JARROD WATTS
Polygon Labs

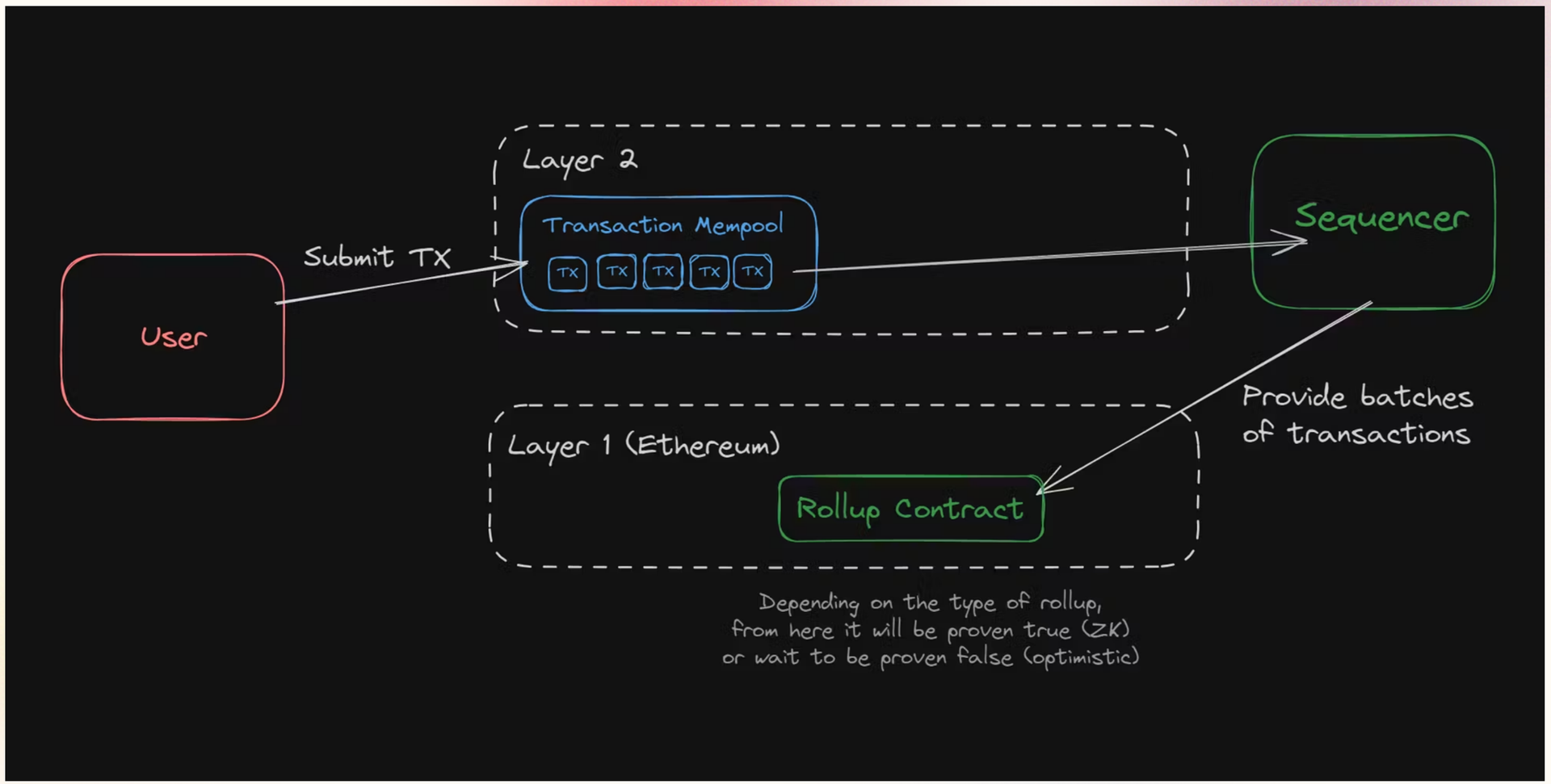
# AggLayer

ATOMIC COMPOSABILITY ACROSS AGGREGATED CHAINS,
WITHOUT SACRIFICING SOVEREIGNTY.

# What We'll Cover

| | |
|---|---|
| **WTF is a sequencer?** | Basics of L2 sequencers, what they do |
| **Centralized Sequencers** | Explanation + pros & cons of centralized sequencers |
| **Decentralized Sequencers** | Comparison of centralized sequencers vs. decentralized sequencers |
| **Shared Sequencers** | Evolution of decentralized sequencers into shared sequencers |
| **Polygon AggLayer** | Polygon's new AggLayer vs. shared sequencers |
| **Final Thoughts** | Closing discussion |

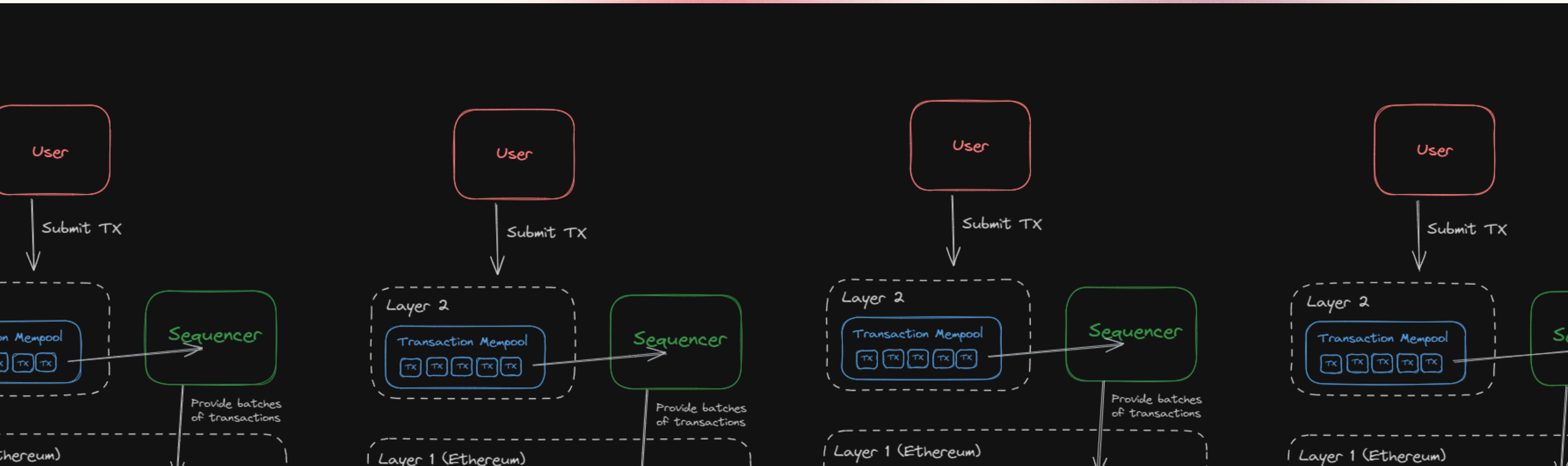
# Sequencers Explained



# Role of a Sequencer

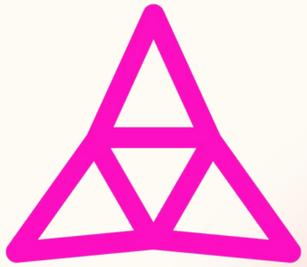
1. Read transactions from the L2 mempool & **execute** them
2. **Batch** transactions together, and send the batches to L1

# Sequencer Landscape Today



# Cross-L2 Interoperability?





## Operator

**The system has a centralized sequencer**

While proposing blocks is open to anyone the system employs a privileged sequencer that has priority for submitting transaction batches and ordering transactions.

MEV can be extracted if the operator exploits their centralized position and frontruns user transactions.



# Centralized = Good?

- “Centralized” - what?
- Offers best performance
- Most users on L2 care most about performance

## Ok but how?

- Consistent near-instant finality on the L2
- Batches are likely always going to be valid = less time to prove = less time to withdraw

# Centralized = Bad?

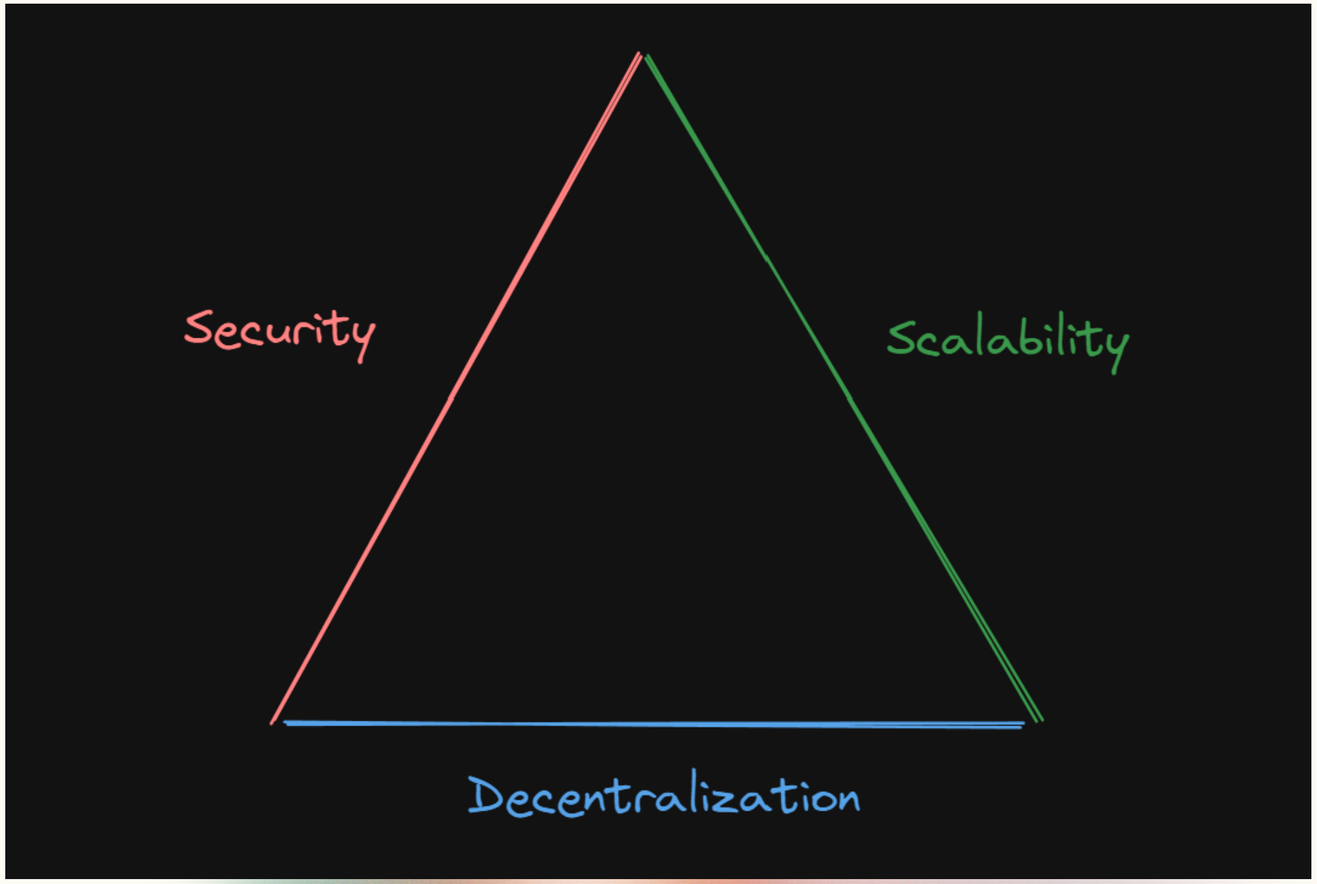
## Today's Arbitrum Sequencer Downtime: What Happened?

Offchain Labs · Follow

2 min read · Jan 10, 2022

411

# As always, it depends



# If 💩 hits the fan:

---

**Users can force any transaction**

Because the state of the system is based on transactions submitted on-chain and anyone can submit their transactions there it allows the users to circumvent censorship by interacting with the smart contract directly. Anyone can become a Proposer after approximately 6d 8h (45818 blocks) of inactivity from the currently whitelisted Proposers.

SequencerInbox.sol#L125 - Etherscan source code, forceInclusion function
Sequencer Isn't Doing Its Job - Arbitrum documentation

# Centralized Sequencers

## Built for Performance

Consistently performs its job well. Great UX.

## Obviously, not decentralized

A single point of failure

## No Interoperability

Users cannot perform cross-L2 transactions

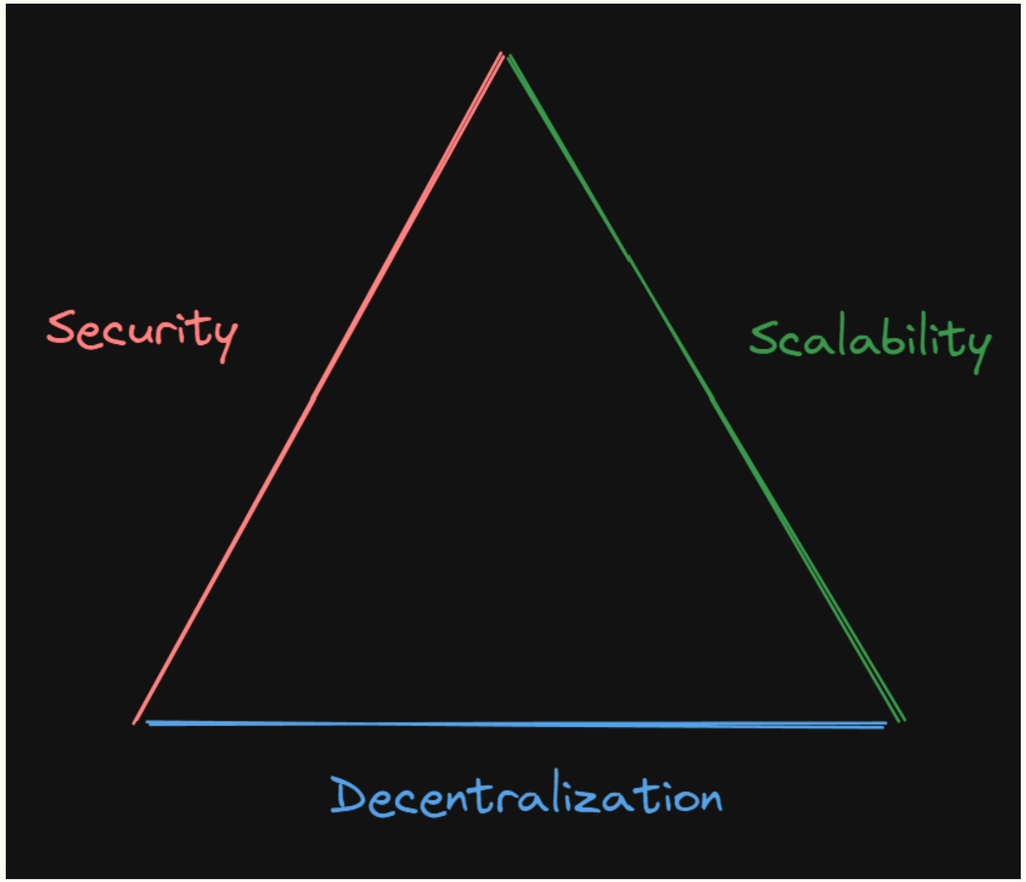

DECENTRALIZED SEQUENCERS

# Decentralized = Good Right?

- Essentially just reverse the pros and cons of centralized sequencers

- Centralized = more performant, less decentralized
- Decentralized = less performant, more decentralized

- Decentralized is not strictly just good/better



# Decentralized Sequencers

## Built for Decentralization

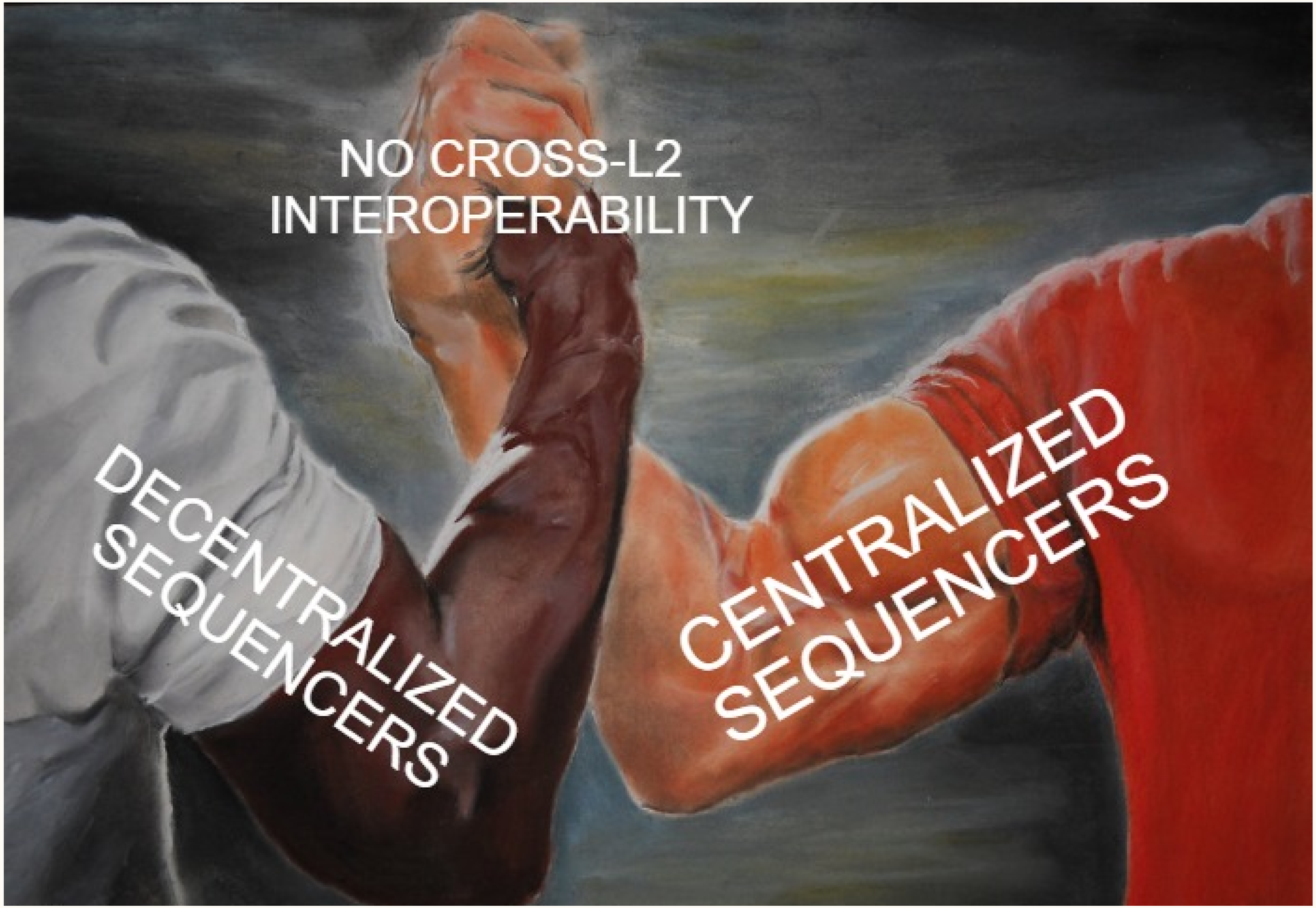
Doesn’t have a single point of failure

## Not as performant

Calling upon a pool of nodes is less reliable than an instance you operate

## No Interoperability

Users cannot perform cross-L2 transactions


NO CROSS-L2
INTEROPERABILITY
DECENTRALIZED
SEQUENCERS
CENTRALIZED
SEQUENCERS

# Users & Developers

TVL **$83.65M** ▼ 14.81%

| | |
|---|---|
| Canonical | **$83.65M** (100%) |
| External | **$0.00** (0%) |
| Native | **$0.00** (0%) |

View TVL Breakdown

TVL **$532M** ▼ 5.73%

| | |
|---|---|
| Canonical | **$532M** (100%) |
| External | **$0.00** (0%) |
| Native | **$0.00** (0%) |

View TVL Breakdown

TVL **$5.02B** ▼ 17.32%

| | |
|---|---|
| Canonical | **$2.19B** (44%) |
| External | **$0.00** (0%) |
| Native | **$2.83B** (56%) |

View TVL Breakdown

TVL **$9.98B** ▼ 12.66%

| | |
|---|---|
| Canonical | **$6.41B** (64%) |
| External | **$29.97M** (0%) |
| Native | **$3.54B** (36%) |

View TVL Breakdown

# Goal: enable cross-chain transactions for users


SHARED SEQUENCERS

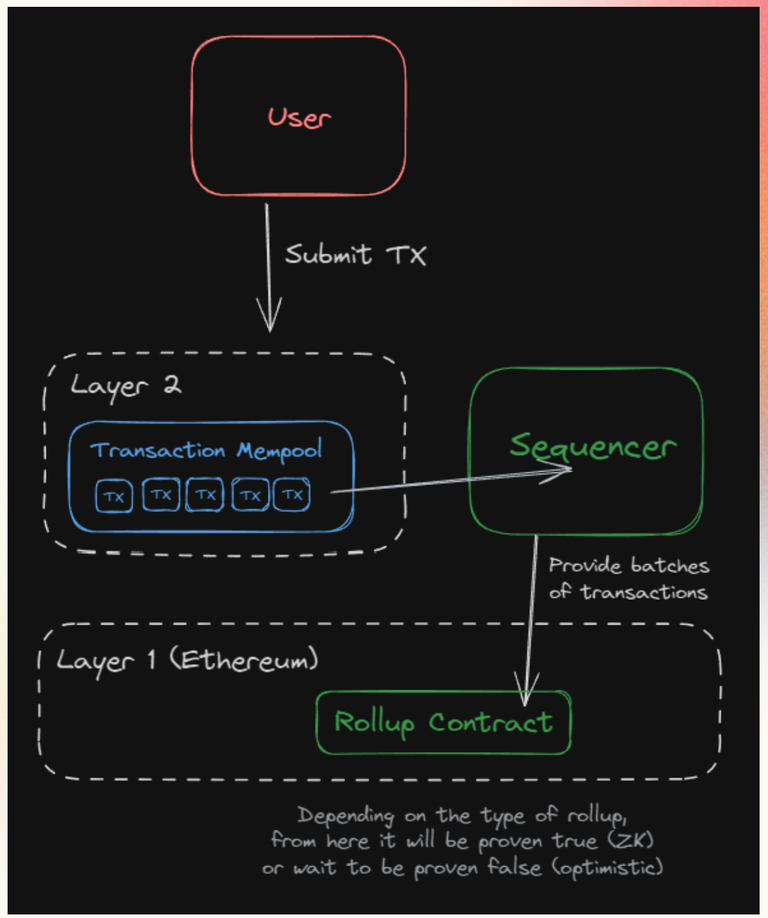
# WTF is shared sequencing?

Shared sequencers operate a separate blockchain to perform the role of a **decentralized sequencer** for multiple rollups to introduce **interoperability** advantages.

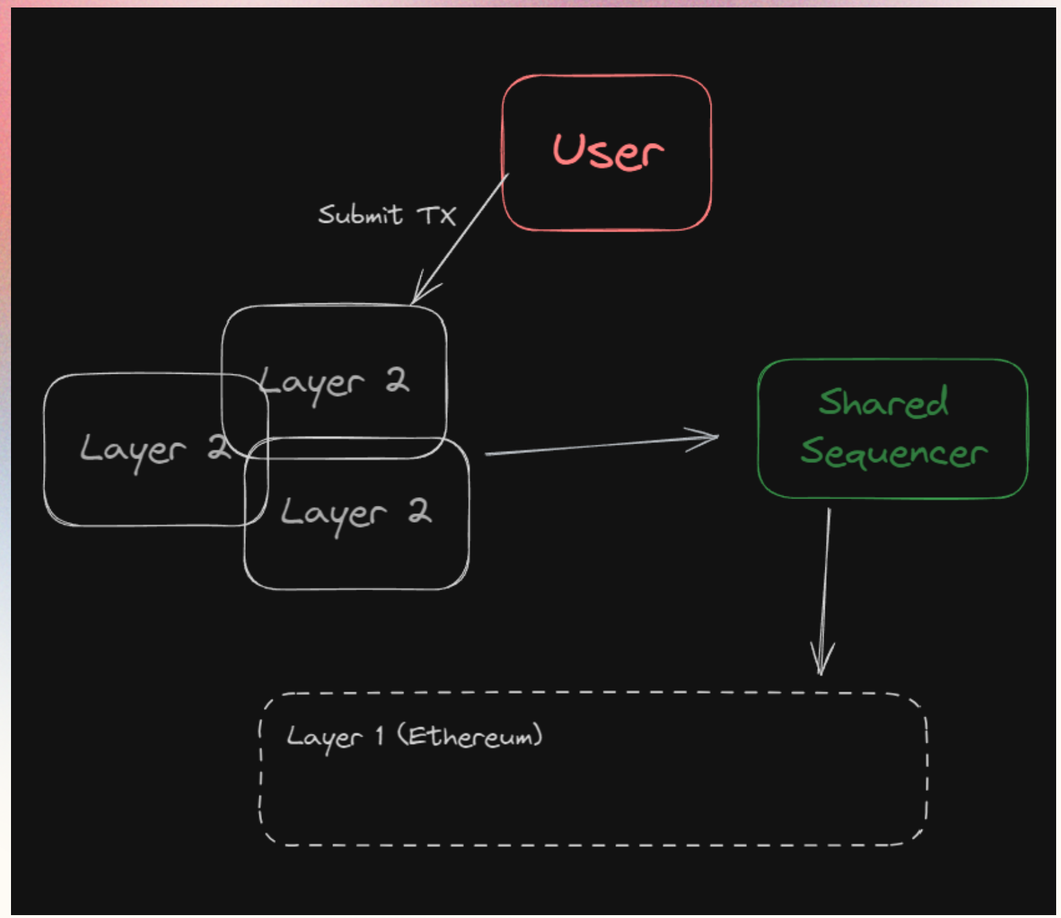
- Decentralized sequencers = serve **one** chain
- Shared sequences = serves **multiple** chains

# Shared vs Decentralized

## Decentralized




## Shared



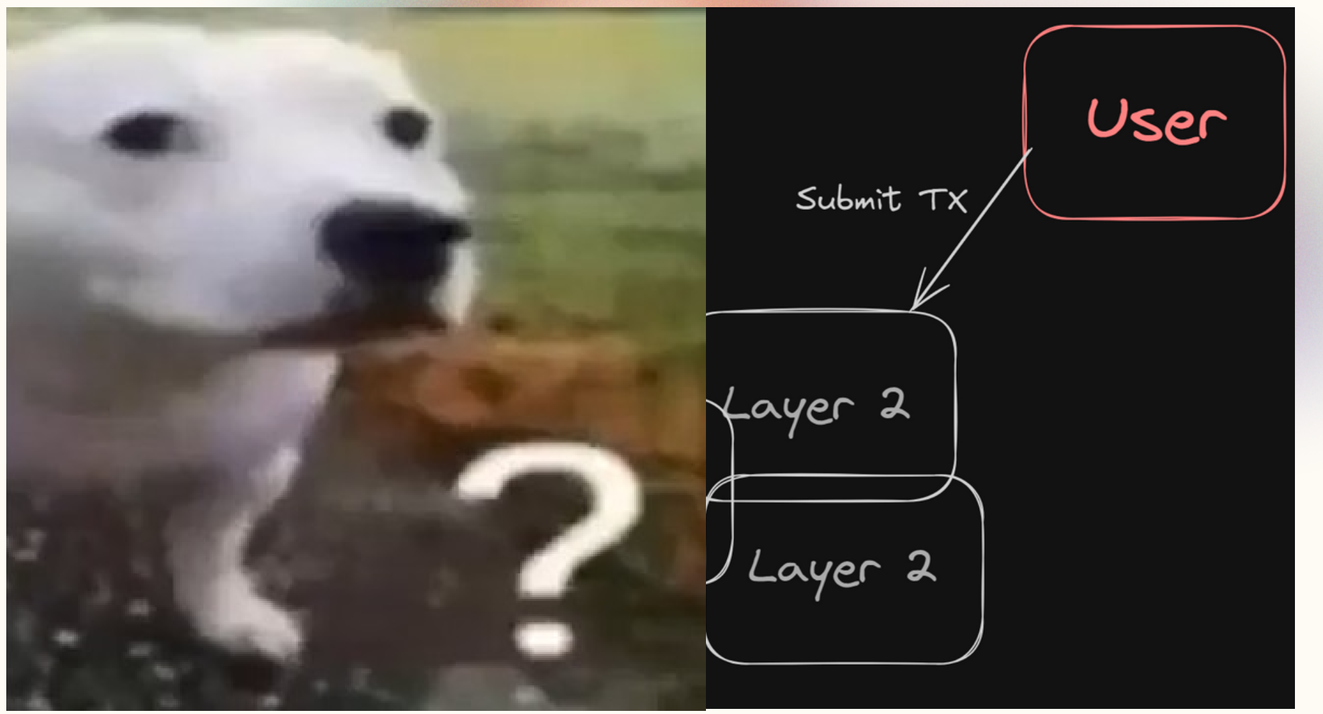
# Why shared sequencing?

**Atomic transactions**: Submit a set of transactions or a “bundle” to multiple different chains.

Either ALL or NONE of those transactions will execute successfully.

Example use cases:

- Performing a cross-chain transfer
- Performing cross-chain DEX arbitrage



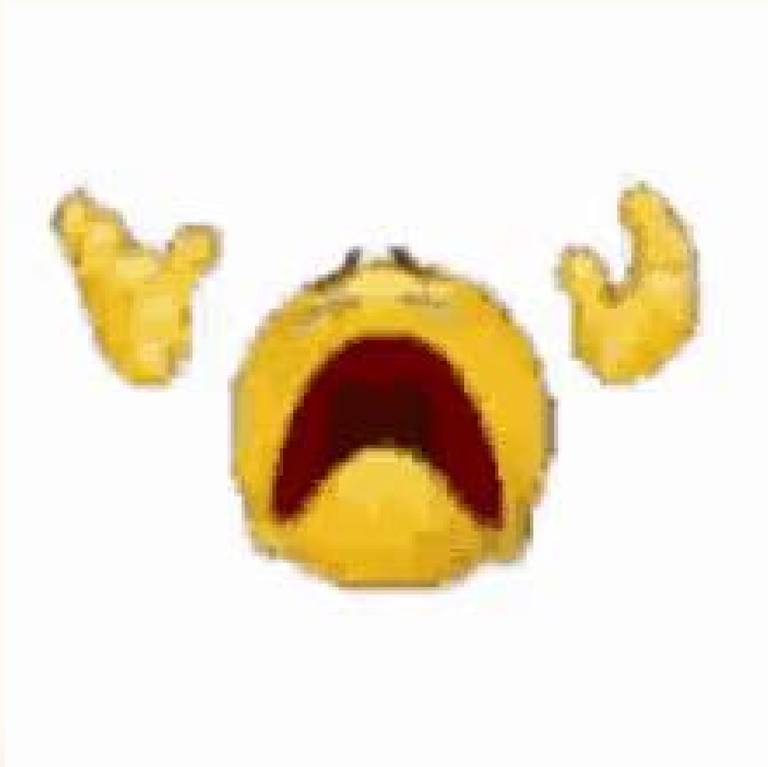
# Why NOT shared sequencing?

To use a shared sequencer, you **give up** control or “sovereignty” of your sequencer to achieve the interoperability benefits.

This limits your ability as an L2 to:

- Run a centralized sequencer for performance
- Have control over the MEV collection
- Have control over censorship resistance
- Upgrade or modify your operator

You basically give the responsibility of your chains’ sequencer away to another network.



# Shared Sequencers

## Decentralization + Interoperability

Enables new use cases while also maintaining decentralization

## Not as performant

Still not as performant as centralized sequencers

## Loss of Sovereignty

L2 developers give up control of their sequencer

# ✨ Polygon AggLayer ✨

# Polygon AggLayer

The AggLayer (aggregation layer) enables the same **atomic** transactions and cross-rollup **interoperability** as shared sequencers **without** forcing rollup developers to give up control of their sequencers & sovereignty.

This way, L2s get both:

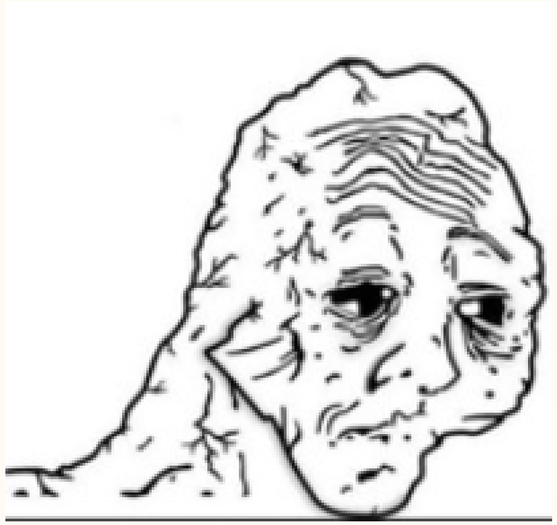
- Freedom to operate their sequencer how they want.
- Interoperability between any L2s that opt into the AggLayer.

# Is this just a Polygon shill? 😕

ANY alt L1 or L2 can connect to the AggLayer.

Not just Polygon CDK chains.

# Why AggLayer?



## L2 Developers

Tap into a huge pool of liquidity and community from other AggLayer chains

## App Developers

Users from other chains can use your app without directly bridging to the L2 your app is on

## End Users

A better UX where users are not siloed to individual L2s

# How’s it Work?

- Each chain’s sequencers execute transactions as usual (centralized or decentralized)
- Each chain’s sequencer generates a ZK proof of its updated state
- Each chain sends the proof to the AggLayer rather than Ethereum
- The AggLayer aggregates all of the proofs and generates a single, aggregated ZK proof.
- This aggregated proof guarantees proofs for all chains are valid.
- The AggLayer posts that proof to Ethereum

# Wait, Aren’t ZK Proofs Slow? 

- It’s not realistic to wait for all ZK proofs before anything can happen (30-60mins)
- **Optimistic Confirmations**: A chain can send a batch of transactions *without* a proof to AggLayer
  - Another chain, chain B, can act on that unproven information on Chain A, optimistically
  - Later, chain A either:
    - Doesn’t post a proof: Chain B is forced to rollback. Chain A gets slashed.
    - Does post a proof: the optimistic system is now backed up by ZK proofs.

# Atomic Cross-Chain Transactions

- Users can submit a bundle of transactions to many chains, with the guarantee that all transactions will be successfully executed, or none will be included.

- How?

1. Users submit atomic bundles to the AggLayer
2. Transactions in the bundle are forwarded to their respective chains
3. The chain executes those transactions
4. Each chain generates a ZK proof for the block that contains the bundle.

# Roadmap

Feb:

- Unified bridge

April:

- Proof aggregation

Later in 2024:

- Optimistic confirmations
- Atomic cross-chain transactions