Abstract: Ethereum is going through the "puberty phase" of the rollup-centric roadmap. The rapid L2 scalability growth spurt has led to awkward fragmentation of composability, with a significant breakdown of network effects. L2s today are silos delineated by their sequencers, at bridging distance from each other as well as the L1.

Thankfully this fragmented state of affairs is temporary—we can do better than slow and asynchronous bridges. With shared sequencing plus real-time settlement we can (re)establish universal synchronous composability and (re)activate Ethereum's network effects with maximum thrust. Shared sequencing combines strengths of the modular and the monolithic theses.

We know Ethereum L1 as a data availability layer. Few realise Ethereum L1 can also provide shared sequencing for rollups, with preconfirmations. There's a simple design with no hard fork required. Such an Ethereum shared sequencer would be maximally secure and credibly neutral, and would enjoy a half-trillion dollar TVL head start. Ethereum L1 as the canonical shared sequencer—isn't that a compelling and unifying Schelling point? :)

Part 1: fragmentation

* universal synchronous composability (USC) is lost
  - USC is when contracts can arbitrarily call other contracts synchronously
  - USC is the ultimate form of composability
    * a luxury that Ethereum enjoyed until recently
    * a luxury that competing chains like Solana are making the most of
  - synchronous composability across L2s is only possible with shared sequencing
    * every individual sequencer delineates its own separate synchronous zone
    * shared sequencing is necessary, but not sufficient
      - we also need real-time settlement, i.e. the ability to instantly go in and out of rollups
        * zk rollups are necessary for instant state reads and withdrawals
          - also need real-time proving
            * possible with advanced SNARK recursion and hardware acceleration
          - optimistic rollups can use liquidity providers to partially simulate synchronous composability
* advantages of universal synchronous composability
  - TLDR: we all become closer and everything becomes better
    * proximity, efficiency, robustness
  - shared liquidity
    * like 1inch or Matcha but across L2s
    * more efficient trade execution
  - gas efficiency
    * deduplication: no need for many Uniswap and Aave instances
      - natural consolidation, more gas efficiency, less friction
  - bridge-less bridging ("unbridging")
    * asynchronous bridges are brittle
      - bridges are highly complex and often have vulnerabilities
        * many bridges have been drained
      - often come with additional trust assumptions in their governance or multisigs
  - single preconfirmation counterparty
    * extremely high economic guarantees from a single entity
      - much cleaner and robust
  - simpler and safer complex dApps and superstructures
    * imagine a decentralised marketplace
      - ENS for identity at L1
      - ratings and reputation on Base
      - escrow agents on Arbitrum

        - insurance providers on Optimism
    - easier bootstrapping
        * new L2s don't have to start from scratch, lower barrier to entry
            - no need to redeploy defi apps like Uniswap and Aave
                * directly tap into existing liquidity
                * no need to pay for costly liquidity mining incentives
            - no need to pay again for critical infrastructure
                * example: Chainlink
                    - oracle updates are immediately available across the synchronous zone
                * dirty industry secret: the Chainlink monopoly
                    - Chainlink is required for defi to thrive within a rollup
                    - Chainlink gets away charging high rent from rollups given their monopoly
                    - Chainlink uses opaque backroom deals protected by NDAs
    - extensibility by default
        * every new entrant extends the ecosystem, growing the pie
            - no need to choose an ecosystem over another
                * better devex: no need to place all chips on one execution provider
            - less cognitive friction for everyone
        * deeper exploration of niche VMs
            - a single VM (e.g. the EVM) cannot cater for all niches
                * friendly to the long-tail of VMs
                * specialisation of VMs
                    - e.g. friendly to SNARKification (e.g. Cairo)
                    - e.g. friendly to formal verification (Move?)
        * innovation not bottlenecked by the pace of governance
    - smooth upgrades and migrations
        * Uniswap-style governance-free ugprades
            - seamless transition from v1, to v2, to v3
                * shared liquidity through shared sequencing
                    - contrast that with the fragmented liquidity of L1 Uniswap vs L2 Uniswaps
        * especially important as rollups ossify and limit or remove governance
    - collaborative competition ("coopetition")
        * extremely well funded giants (Arbitrum, Optimism, zkSync) will inevitably compete
            - deca-billion ecosystems fighting for scarce talent, users, liquidity
                * a neutral shared sequencer can avoid hundreds of billions of dollars of deadweight loss
                    - 1+1 = 3
            - perfectly safe for ecosystems to compete on tooling, VMs, tokenomics, governance, biz dev
                * for the sake of the industry, don't fight on sequencing
                * promising early signs
                    - Arbitrum "Orbit" charges 10% for licensing rights
                    - Optimism "Superchain" charges 15% for shared governance (Law of Chains)
                    - Matter Labs "hyperchains", Polygon "supernets" show no signs of competing on sequencing
    - cultural and memetic unity
        - shared sequencing is Ethereum "healing" with market-driven incentives
        - simple, beautiful, compelling shared vision and narrative
        - "united chains of Ethereum"
* measuring losses of network effects
    - Melcalfe's law
        * strengh of network effects are quadratic in size
    - example 1: cross-rollup fragmentation
        * Arbitrum TVL ($10B) is half of total rollup TVL ($20B)
            - a 2^2 = 4x loss in potential network effects
    - example 2: L1-L2 fragmentation

        * Arbitrum TVL ($10B) is 1/50th of Ethereum's TVL ($500B)
            - a 50^2 = 2,500x loss in potential of network effects

Part 2: shared sequencing

* creating superstructures
    - shared sequencing "glues" constituent L2s to form a super-L2
        * validiums and other L2s can also join the superstructure
            - doesn't have to be a super-rollup
        * kintsugi metaphor
            - beautifully repairing a piece of broken pottery using gold
                * end result is even more beautiful than the original pottery
* super-everything
    - we can add a "super-" prefix to every concept in the sequencing pipeline
        * super-transaction: a transaction that atomically touches multiple executions
        * super-bundle, super-block
        * super-builder, super-searcher, etc.
* scaling sequencing with PBS
    - heavy-duty shared sequencing is naturally done offchain by sophisticated super-builders and super-searchers
        * shared sequencing is not a scalability bottleneck
            - data availability and execution are the true bottlenecks
            - only super-builders who can sequence most efficiently survive
                * likely much more efficient that today's sequencers operated by rollup teams
                * super-builders will have all the bandwidth, compute, algorithms to sequence optimally
        * important to remember that super-builders, despite the sophistication, are untrusted
* giving up sequencing control will take time
    - initially rollup teams want to retain control over sequencing
        * simplicity
            - deploying a centralised sequencer is the easiest way to get started
            - preconfirmations are especially easy
        * security training wheels
            - hedge against execution verifier vulnerabilities
                * if the sequencer filters invalid blocks only valid block will make it to the verifier
                    - there will be fraud proof verifier and SNARK verifier bugs
                * sequencer-verifier security in depth
                    - like a 2-of-2 multisig
        * MEV protection
            - centralised sequencers can provide MEV protection to users
                * protect swappers from toxic sandwiching
                * protect liquidity providers from toxic flow
                    - LPs are sitting ducks
                * Arbitrum, Optimism, Base all provide basic MEV protection to users
    - need to be patient
        * thing like multi-proving and encrypted mempools will help the maturation process
            - they will accelerate both the decentralisation and the sharing of sequencers
* incentives towards decentralising the sequencer
    - regulatory liability
        * securities concerns with the SEC
        * censorship concerns with the OFAC SDN sanctions list
            - centralised operators could be forced to censor sanctioned contracts
                * Offchain Labs, Optimism Foundation, Coinbase Cloud are targets
    - devops burden

        * rollup teams are specialised in execution, not sequencing
            - centralised sequencers go down accidentally
            - centralised sequencers don't handle spikes gracefully
    - incremental sequencer decentralisation
        * centralised -> federated -> decentralised
* are L2 ecosystems incentivised to adopt a shared sequencer?
    - fair question: isn't the sequencer a cash cow that L2 ecosystems will want to control
        * this is largely a misconception!
        * sequencers today are collecting L1 data fees and L2 execution fees
            - the vast majority of the value originates outside of the sequencer
                * value comes from L1 data availability and L2 execution
                * the sequencer just happens to be the user-facing fee collector
                    - L2s have full sovereignty over execution fees
            - "sequencer fees" is basically MEV
                * rollups don't really want to collect toxic MEV
                * MEV is much smaller than execution fees on L1
                    - 800 ETH/day of MEV (MEV-boost)
                    - 3,200 ETH/day of execution base fees (EIP-1559)
                * application-level improvements will dramatically reduce MEV
                    - sandwiching will go away
                        * Uniswap X, encrypted mempools
                            - auction off the arbitrage, rebate proceeds to swappers
                    - CEX-DEX arbitrage will go away
                        * Sorella, Oval
                            - auction off the arbitrage, rebate proceeds to LPs
        * personal thesis: the MEV gambit
            - best guess: MEV will be less than 1% execution fees
            - MEV gambit: give up a tiny bit of MEV, get a ton more execution fees
            - Espresso is a hedge in case I'm wrong and MEV will be significant
                * Espresso has a shared sequencer design that redistributes MEV back to rollups
* desiderata for the canonical shared and decentralised sequencer
    - economic security
        * for censorship resistance, liveness
            - a 51% attacker can farm highly toxic MEV
                * manipulate oracles, lending markets, DEXes, etc. with censorship
            - a 51% attacker take down the shared sequencer and force mass exits
                * longevity at risk—network effects would reset
    - credible neutrality
        * every rollup and their competitors must feel comfortable opting in
    - incentive-alignment
        * high network effects
        * possible rebates

Part 3: the Ethereum shared sequencer

* the Ethereum shared sequencer hides in plain sight
    - common knowledge that Ethereum provides data availability
    - relatively unknown that Ethereum can also provide shared sequencing
        * L1 proposers can be given the right to sequence rollup transactions
            - with PBS we can have the best of decentralisation and sophistication
                * low-powered and decentralised validators
                * trustless yet sophisticated block builders doing the heavy lifting
        * even less known that Ethereum can provide preconfirmations

    - generalising the Ethereum rollup-centric vision to an Ethereum L2-centric vision
        * may evolve to an Ethereum sequencer-centric vision
* L1 sequencing has huge potential
    - maximum security
        * 29M ETH ($70B) of economic security
            - maximum censorship resistance and longevity
        * Ethereum's security is already assumed for settlement and data availability
            - the Ethereum shared sequencer will live as long as Ethereum itself
            - it's a sunk cost, no new weakest link, no new security assumption
    - maximum credible neutrality
        * Ethereum is accepted as credibly neutral
            - no new token, no new brand: just plain old Ethereum
            - trusted brand, plenty of Lindy
            - reinforces ETH as the unit of account for Ethereum fees
    - maximum network effects
        * largest TVL, by a 50x margin (relative to Arbitrum)
        * some assets will likely never migrate their root of trust
            - ENS names
            - immutable NFTs like CryptoPunks
            - whale portfolios
    - maximum simplicity
        * no need for external sequencer or consensus mechanism
        * reuses L1 proposers and PBS
* key recent breakthrough: we can add preconfirmations to the L1 sequencer
    - write up here: https://notes.ethereum.org/WLuNFaliQiqw7Zhd-7AnmQ?view
    - simple construction
        * allow L1 proposers to pledge collateral to provide shared sequencing and preconfirmations
            - L1 attesters can optionally restake too; see Espresso's design
        * first such L1 proposer in the lookahead is given sequencing rights
            - requires ~10% of L1 proposers to opt in
                * most L1 proposers will opt in, similar to MEV-boost, as it's rational to do so
            - prior proposers can execute preconfirmed transactions and include non-preconfirmed transactions
                * in practice most preconfirmed transactions will execute in the next slot
        * slash shared sequencer for preconfirmation promises that were not honoured
            - two different types of faults
                * safety: a conflicting transaction was executed
                * liveness: no conflicting transaction, but a missed slot
    - no hard fork required
        * I previously thought we need inclusion lists—not true
            - a modification to MEV-boost is sufficient
        * execution tickets will make L1 preconfirmations even better
            - almost every L1 proposer will be a preconfirmer
    - massive opportunity for entrepreneurs (and Ethereum!)
        * a personal focus for 2024
            - happy to provide advice and support on shared sequencing