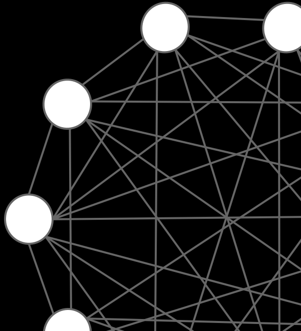
₿6.15

@anilsaidso

Bitcoin Terminology



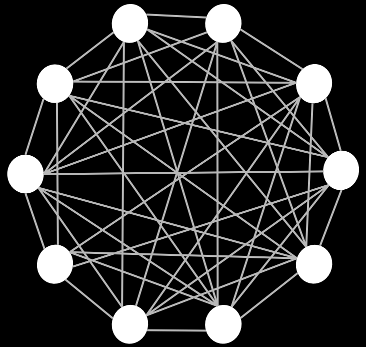
Contents

- ₿ Bitcoin
- ⚡ Lightning Network
- N.O.S.T.R.

₿6.15

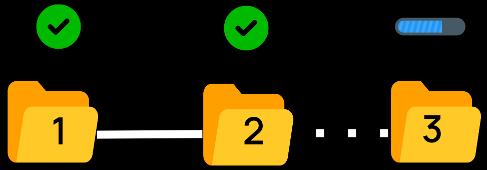
bitcoin

money



the network

connected nodes



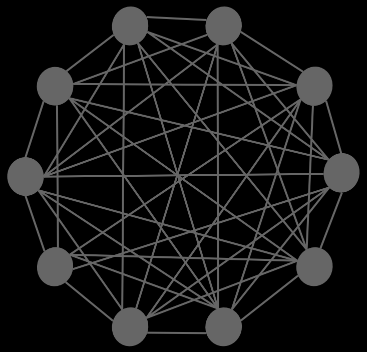
the blockchain

linked record of verified tx’s

₿6.15

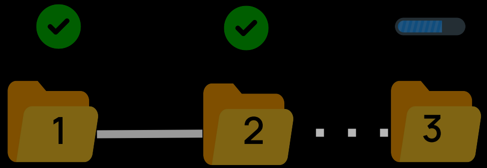
bitcoin

money



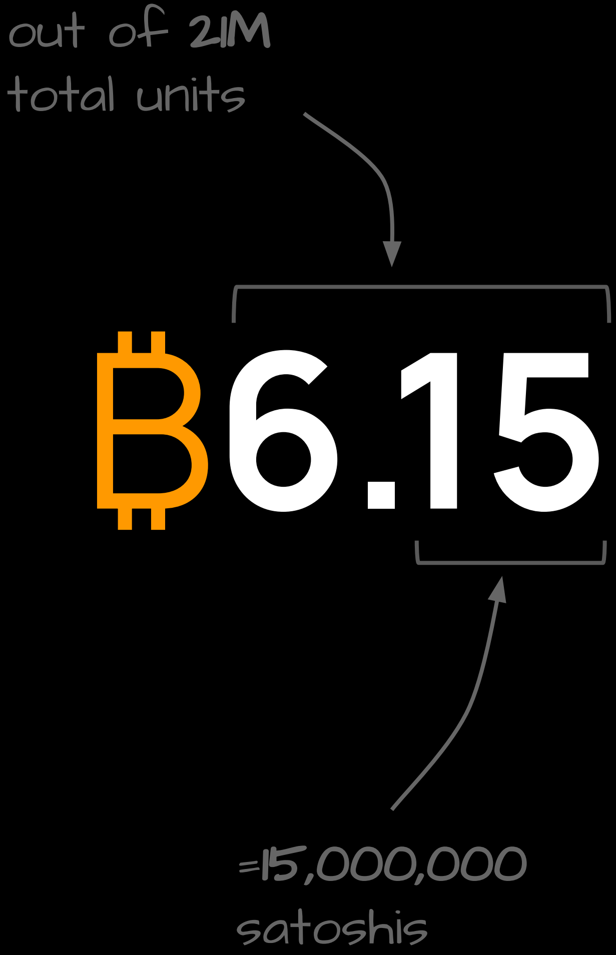
the network

connected nodes



the blockchain

linked record of verified tx’s



out of 21M total units

₿6.15

=15,000,000 satoshis

bitcoin

unit of account

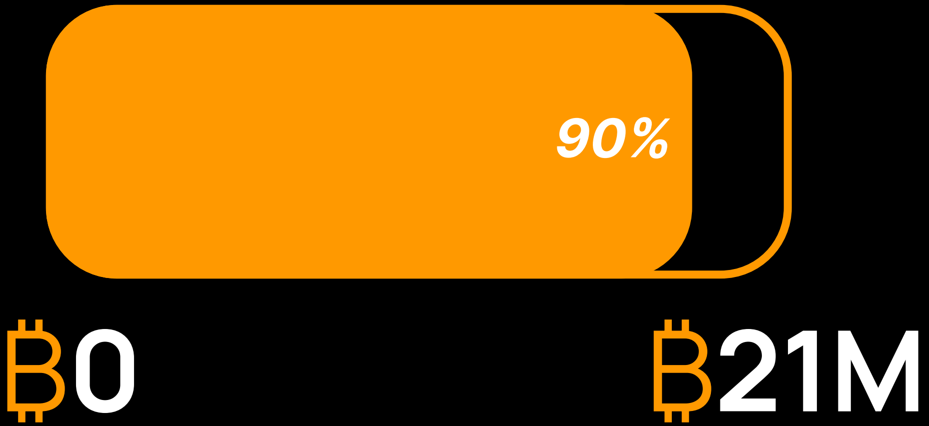
native to the bitcoin blockchain

satoshi

100,000,000
sats

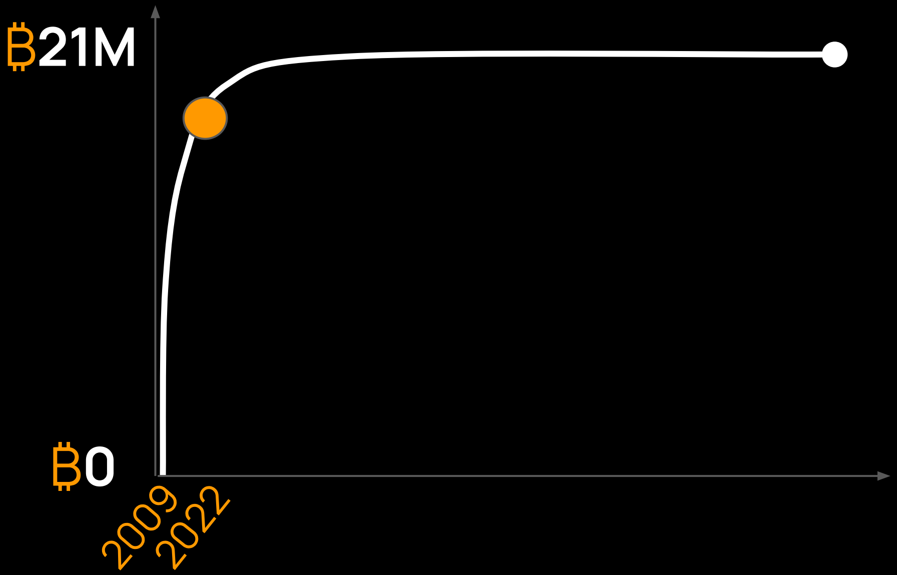
1 BTC

A bitcoin is divisible into 100 million smaller units called satoshis (or *sats*)



terminal supply

The maximum amount of bitcoin that will ever exist once all has been mined



supply schedule

pre-programmed timetable for issuance of new bitcoin

₿50

₿25

₿12.5

➡ **₿6.25**

₿3.125

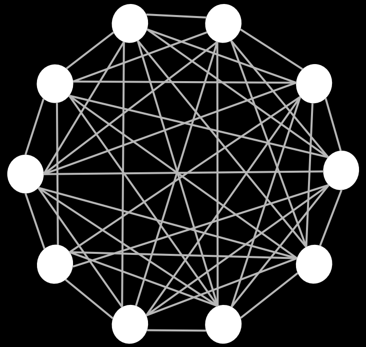
halving

every 210,000 blocks (~4yrs), the rate of new bitcoin issued per block is reduced by 50%

₿6.15

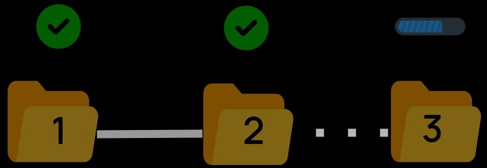
bitcoin

money



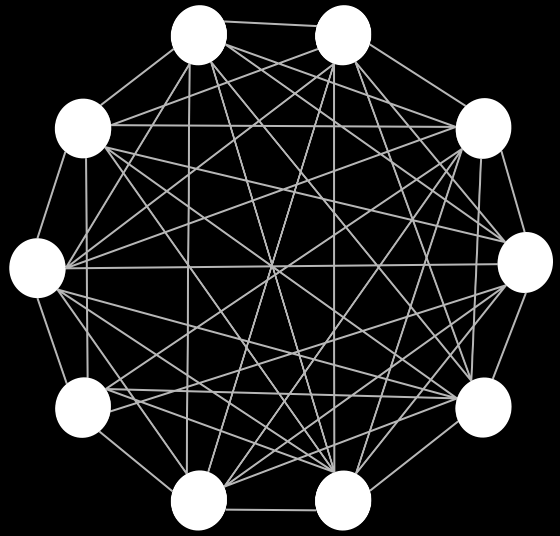
the network

connected nodes



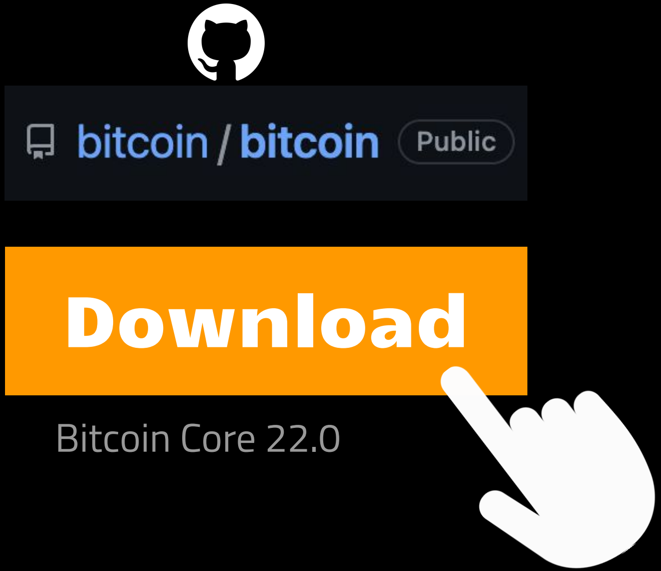
the blockchain

linked record of verified tx's



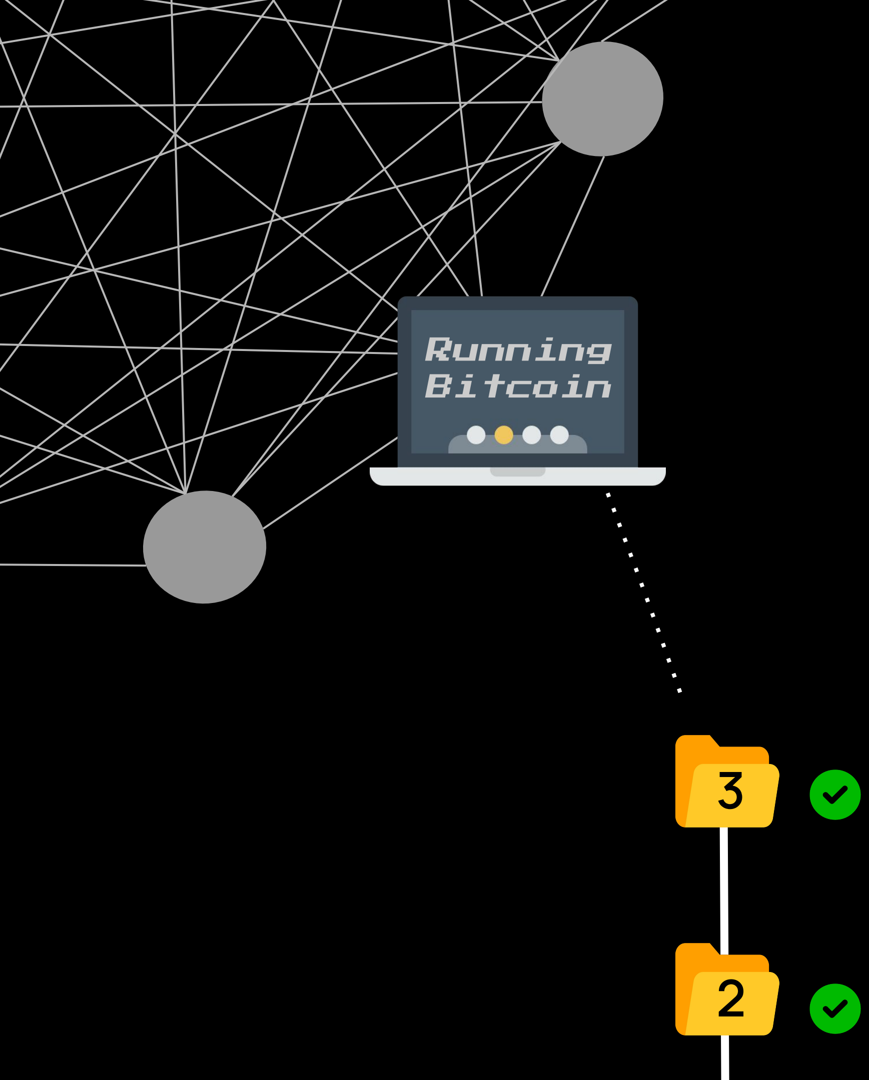
bitcoin's network

connected nodes following a common set of rules



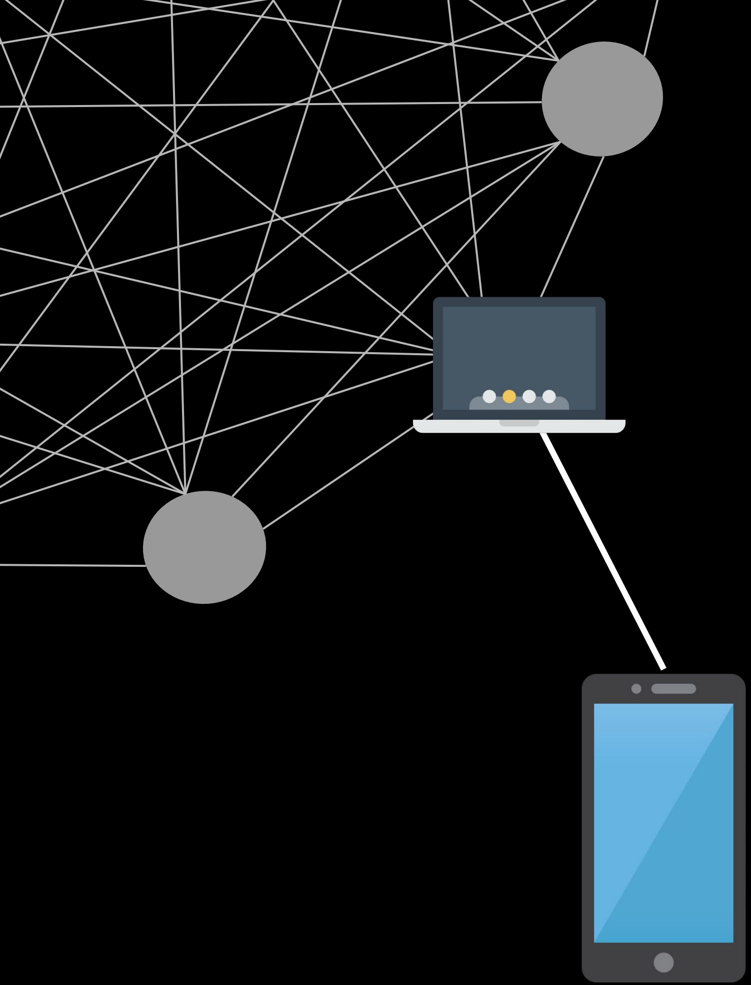
bitcoin software

open-source software that codifies the ruleset



full node

- runs bitcoin software
- maintains a complete copy of the blockchain
- enforces the network’s rules



light client

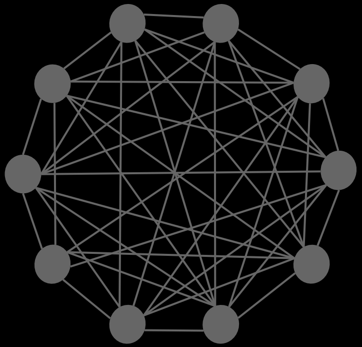
connects to a full node to interact with the network

stores only partial records to save on disk space

₿6.15

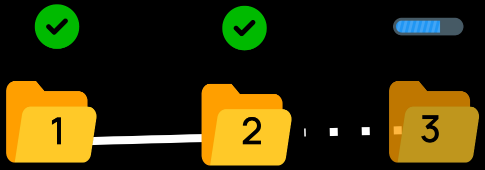
bitcoin

money



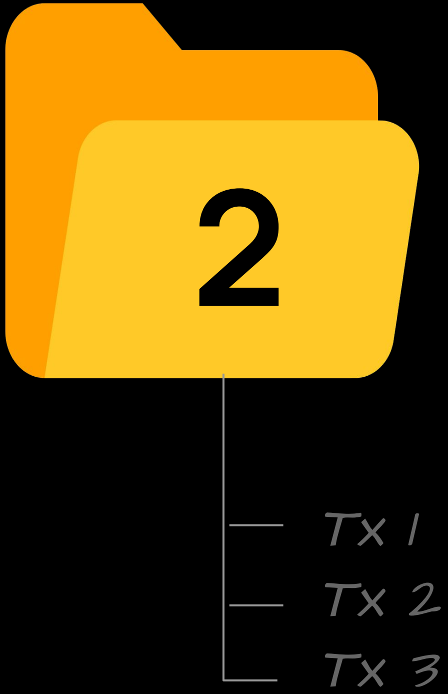
the network

connected nodes



the blockchain

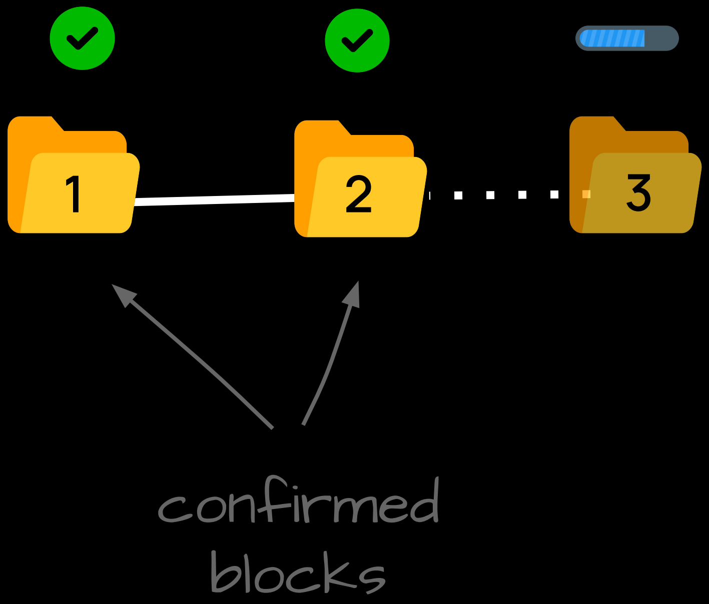
linked record of verified tx's



block

time-stamped batch of confirmed transactions

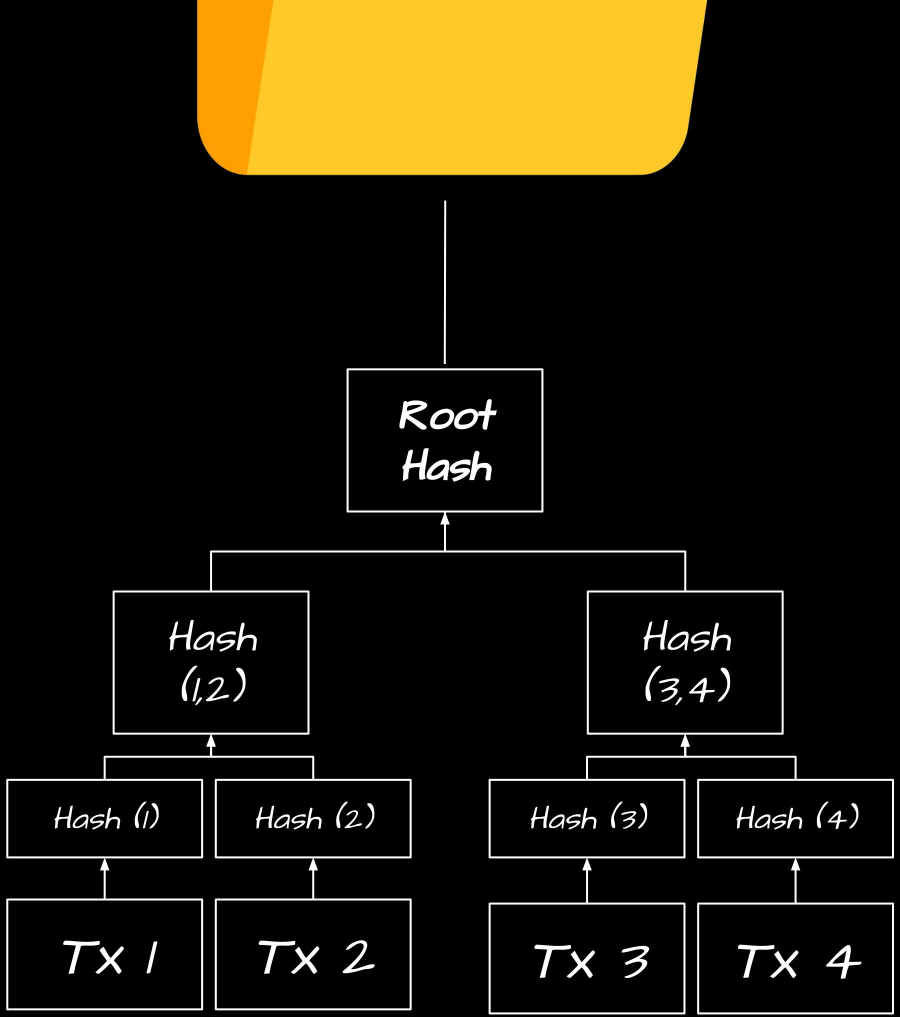
every 10 minutes on avg.



bitcoin's blockchain

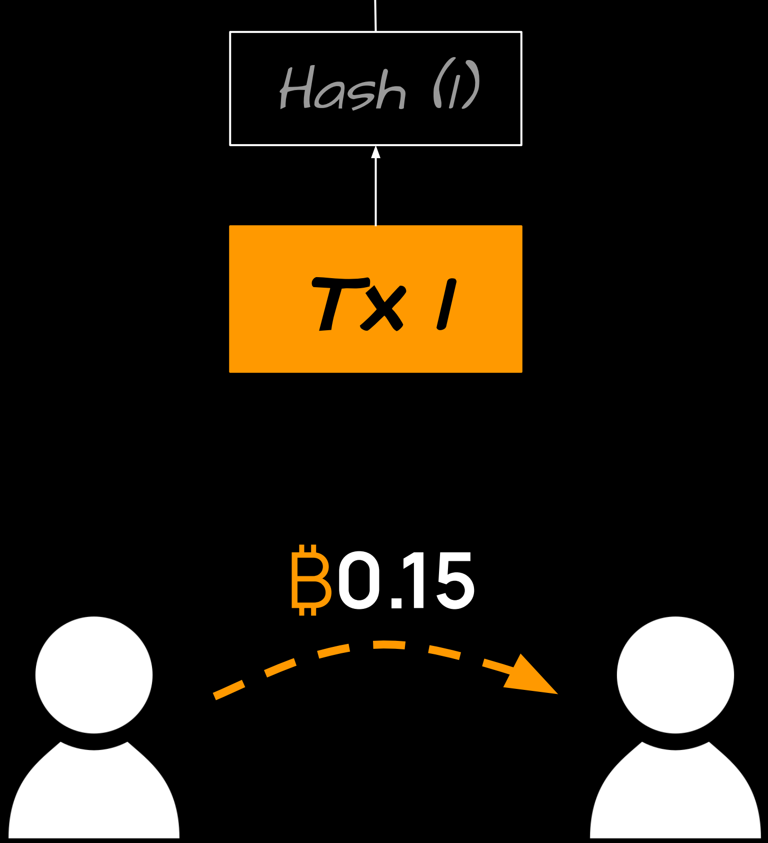
sequentially-linked blocks

historical record of all confirmed transactions



Merkle tree

data structure that helps reduce storage space and easily prove transaction validity



transaction

transfer of ownership of bitcoin between network participants

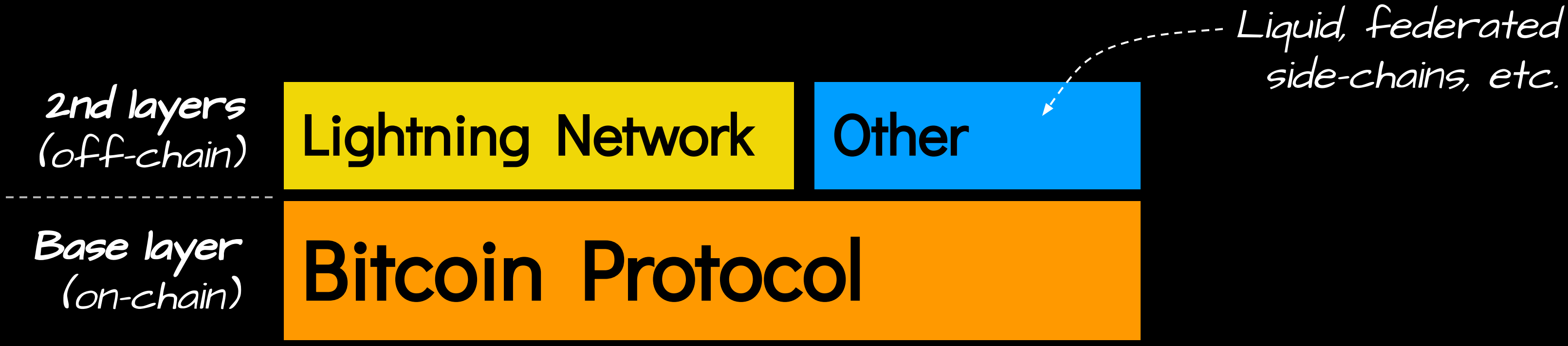
cryptographically signed by the sender

Lightning Network Basics

@anilsaidso

Lightning Network

Protocol that enables scalability via instant off-chain payments.



2nd layers (off-chain)

Lightning Network

Other

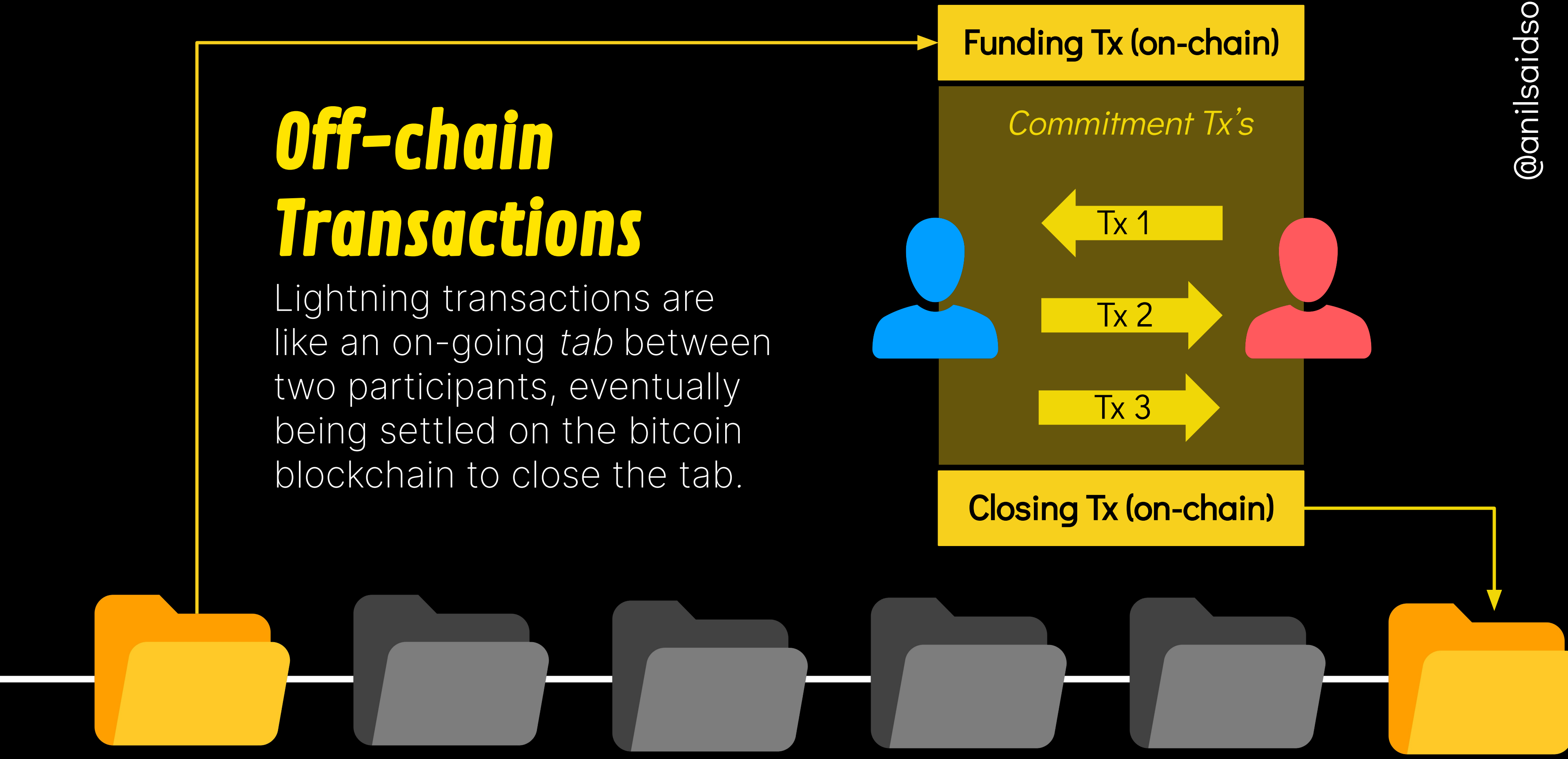
Liquid, federated side-chains, etc.

Base layer (on-chain)

Bitcoin Protocol

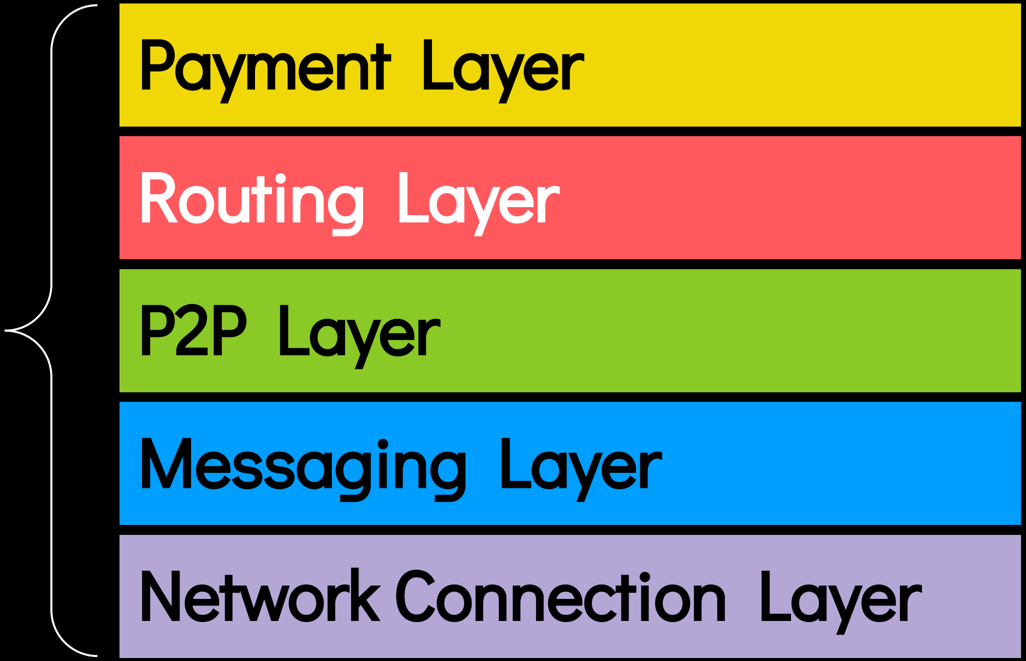
Off-chain Transactions

Lightning transactions are like an on-going *tab* between two participants, eventually being settled on the bitcoin blockchain to close the tab.



Lightning Network

The LN protocol suite is comprised of **five layers**



Payment Layer

Routing Layer

P2P Layer

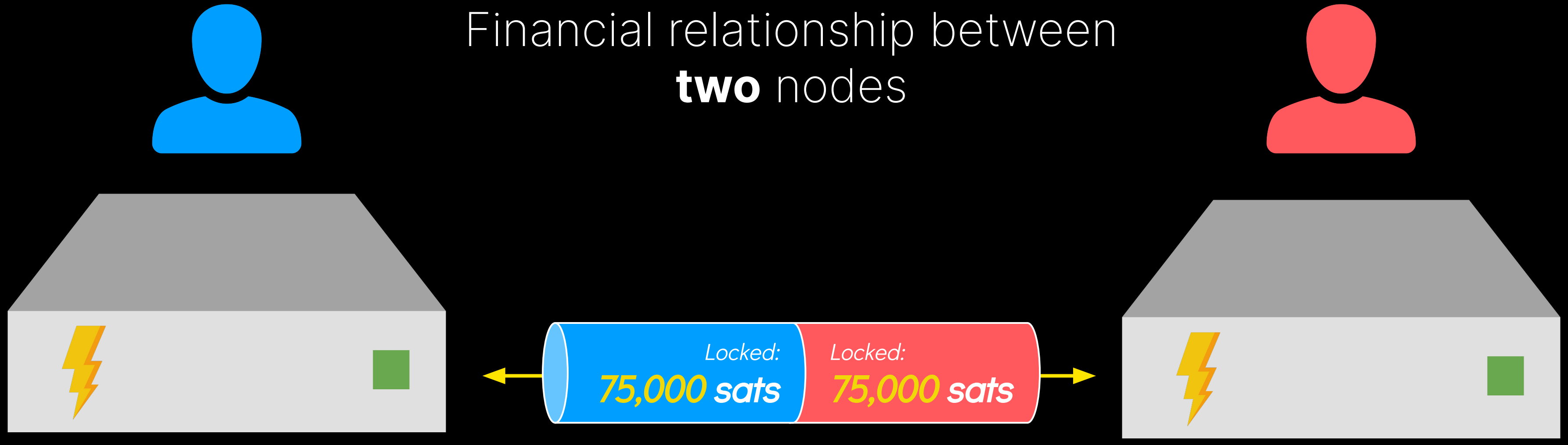
Messaging Layer

Network Connection Layer

@anilsaidso

Payment Channel

Financial relationship between **two** nodes

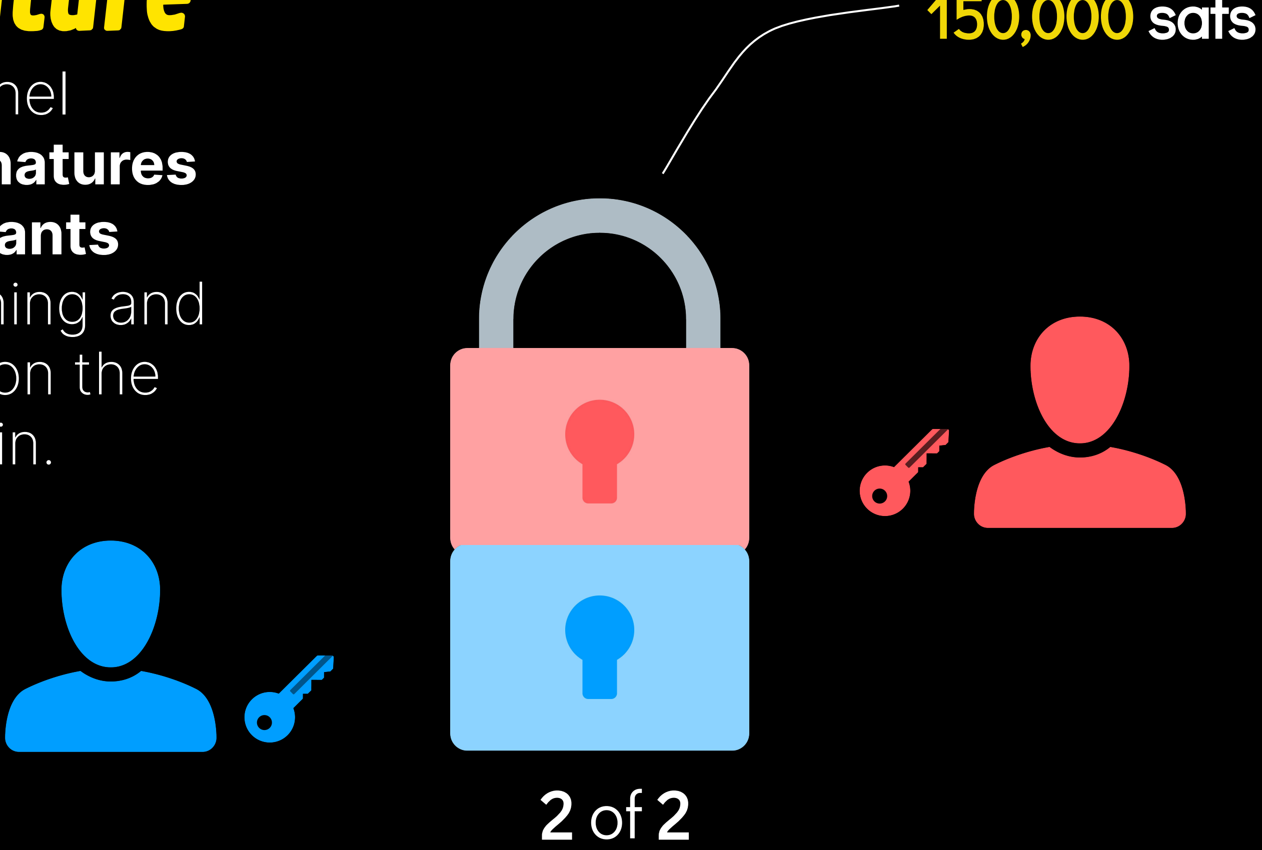


Total Channel Capacity:

150,000 sats

Multisignature

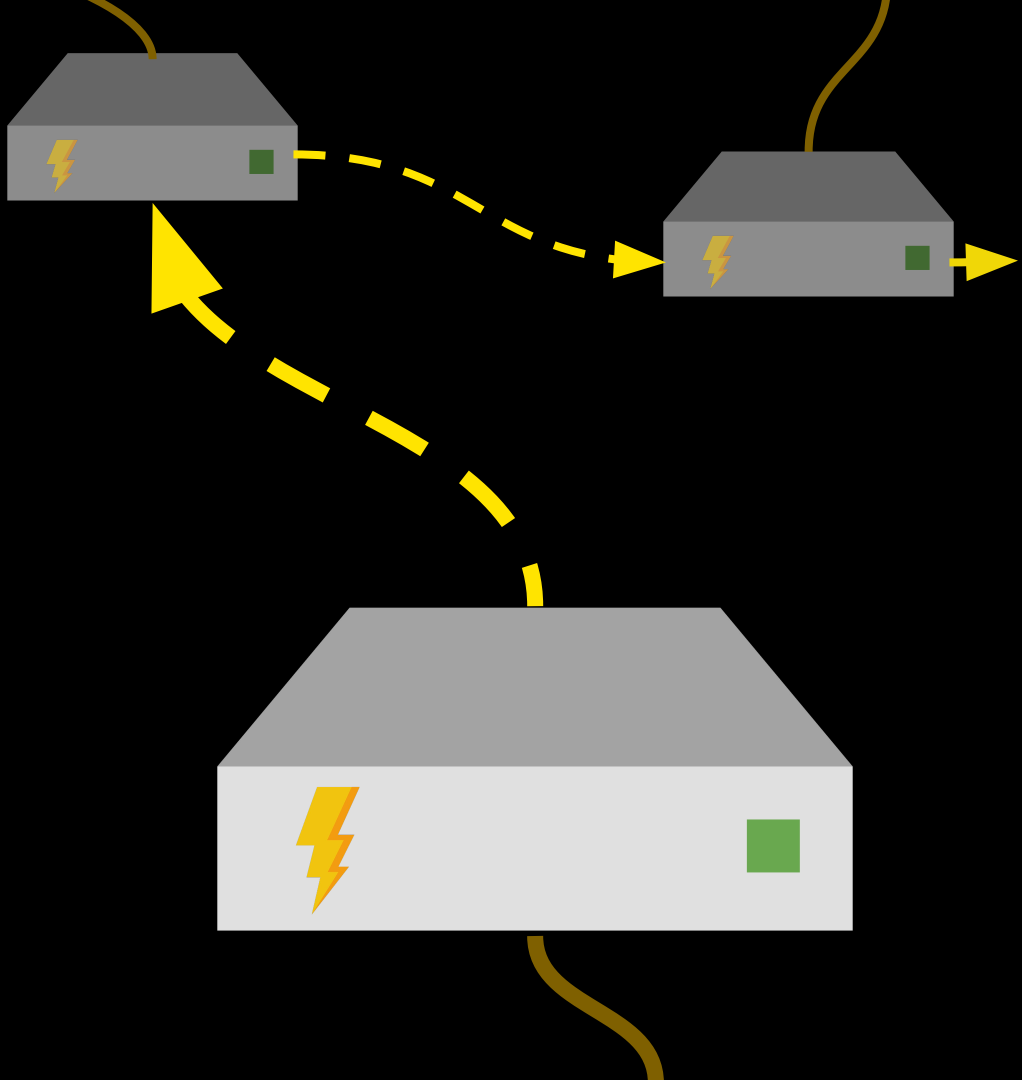
A payment channel requires the **signatures of both participants** (2-of-2) for opening and final settlement on the bitcoin blockchain.



@anilsaidso

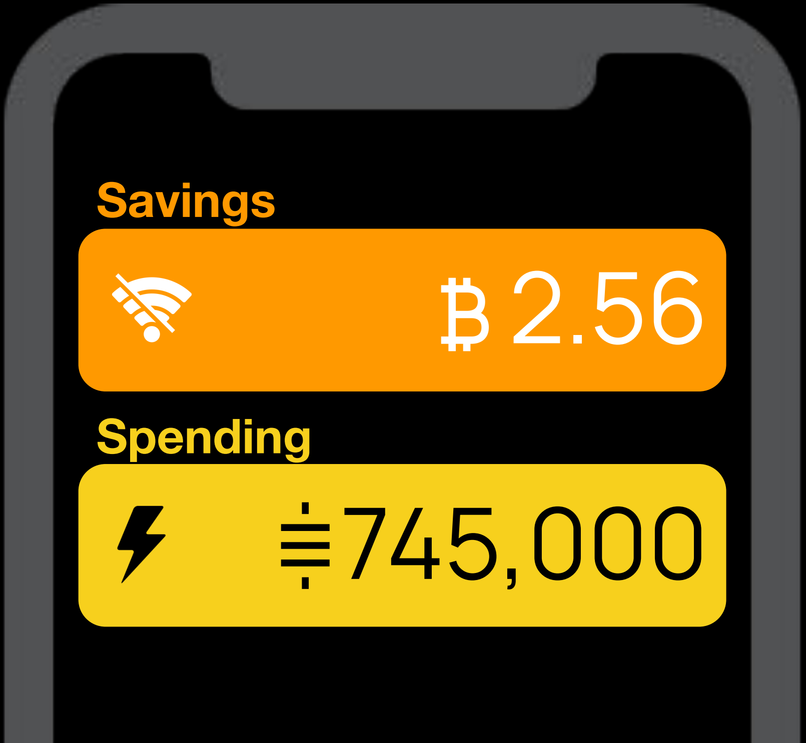
Payment Routing

Lightning payments occur off-chain, hence all payments must be **forwarded** (*routed*) to their final destination



Lightning Wallet

A lightning wallet is **always online**. It should not be used to store large amounts. Only top up your lightning wallet with funds that you plan to spend in the near future.



@anilsaidso

NOSTR

Basics

@anilsaidso

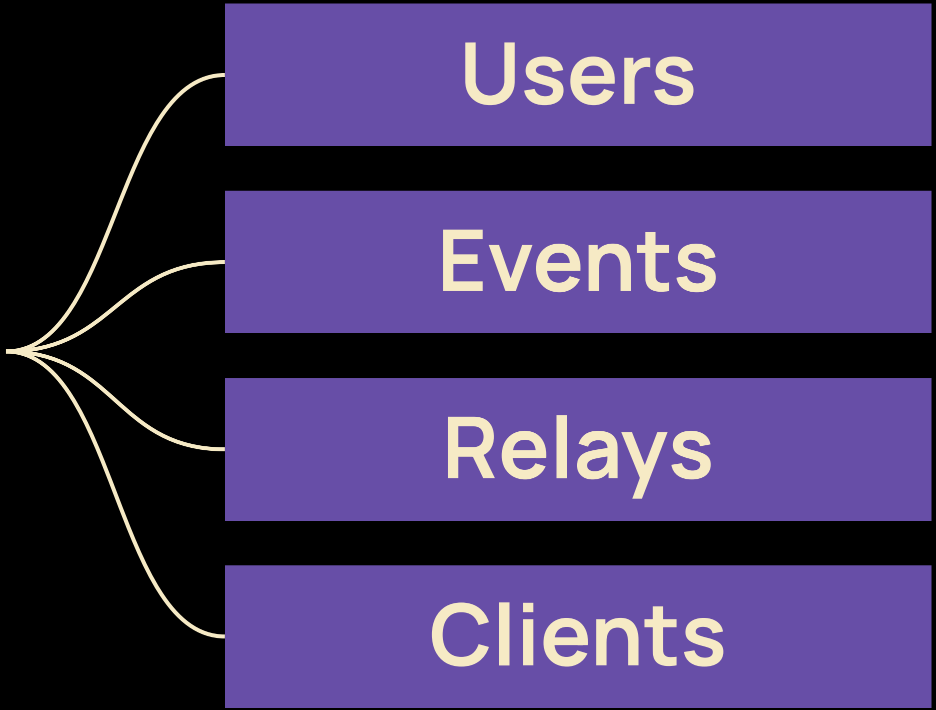
N.O.S.T.R.

Notes and Other Stuff Transmitted by Relays

An **open protocol** for censorship-resistant communication networks created by @fiatjaf

@anilsaidso

WHAT MAKES UP NOSTR



- Users
- Events
- Relays
- Clients

👤 Users

Similar to the bitcoin protocol, nostr is **permissionless**.

To use the protocol a user generates a key pair:
public key & **private key**

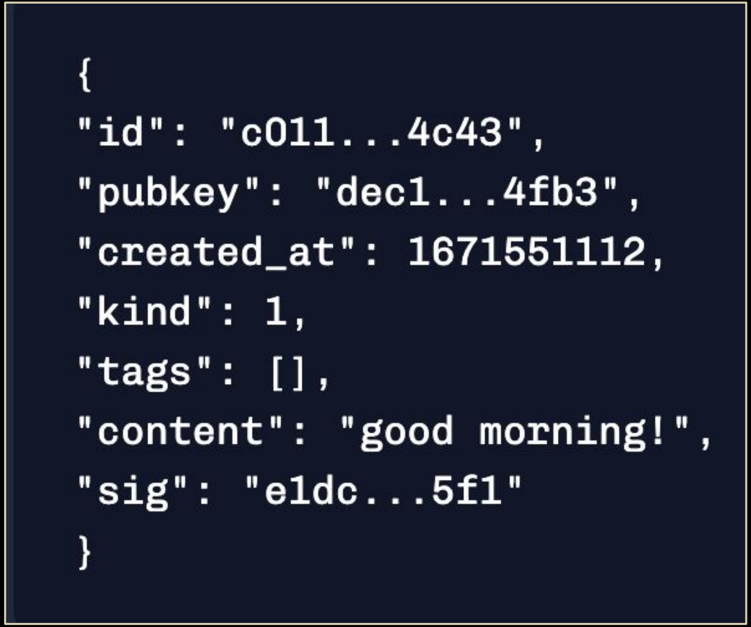
Public Key
Like a username, it’s how others can find you.

Private Key
Like a password, it’s used for signing messages to prove authenticity.
***DO NOT SHARE**

✏️ Events

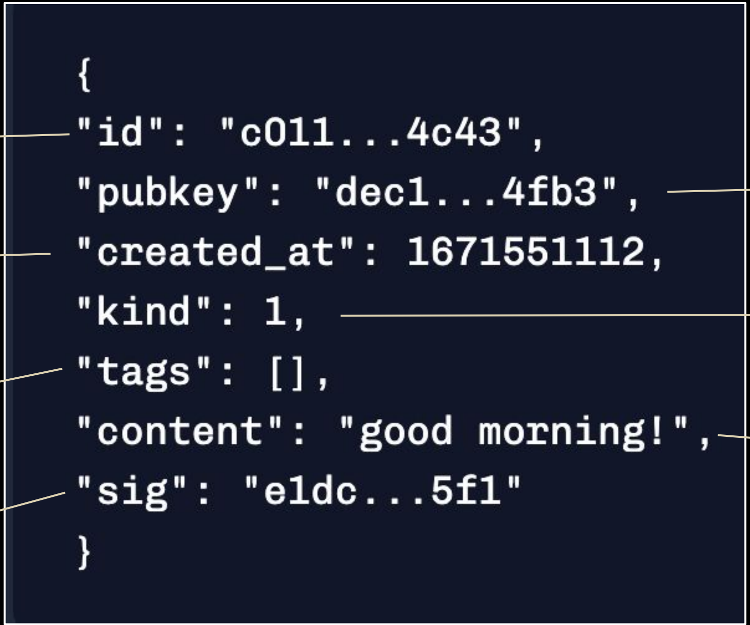
Nostr is a protocol for **packaging** simple text-based objects.

These are called *events*.



```
{
"id": "c011...4c43",
"pubkey": "dec1...4fb3",
"created_at": 1671551112,
"kind": 1,
"tags": [],
"content": "good morning!",
"sig": "e1dc...5f1"
}
```

Anatomy of an Event



```
{
"id": "c011...4c43",
"pubkey": "dec1...4fb3",
"created_at": 1671551112,
"kind": 1,
"tags": [],
"content": "good morning!",
"sig": "e1dc...5f1"
}
```

Event's unique identifier

User

Timestamp

Type of event *(e.g. plain text note, E2E msg, etc.)*

Tags
(e.g. specify post as a reply to event XYZ)

Message

Proof of authenticity
(signed with user's private key)

💻 Relays

Posting content is not broadcast to all users, nor sent directly to a particular recipient (P2P).

Instead, it is sent to a **relay server**, readable by users also connected to that common relay.

Relays can be public/private, free/paid, or application-specific.

@anilsaidso

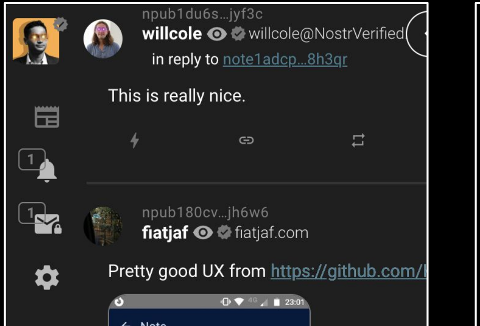
Clients

Users interact with the nostr protocol through a *client*.

You can use any client you wish or even build your own.

Mobile

Web *(browser)*

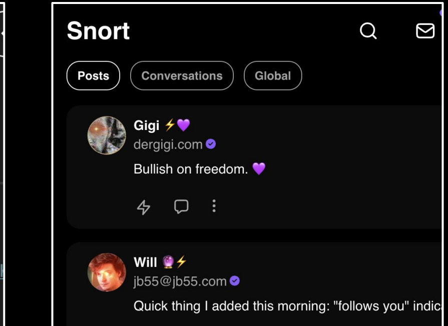


npub1du6s...jyf3c
willcole willcole@NostrVerified
in reply to note1adcp...8h3qr

This is really nice.

npub180cv...jh6w6
fiatjaf fiatjaf.com

Pretty good UX from https://github.com/



Snort

Gigi ⚡💜
dergigi.com

Bullish on freedom. 💜

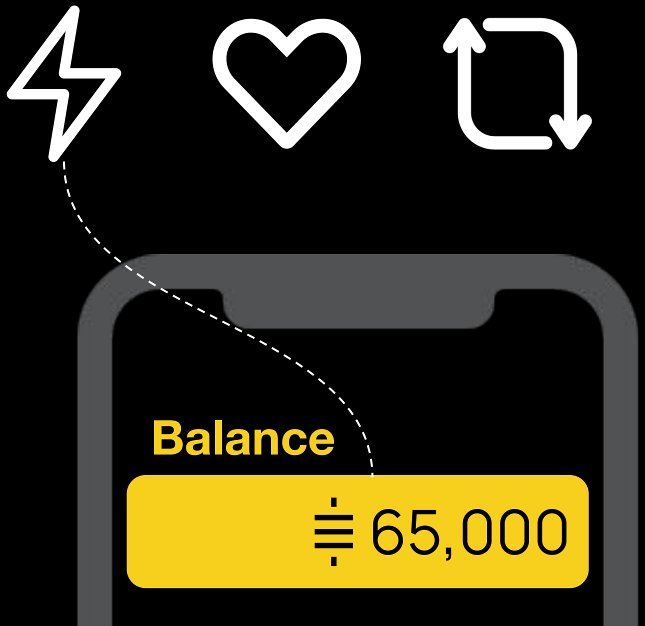
Will 🔮⚡
jb55@jb55.com

Quick thing I added this morning: "follows you" indic

Zaps

As an open protocol, Nostr is interoperable with *other* open protocols such as Lightning.

When using compatible clients, users can show their appreciation for content by *zapping* a post (tipping in bitcoin).



Anil

@anilsaidso