

# Upcoming Bitcoin Protocol Improvements

Gloria Zhao

6B00 2C6E A3F9 1B1B 0DF0 C9BC 8F61 7F12 00A6 D25C

“upcoming” = ?

“protocol improvements” = ?



“upcoming” = next year?

“protocol improvements” = soft forks?



**Soft forks planned for activation this year:**



**Thanks!**

**just kidding.**



“protocol improvements” = ?

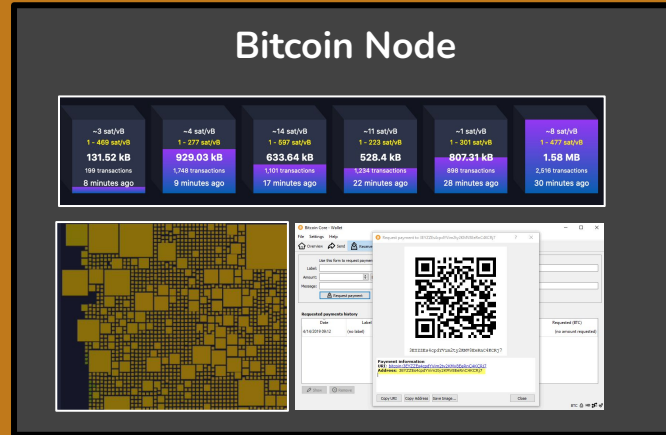
# What is a Bitcoin node?



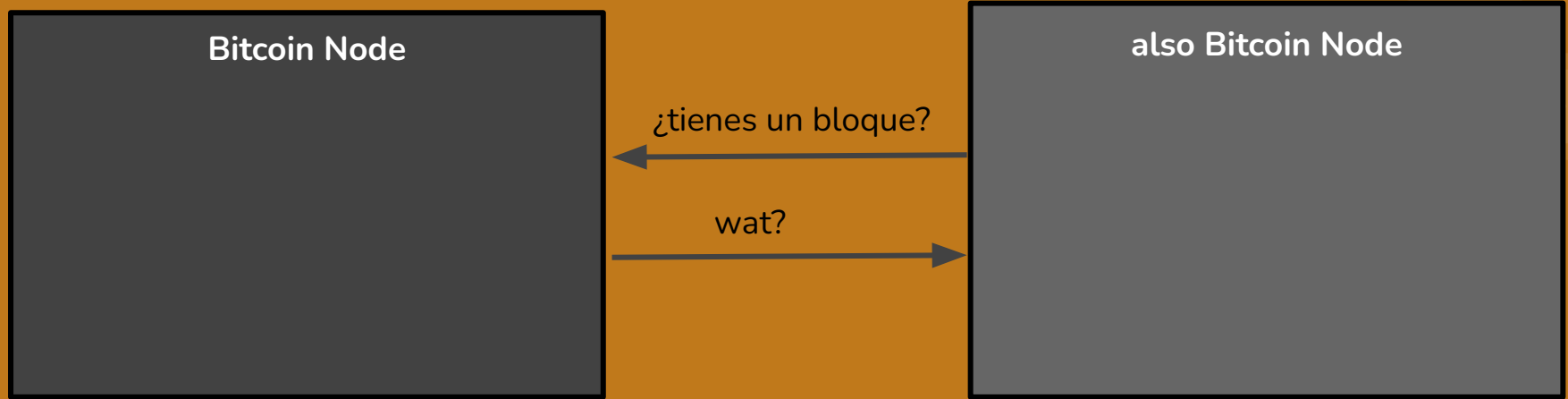
Bitcoin Node



# Does it need a block chain? A mempool? A wallet?



# Bitcoin node = speaks Bitcoin p2p protocol

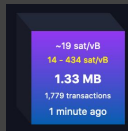


# Bitcoin node = speaks Bitcoin p2p protocol



# Bitcoin node = speaks Bitcoin p2p protocol

## Bitcoin Node



send BLOCK to peer4

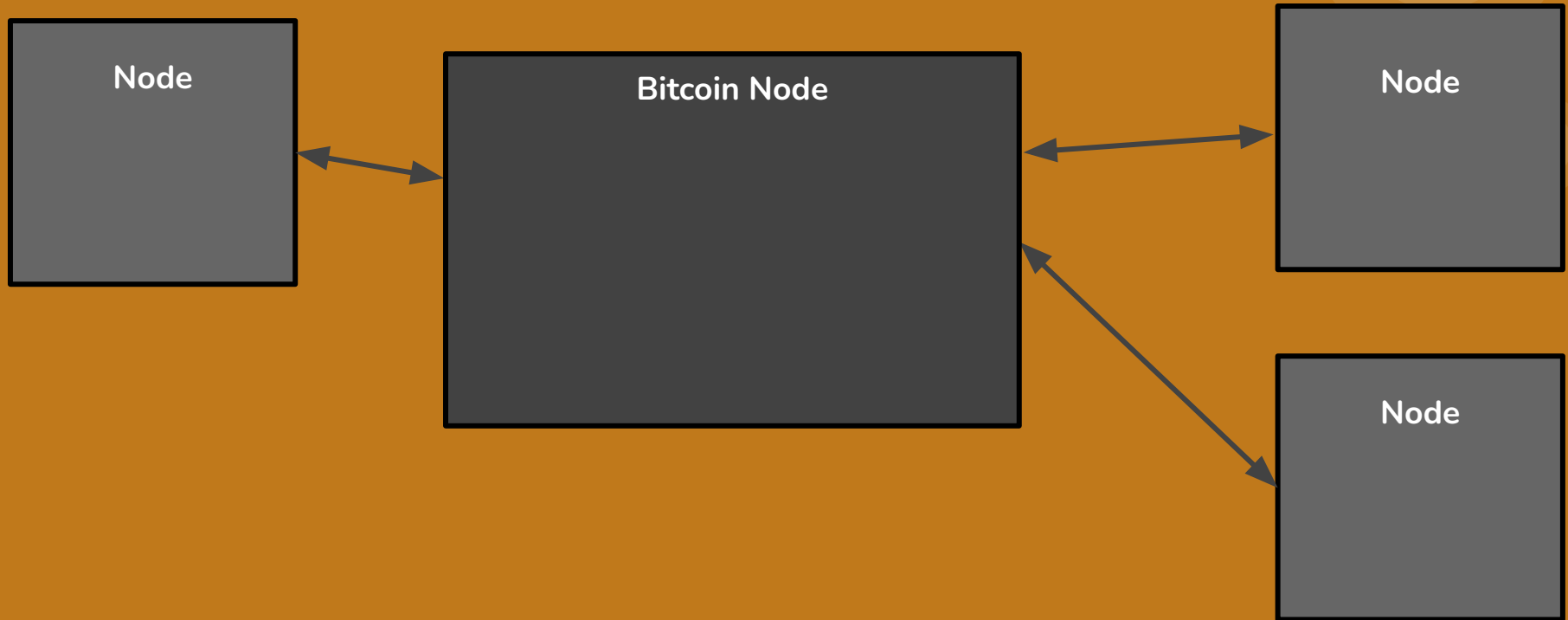
```
00601b20fce6588d59003f90817c8a
f9087482d89e64b6db462f0400000
00000000000001d3d841946699c1f
cf7fd17ebc913d7945be715ddaa77a
85f6bbcca69059ddd032e5646329a4
07170268933ffdf306010000000001
010000000000000000000000000000
000000000000000000000000000000
0000000000ffffffffff5803519f0b1b4d
696e656420627920416e74506f6f6
c38373412000b02d439791dfabe6d
6dd5e2ffc665a2e7cf2659c62795d54
72cb9a0a69ac35645e4577b236681
6a7298020000000000000005f55000
009310000000000000ffffffffff04d407b
0260000000017a9144b09d828dfc8
baaba5d04ee77397e04b105...
```

## also Bitcoin Node

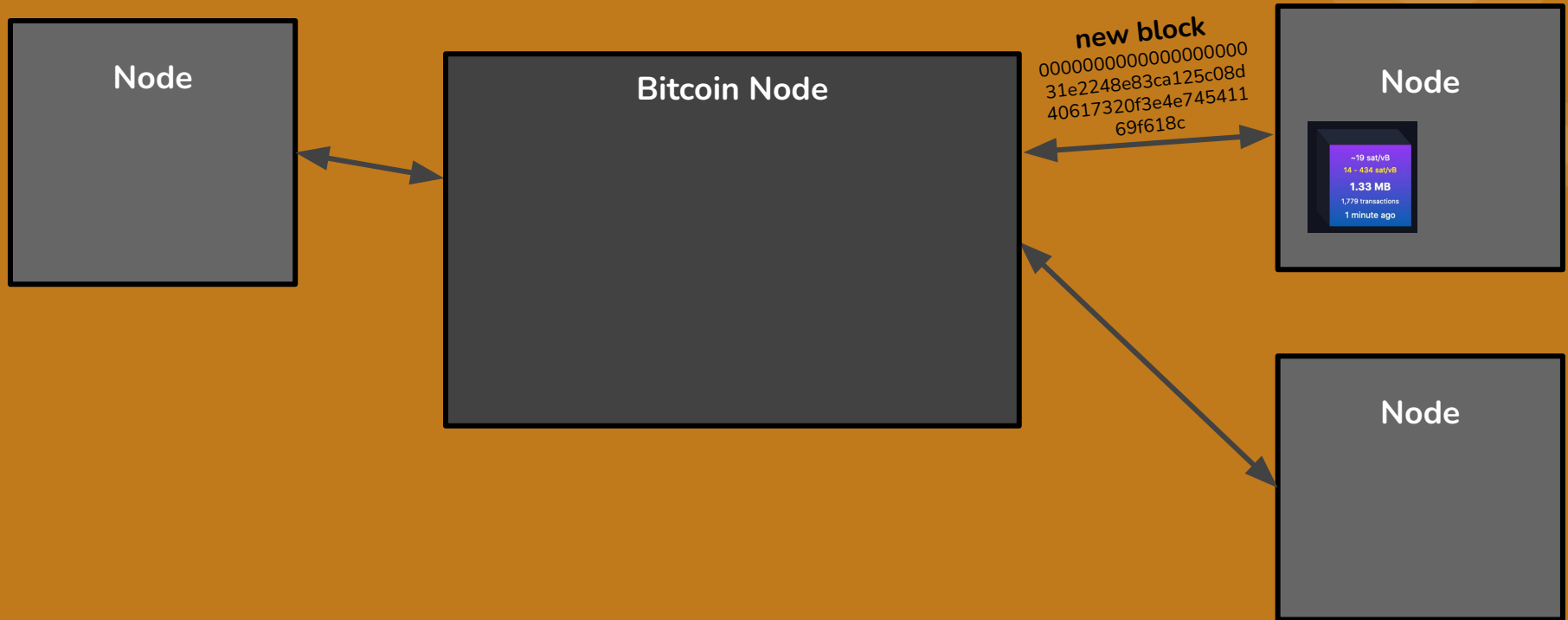


received BLOCK from peer1

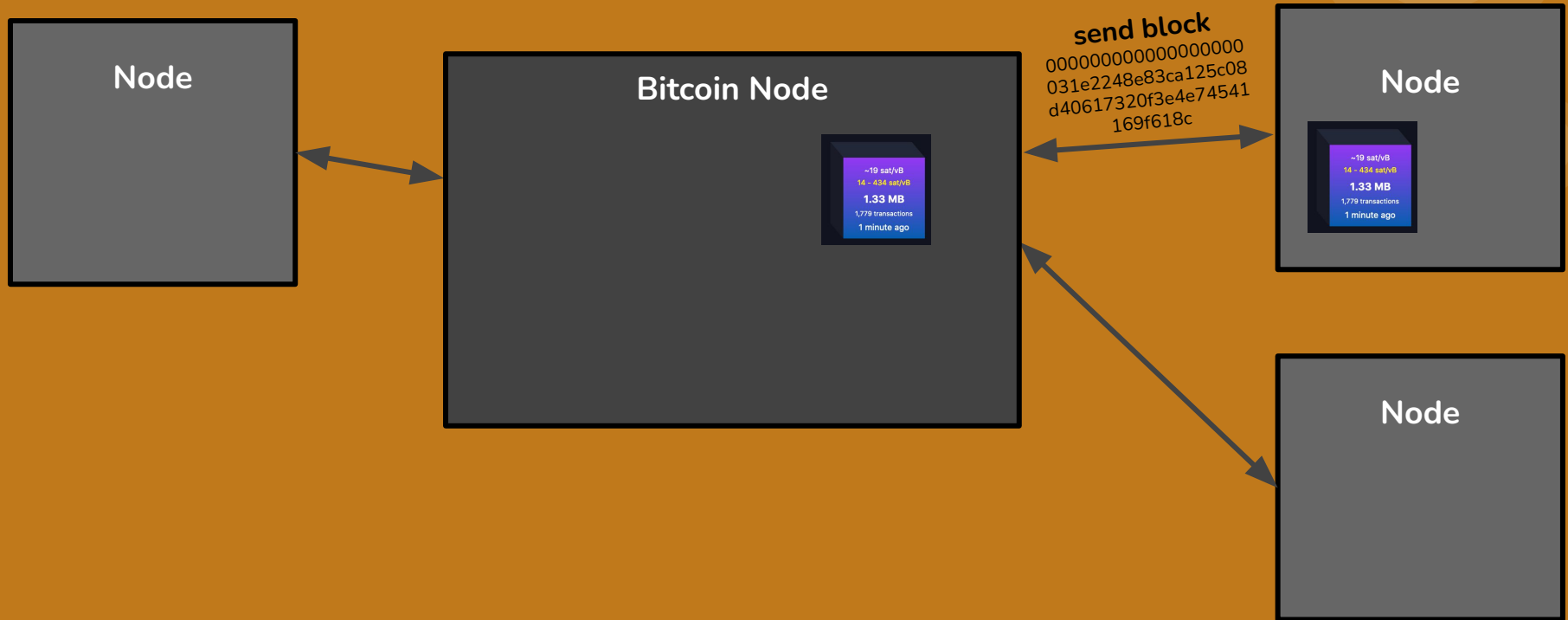
**Bitcoin node = speaks Bitcoin p2p protocol**



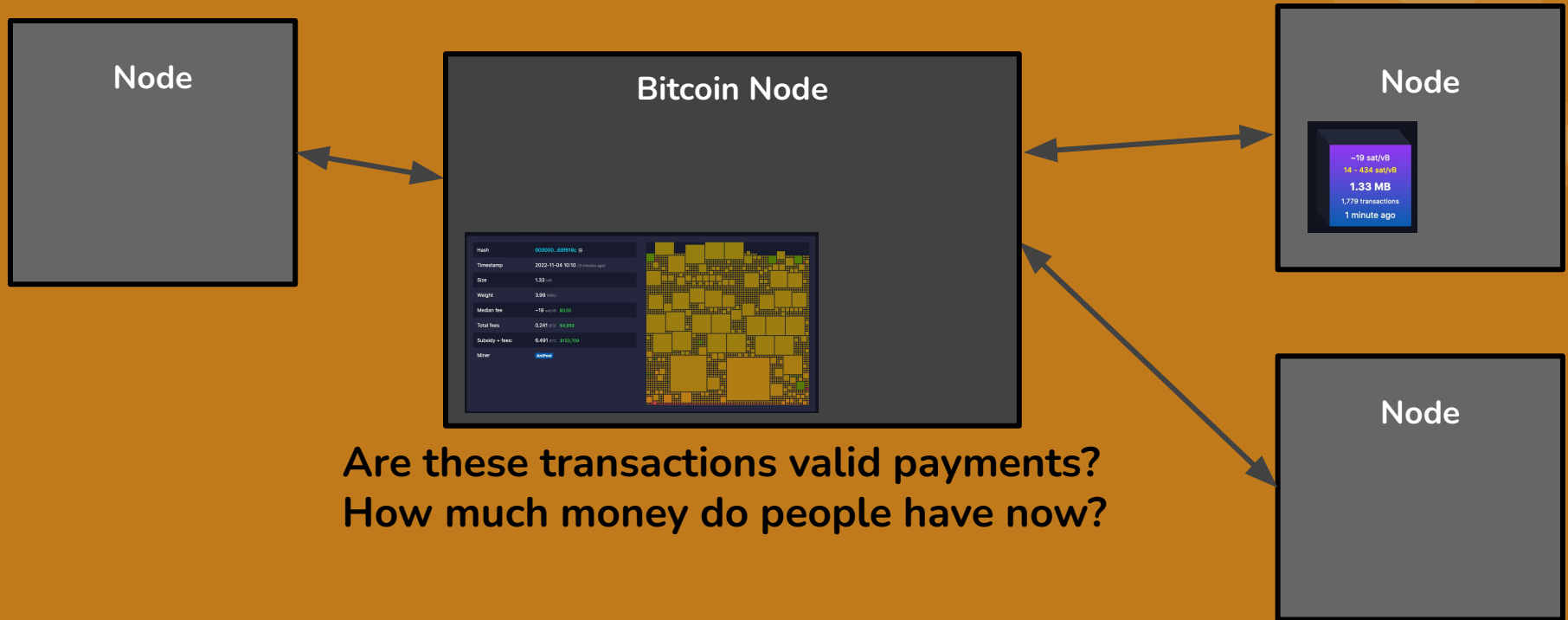
# Bitcoin node = speaks Bitcoin p2p protocol



# Bitcoin node = speaks Bitcoin p2p protocol



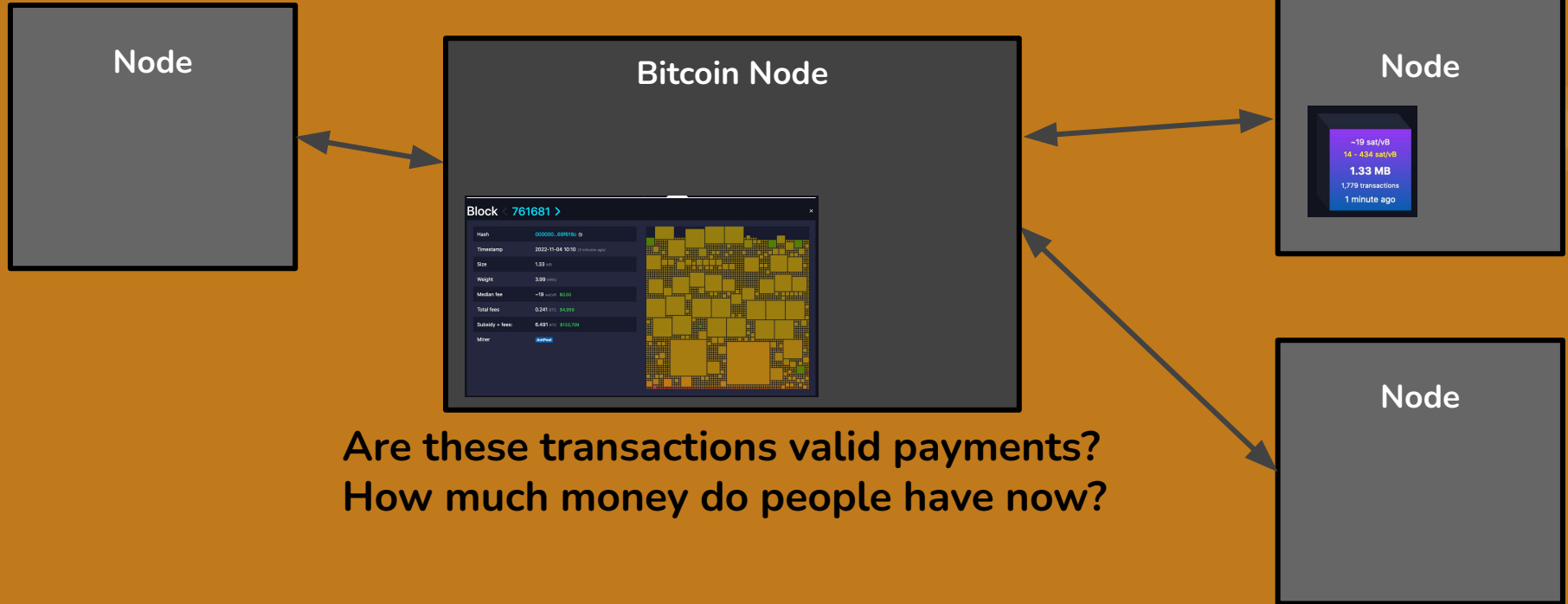
# Bitcoin node = speaks Bitcoin p2p protocol






*fully validating*

**Bitcoin node = speaks Bitcoin p2p protocol  
and applies consensus rules**





A fork happens when we change the way nodes answer:

Are these transactions valid payments?  
How much money do people have now?

No forks in this presentation



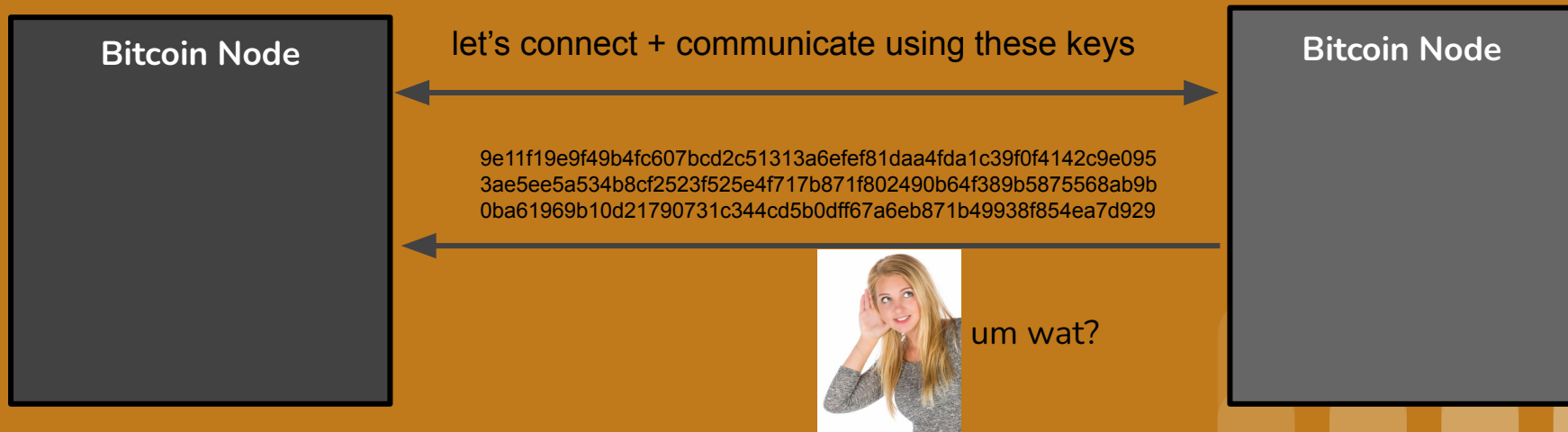
# Encrypted p2p Transport

Node communication is currently unencrypted:



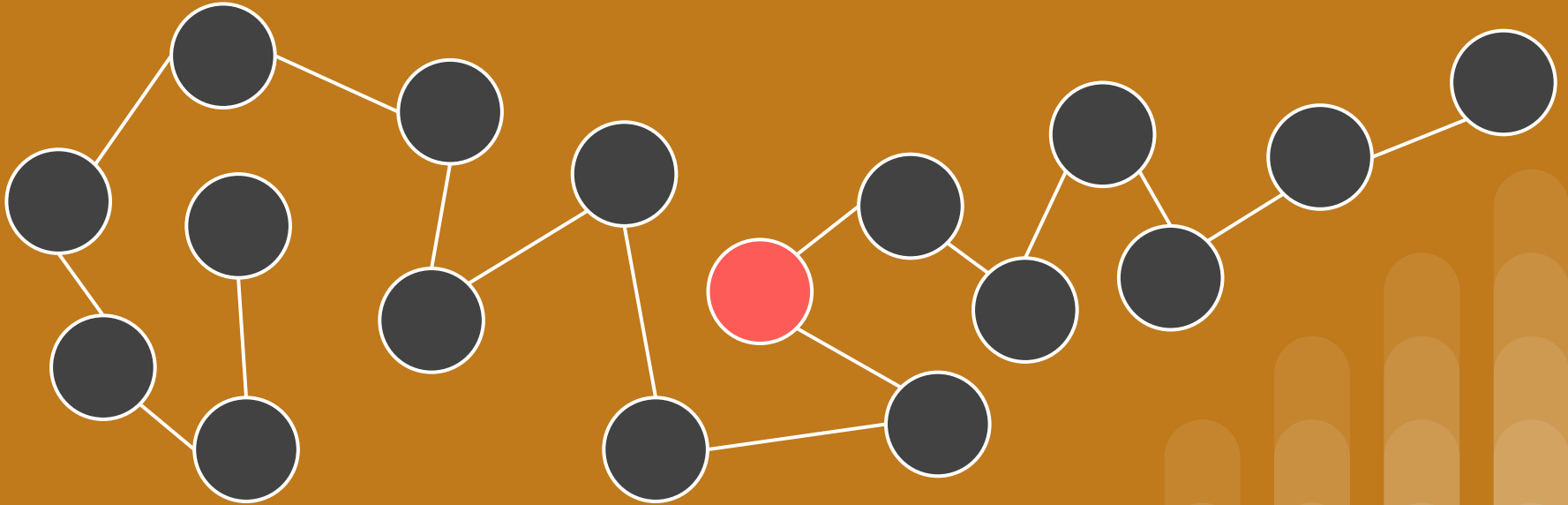
# Encrypted p2p Transport

- Protects nodes from Man in the Middle attacks
- Improves privacy by making passive eavesdropping more expensive



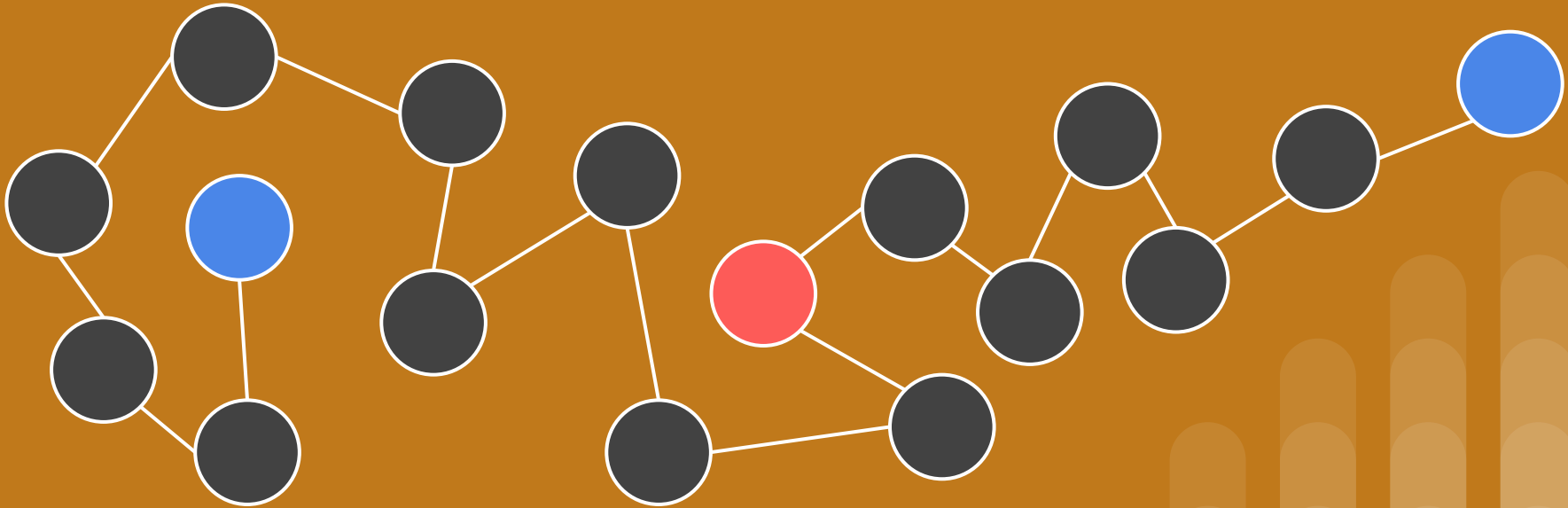
# Transaction Reconciliation (Erlay)

- More peers is better. Avoid getting eclipsed; avoid network splits.



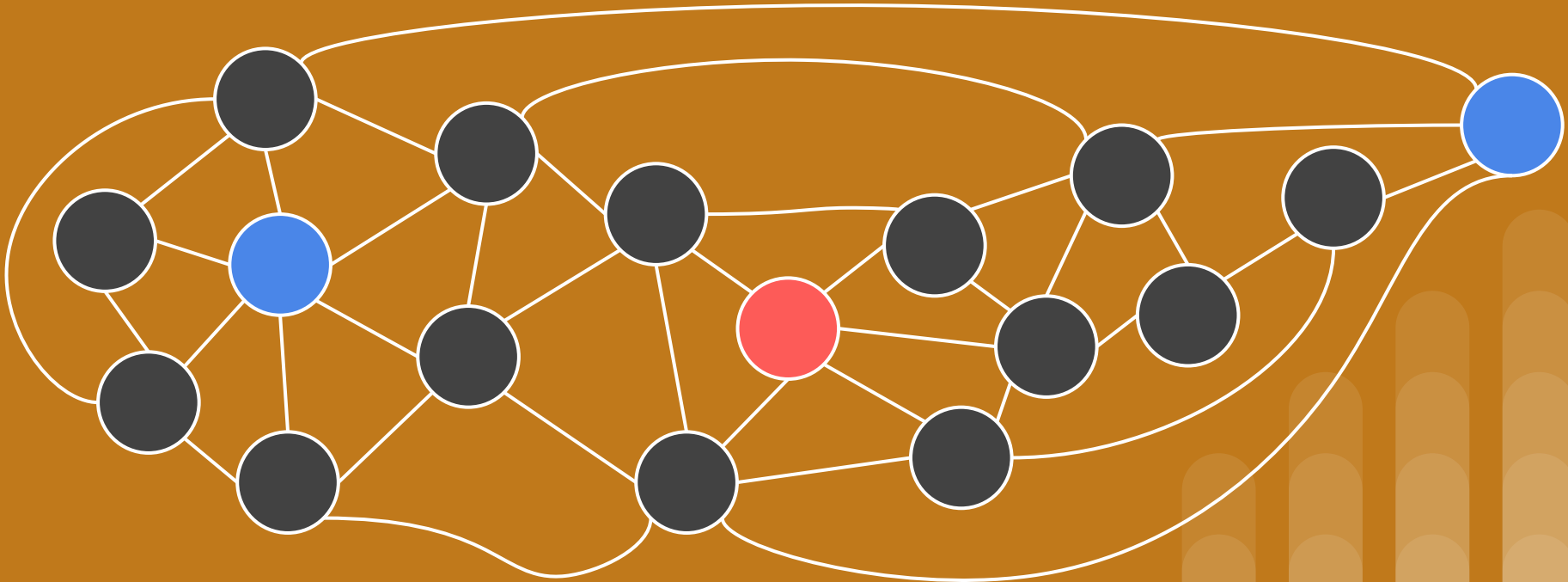
# Transaction Reconciliation (Erlay)

- More peers is better. Avoid getting eclipsed; avoid network splits.



# Transaction Reconciliation (Erlay)

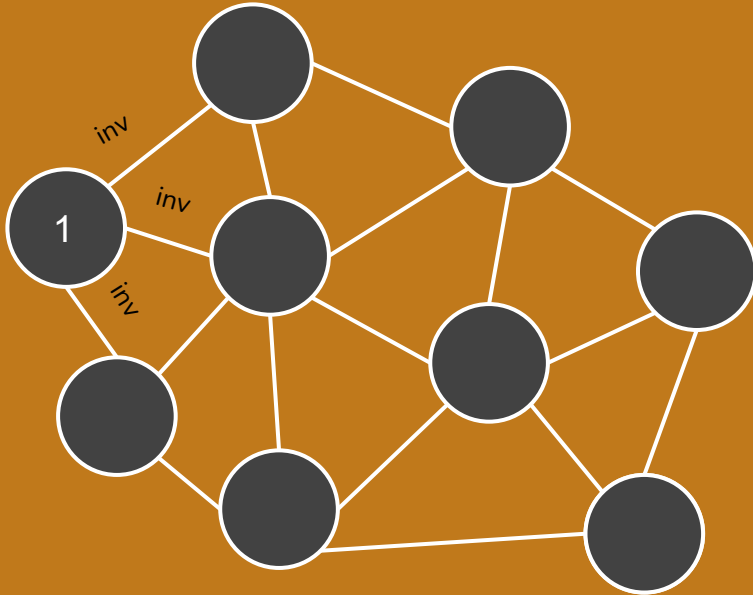
- More peers is better. Avoid getting eclipsed; avoid network splits.





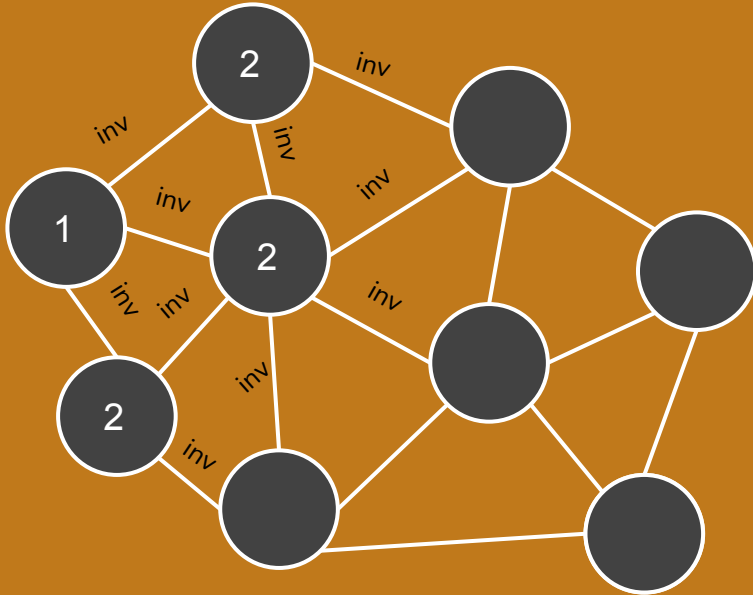
# Transaction Reconciliation (Erlay)

Legacy: announce to all peers



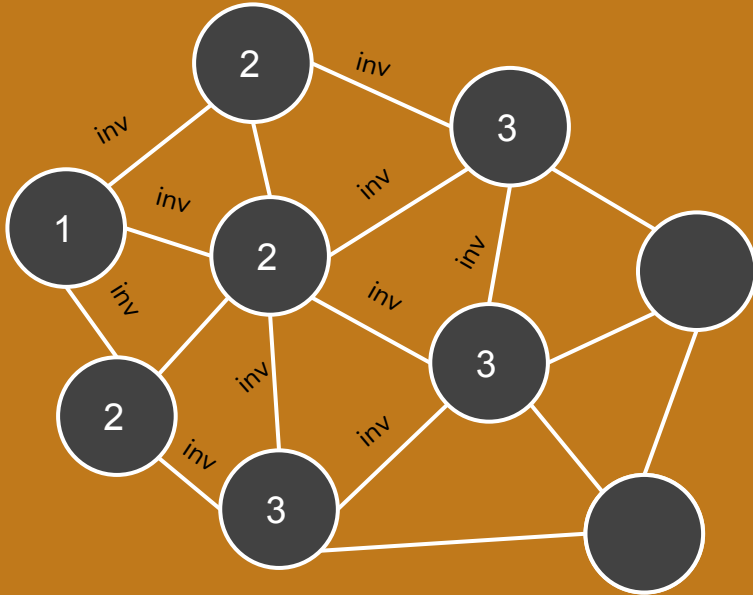
# Transaction Reconciliation (Erlay)

Legacy: announce to all peers



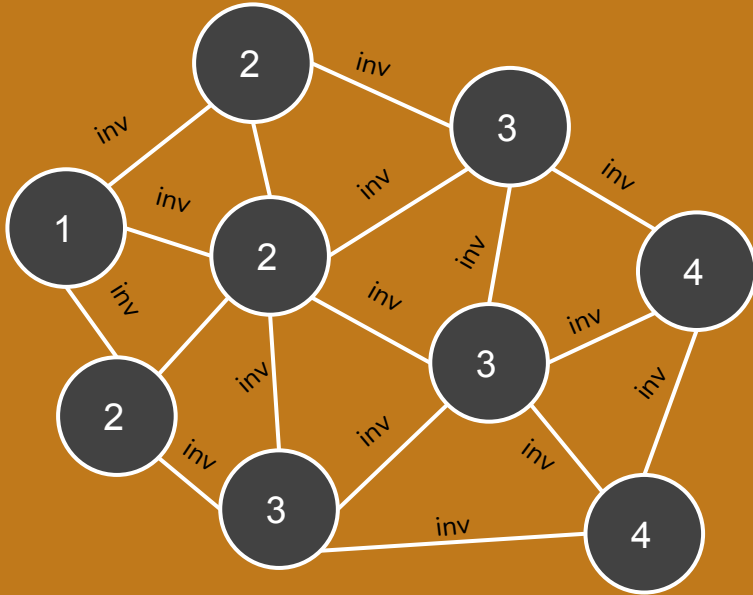
# Transaction Reconciliation (Erlay)

Legacy: announce to all peers



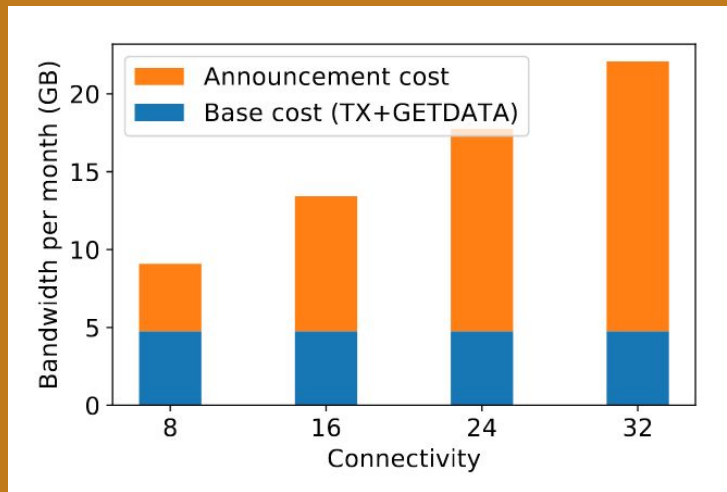
# Transaction Reconciliation (Erlay)

Legacy: announce to all peers



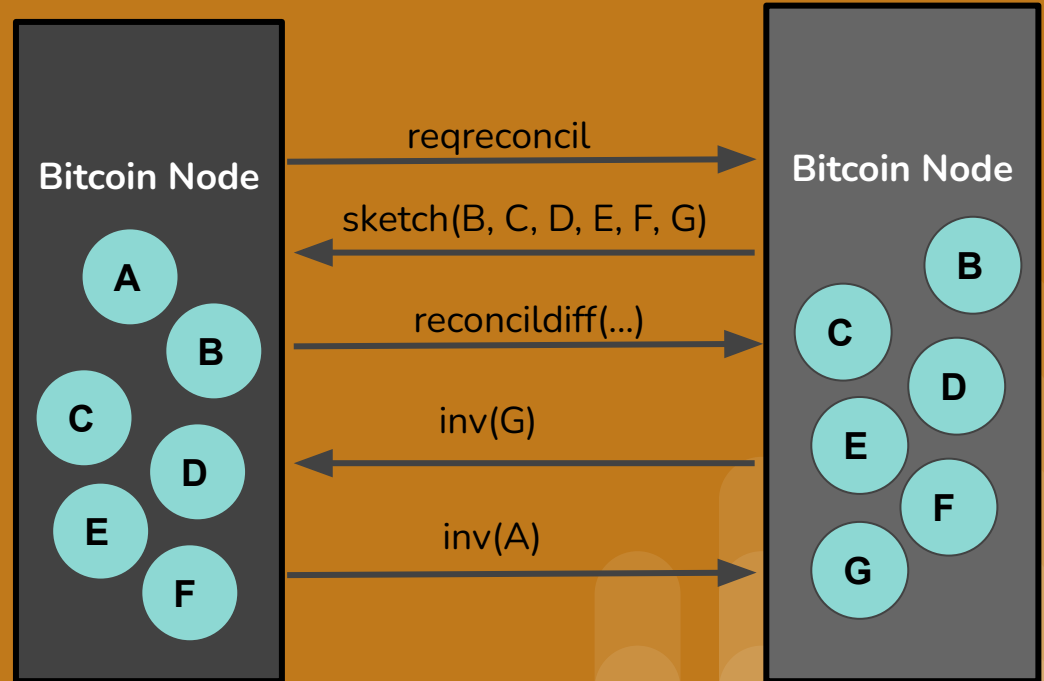
# Transaction Reconciliation (Erlay)

- Increasing the number of outbound peers increases bandwidth usage
- Transaction announcement scales particularly poorly



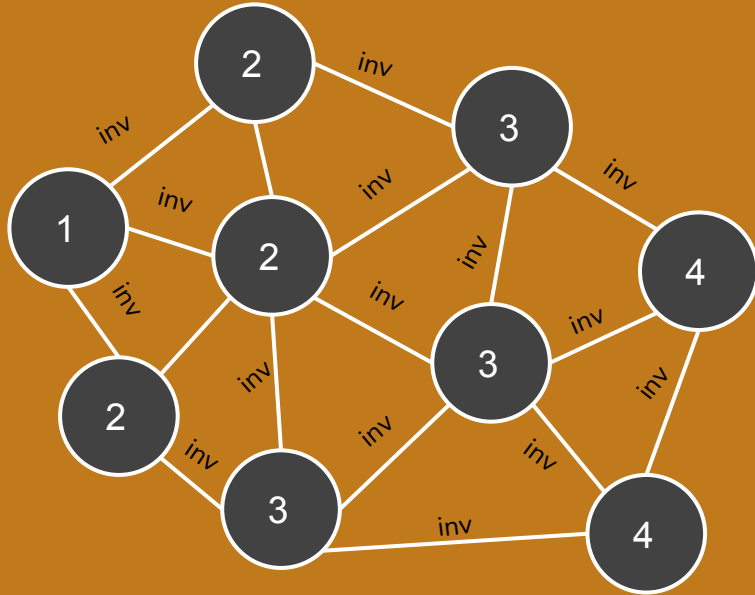
# Transaction Reconciliation (Erlay)

Erlay: periodically reconcile

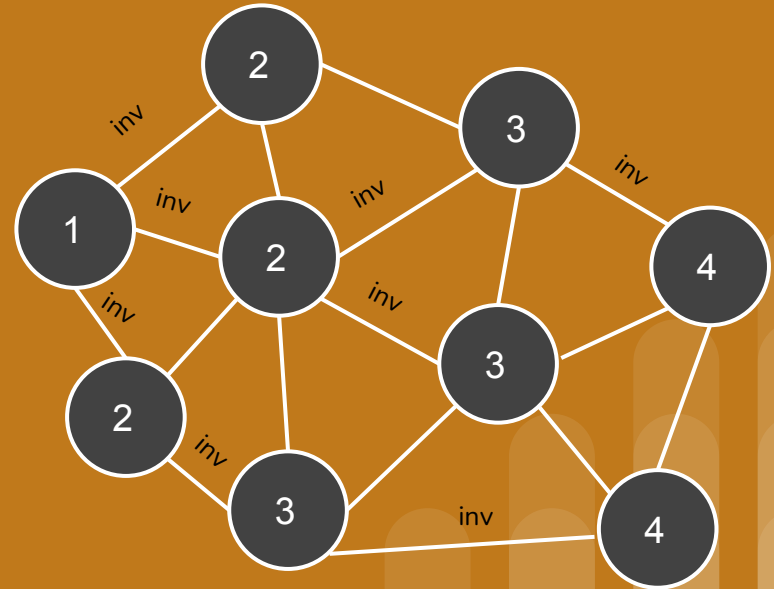


# Transaction Reconciliation (Erlay)

Legacy: announce to all peers

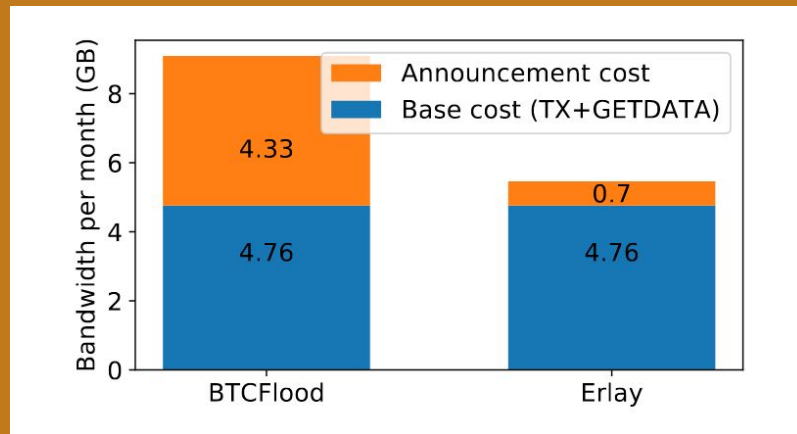
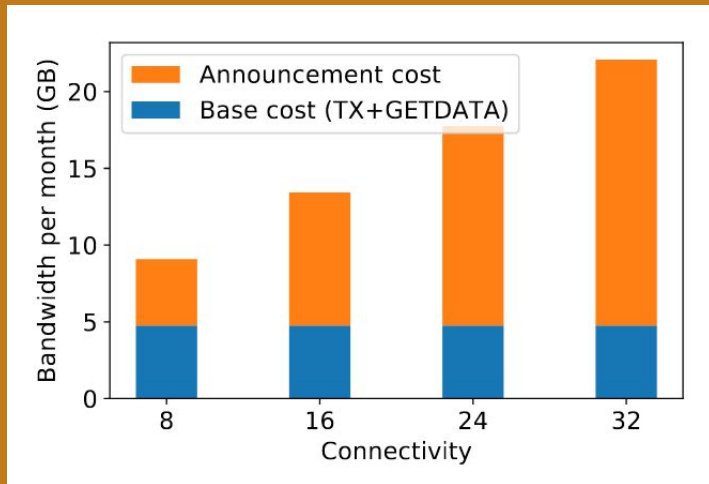


Erlay: periodically reconcile



# Transaction Reconciliation (Erlay)

- Reduces bandwidth used for announcing transactions
- Potentially allows higher peer-to-peer connectivity, helping prevent eclipse attacks

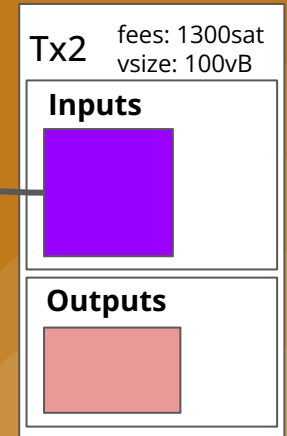
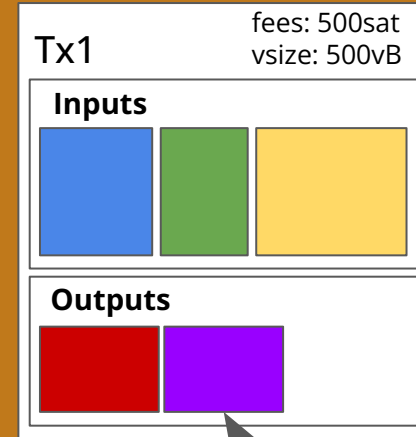


<https://arxiv.org/pdf/1905.10518.pdf>



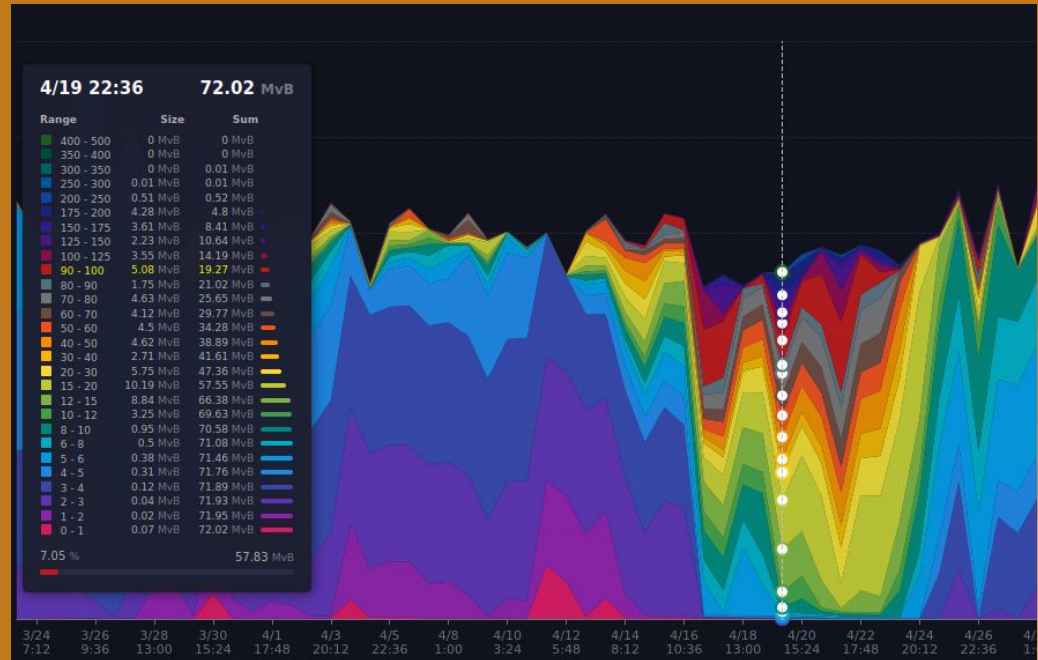
# Package Relay

- Users can fee-bump a tx by spending output in another high-fee tx (CPFP)



# Package Relay

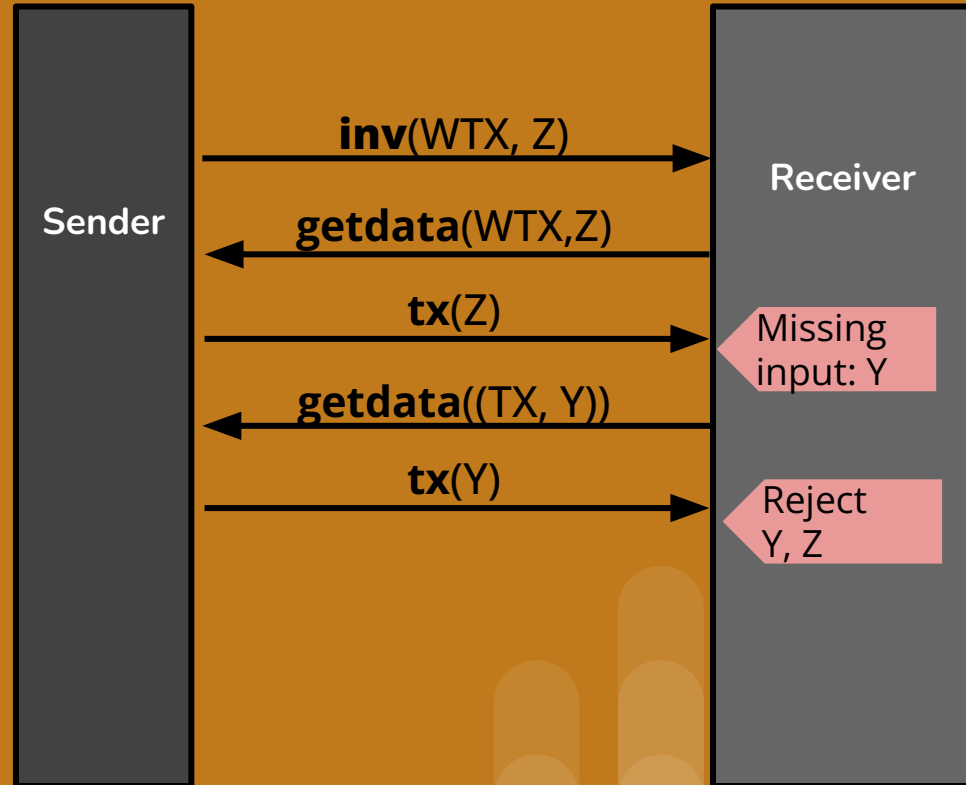
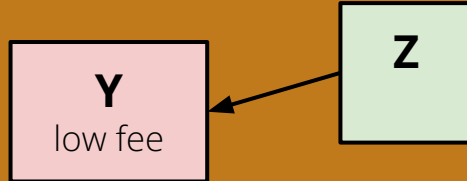
- Caveat: Transactions below a mempool's minimum feerate are always rejected



<https://node210.bitcoin.wiz.biz/graphs#1y>

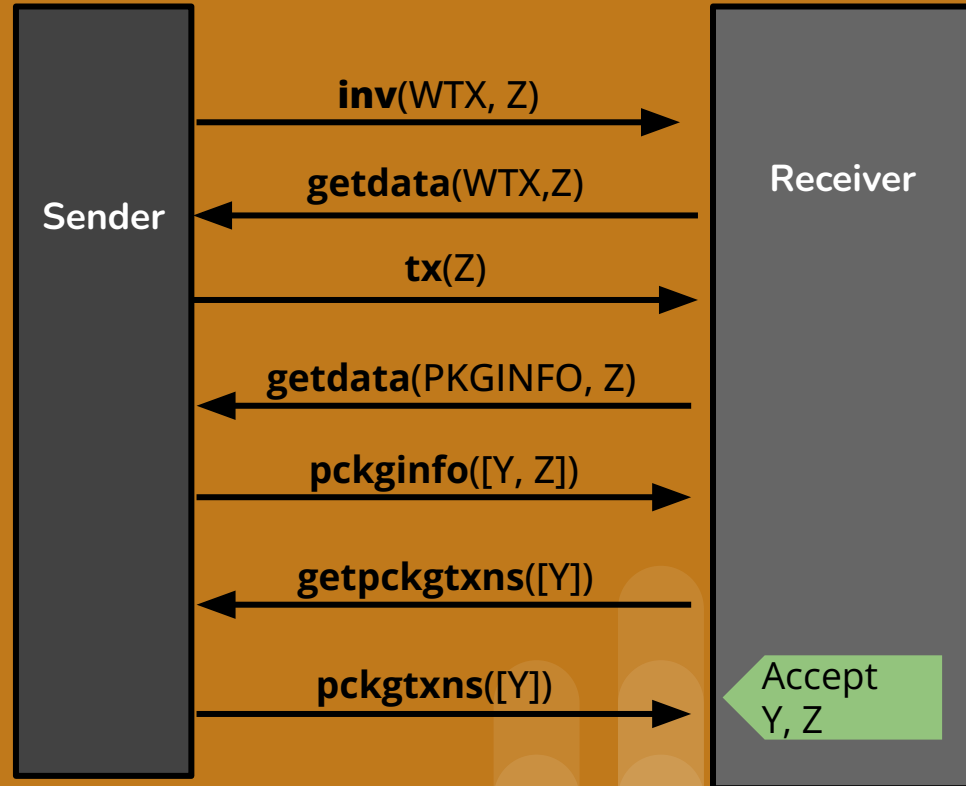
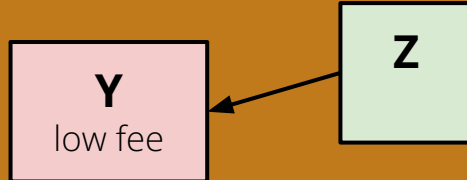
# Package Relay

- transactions may be out of order
- an “orphan” transaction spends UTXOs that the node has not encountered yet
- dealing with orphans currently requires requesting by txid



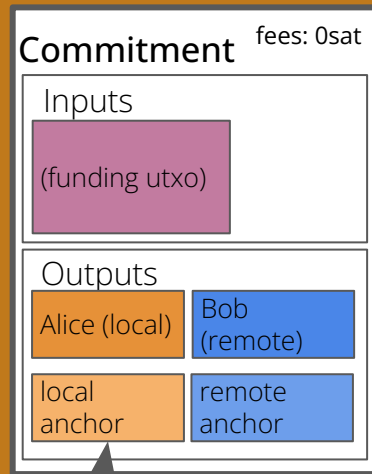
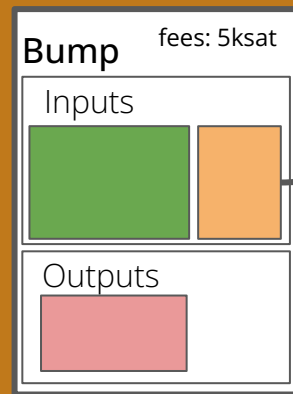
# Package Relay

Package Relay = send and validate dependent transactions together



# Package Relay

- Makes fee bumping more reliable
- Allows 0 fee commitment transactions, saving money on non-malicious unilateral channel close
- Makes transaction relay more robust



## **P2P protocol improvements allow nodes to:**

- communicate more quickly
- communicate more efficiently
- communicate more securely
- communicate information they couldn't before

**... and make Bitcoin better for everyone!**

**“Only nerds participate in  
protocol development”**

# #1: Subscribe to the Optech newsletter

- Weekly newsletters
- Summaries of technical topics
- Workshops for businesses
- Written / reviewed by devs
- Translated

[bitcoinops.org](https://bitcoinops.org)



## Bitcoin Optech Newsletter #222

Oct 19, 2022

This week's newsletter describes the block parsing bug affecting BTCD and LND last week, summarizes discussion about a planned Bitcoin Core feature change related to replace by fee, outlines research about validity rollups on Bitcoin, shares an announcement about a vulnerability in the draft BIP for MuSig2, examines a proposal to reduce the minimum size of an unconfirmed transaction that Bitcoin Core will relay, and links to an update of the BIP324 proposal for a version 2 encrypted transport protocol for Bitcoin. Also included are our regular sections with summaries of changes to services and client software, announcements of new releases and release candidates, and descriptions of notable merges to popular Bitcoin infrastructure projects.

### News

- **Block parsing bug affecting BTCD and LND:** on October 9th, a [user](#) created a [transaction](#) using [taproot](#) with a witness containing nearly a thousand signatures. The consensus rules for taproot don't place any direct limits on the size of witness data. This was a design element discussed during taproot's development (see [Newsletter #65](#)).

Shortly after the large-witness transaction was confirmed, users began to report that the BTCD full node implementation and LND Lightning Network implementation were failing to provide data from the most recent blocks that were available to Bitcoin Core full nodes. For BTCD nodes, this meant that transactions which had been recently confirmed were being reported as still unconfirmed. For LND, it meant that new channels that had recently become ready to use weren't being reported as fully open.

A developer for both BTCD and LND fixed the problem in BTCD's code, which LND uses as a library, and quickly released new versions for both [LND](#) (as mentioned in [last week's newsletter](#)) and [BTCD](#). All users of BTCD and LND should upgrade.

Until a user upgrades their software, they will suffer the lack-of-confirmation problems described above and may also be vulnerable to several attacks. Some of those attacks require access to significant hash rate (making them expensive and, hopefully, impractical in this case). Other attacks, particularly those against LND users, require the attacker to risk losing some of their funds in a channel, which is also hopefully a sufficient deterrent. We again recommend upgrade and, further, we recommend that anyone using any Bitcoin software sign up for security announcements from that software's development team.

After the above disclosures, Loki Verloren [posted](#) to the Bitcoin-Dev mailing list to suggest that direct limits be added to taproot's witness size. Greg Sanders [replied](#) to note that adding limits now would not only increase code complexity but could also lead to people losing their money if they already received bitcoins to a script which requires a large witness to spend.

- **Transaction replacement option:** as reported in [Newsletters #205](#) and [#206](#), Bitcoin Core merged support for

[Bitcoin Optech](#) [About](#) [Publications](#) [Topics](#) [Workshops](#) [Compatibility](#) [Dashboard](#)

[/ home / topics /](#)

## MuSig

**MuSig** is a protocol for aggregating public keys and signatures for the schnorr digital signature algorithm.

MuSig allows multiple users each with their own private key to create a combined public key that's indistinguishable from any other schnorr pubkey, including being the same size as a single-user pubkey. It further describes how the users who created the pubkey can work together to securely create a [multisignature](#) corresponding to the pubkey. Like the pubkey, the signature is indistinguishable from any other schnorr signature.

Compared to traditional script-based multisig, MuSig uses less block space and is more private, but it also requires more interactivity between the participants. As of August 2021, there are three protocols in the MuSig family:

- **MuSig** (also called MuSig1), which should be simple to implement but which requires three rounds of communication during the signing process.
- **MuSig2**, also simple to implement. It eliminates one round of communication and allows another round to be combined with key exchange. That can allow using a somewhat similar signing process to what we use today with script-based multisig. This does require storing extra data and being very careful about ensuring your signing software or hardware can't be tricked into unknowingly repeating part of the signing session.
- **MuSig-DN** (Deterministic Nonce), significantly more complex to implement. Its communication between participants can't be combined with key exchange, but it has the advantage that it's not vulnerable to the repeated session attack.

## Primary code and documentation

- [MuSig paper](#)
- [MuSig2 paper](#)
- [MuSig-DN paper](#)

## Optech newsletter and website mentions

2022

- [Disclosure of security vulnerability in MuSig2 as described in a draft BIP](#)
- [Discussion about designing LN upgrades to support recursive MuSig2](#)
- [MuSig2 implementation notes](#)
- [LND #6361 adds support for MuSig2 signing](#)
- [Proposed BIP for MuSig2](#)
- [Proposal to use MuSig2 in the LN gossip protocol](#)



# #2: Attend a BitDevs in your city (or start one!)

[bitdevs.org/cities](https://bitdevs.org/cities)

## SF Bitcoin Devs

Meetup Twitter GitHub RSS

### Socratic Seminar 27

2022-06-27

#### Preamble

- Asking questions is encouraged, even if you think they're dumb!
- [Chatham House Rule](#)
- [Bitcoiner Jobs](#)

#### New Work & Research

- Hertzbleed: New sidechannel attack just dropped. First speculative execution, now dynamic frequency scaling. Is no hardware optimization technique safe?
- Alex Leishman writes a Twitter thread on Bitcoin MEV. Tejaswari Nadhalli responds with a link to a paper he wrote on MEV opportunities afforded by HTLCs. @roasbeef responds with a follow up paper that reworks the MEV-resistant HTLC construct proposed in the previous paper.
- Craft Store Bitcoin: Explaining the UTXO set with crafting supplies. Good for explaining how the world's best money works to small children.
- Web 5: What if we rebuilt the web using GPG and BitTorrent but with good UX.
- Murch discusses the waste metric, a simple metric for coin selection which tries to capture coin-selection goals in a way that is sensitive to current fee market conditions relative to expected long-term fee market conditions.
- RIDDLE: ring signatures over UTXOs for anti spam sybil.

#### Privacy

- Dmix: a decentralized mixer. Does it work? I dunno, I'm not smart enough to understand the paper.
- pln: A very private Lightning wallet that spins up a node for every channel. No receive only send.
- Justin Moon discusses Minimint, a federated Chaumian Lightning bank. Like Wallet of Satoshi but more private and less trusted.

#### Lightning

- Validating Lightning Signer: avoiding blind signing in Lightning. The list of policy controls is especially interesting.
- Summary of Lightning Network Summit 2022
- ZmnsCPxj discusses using routing fees to signal available liquidity.
- Lightspark: Serious Investors™ discover the Lightning Network after realizing that Libra is never going to happen.
- Rust Lightning and Sensei add 0-conf channels. Let's get reckless in this mf.
- Lightning node operator medium\_of\_exchange gets hacked because vultr can't defend against social engineering.
- Intentionally triggering a justice transaction: fiatjaf has a channel that he wants to close, but it doesn't have many sats on his side, and he doesn't want a small UTXO. So he intentionally broadcasts and old state, so lnbiz takes everything in a penalty transaction.
- ln.cash: Lightning Network sats dead-drop
- vpn.sovereign.engineering: top-up your Mullvad VPN w/Lightning

#### Applications

- Discreet log protocol has hard forked to improve message serialization.
- DLC Channels
- Some nice Taro docs over here

## BitDevs NYC

- 📍 New York, NY
- 👤 4,950 members · Public group
- 👤 Organized by J and 6 others

Share: [f](#) [t](#) [in](#) [✉](#)

Join this group

...

### Organizers



J and 6 others

[Message](#)

### Members (4,950)

[See all](#)





# Thank you!

@glozow

