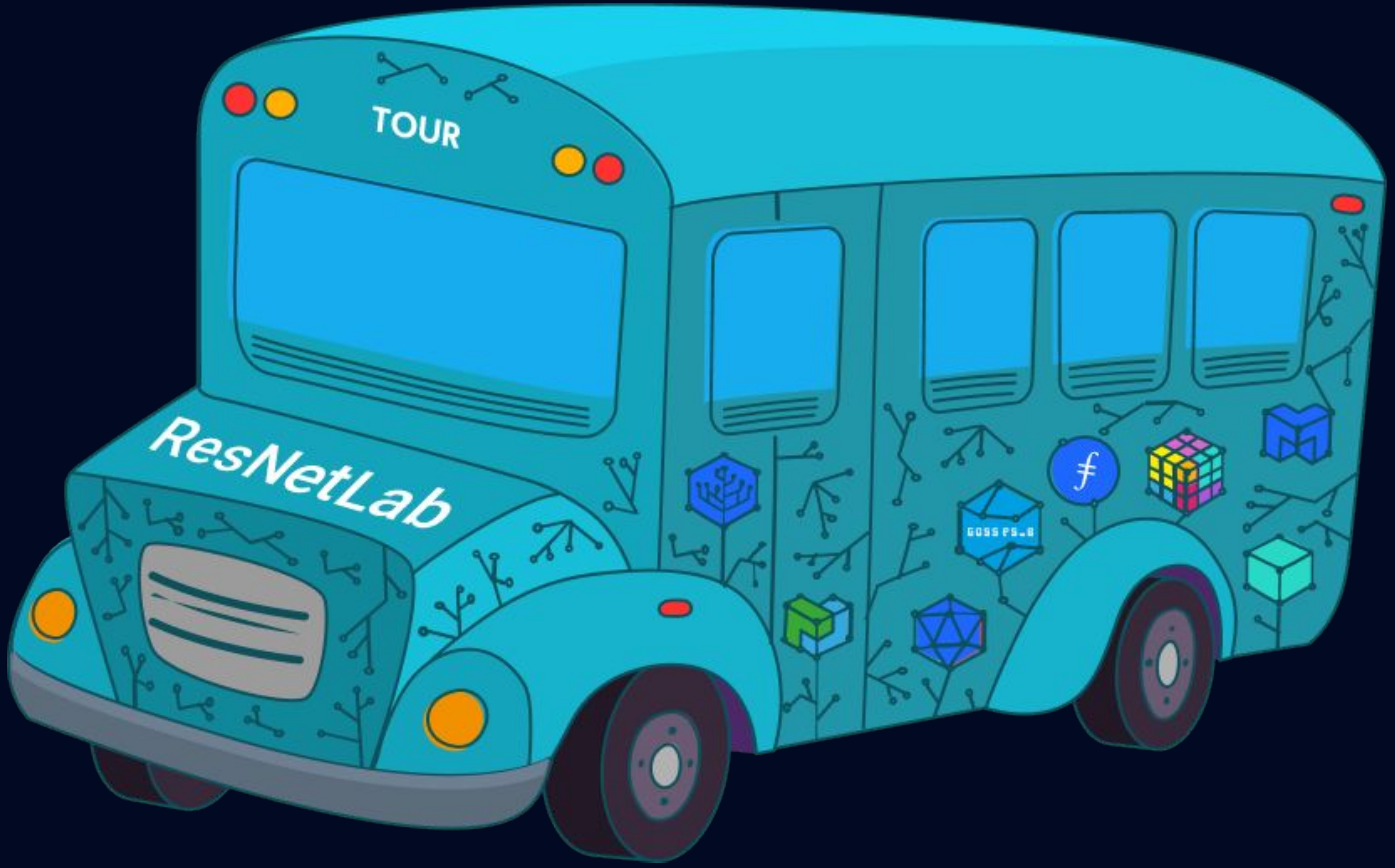
Faster

Booming ecosystem of applications

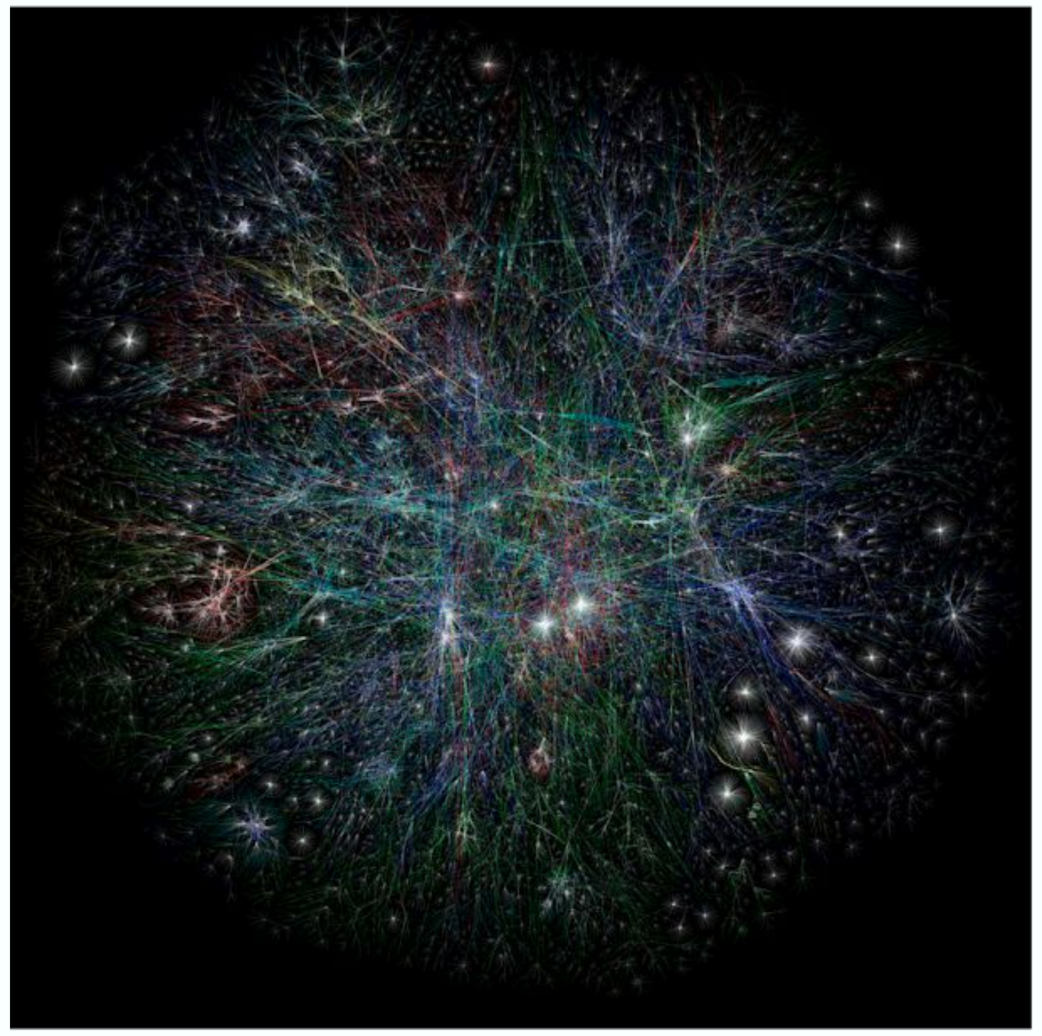
IPFS
Data
orbit db
qri.io
arbore
Catena BLOCKCHAIN EXPLORER
Identity
civic
ION
NOMIOS
ZINC
handshake
Productivity
berty
PeerPad
PEERGOS
Marketplace
OpenBazaar
ORIGIN
Bounty0x
Ethlance
NAME BAZAAR ENS ETH
PORTION
NFT
dlux DECENTRALIZED LIMITLESS USER EXPERIENCES
DIGITAL ART CHAIN
Decentraland
Glossy
mokens
Content
D.tube
EVERIPEDIA
BitTube
UNSTOPPABLE DOMAINS
ALEXANDRIA
dlive
Matters
Viewly
VIULY.COM
DSOUND
ujo
AUDIUS
Other
textile
MÓIBIT
TEMPORAL
#KarmaPay
SprtsHub
Simple As Water
ipwb
WINGS
adchain
MONITOR CHAIN
IPSE
IKU
Dappkit
FILESTORM
点对点科技 DECENTRAL TECH
kauri.io
INFURA
Social Media
3BOX
AKASHA
Indorse
Identifi
KARMA
Peepeth
busy
Governance
GovBlocks
Democracy Earth
ARAGON ONE
Exchange
Dether
faa.st
Swap.online
Finance
REQUEST NETWORK
Bloom
colu.
SETTLE
MARKETPROTOCOL
Integrations & Collabs
openSUSE
Guix
NETFLIX
CLOUDFLARE
Prediction and betting
AUGUR
CryptoBets
PLAY 2 WIN
VIRTUE POKER
Crypto Dice
MÖBIUS 2D

Module:
Welcome to Web 3.0

ResNetLab on Tour



Web 3.0 is the Read-Write-Trust-Verifiable Web



Internet

wires, network

Web 1.0

read-only static

Web 2.0

read-write interactive

Web 3.0

read-write-trust verifiable

IPFS: Distributed Web Protocol

IPLD: authenticated data model & formats

Multiformats: future-proofing formatting rules

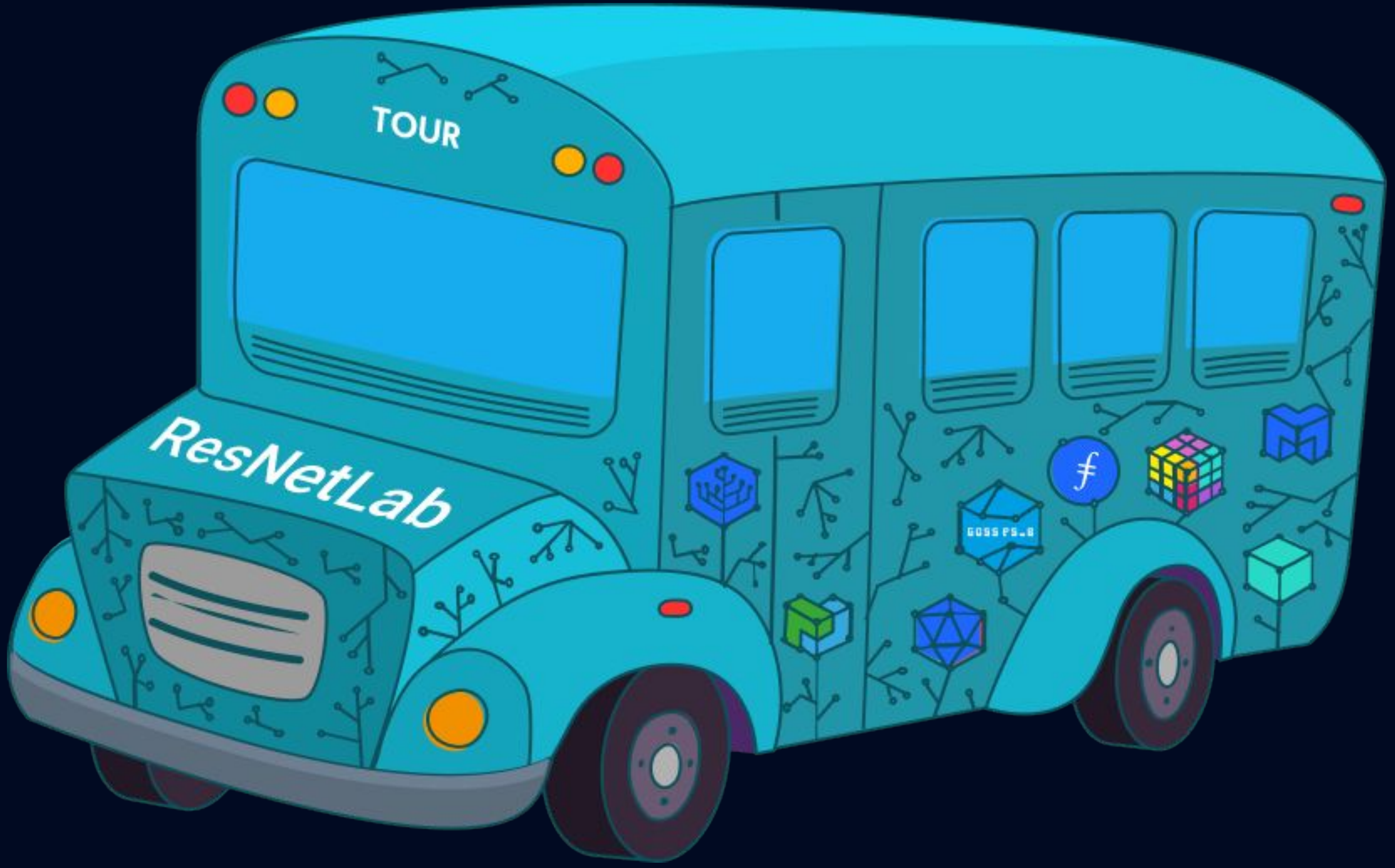
libp2p: modular p2p networking library

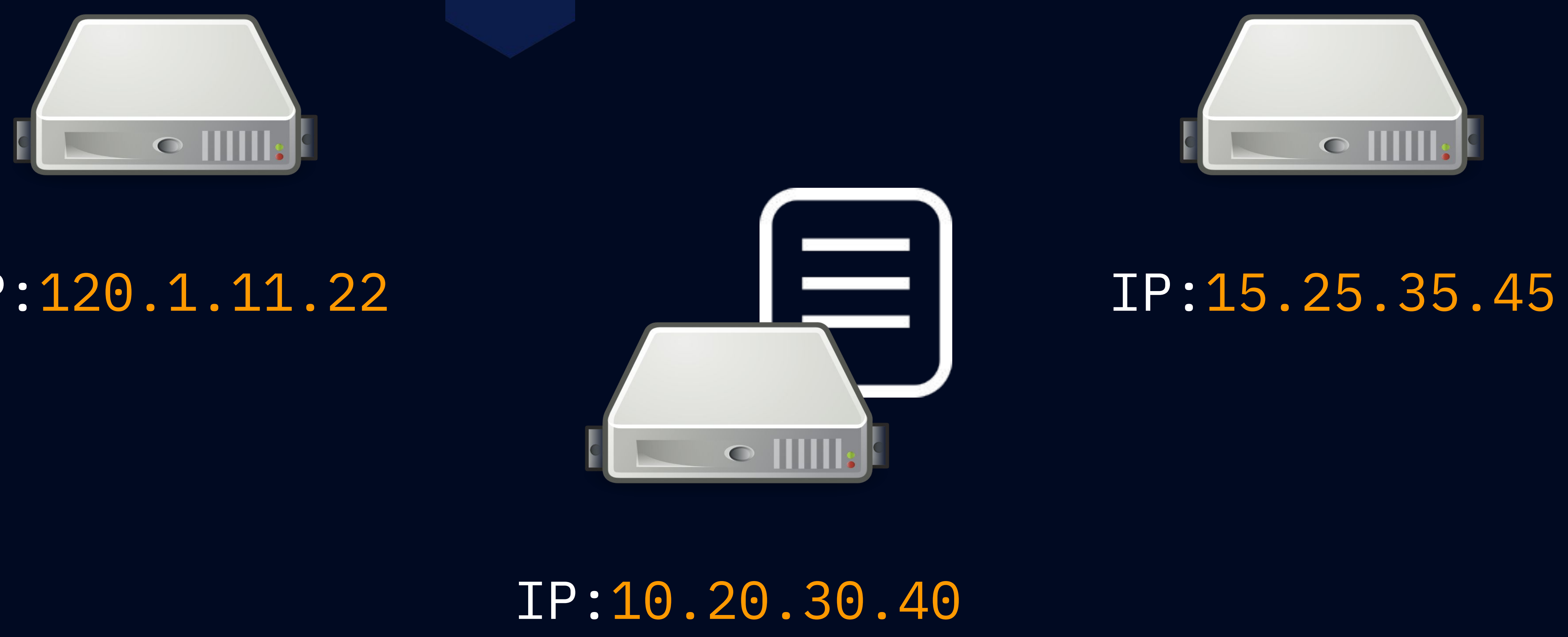
IPFS uses libp2p, IPLD and Multiformats to provide content-addressed decentralized storage.

Module: Content Addressing in IPFS

ResNetLab on Tour

ResNetLab





IP:120.1.11.22

IP:15.25.35.45

IP:10.20.30.40

http://example.com/cat.png

http://10.20.30.40/cat.png

location

/ipfs/QmW98pJrc6FZ6

content

ipfs://QmW98pJrc6FZ6

IPFS Components

CONTENT ADDRESSING

- **Anatomy of the IPFS CID**
- Chunking
- Linking Chunks in Merkle DAGs
- From Data to Data Structures with IPLD

CONTENT DISCOVERY & ROUTING

- Routing & Provider Records
- DHT-based Routing
- Gossip-based Routing

CONTENT EXCHANGE

- Bitswap
- GraphSync

MUTABLE NAMES & MESSAGE DELIVERY

- Dynamic Data
- IPNS
- PubSub
- CRDTs

Content Identifier

CIDs are:

- *the most fundamental ingredient of the IPFS architecture*
- used for **content addressing**
- used to name every piece of data in IPFS
- a **hash** with some **metadata**
- **self describing**

CIDv0: **Qm**S4ustL54uo8FzR9455qaxZwuMiUhyvMcX9Ba8nUH4uVv

CIDv1: **bafy**beibxm2nsadl3fnxv2sxcxmxaco2jl53wpeorjdzidjwf5aqdg7wa6u

Binary Breakdown

Anatomy of a CID

How to interpret the data

Hash function

CID-V1

dag-pb
(0x70)

sha-256
(0x12)

128 | 2

Actual Content Hash!

0000000101110000 00010010100000000000000010110010010…

CID Version

Multicodec

Multicodec

Length

<- IPLD encoding ->

bafybeigdyrzt5sfp7udm7hu76uh7y26nf3efuylqabf3oclgtqy55fbzdi

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

Visit: cid.ipfs.io

CIDs are Immutable links

Deduplication
Identical data can be identified by its address

Self-certification
Content is authenticated by its address

Integrity checking
If the content changes, its address also changes

IPFS Components

CONTENT ADDRESSING

- Anatomy of the IPFS CID
- **Chunking**
- Linking Chunks in Merkle DAGs
- From Data to Data Structures with IPLD

CONTENT DISCOVERY & ROUTING

- Routing & Provider Records
- DHT-based Routing
- Gossip-based Routing

CONTENT EXCHANGE

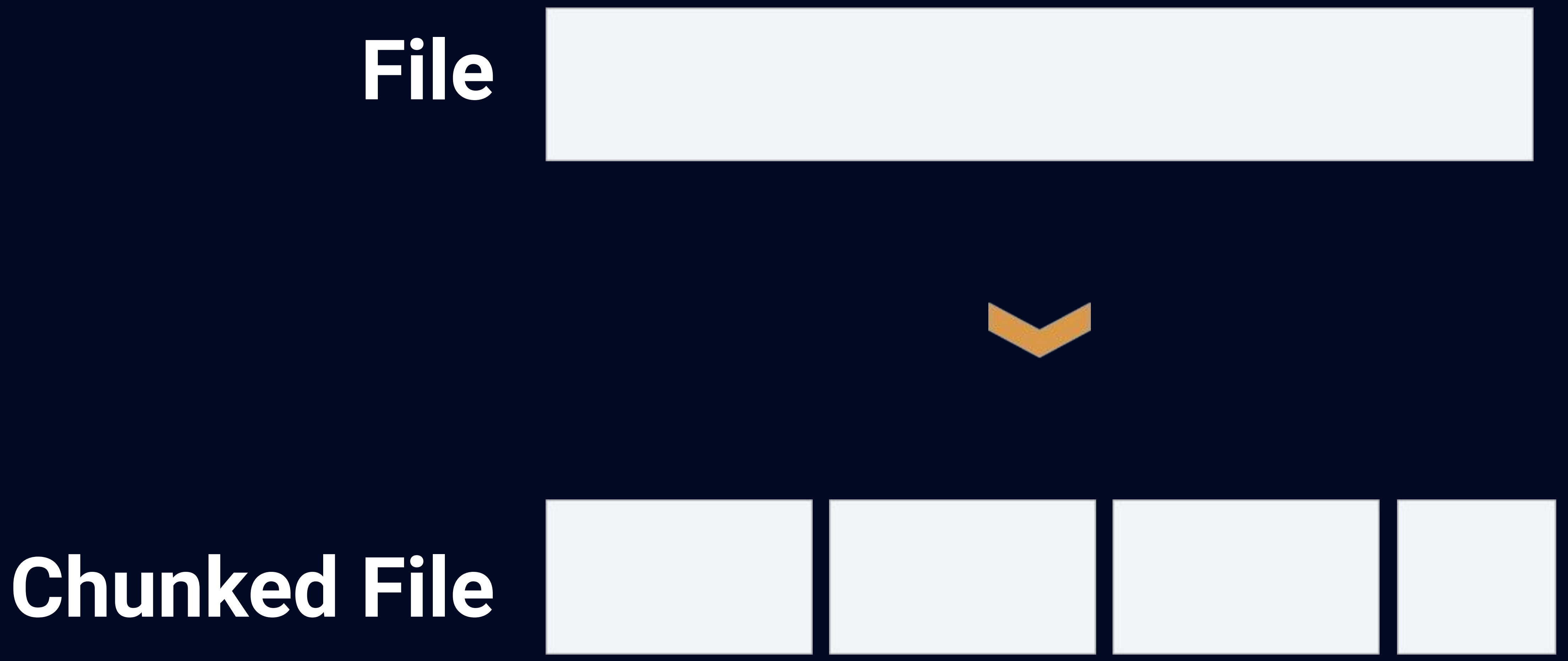
- Bitswap
- GraphSync

MUTABLE NAMES & MESSAGE DELIVERY

- Dynamic Data
- IPNS
- PubSub
- CRDTs

Content Addressing

Chunking

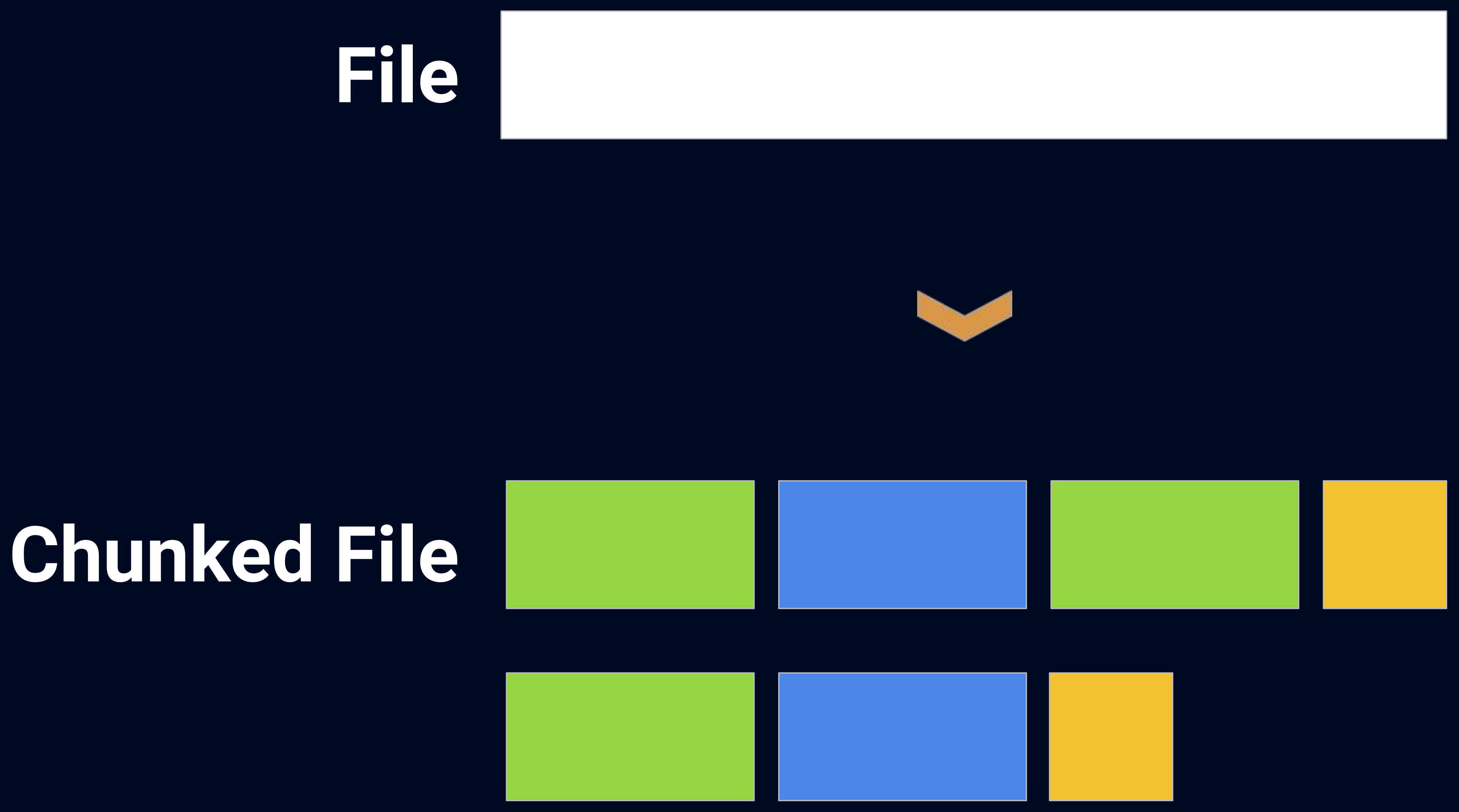


- Deduplication
- Piecewise Transfer
- Random Access

Each chunk is individually addressed and identified by its own hash

Content Addressing

Chunking

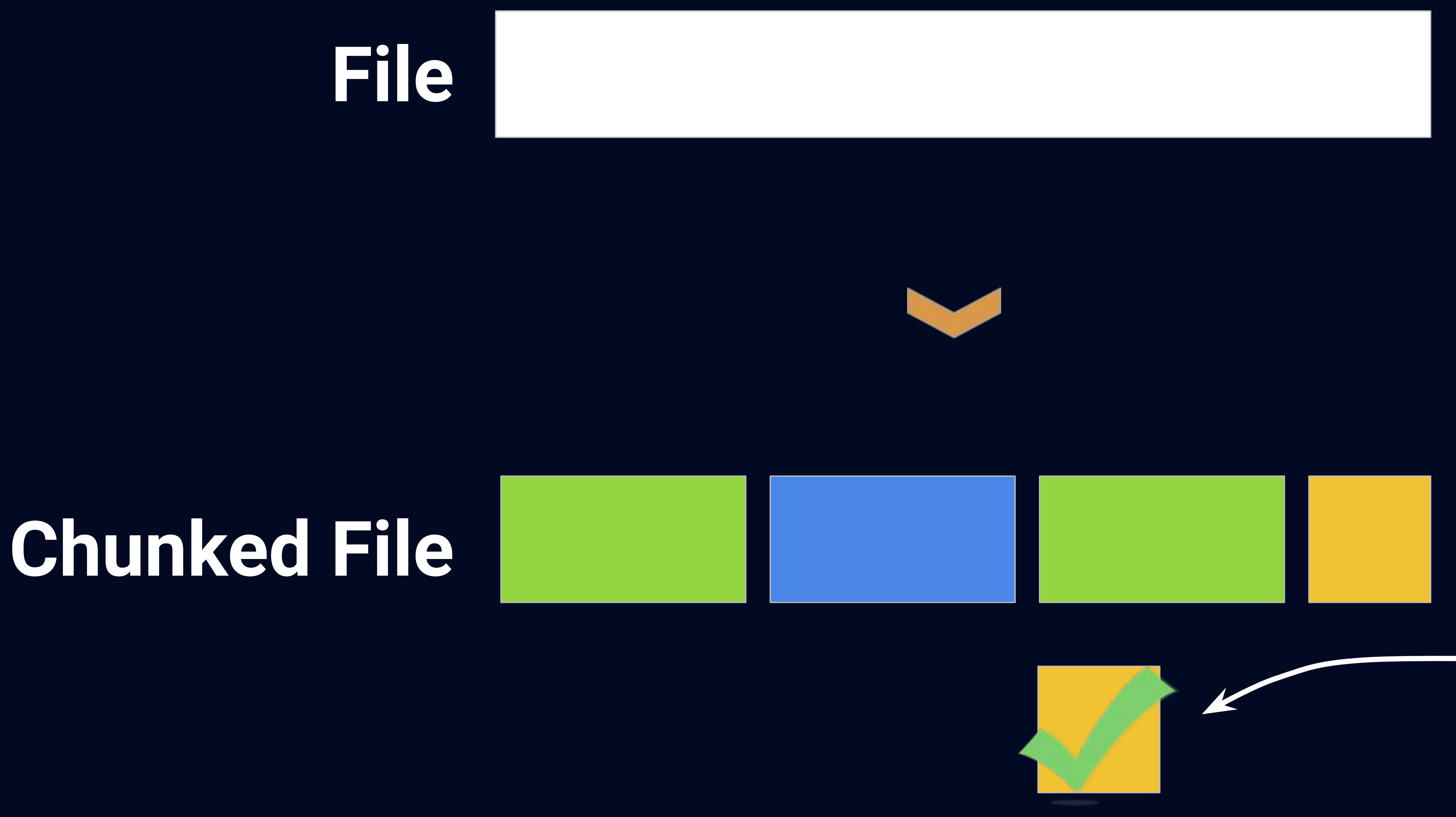


Optimise **storage** requirements

- **Deduplication**
- Random Access
- Piecewise Transfer

Content Addressing

Chunking



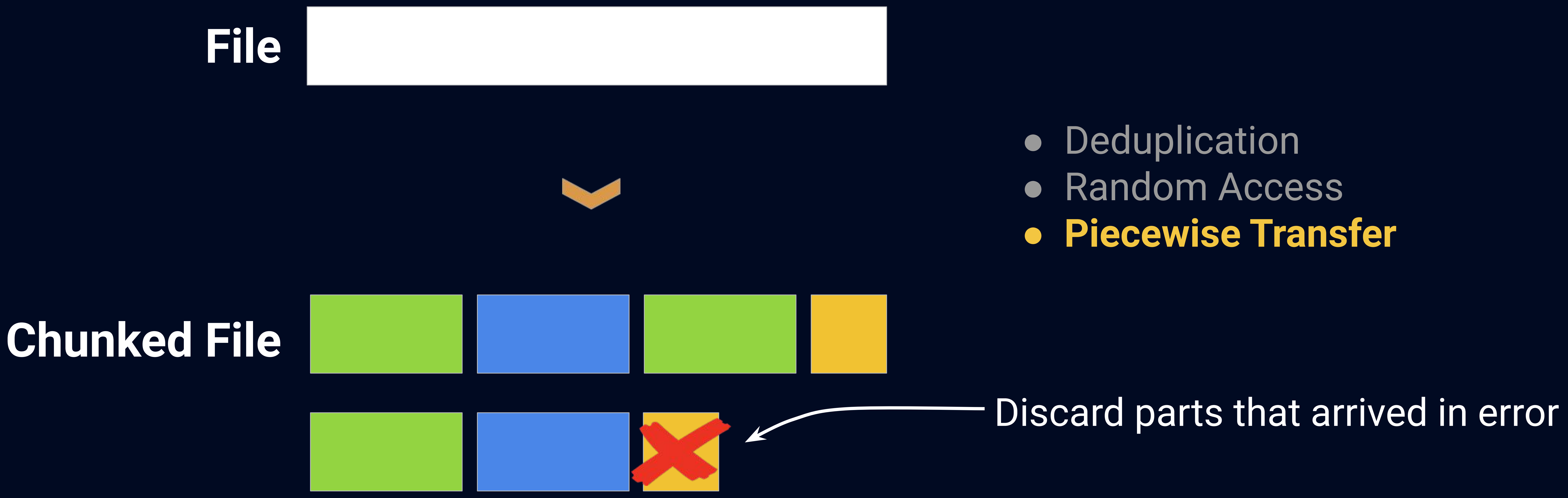
- Deduplication
- **Random Access**
- Piecewise Transfer

Fetch the parts you need only

Optimise **bandwidth** requirements

Content Addressing

Chunking



Identify errors without having to fetch the whole file

- Deduplication
- Random Access
- **Piecewise Transfer**

IPFS Components

CONTENT ADDRESSING

- Anatomy of the IPFS CID
- Chunking
- **Linking Chunks in Merkle DAGs**
- From Data to Data Structures with IPLD

CONTENT DISCOVERY & ROUTING

- Routing & Provider Records
- DHT-based Routing
- Gossip-based Routing

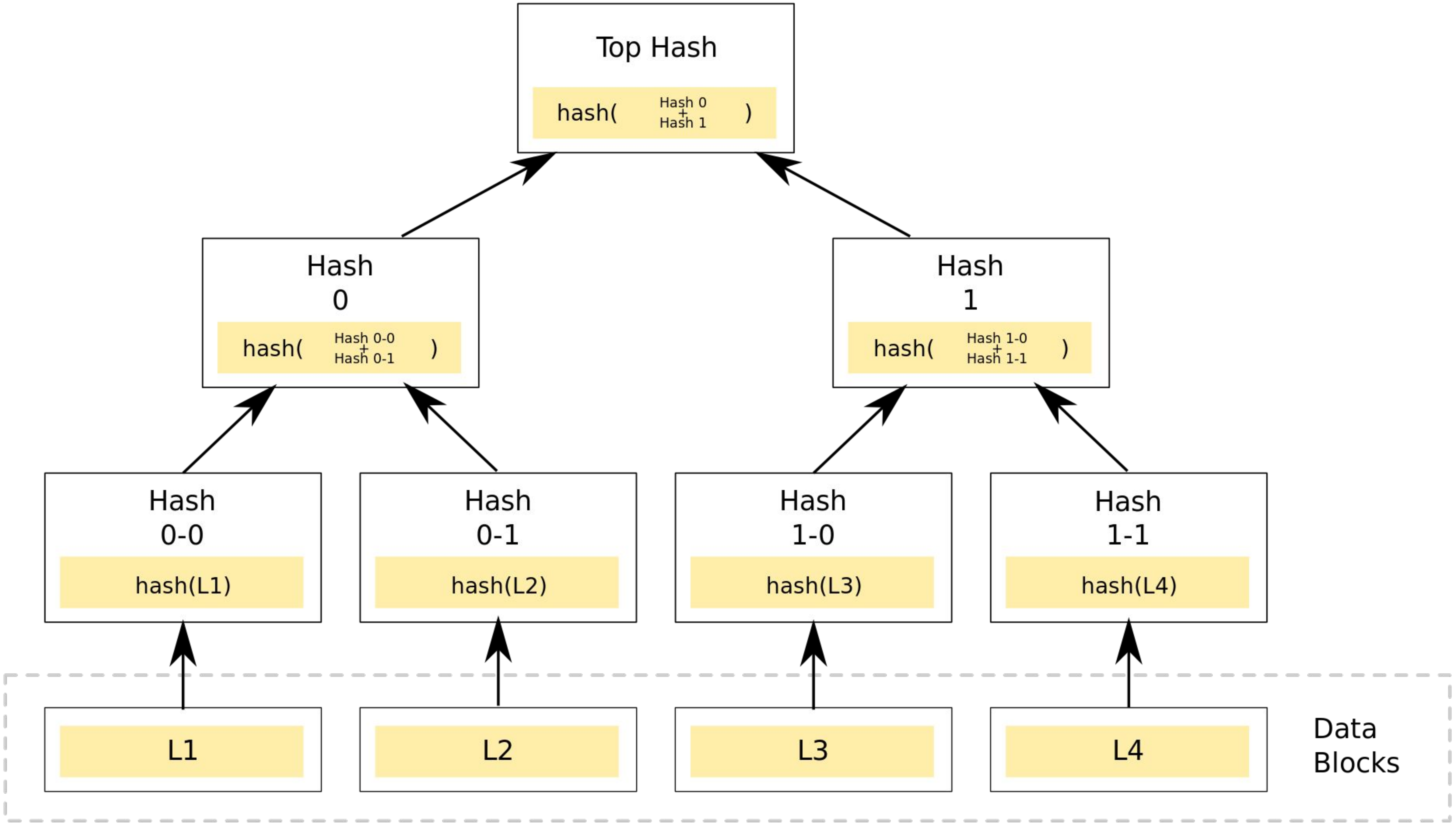
CONTENT EXCHANGE

- Bitswap
- GraphSync

MUTABLE NAMES & MESSAGE DELIVERY

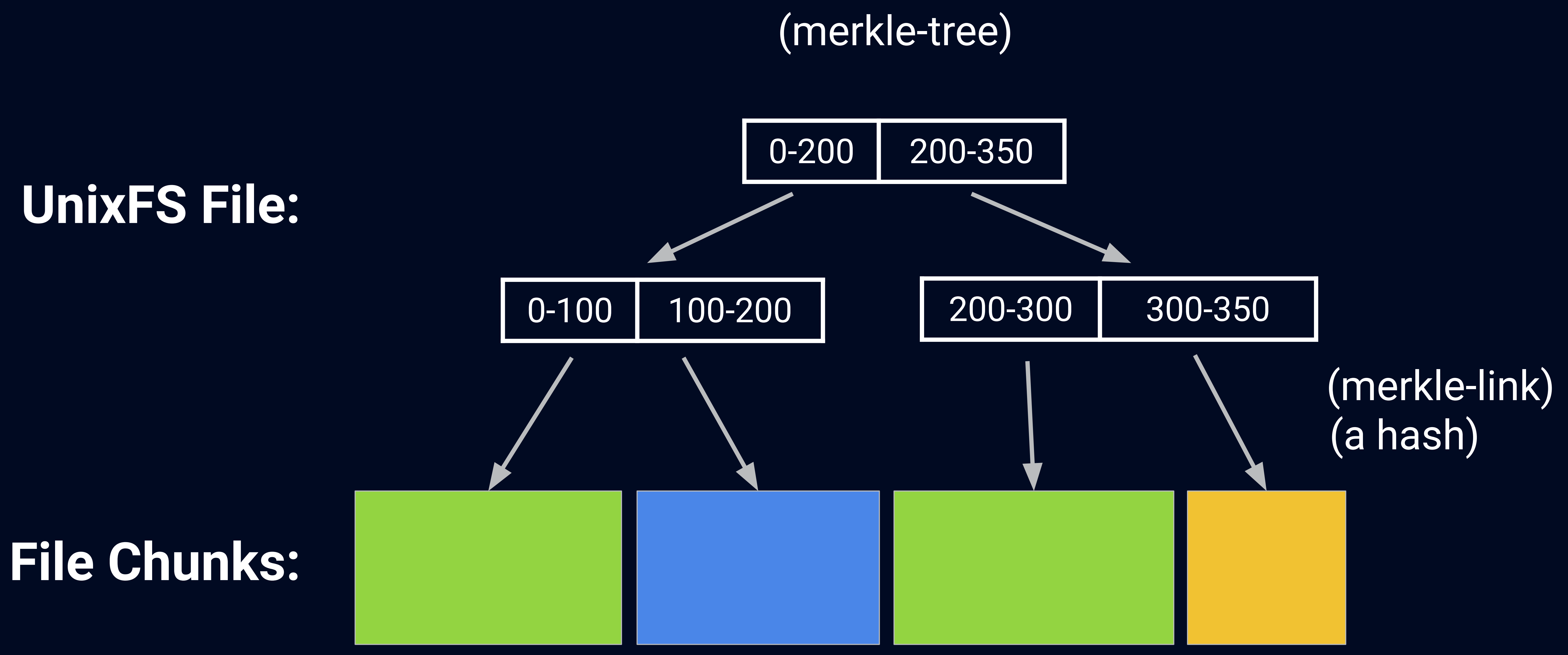
- Dynamic Data
- IPNS
- PubSub
- CRDTs

Merkle Trees



Content Addressing

Linking Chunks in a Tree

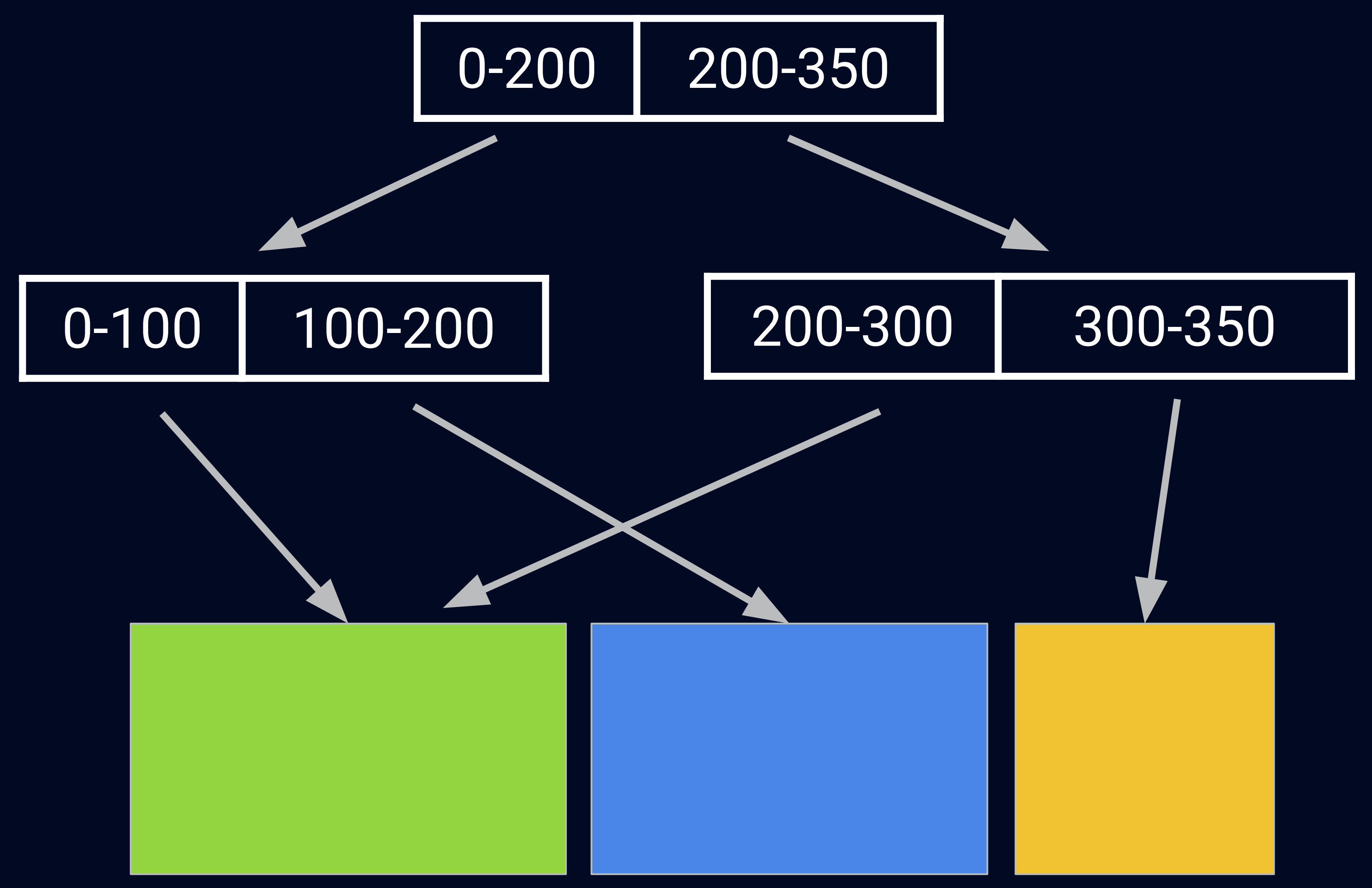


Content Addressing

Linking Chunks in a DAG

(merkle~~-tree~~-***dag***) - directed acyclic graph

UnixFS File:



0-200 | 200-350

0-100 | 100-200

200-300 | 300-350

(merkle-link)
(a hash)

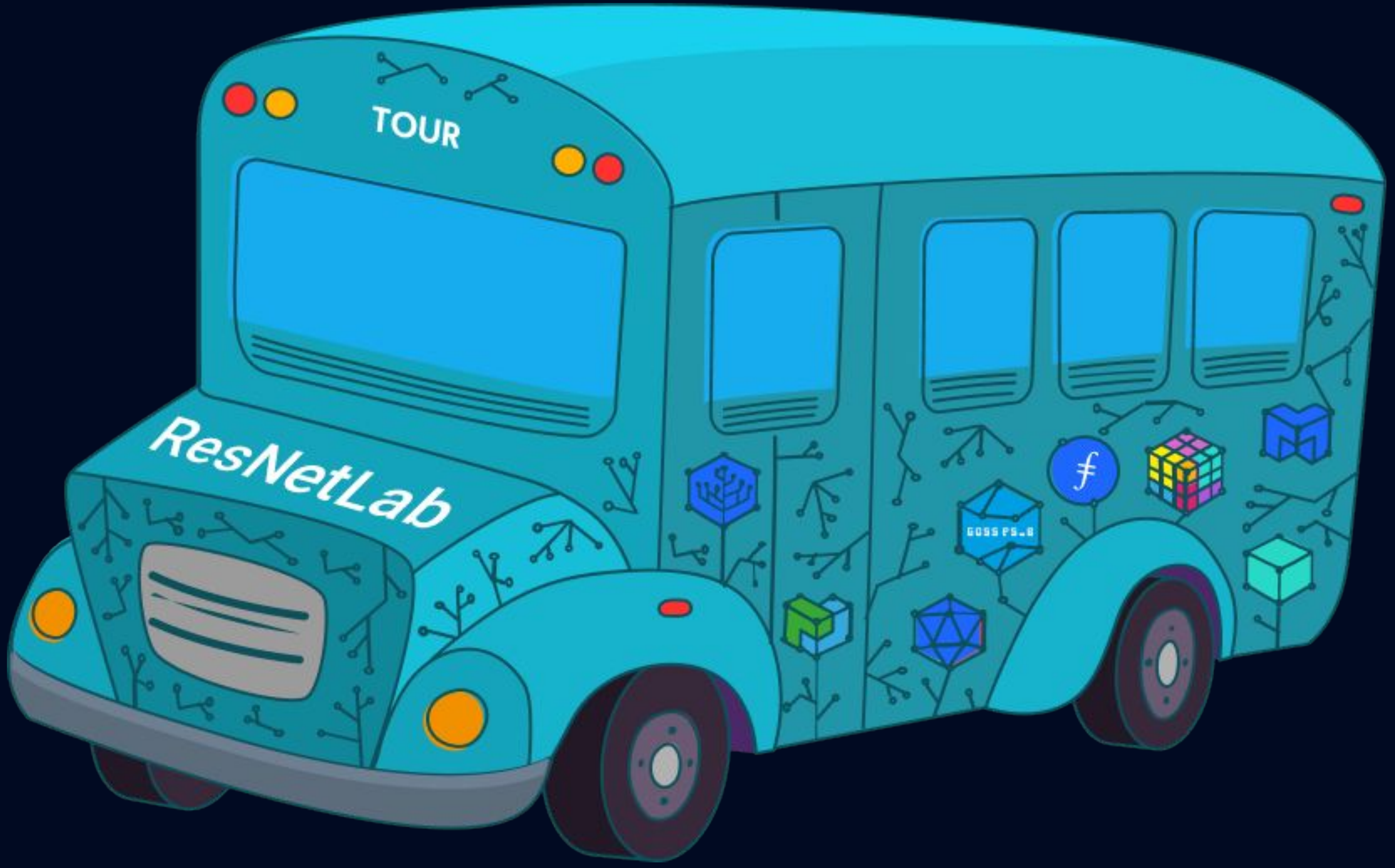
File Chunks:

Merkle DAGs are graph data structures where each node is content-addressed

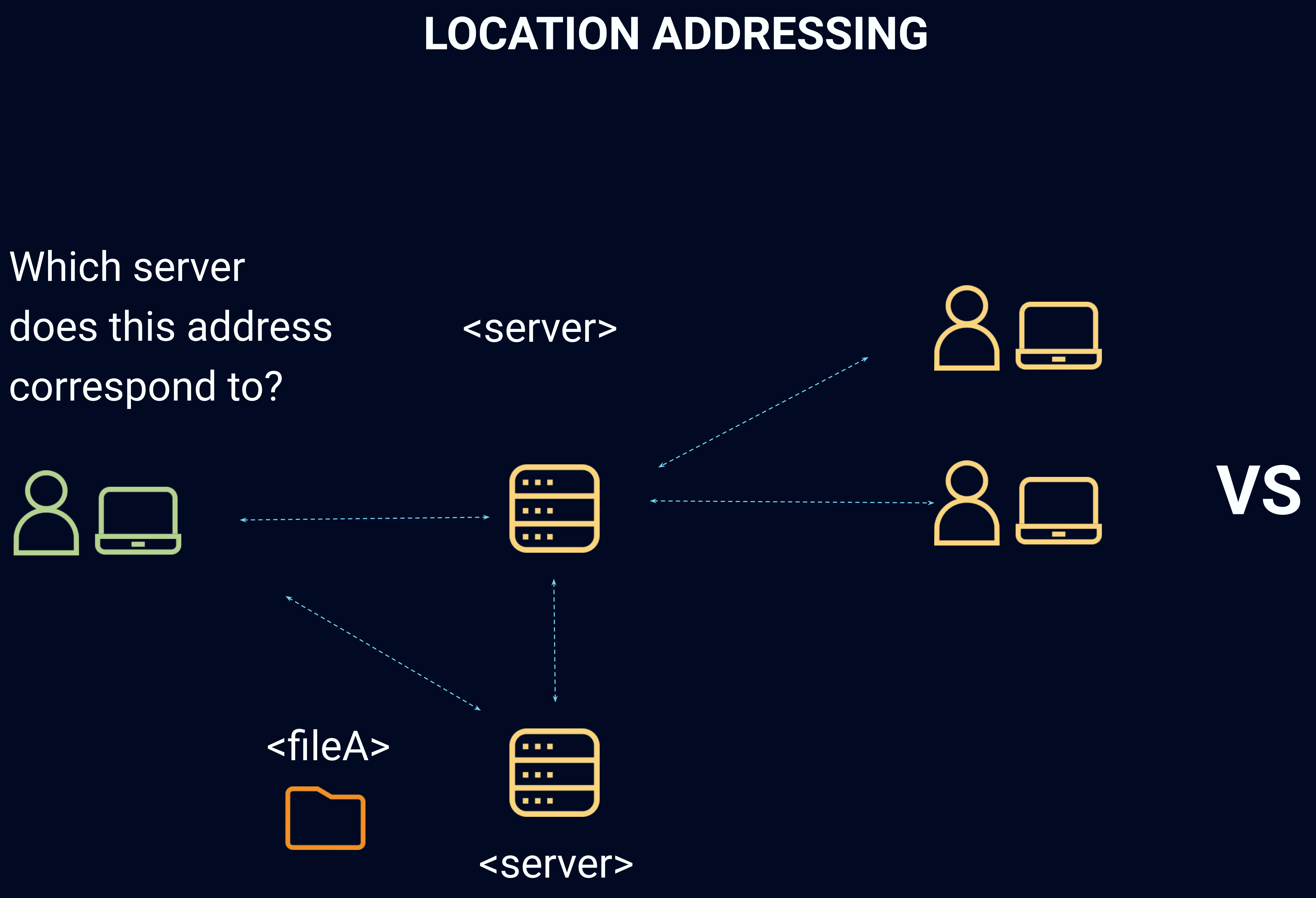
Visit: dag.ipfs.io

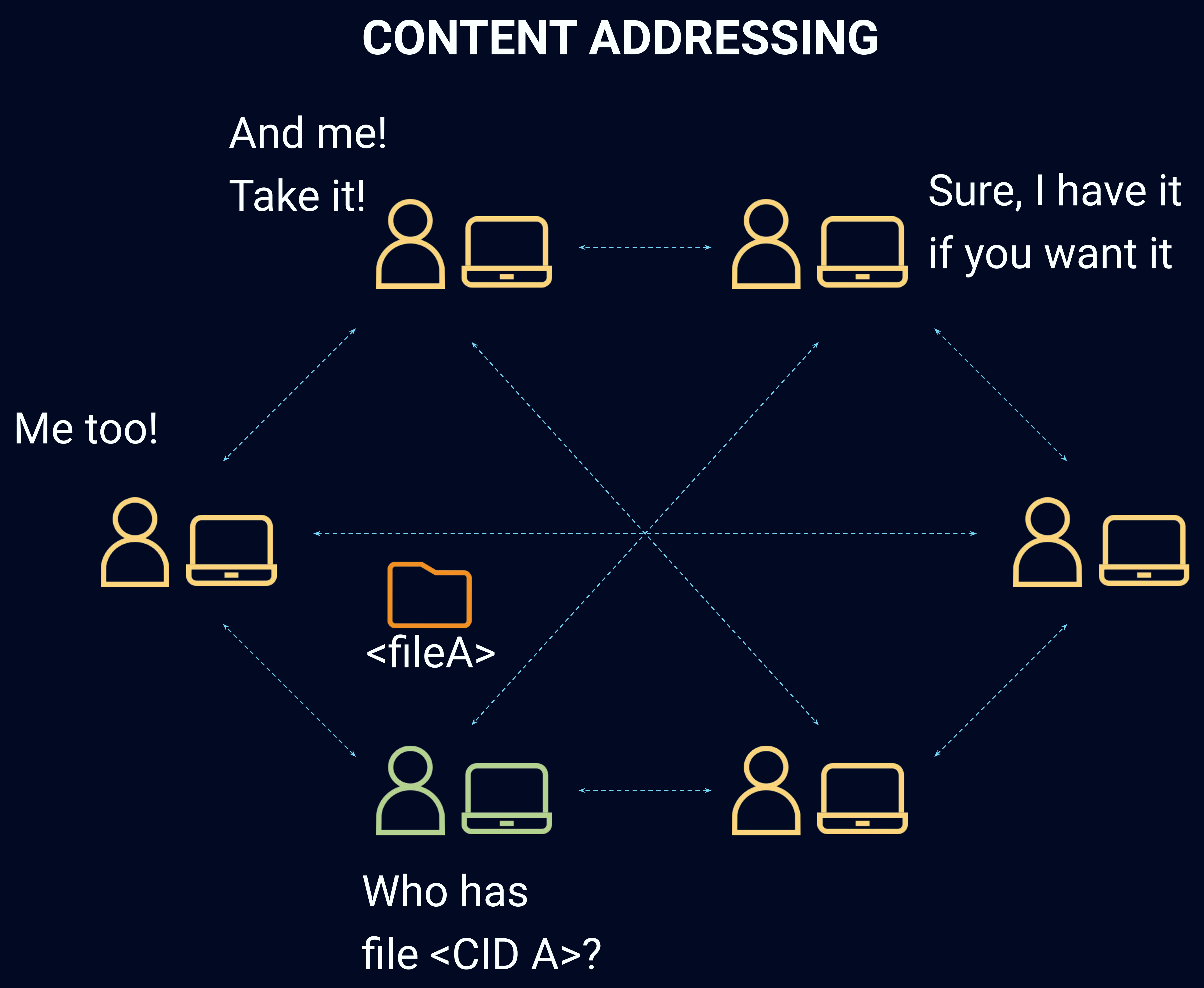
Module: Content Routing

ResNetLab on Tour



Location Addressing vs Content Addressing





The challenge of content routing in P2P networks

- There is no central entity orchestrating the storage and discovery of content.
- There is no central directory to find how to reach every peer in the network.
- P2P networks present high node churn.
- Thousands of peers and millions of content item!

Challenges include...

Discovering peers in the network

Finding peers storing the content

Contacting these peers to request the content

Doing it all in scalable way!

IPFS Components

CONTENT ADDRESSING

- Anatomy of the IPFS CID
- Chunking
- Linking Chunks in Merkle DAGs
- From Data to Data Structures with IPLD

CONTENT DISCOVERY & ROUTING

- **Routing & Provider Records**
- DHT-based Routing
- Gossip-based Routing

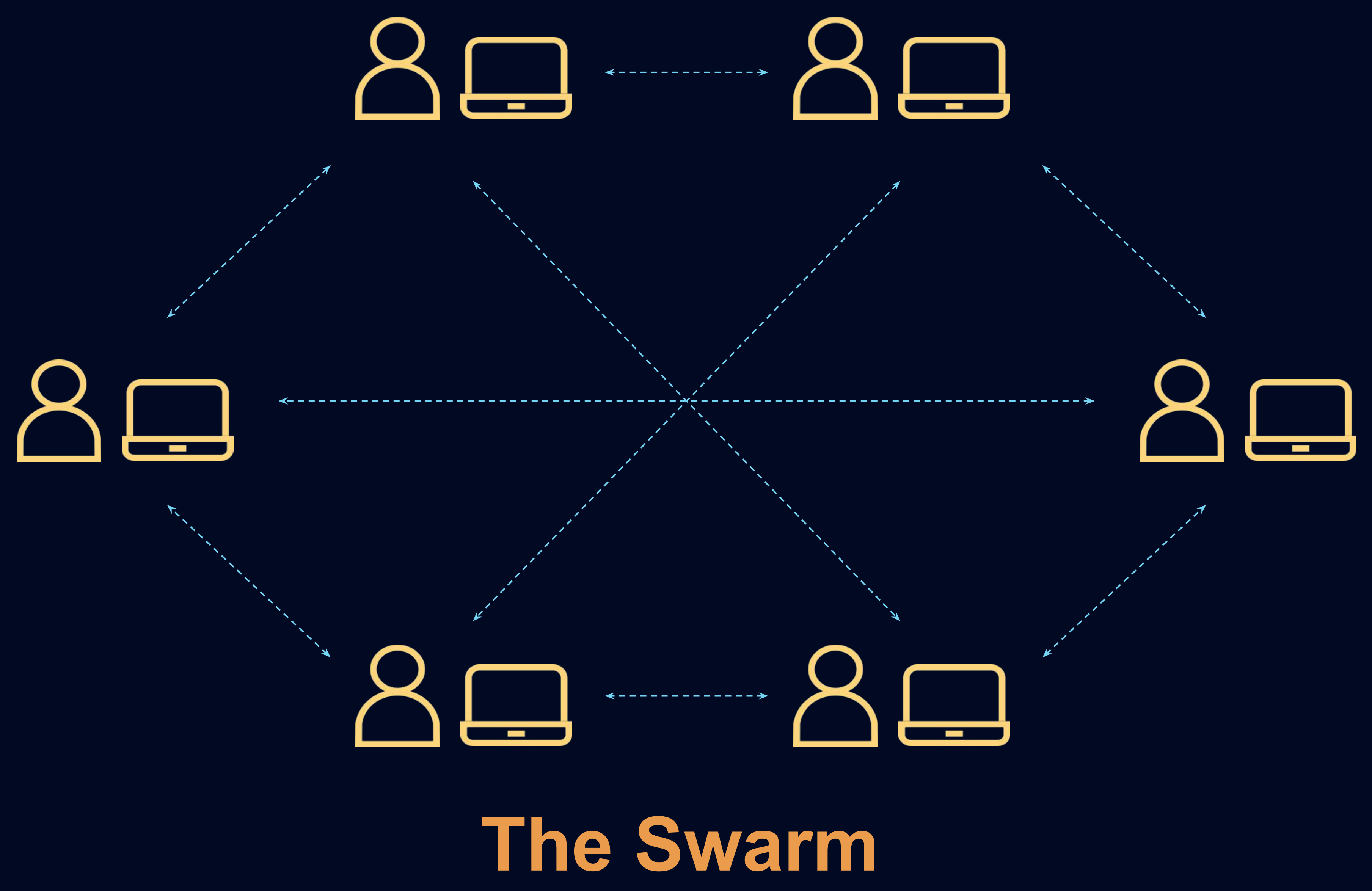
CONTENT EXCHANGE

- Bitswap
- GraphSync

MUTABLE NAMES & MESSAGE DELIVERY

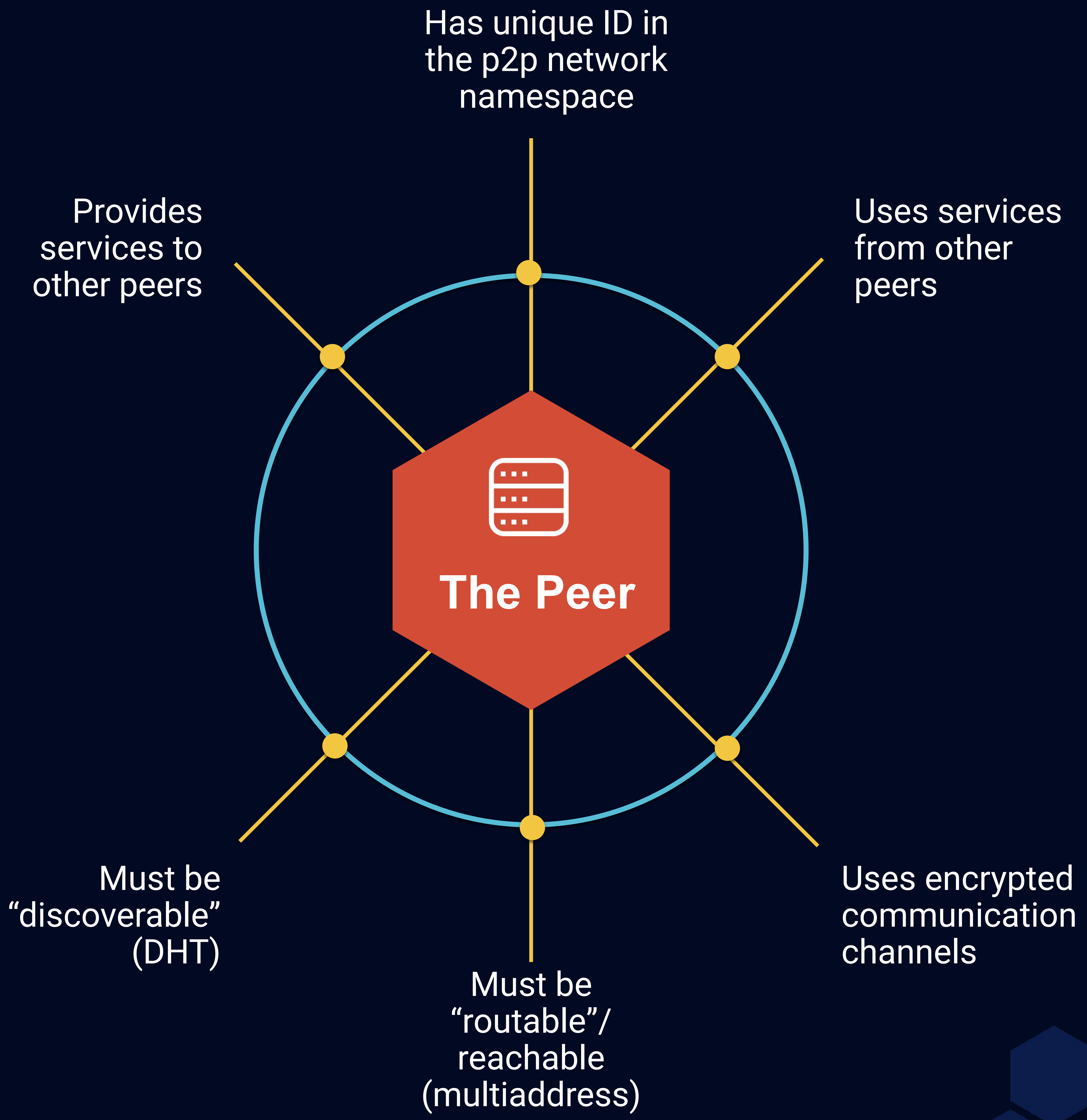
- Dynamic Data
- IPNS
- PubSub
- CRDTs

Peer Routing



Every peer uses a cryptographic key pair, for the purpose of

- Identity: unique name in the network
  "QmTuAM7RMnMqKnTq6qH1u9JiK5LqQvUxFdnrcM4aRHxeew"
- Channel security (encryption)


Has unique ID in the p2p network namespace
Provides services to other peers
Uses services from other peers
The Peer
Must be "discoverable" (DHT)
Uses encrypted communication channels
Must be "routable"/ reachable (multiaddress)

Content Routing Interface in libp2p/IPFS

- Design goals
    - Reliable: any content can be found
    - Scalable and fast: The performance of queries are not affected by the size of the network
    - Resistant to node churn and sybil attacks

- Two design approaches
    - DHT-based: libp2p KadDHT
    - Gossip-based: Bitswap, PubSub

- Operations
    - Provide: Make content available for other peers
    - Resolve: Find the peers storing the content
    - Fetch: Fetches content from a provider

IPFS Components

CONTENT ADDRESSING

- Anatomy of the IPFS CID
- Chunking
- Linking Chunks in Merkle DAGs
- From Data to Data Structures with IPLD

CONTENT DISCOVERY & ROUTING

- **Routing & Provider Records**
- **DHT-based Routing**
- Gossip-based Routing

CONTENT EXCHANGE

- Bitswap
- GraphSync

MUTABLE NAMES & MESSAGE DELIVERY

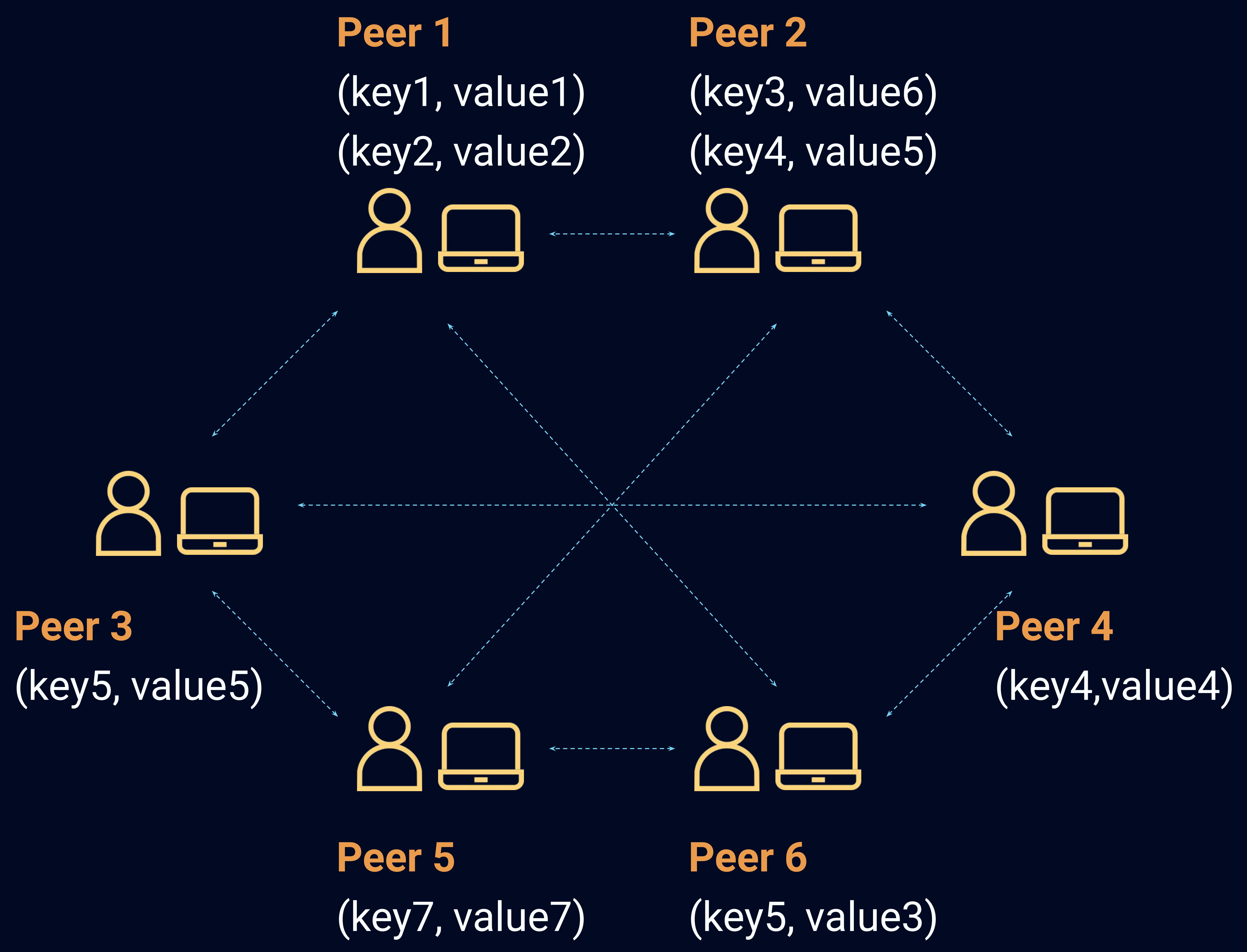
- Dynamic Data
- IPNS
- PubSub
- CRDTs

The DHT

- A DHT provides a 2-column table (key-value store) maintained by multiple peers.

- Each row is stored by peers based on similarity between the key and the peer ID. We call this “distance”:
  - A peer ID can be “closer” to some keys than others
  - A peer ID can be “closer” to other peers.

- The DHT is used in IPFS to provide:
  - Peer routing *(PeerID, /ipv4/1.2.3.4/tcp/...)*
  - Content Discovery *(ContentID, PeerID)*
  - IPNS Records *(IPNS key, IPNS Record)*



Peer 1
(key1, value1)
(key2, value2)

Peer 2
(key3, value6)
(key4, value5)

Peer 3
(key5, value5)

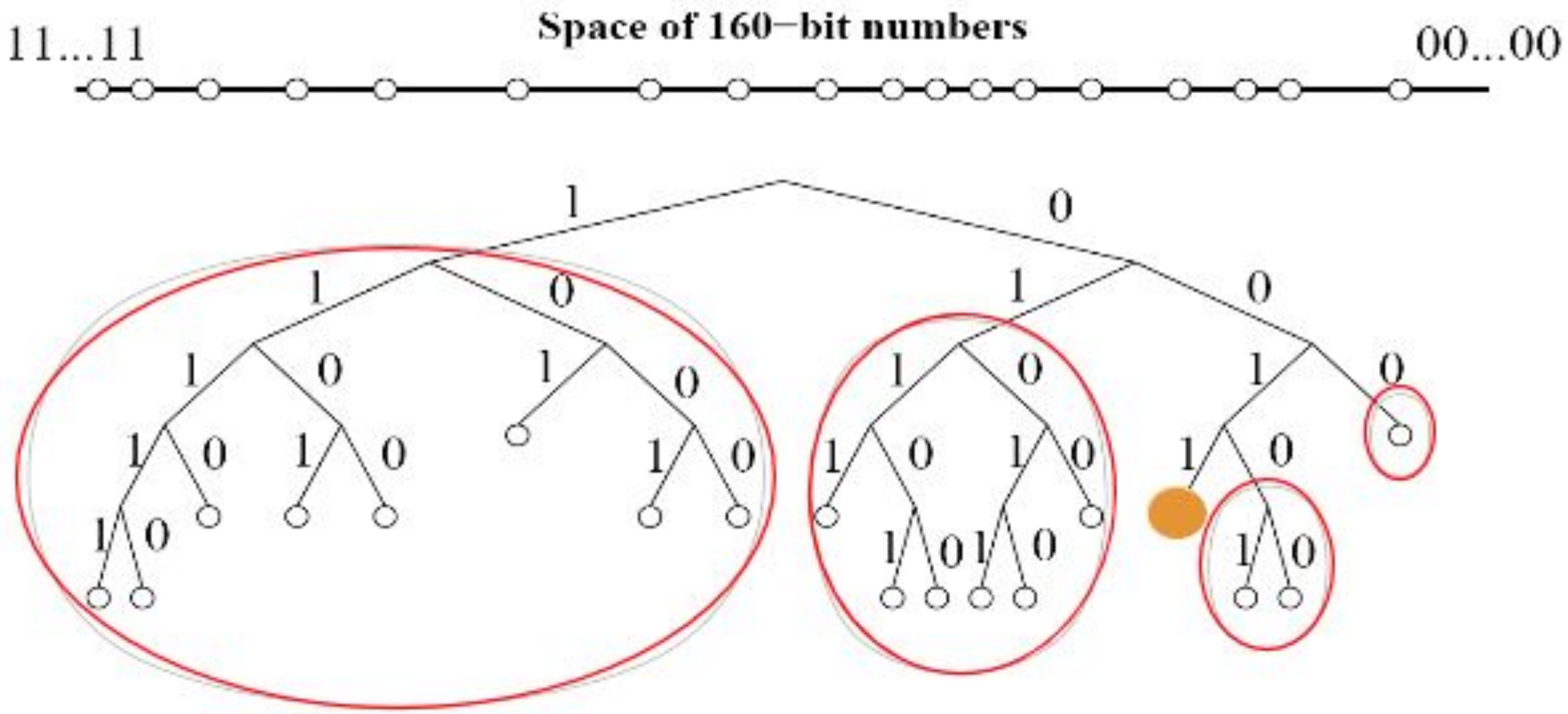
Peer 4
(key4,value4)

Peer 5
(key7, value7)

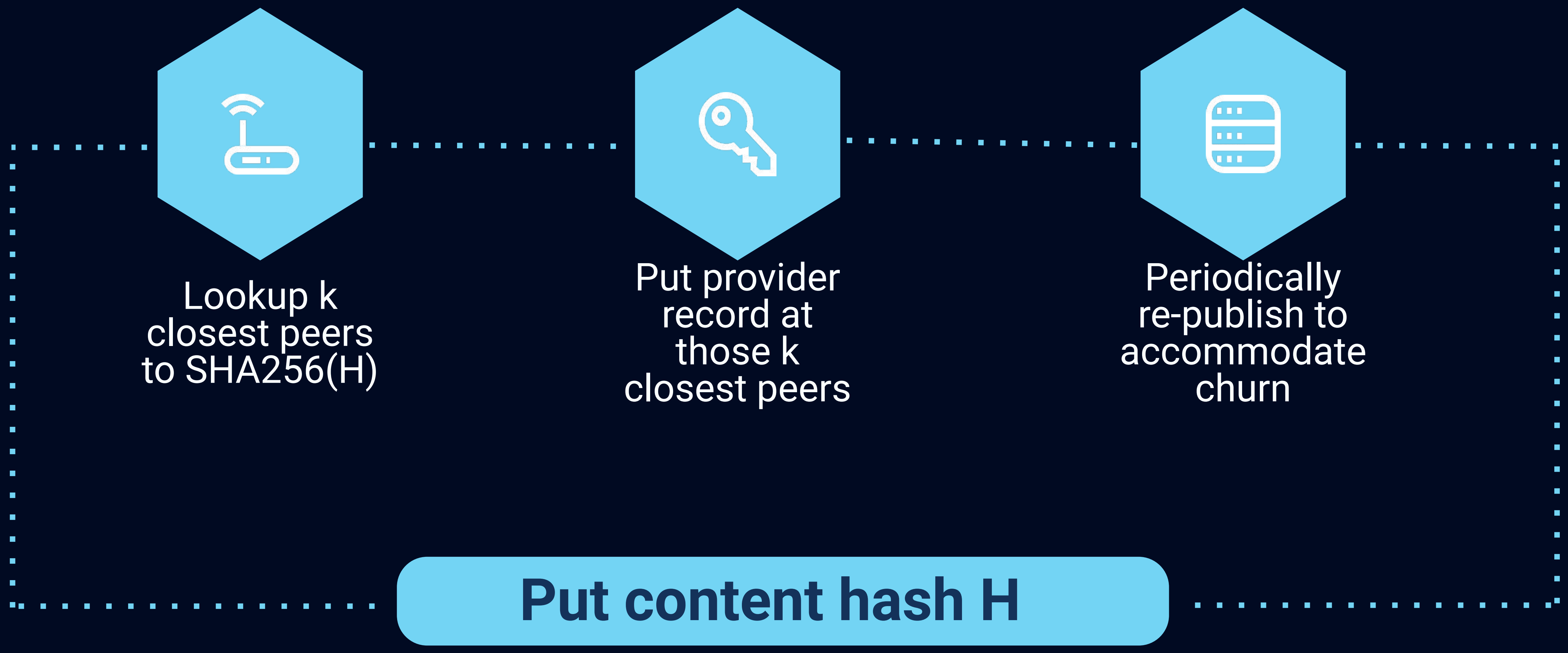
Peer 6
(key5, value3)

Inspired by Kademlia DHT

- IPFS uses an adaptation of the Kademlia DHT:
  - 256 bits address space - SHA256
  - Distance between two object through XOR
    - distance(a, b) = a XOR b = distance(b,a)
  - It uses tree-based routing (figure)
  - The binary tree is divided into a series of successively lower subtrees. Each contain a k-bucket (list of nodes with that prefix)
  - Initiates parallel asynchronous queries to avoid waiting for offline nodes.

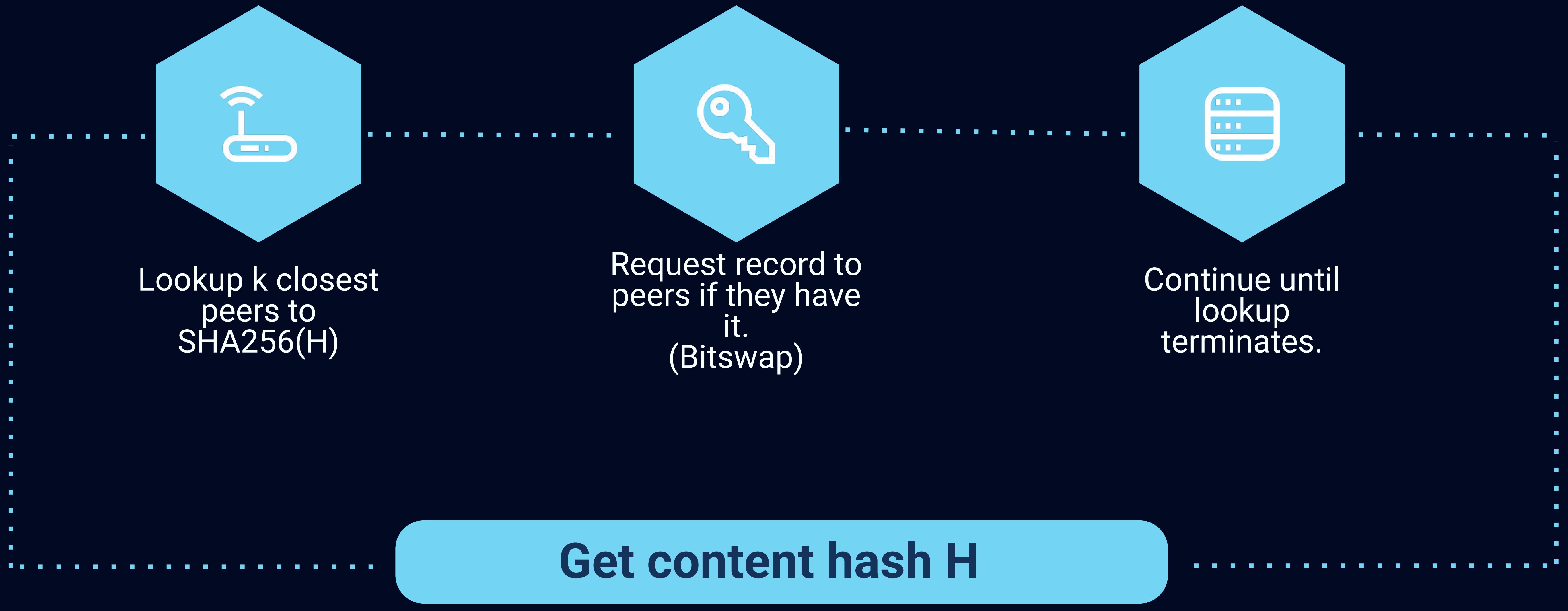


Providing Content


Lookup k closest peers to SHA256(H)
Put provider record at those k closest peers
Periodically re-publish to accommodate churn
Put content hash H

- Content is not replicated or uploaded to any external server. The content stays local on the user's device.
- It is the Content Identifier (CID) together with a pointer to the user's machine that is made known to the network.
  - This tuple is called the provider record and is added to 20 peers.
    - Provide records expire (i.e. they're not provided by peers) after 24 hours to account for *provider churn.*
    - Provider records are re-published after 12 hours (by providers) to account for *peer churn* (i.e. make sure close to 20 peers still store the record).

Resolving Content


Lookup k closest peers to SHA256(H)
Request record to peers if they have it. (Bitswap)
Continue until lookup terminates.
Get content hash H

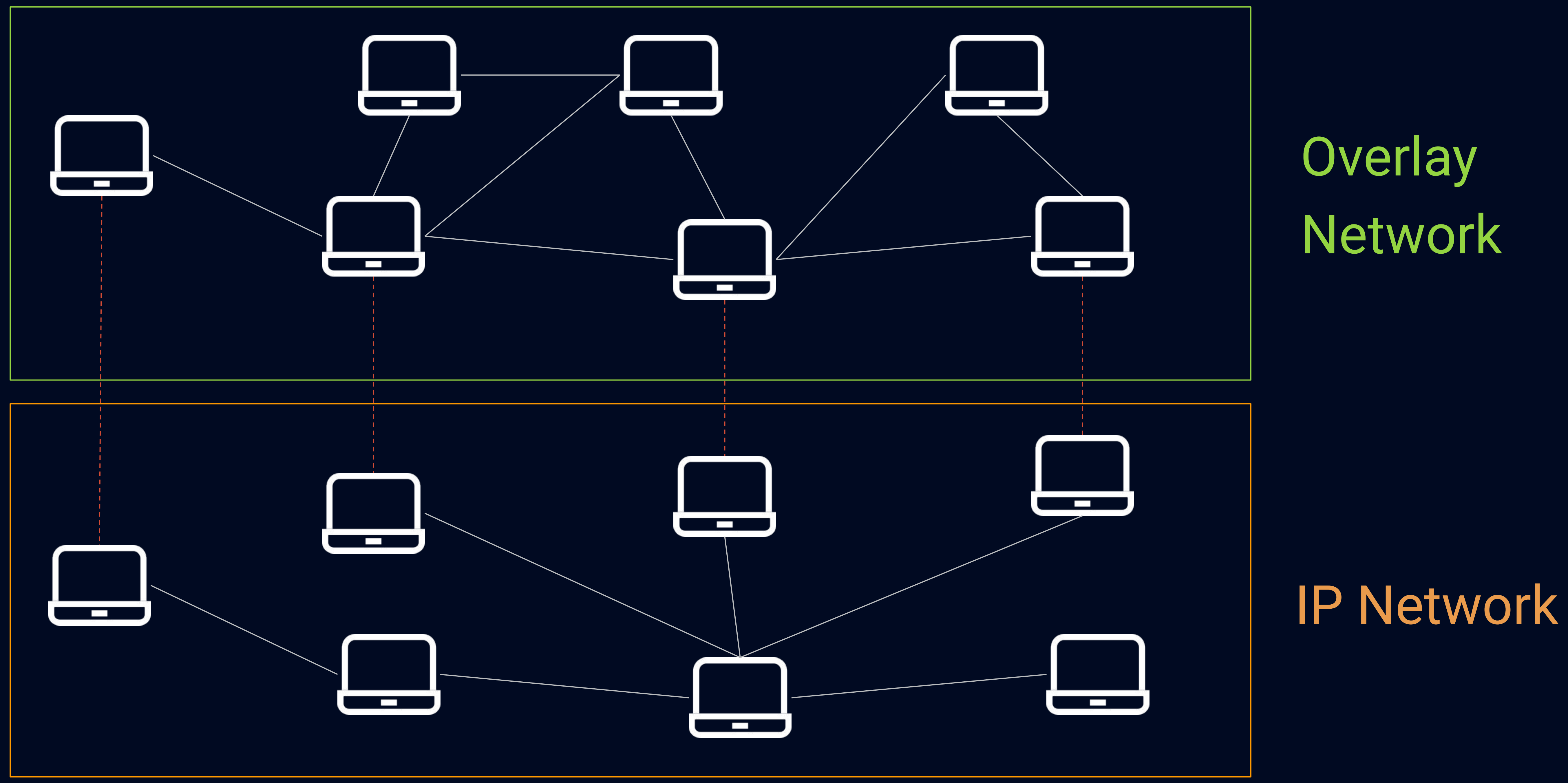
Multi-round iterative lookups

- Content Discovery ***(Resolve)***: Contact k closest peers to the CID. If they have the object they send it back, if not they respond with the provider record.
- Peer Discovery: A peer may not know the multiaddress for the peer in the provider record so it needs to perform a new DHT query to find the peer's network addresses.
    - Routing tables refresh every 10 min. This usually determines if a new walk is needed to get the peer's contact information.
- Peer Routing: Use the multiaddress of the provider to contact it.

Pros and Cons of using a DHT

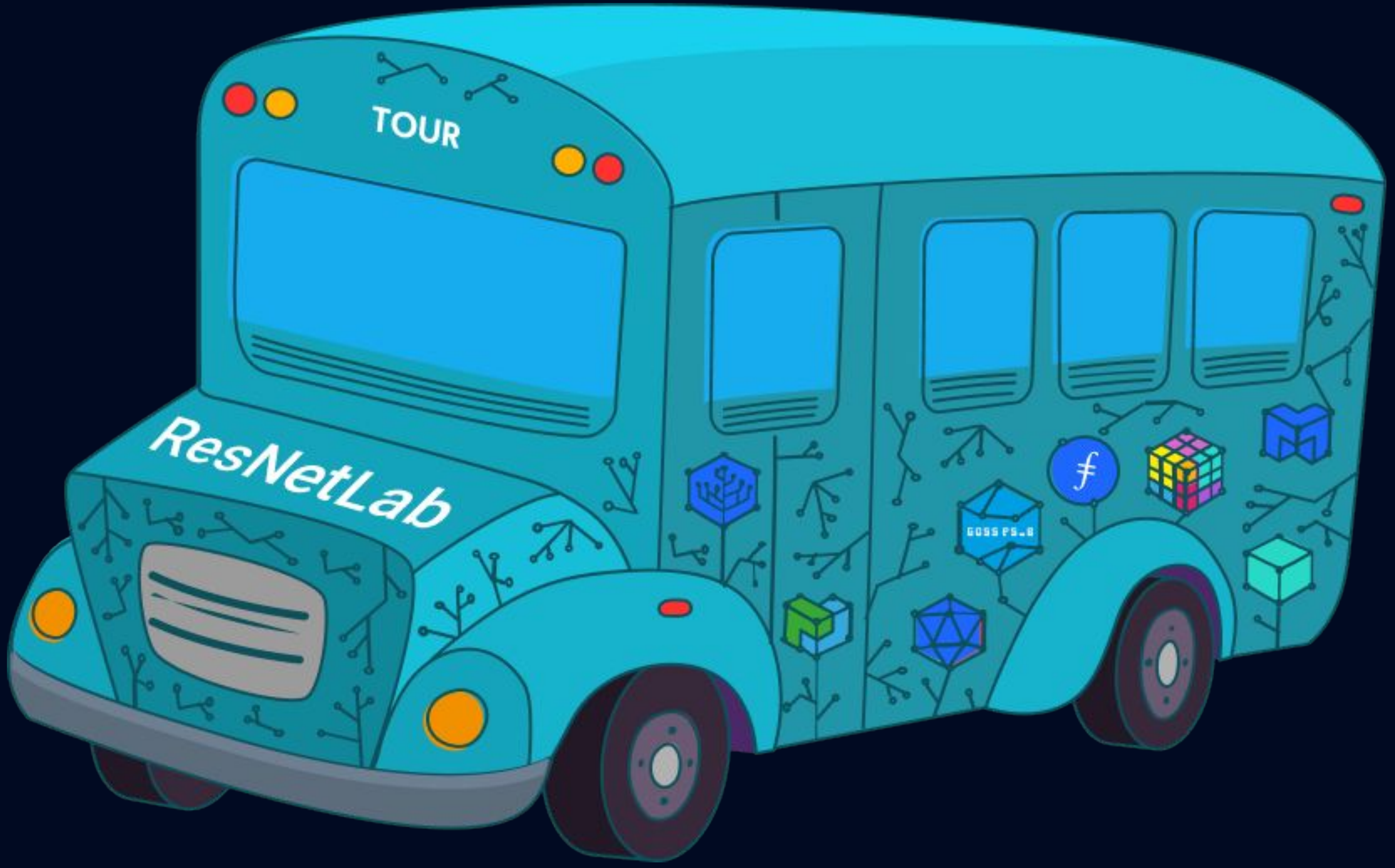
- Fault tolerant. Resistance to churn.
- Finds peers 100% probability (as long as they are reachable).
- Ensures freshness of the routing information.

- Can be slow in network with a large number of peers.
  - Lookup O(logN); may require several hops to find peers.
- DHT proximity ≠ Spatial proximity



Module: Content Exchange

ResNetLab on Tour



IPFS Components

CONTENT ADDRESSING

- Anatomy of the IPFS CID
- Chunking
- Linking Chunks in Merkle DAGs
- From Data to Data Structures with IPLD

CONTENT DISCOVERY & ROUTING

- Routing & Provider Records
- DHT-based Routing
- Gossip-based Routing

CONTENT EXCHANGE

- **Bitswap**
- GraphSync

MUTABLE NAMES & MESSAGE DELIVERY

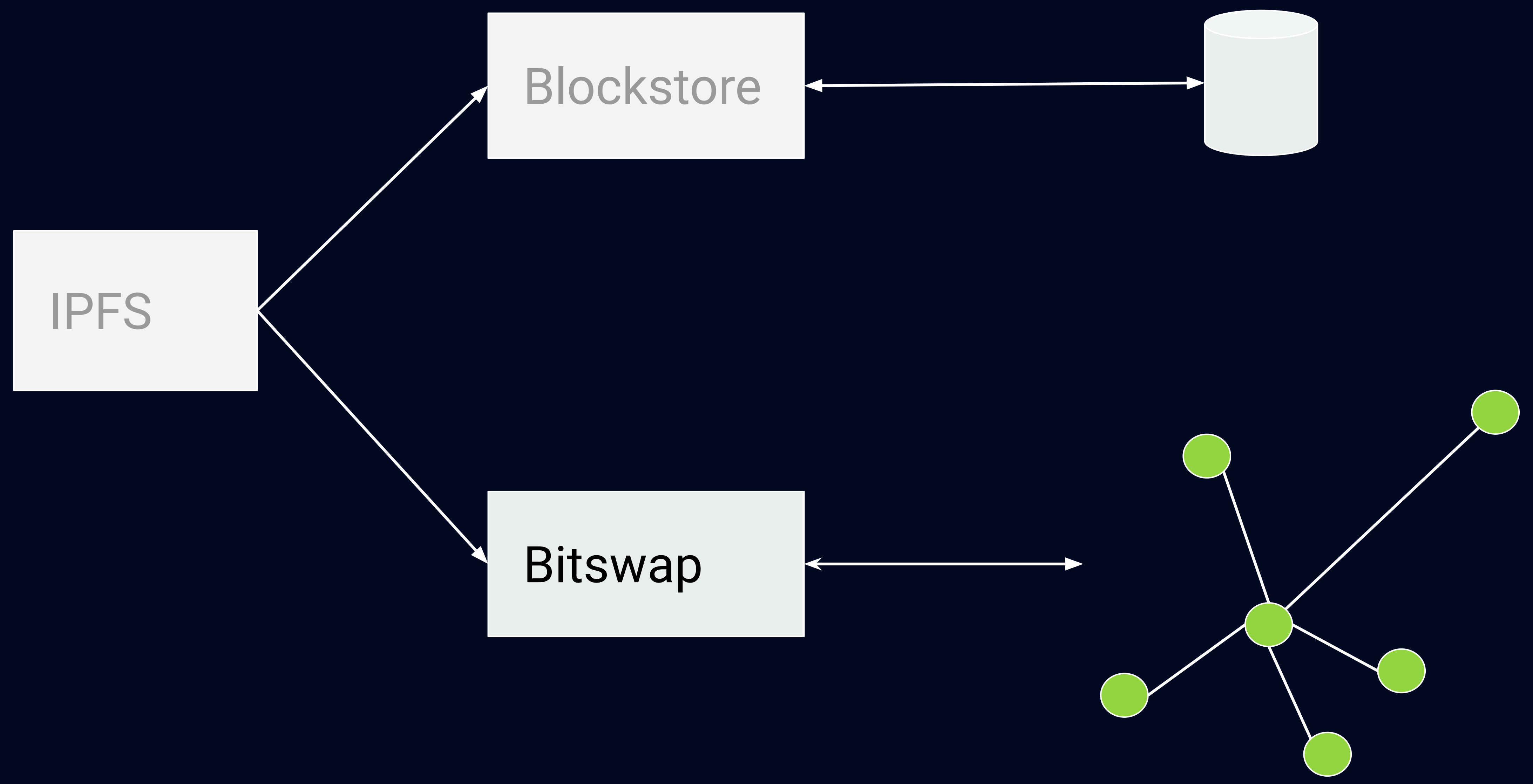
- Dynamic Data
- IPNS
- PubSub
- CRDTs

The IPFS Example

Bitswap Operation

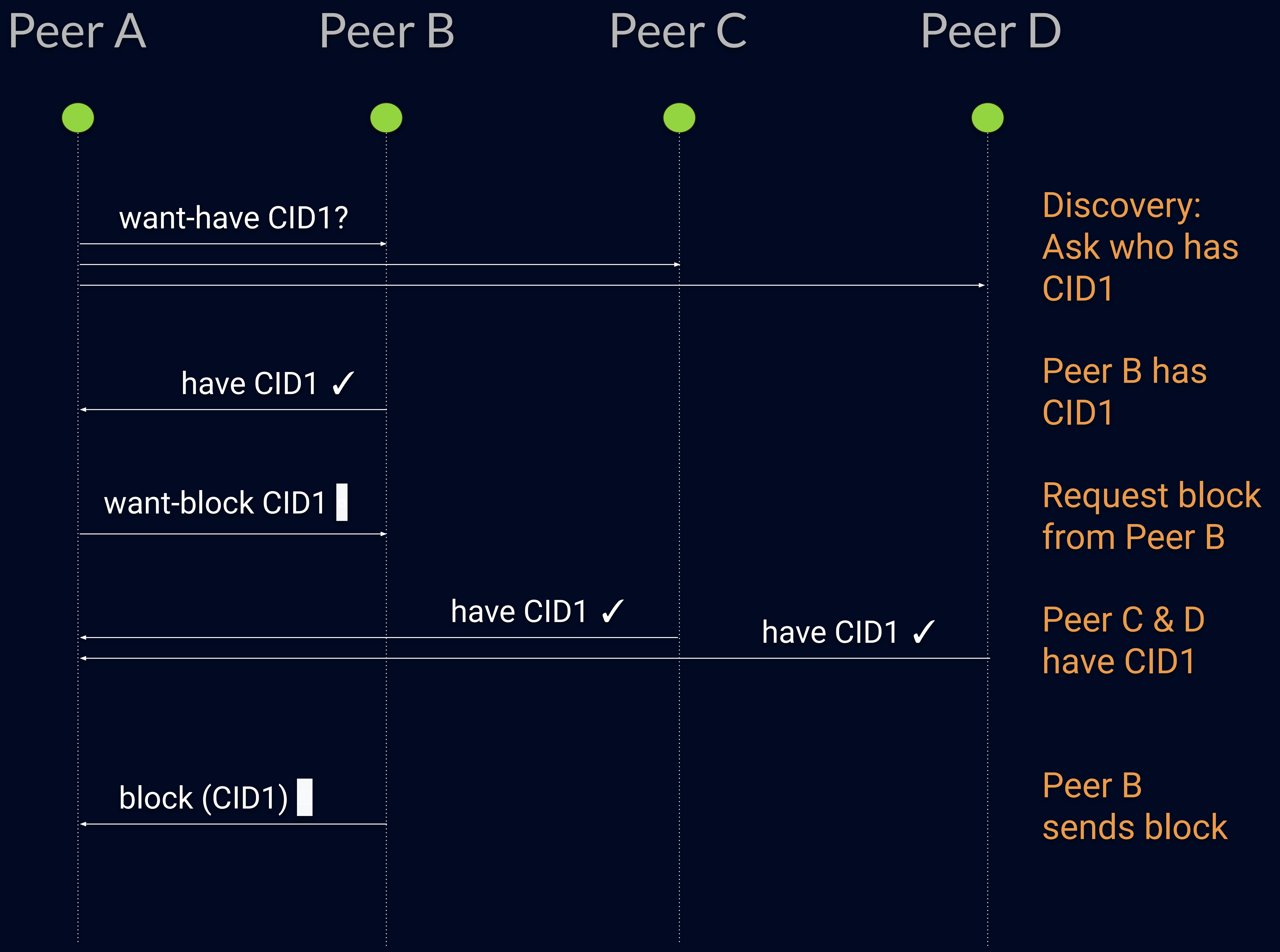
- IPFS asks Bitswap for blocks
- Bitswap fetches blocks from the network

- Message-oriented protocol
  - Requests: WANT-HAVE / WANT-BLOCK / CANCEL
  - Responses: HAVE / BLOCK / DONT_HAVE



Root Block

- **HAVE message**
  - Sometimes we don't want a whole block
  - We just want to know who has a block (eg for discovery)
- **Two kinds of WANT messages**
  - WANT-HAVE
  - WANT-BLOCK
- If the block is small enough, reply with BLOCK instead of HAVE



Discovery: Ask who has CID1

Peer B has CID1

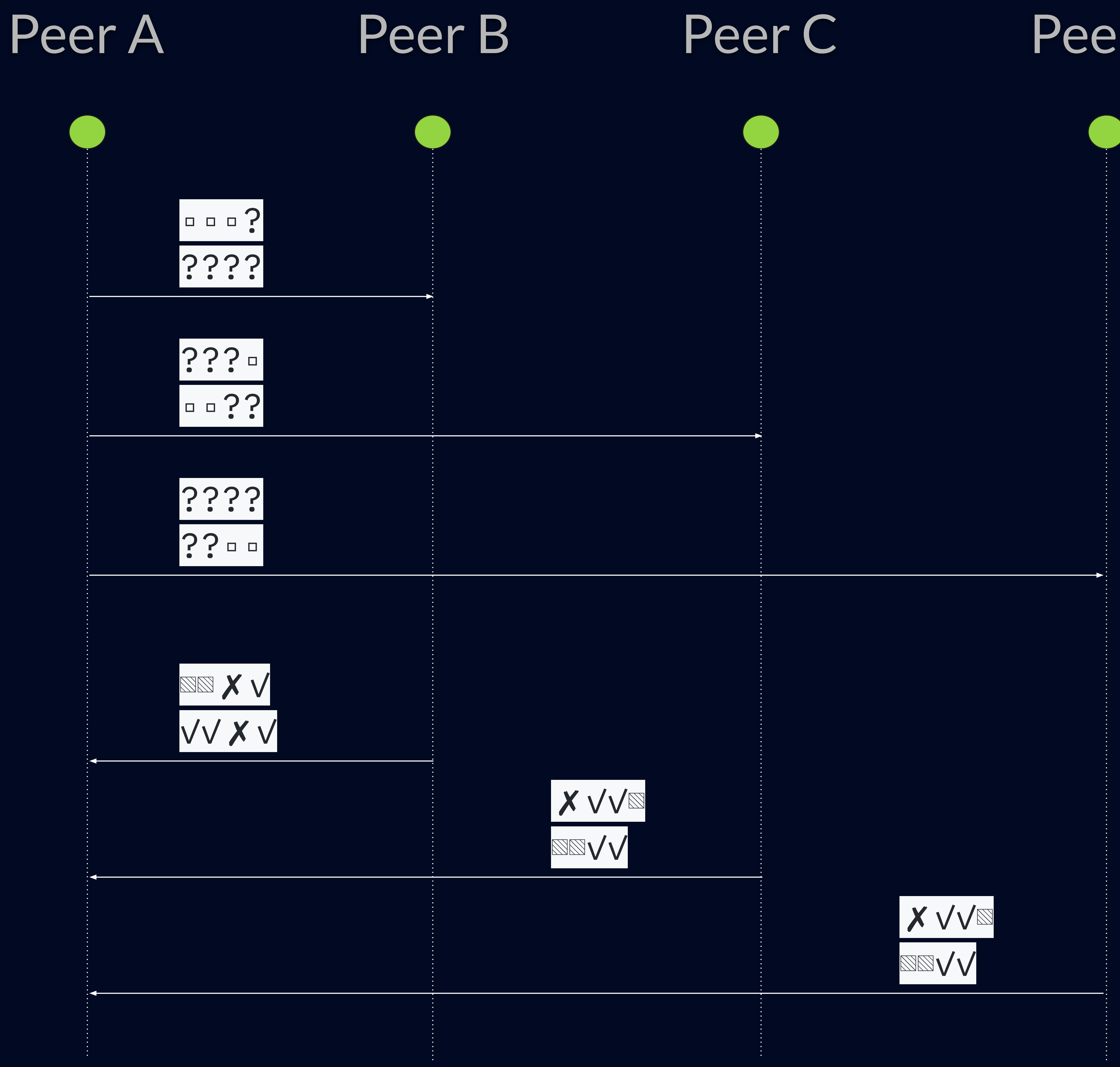
Request block from Peer B

Peer C & D have CID1

Peer B sends block

Subsequent Requests

- DONT_HAVE message
  - Allows peer to indicate that it does not have a block
- Requests:
  - WANT-BLOCK
  - WANT-HAVE
- Respond with combination of
  - HAVE, DONT_HAVE
  - BLOCK



Peer A sends either

□ want-block
? want-have

for each CID that it wants

Peers B, C & D respond with

▨ Block
✓ HAVE
✗ DONT_HAVE

for each CID

THE IPFS STACK

IPFS is the result of combining multiple blocks commonly used to build distributed applications into a distributed-storage application.

IPFS uses libp2p, IPLD and Multiformats to provide content-addressed decentralized storage.

IPFS

LIBP2P

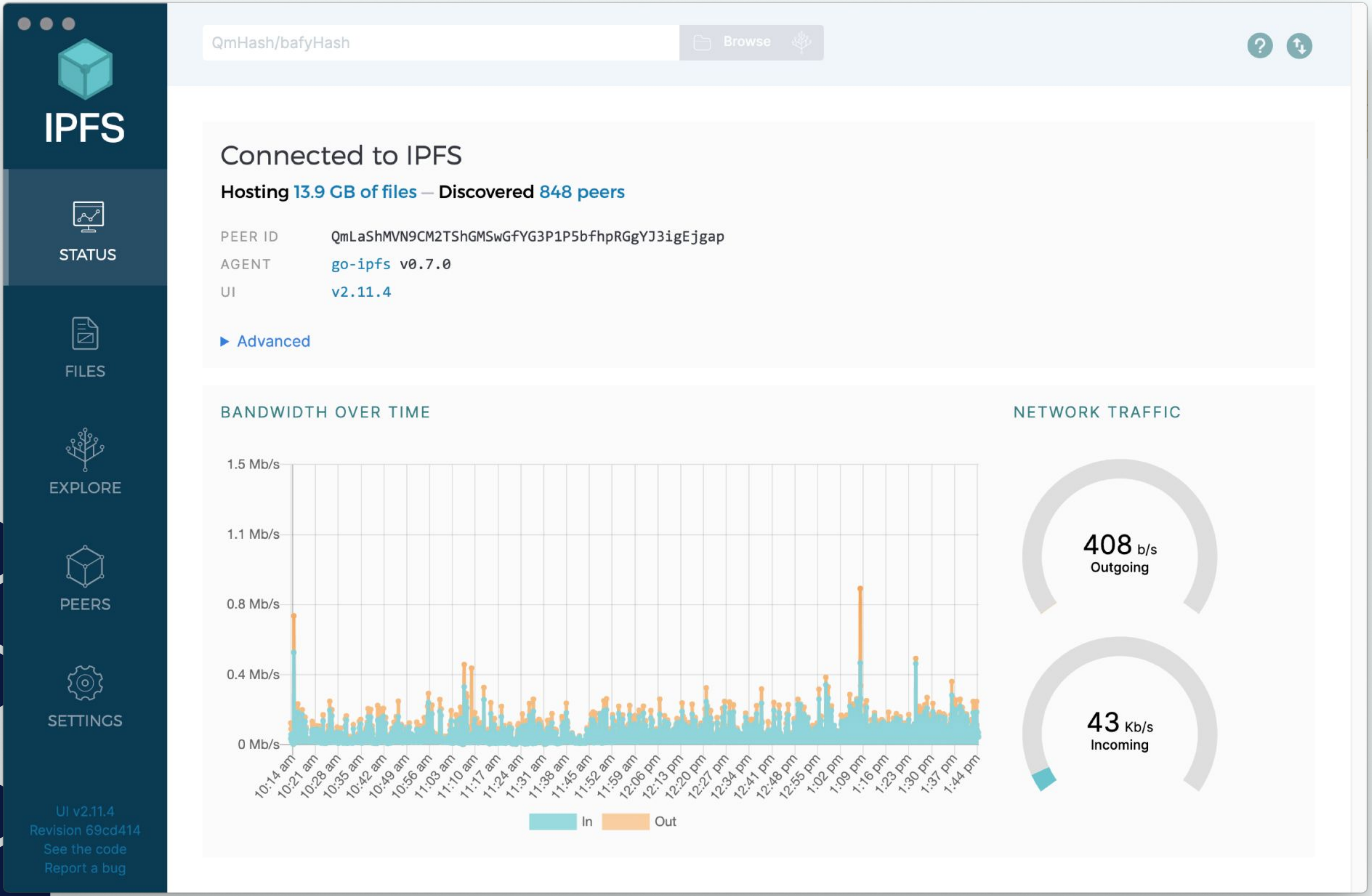
libp2p is the peer-2-peer network-layer stack that supports IPFS. It takes care of host addressing, content and peer discovery through protocols and structures such as DHT and pubsub.

IPLD

IPLD (InterPlanetary Linked Data) provides standards and formats to build Merkle-DAG data-structures, like those that represent a filesystem.

Multiformats

Multiformats provides formatting rules for self-describing values. These values are useful both to the data layer (IPLD) and to the network layer (libp2p)



Connected to IPFS

Hosting 13.9 GB of files — **Discovered** 848 peers

PEER ID QmLaShMVN9CM2TShGMSwGfYG3P1P5bfhpRGgYJ3igEjgap

AGENT go-ipfs v0.7.0

UI v2.11.4

▶ Advanced

BANDWIDTH OVER TIME

NETWORK TRAFFIC

408 b/s Outgoing

43 Kb/s Incoming

UI v2.11.4
Revision 69cd414
See the code
Report a bug

IPFS Desktop

Booming ecosystem of applications

IPFS
Data
orbit db
qri.io
arbore
Catena BLOCKCHAIN EXPLORER
Identity
civic
ION
NOMIOS
ZINC
handshake
Productivity
berty
PeerPad
PEERGOS
Marketplace
OpenBazaar
ORIGIN
Bounty0x
Ethlance
NAME BAZAAR ENS ETH
PORTION
NFT
dlux DECENTRALIZED LIMITLESS USER EXPERIENCES
DIGITAL ART CHAIN
Decentraland
Glossy
mokens
Content
D.tube
EVERIPEDIA
BitTube
UNSTOPPABLE DOMAINS
ALEXANDRIA
dlive
Matters
Viewly
VIULY.COM
DSOUND
ujo
AUDIUS
Other
textile
MÓIBIT
#KarmaPay
TEMPORAL
SprtsHub
Simple As Water
ipwb
WINGS
adchain
MONITOR CHAIN
IPSE
I K U
Dappkit
FILESTORM
点对点科技 DECENTRAL TECH
kauri.io
INFURA
Social Media
3 BOX
AKASHA
Indorse
Identifi
KARMA
Peepeth
busy
Governance
GovBlocks
Democracy Earth
ARAGON ONE
Exchange
Dether
faa.st
Swap .online
Finance
REQUEST NETWORK
Bloom
colu.
SETTLE
MARKET PROTOCOL
Integrations & Collabs
openSUSE
Guix
NETFLIX
CLOUDFLARE
Prediction and betting
AUGUR
CryptoBets
PLAY 2 WIN
VIRTUE POKER
Crypto Dice
MÖBIUS 2D

Earn Filecoin for hosting files

The time to earn has arrived. Now anyone can become a cloud storage provider and make money from open hard drive space.

ResNetLab On Tour 📼

- **Unlimited free access to the content:** https://research.protocol.ai/tutorials/resnetlab-on-tour/
- **5 Core Modules** and over **8 Elective Modules** to be released over time
- Core Modules designed to equip you with everything in order to understand
  - **Content Addressing**
  - **Content Routing**
  - **Exchange of Content**
  - **Mutable Content**
- If you are an event organizer and/lecturer, feel empowered to take away the materials and **organize your own local event**! Let us know if you need help.

A Few Pointers

- **Docs:** https://docs.ipfs.io
- **Video tutorials:** https://research.protocol.ai/tutorials/resnetlab-on-tour/
- **Interactive Coding and Non-Coding Tutorials:** https://proto.school
- **Discussion Forums:**
  - **IPFS:** https://discuss.ipfs.io
  - **libp2p:** https://discuss.libp2p.io

The Ecosystem

→ An arsenal of projects and platforms for **experimentation, research and development.**

→ **A great community** to collaborate with.

→ **Top quality research teams** to inspire and get inspired from.

→ Many **collaboration opportunities.**

→ **Exciting challenges** to overcome.

→ Lots of **open positions** and **funding opportunities.**

ResNetLab

Get in touch!
yiannis@protocol.ai

resnetlab@protocol.ai   https://github.com/protocol/ResNetLab/discussions