Decentralized Starknet

“Starknet is an L2 network that enables Ethereum to scale securely and dapps to achieve unlimited scale for transactions and computation”. But if Starknet is a layer two (L2) network built on top of Ethereum (which has a network of a massive 800,00 validators), does it still need to be decentralized to be considered a credible scaling solution ? We will take a deep dive into this question and try to understand what lies ahead in Starknet’s path to decentralization.

There are several properties that are to be expected from a blockchain:***immutability***, ***availability*** and ***censorship resistance***. Zero-knowledge proofs (or validity proofs) which Starknet is built on guarantees one important property and that is integrity of computation. Whenever a transaction is sent to Starknet, it is:

1. received on the sequencer which orders the transaction it receives and batches them for the prover;
2. the prover takes a batch of transactions and generates a certain *proof of integrity* - the proof that guarantees only transactions signed by the sender can be accepted;
3. the proof, in turn, is sent to and verified by the Ethereum network (and its massive validator architecture)

Cryptography behind these validity proofs vastly reduces the time of verification of these proofs on Ethereum which helps Starknet achieve unlimited scalability. The proofs, in turn, ensure that any transaction that is included on the L2, is signed by the sender of the transaction, thereby guaranteeing **immutability**.

**Availability** of a chain implies that for any user who wishes to view or alter the state of a chain is able to do it on-demand. In a secure immutable chain, a user shall not have to worry about the whole chain going down. Achieving a perfect (or near-perfect) uptime can be achieved by decentralizing the sequencer and the full node systems and this is something that we shall see on Starknet sometime in the future. As of September 2023, there are already 3 versions of Starknet full node and 2 versions of the sequencer implementations that are live/under development.

If a chain is secure and always online, there is still the problem of **censorship resistance** that must be considered when talking of a purely decentralized blockchain. Censorship resistance means that the network is designed in such a way that no single entity can restrict the data being added to or retrieved from the blockchain.

[Need to ask Ilya for more information on this]

## Decentralization of full nodes

A variety of implementations for full nodes on Starknet acts as an important indicator of access decentralization to on-chain data. By running different full node implementations, participants contribute to the overall security, resilience, and decentralization of the network. Having independent teams work on their own software reduces the chance of having a single point of failure in case any one of the implementations has an unnoticed bug.Other than that multiple implementations foster innovation and test the network for speed and stability. Running an own full node allows users to make [RPC calls](https://docs.infura.io/networks/starknet/json-rpc-methods), estimate gas fees for mainnet transactions, check status of submitted transactions, use the Trace API, etc. Currently, 3 different implementations of Starknet full node are under development/production:

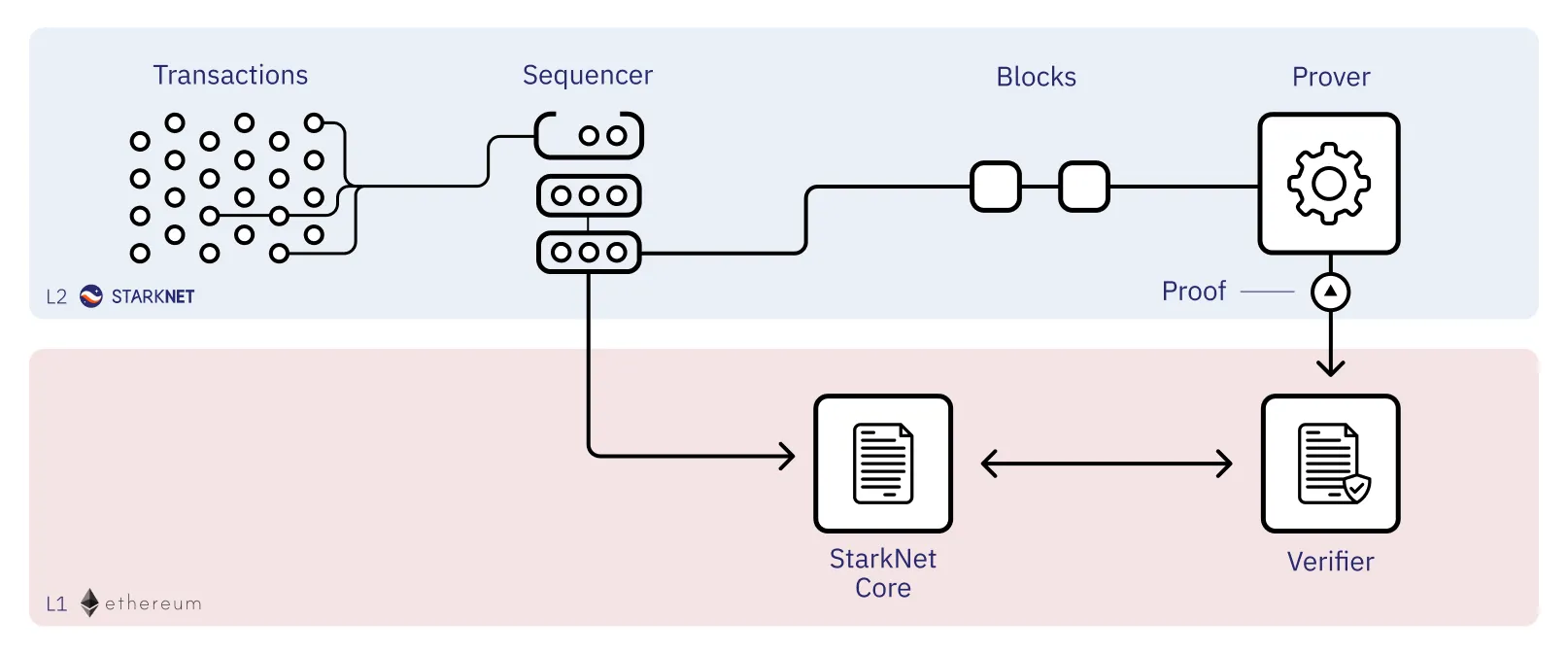
| Implementation | Developed by | Language | Open Source ? |
| --- | --- | --- | --- |
| [Pathfinder](https://github.com/eqlabs/pathfinder) | Equilibrium | Rust | ✅ |
| [Juno](https://github.com/NethermindEth/juno) | Nethermind | Go-lang | ✅ |
| [Papyrus](https://github.com/starkware-libs/papyrus) | StarkWare | Rust | ✅ |

## Sequencer decentralization

Sequencers on Starknet play the pivotal role of receiving transactions, ordering them into a block, and forwarding on to the prover. The sequencer carries the responsibility of fair ordering of transactions (meaning of “fair” he is often a [topic of debate](https://medium.datadriveninvestor.com/focus-on-users-or-profits-a-comprehensive-introduction-to-mev-and-l2-sequencers-135afda229e)) and any outage of the sequencer can cause the network to go offline (and not produce any new blocks). By distributing the role of sequencing across different entities, the system becomes more resilient to outages and increases fairness of transaction ordering by mitigating the risk of single point of manipulation.

Along with the legacy Tendermint-based Starknet sequencer, [Madara](https://github.com/keep-starknet-strange/madara) is another sequencer available on Starknet. Madara, developed by the Keep Starknet Strange team (StarkWare Exploration Team), utilized the Substrate framework to amplify the capabilities of Cairo VM. Substrate is an open-source Rust-based framework used for building blockchains - Polkadot, Kusama, Acala network and many other chains are built using the Substrate framework. Moreover, Madara enables appchain development built on top of Starknet (a.k.a. L3 chains). Developers can create appchains for specific use cases and exponentially increase the throughput. In the future, the team at Madara might utilize storage proofs by Herodotus to achieve cross-chain interoperability.

## Permissionless Provers



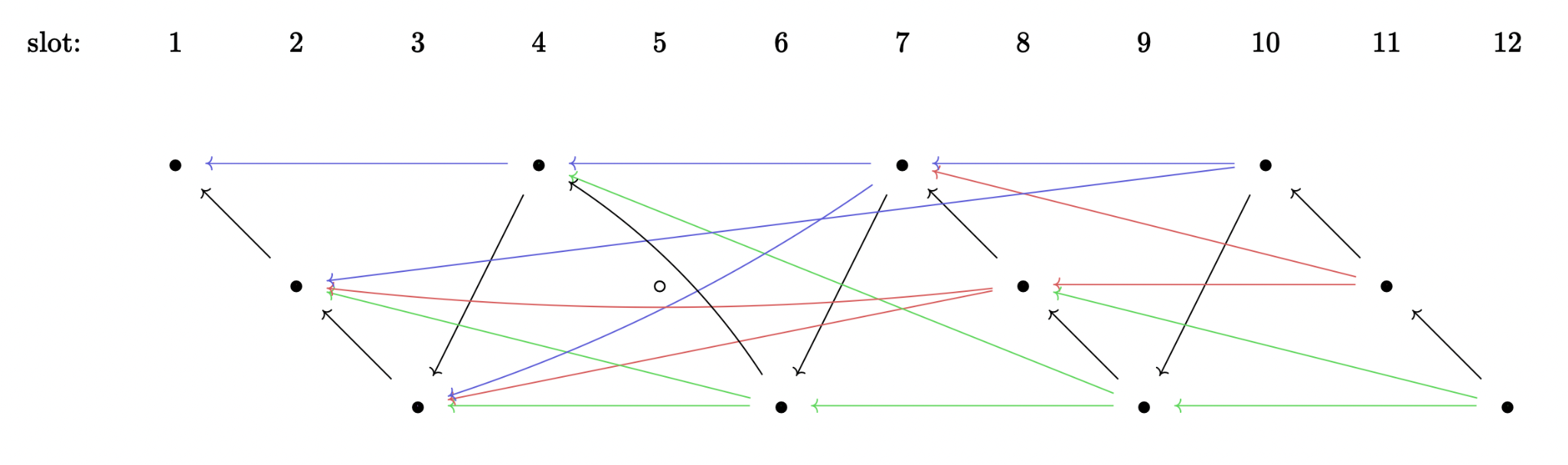
The main caveat of liveness in a system is that the whole system is as strong as the weakest link. In the path from a transaction being signed by the user on a rollup and the state change being confirmed on the base layer, the transaction goes through the steps described previously. One of the most crucial yet resource-intensive steps in this journey on the L2 is the role played by the prover. Although decentralizing the block proposal layer is a significant step, it's not enough on its own. We need to similarly decentralize the proving layer to safeguard against liveness attacks that could compromise the integrity and functionality of the network.

The recent open sourcing of the Starknet Stone Prover (the legacy prover that has been live since 2020) on the [Starknet Summit](https://summit23.starknet.io) was a big step towards decentralization of the prover architecture. Other than that there are at least 2 more implementations of the prover that are under development - [Platinum](https://github.com/lambdaclass/lambdaworks_stark_platinum/tree/main) and [Sandstorm](https://github.com/andrewmilson/sandstorm). Similar to the point made in the chapter about sequencers, multiple prover implementations decrease the chance of a single point of failure in the ecosystem and increases the chances of a 100% for the chain.

| Implementation | Developed by | Open Source ? |
| --- | --- | --- |
| [Stone](https://github.com/starkware-libs/stone-prover) | Equilibrium | ✅ |
| [Platinum](https://github.com/lambdaclass/lambdaworks_stark_platinum/tree/main) | LanbdaClass | ✅ |
| [Sandstorm](https://github.com/andrewmilson/sandstorm) | Andrew Milson | ✅ |

Today, Starknet has a single sequencer and a single prover that maintain the entire infrastructure of Starknet (and even then, Starknet has managed to become the largest L2 by throughput). Having multiple implementations of each brings us one step closer rto decentralization but it is not enough on its own. We need to similarly decentralize the *proving layer* to accommodate safeguard against liveness and censorship attacks.

A recent improvement [proposal](https://community.starknet.io/t/starknet-decentralized-protocol-vii-chained-proof-protocols-braiding/18831) for the proving architecture accommodates multiple prover entities enabling a decentralized proof production protocol. For gaining a deep understanding of the proposal, you are invited to read the proposal on Starknet Community page. However, let us give a superficial insight into the suggested solution by looking at the diagram [here](https://q.uiver.app/?q=) (also given below).



Black arrows represent the blockchain and colored arrows represent proofs

Similar to a slot schedule that is on Ethereum, a deterministic slot proposal schedule will be allocated to each slot of time on Starknet. The slots are also divided into k strands (k=3 in the diagram above) and each slot leader proposes a block along with the proof for the previous block in its strand. More complex chaining algorithms are discussed in the post to handle cases of missed slots and develop more efficient proving.

For this system to be truly effective, a free market for proving needs to be there. Only with a market-driven approach for requesting proofs, the incentives for individual provers to maintain a consistently high level of availability become strong enough to ensure a robust, resilient, and efficient network.

On top of it provers may be disincentivised from going offline or trying to submit false proofs by slashing. Provers will also need to stake collateral which could be slashed in case of very low uptime or any malicious behavior.

## Other protocol upgrades

As discussed in a [post](https://community.starknet.io/t/simple-decentralized-protocol-proposal/99693) by [Ilya Volokh](https://www.linkedin.com/in/ilia-volokh-ikv/?originalSubdomain=il) the path to decentralization of Starknet has multiple components. Some of those we have discussed above, but some are still left unexplored.

One of the key features to be implemented in the roadmap for Starknet is a proposed leader election protocol for Starknet. Similar to how slot leaders are deterministically defined for future slots, leaders will be assigned for specific slots in time that are supposed to produce blocks for Starknet. The leaders will be randomly (Randomness will be inherited from the L1 set of validators) assigned to slots. Users will be able to stake assets (STRK token) via the "bookie" contract, which serves as the staking contract on Ethereum's L1.

The next in line is the implementation of Tendermint consensus protocol for Starknet. The Tendermint protocol has been the foundation for the widely used Cosmos protocol and its infrastructure of [appchains](https://cosmos.network/appchains/). Starknet aims to build on the Tendermint algorithm with some modifications for integrating into Starknet. These include *transaction streaming* (Unlike traditional Tendermint where each block is broadcast in one chunk, this variant will stream transactions to improve throughput and reduce latency), *coupling sequencing and execution* (integration of sequencing directly with execution of transactions), and *recovery from corrupted consensus* (introduction of a fork choice rule invalidates a Starknet block if it hasn't been finalized on L1 (Layer 1) for a certain number of hours).

## Decentralization of Decision making and Governance

Protocol upgrades (such as the ones discussed above) are necessary for the continuous incremental upgrades to the Starknet ecosystem. However, just as a country needs a government to be run, a network needs decentralized decision-making infrastructure to be set up for successful operations. In order for network participants and relevant stakeholders to cast their opinion in every decision, governance infrastructure needs to be established.

On Starknet each major protocol upgrade goes through a governance decision-making process. Every protocol version upgrade is first deployed to Starknet testnet. Voters will then have a few days period to examine and test the upgraded version as it runs on Goerli. After testing, governance participants can vote “Yes” or “No” on the upgrade proposal (e.g. [this](https://snapshot.org/#/starknet.eth/proposal/0xe3666dcfaecdd871bbad1c2f882861d2bb0a26eb27aeab36679c498f2a1c86c2) is the voting page for one major recent upgrade of Starknet - Alpha v0.12.0). All upgrades are discussed on the [Starknet Community Forum](https://community.starknet.io).

Moreover, not all SNIPs (StarkNet Improvement Proposals) need to come from Starknet. Some proposals and discussions ([transaction v3 SNIP](https://community.starknet.io/t/transaction-v3-snip/98228/1), [volition: hybrid DA solution](https://community.starknet.io/t/volition-hybrid-data-availability-solution/97387)) have been initiated by developers out of the Starkware core team. This signifies increasing interest of developers around Starknet and reinforces decentralization of the system to encourage creativity and active engagement.

## Governance Hub and Diverse Global Team

With the ever-increasing pace of innovation on Starknet frequently required decision-making processes have underlined the need for a Starknet governance hub (similar to Maker DAO’s [voting portal](https://vote.makerdao.com)) that streamlines and conveniently allows governance participants to vote on proposals, view latest updates, and engage in discussions.

Globally diverse teams within StarkWare encourage multiplicity of perspectives, opinions and skills. Diverse in thought and understanding allows for the protocol to scale with global needs of the users. Such a geographic diversity acts as an additional layer of decentralization such that no entities located in one locality have too much power of development of the network. The benefit expands beyond the decentralization of developers behind Starknet since globally available teams would also invite contributions from a wider range of stakeholders from different corners of the world.

In conclusion, Starknet's journey towards full decentralization is not just a technical, but also a community-driven process. With advancements in all the key components of the infrastructure, Starknet is shaping up to be a robust, censorship-resistant, and highly available Layer 2 solution.