FCAT

Behind the Blocks: How Solana Staking Powers the Network

By Mia Galvin & Kevin Kelly

October 2025

As a rapidly evolving, high-speed blockchain, Solana processes tens of millions of transactions daily. By locking up SOL, stakers earn rewards while enabling validators to sustain this throughput and secure the network. This report explores the mechanics and economics behind staking—and why it's vital to Solana's speed, security, and decentralization.

The Solana Staking Ecosystem

The Solana network is supported by approximately 1,000 nodes distributed across a diverse, global set of validators who secure the network and contribute to its decentralization. These nodes collectively stake 67% of the circulating SOL supply—roughly ~400m staked SOL out of a ~600m supply—with the majority of stake concentrated in the US, Netherlands, U.K., and Germany.[1]

Solana runs on a delegated proof-of-stake (DPoS) model in which SOL token holders may choose to natively stake and earn rewards by delegating to a validator. Serving as the backbone of the network, validators participate in consensus by continuously voting on the state of the blockchain, verifying transactions, and producing blocks.

It's important to understand what "decentralization" actually means on a given network.

Solana's decentralization depends on both the distribution of staked SOL and the diversity of its validators. At the time of writing, the top 21 validators control more than 33% of total staked SOL, a **superminority** comprised mostly of large entities, including major centralized exchanges (CEXs) and blockchain infrastructure providers.

Within PoS blockchains, a "superminority" refers to the minimum number of nodes that hold enough staking power to compromise or disrupt a network.

Beyond stake distribution, **software diversity** (otherwise known as "client diversity") also plays a critical role in network resilience. If too many validators run the same client software, a single bug or exploit could lead to widespread performance issues.

Today, the Jito-Agave client represents over 70% of Solana stake. This is in direct contrast to Ethereum, on which no client has over 66% share of the network.[2] Multiple alternative client implementations are currently in development, and as Solana matures, broader adoption is likely to help improve fault tolerance and reduce systemic risk.

There are four main ways for users to stake their SOL

Custodial Staking: Users can stake through custodians (such as a CEX or institutional provider) that manage staking wallets and tokens on their behalf.

Liquid Staking: Users may stake their SOL to a smart contract or staking pool and receive a liquid token (LST) in return—these tokens can be used in DeFi while still earning staking rewards after protocol and validator fees are applied.

Delegated Staking: Users can hold their SOL in a self-custody (or "non-custodial" wallet) and delegate to chosen validator(s) to earn rewards. This increases validators' stake-weight and, therefore, their voting power, while users retain full control of private keys and funds.

Self-Staking: Advanced users may run their own validator node to self-stake and attract external delegators—this option requires technical expertise, significant hardware resources, and ongoing maintenance.

An Overview of Staking Mechanics

Protocol Architecture

The Solana blockchain combines two consensus mechanisms—Delegated Proof-of-Stake (DPoS) and the network's unique Proof-of-History (PoH)—to maintain fast block production and transaction confirmation times. PoH functions as a globally-synchronized clock that timestamps transactions and allows nodes to quickly and accurately verify transactions, while DPoS allows users to stake their SOL with validator nodes who secure the network.

Solana slots are around 400ms, and each leader validator is assigned four consecutive slots (1.6 seconds), during which they produce blocks and earn block rewards.[3] There are 432,000 slots in one Solana epoch, which takes around two days at peak network efficiency.

On the Solana network, Remote Procedure Call (RPC) nodes receive transactions from users or decentralized applications (dApps). These are then efficiently forwarded using the Gulf Stream protocol and leader schedule, which together eliminate the need for a traditional mempool—a waiting area in which incoming transactions typically queue. Instead, transactions are directly routed to the current and upcoming leaders, specifically their Transaction Processing Unit (TPU), allowing for faster validation and block production. Solana also leverages QUIC, a transport layer protocol, to reduce latency and optimize transaction forwarding.[4] The standard Solana Labs client, Agave, prioritizes transaction traffic from trusted RPC nodes that peer with validators, using a system called Stake-Weighted Quality of Service (SWQoS).

If the leader is running the **Jito-Solana**[5] client, transactions are directly forwarded from the RPC nodes to the Jito Relayer, which functions as an outsourced TPU proxy, verifying transactions and reducing the load on the validator. The Relayer then submits transactions to both the validator and the Jito Block Engine. The Block Engine connects MEV searchers and validators through an off-chain auction space, receiving transaction bundles from MEV searchers, simulating the highest-paying combination, and then forwarding to the leader validator for inclusion in their block.

Since **Jito-Solana** is a fork of the original Agave client, this process incorporates similar mechanisms found in the validator's TPU, while also improving filtering, spam reduction, and reward capture.

Only the leader validator for a given slot will receive transactions and assemble them into a block, following the structure detailed below:

- The leader schedule is pre-determined by stake-weight; RPC and validator nodes know selected leaders in advance and can prepare accordingly.
- Once the leader produces its assigned block—containing verified transactions including MEV bundles—it's broken into shreds.
- These shreds are broadcasted through the Turbine mechanism (a tree-like structure based partially on stake-weight) to non-leader validators, who then verify the block.
- Once ready, the validators participate in consensus, and if the block is voted on by a stake supermajority (≥ 66%), it is finalized.

Validators also continuously exchange data and information with their peers through the gossip network—a peer-to-peer (P2P) communication system between nodes that allows for maintenance of the cluster's health.

Delegation on Solana involves both a warm-up and cool-down period. When initially delegating to a validator, it takes an epoch for SOL to fully activate and begin earning rewards (the "warm-up"). Similarly, when a user chooses to unstake from a validator, the funds do not automatically become inactive, and the unstaked SOL enters a "cool-down" period. This deactivation takes another epoch before the user can fully withdraw funds and return them to a wallet.

For context, a maximum of 25% of the network's total active stake can be activated or withdrawn per epoch,[6] and in an edge-case scenario wherein any stake beyond the limit is withdrawn, the remaining stake is processed during the following epochs.

Validator Operations

Hardware & Software Requirements

Running a Solana validator demands significant resources. At the time of this report, a robust, on-premises server is recommended. Most validators run bare metal servers in data centers given the high processing requirements and high costs from cloud hosting providers.

Recommended server specifications include a high-performance processor with at least 24 cores and high clock speeds of >3.5 GHz, 256GB of RAM, and 4TB of SSD storage across a few NVMe drives.

Competitive validation also requires a stable, high-bandwidth internet connection with at least 10 Gbps symmetrical speed. It is essential that network validators remain synchronized and constantly online, necessitating a reliable power supply and consistent software upgrades. Future software clients in development—namely Firedancer—may reduce complexity and required resources.

By comparison, Ethereum validators use far less processing power and have far lower hardware requirements, as detailed in the table below. This discrepancy is largely related to how the two networks are designed—i.e., Ethereum has slower block times that permit more accessible hardware, whereas Solana block times are over 20 times faster, and that speed is predicated in part upon more efficient hardware built to minimize latency.

Figure 1. Comparative Network Hardware Requirements on Solana & Ethereum

	Solana Validator[7]	Ethereum Validator[8]
CPU	24+ cores	4+ cores
CPU Clock Speed	3.5-4 GHz	3.0+ GHz
RAM	256 GB	32 GB
Storage	4 TB SSD	3 TB SSD
Network	10,000 mbps upload/download	20 mbps upload/download
Clients	Validator	Execution, Consensus, Validator

Validator Responsibilities

Validators have numerous duties across the network tied to consensus, block production, network maintenance, and security. Their primary responsibilities include:

» **Validate and vote on blocks.** Validators participate in ongoing consensus efforts by voting to confirm and validate blocks every ~400ms. Therefore, they must maintain consistent connection with their peers in Solana's gossip protocol, which propagates information across the network.

» **Produce blocks.** When selected as the leader for a given slot, a validator receives incoming transactions and must produce the associated block.

» **Upgrade and monitor software.** Responsible parties regularly update the validator client and monitor validator health to stay compatible for protocol changes and ensure uptime.

» **Secure keys.** Validators must securely manage their keys to prevent unauthorized access, which could compromise performance or result in stolen funds.

Key Management

For Validators

There are three main keys associated with a given validator: identity account, authorized withdrawer, and vote account. All three serve various functions as described in Figure 2. However, in all cases, the identity key must stay hot to perform ongoing validator duties, while the withdrawer account is the most sensitive (as it controls the validator's funds).

At a high level, the identity account acts as the validator's wallet and is similar to a regular user account: it can send and receive SOL and interact with programs. In contrast, the vote account is unique to validators and is used specifically for participating in consensus. It holds the validator's voting history and serves as the reference point for delegators when staking.

Figure 2. Use Cases by Key

Keys	Number of Keys	Use Case(s)
Identity Account (Hot Key)	One (1) for every newly-published validator; can be rotated by the withdrawer key if desired	Serves as the wallet and unique identifier for the validator; signs transactions, verifies blocks, and used to pay for vote fees and sign gossip messages
Authorized Withdrawer (Cold Key)	One (1) per published validator	Used to withdraw staking rewards and can be leveraged to reassign the identity account; has control over validator-level changes
Vote Account (Cold Key)	One (1) per published validator; remains consistent for the validator's total lifetime	Records validator votes and credits for reward calculations, as well as stores validator commission and authorized withdrawer; delegators enter the vote account public key to stake SOL to a validator

For Delegators

In order to delegate to a validator from a non-custodial wallet, users must hold their own private keys. Similar to leading key management practices for validators, best practices for delegators include backing up seed phrases, using cold wallets to store meaningful quantities of SOL long-term, and staying updated with wallet and network changes.

When one delegates from their wallet, a "stake account" is created, assigned to the stake program, and delegated to a validator. The wallet signs this transaction and, by default, is set as both the stake and withdraw authorities. These can be reassigned to different accounts for added security if desired.

A Solana stake account is linked to a single validator's vote account and accumulates staking rewards. Once created, it cannot be topped up with additional SOL, and the funds are locked up. Partial withdrawals are possible by splitting the account and deactivating portions.

The stake authority controls delegation and redelegation, while the withdraw authority controls liquidation of the stake account after deactivation. Losing access to the withdraw authority means users cannot recover staked or unstaked funds and reset other authorities. Experienced stakers often separate stake and withdraw authorities to secure them individually—commonly using hardware wallets or multisig for withdraw authority. For users staking through simple, user-friendly UIs (like Ledger Live or Marinade-native staking), both authorities default to the wallet key, so protecting that key is sufficient.

Economics

At launch, the Solana network established a native token issuance plan to prevent dilution, starting with an 8% inflation rate, which would decrease annually by 15% and ultimately flatten at 1.5%.[9] While this schedule can be used to estimate the inflation rate at any point in time, the actual rate—which reflects net supply growth—differs (and is typically higher) due to the variability of transaction volumes and burning mechanisms. Since SOL doesn't have a fixed supply, 50% of transaction fees were burned until early 2025, when SIMD-0096 ruled to send 100% of priority fees to validators to increase revenue.

Block production rewards (like priority fees and Maximum Extractable Value, or MEV) heavily depend on a Solana validator's **stake-weight**: the staked balance relative to total active staked SOL across the network. The more SOL staked to a validator, the higher likelihood of that validator getting assigned a slot on the leader schedule with the opportunity to produce a block. As less native SOL is issued and inflationary rewards decrease over time, validator earnings will increasingly depend on block rewards.

Revenue Streams

There are three main revenue streams for validators: inflation rewards, block rewards, and MEV rewards.

Inflationary rewards are natively issued SOL funded by the network's inflation rate. Inflationary rewards are distributed at the end of each epoch and have historically comprised the majority of Solana staking rewards, calculated based on staking dynamics and validator performance in consensus.

A Solana validator's uptime allows them to earn vote credits, which are translated into reward points proportional to their staked balance. Upon implementation of **SIMD-0033** on Solana, timely vote credits were added to reduce vote lagging and promote low latency—versus an alternative model in which validators earned a single vote credit per successful vote.

With the introduction of this proposal, validators now have a grace period of two slots for their vote to land and earn a maximum of 16 credits, varying with higher vote latency.[10] The network's staked ratio (total active stake divided by total circulating supply) directly impacts staking yields, as inflation rewards are distributed only to stakers, and higher participation means lower individual yields.

Block rewards are the sum of vote and non-vote transaction fees that validators collect when proposing blocks.[11] Users automatically pay the fixed base fee of 5,000 lamports (0.000005 SOL) per transaction, 50% of which gets sent to the validator who processes them in a given block. They can also incentivize validators to process their transactions by including priority fees (100% paid), calculated in micro-lamports as the current compute unit (CU) price multiplied by the CU limit consumed by the transaction. During periods of network congestion or high activity, the CU price can increase, raising priority fees. Validators also collect vote fees from the vote transactions included in their block.

Maximum extractable value (MEV) rewards refer to the selection and re-ordering of transactions to create a more profitable block with higher rewards. On Solana, the vast majority of validators run the Jito client on their existing hardware, leveraging Jito's Relayer and Block Engine to earn associated revenue.

While there are beneficial forms of MEV (like arbitrage and liquidation protocols solvent), malicious MEV practices do occur, and they can include activities like sandwiching and front-running.

What types of MEV exist on-chain?

Arbitrage exploits price differences for the same asset across different exchanges—e.g., purchasing lower priced SOL on a CEX and selling higher on a DEX. This practice actively helps align prices across markets, improving efficiency and liquidity.

Liquidation occurs in DeFi lending protocols when a borrower's collateral falls below a certain threshold, triggering a liquidation process. Searchers put together transaction bundles that may include debt repayment and collateral seizure, protecting lenders and system health.

Sandwich attacks occur when a bad actor observes a pending transaction likely to move the price of an asset. They quickly place a buy order right before the victim's trade **(front-running)** and a sell order after **(back-running)**, profiting from the price change while the victim suffers a loss.[12]

However, these attacks are more common within the Ethereum ecosystem—especially since Ethereum's public mempool allows for easier influence over transaction ordering. Solana's high throughput, low latency, and lack of a public mempool all serve to restrict visibility into transactions and limit the opportunities for malicious MEV. Jito also promotes healthy MEV activities and filtering through spam prevention, community governance, blacklists, and incentives.

Flow of Funds

As illustrated in Figure 3 below, the three types of staking rewards for validators and delegators flow into different wallets. Rewards for delegators are dependent on the commission rates set by validators: one for inflationary rewards, as well as a second for MEV if they are running Jito—both of which are adjustable.

The validator's vote account, which stores vote data and serves as a reference for delegators to that validator, collects associated percentages before rewards are distributed to delegator stake accounts. Since any SOL in a vote account is not earning rewards, validators can choose to set up automated withdrawal and re-stake mechanisms instead of letting their inactive SOL accumulate.

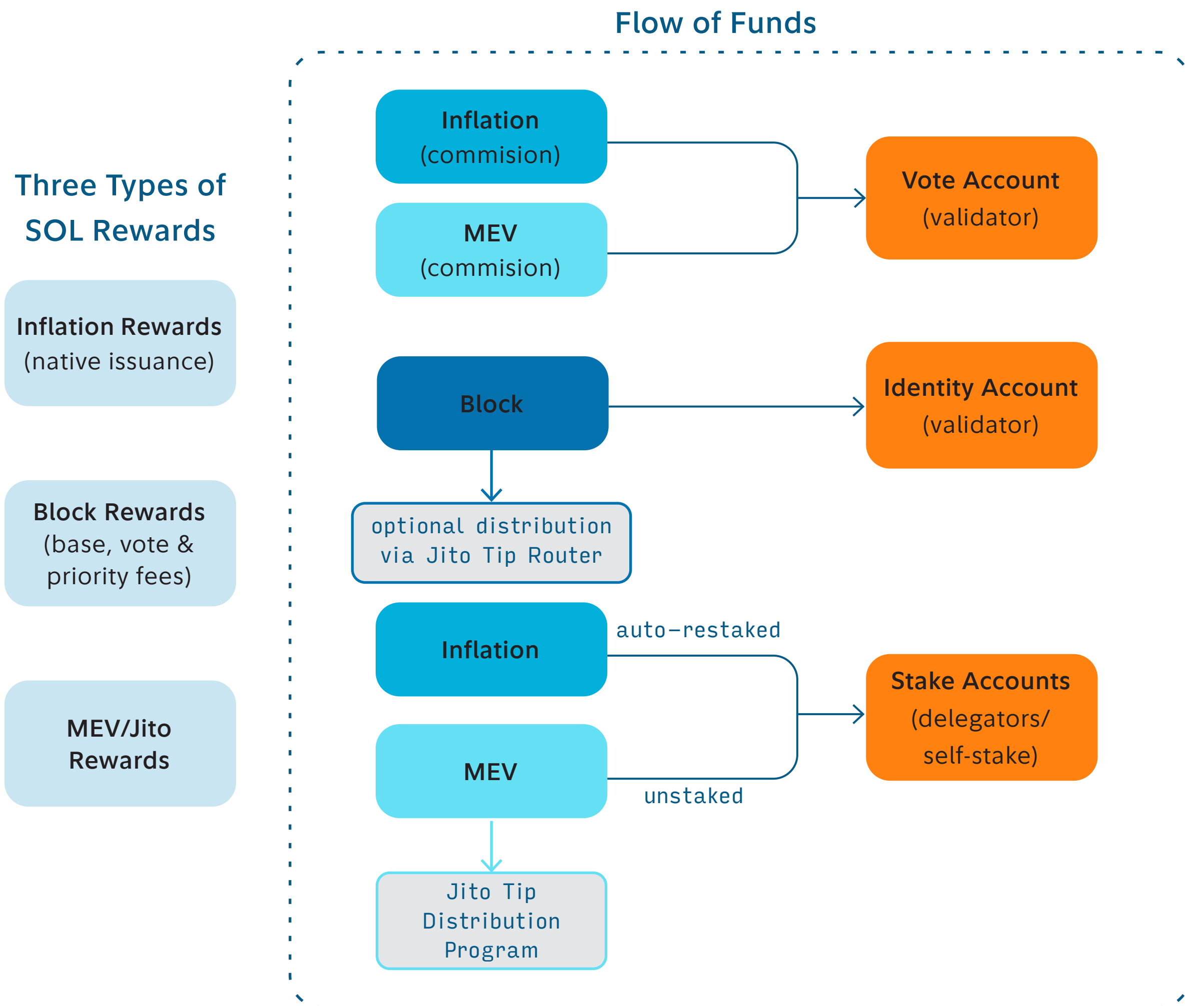
Each time stake is delegated to a validator, a stake account is created, pointing all of the SOL toward the validator's vote account. These are the only accounts in which SOL stake is activated and can begin accruing rewards.

Inflationary rewards are automatically restaked into the delegated stake accounts and distributed proportionally to the amount staked per delegator, compounding each epoch. MEV rewards are also sent to stake accounts by Jito's TipRouter but are airdropped as un-staked SOL, waiting to be claimed by either withdrawing to a user's wallet or re-staking.

The validator's identity account, which also functions as a user wallet, collects all block transaction rewards. It's up to the validator to distribute these rewards to the delegators—however, this can be impractical as a distribution mechanism is not built into the protocol.

However, automated priority fee distribution will be implemented following SIMD-0123 and is currently a voluntary option through Jito's protocol, which does collect fees. The existing identity account also functions to pay validator vote fees, which, at ~1 SOL a day, make up a significant cost for Solana validators.

Figure 3. Flow of Funds on the Solana network



How Users Can Select A Validator & Monitor Performance

Users can choose to delegate to a preferred, trusted validator based on the following criteria:

Validator operator. Validators backed by familiar entities can appear more trustworthy and attract more stake.

Commission rate. A lower commission rate is ideal for delegators to retain more rewards; however, it is important to note that validators who advertise 0% usually retain transaction fees.

Historical uptime. Uptime directly impacts revenue, stake delegation, and reputation, making it important for validator reliability and reward optimization.

Software used and version. Software clients vary in their hardware and revenue implications for delegators; it is also important that the operator stays current with software updates, since falling behind can lead to lower performance.

Additionally, the performance metrics and key performance indicators (KPIs) reported on public dashboards like Validators.app and StakeWiz are helpful for monitoring validator health, with opportunities for greater detail and visualization. These platforms also report validator commission (typically around 5-10%), enhancing transparency of the different commission types, along with undisclosed commission raises.

The uptime (or "participation rate") indicates the amount of time a validator is online and voting, which should ideally be >99%. The skip rate is the percentage of assigned leader slots in which a validator fails to produce a block, meaning that validators maximize block rewards the closer they are to a skip rate of 0%.

Skip rate serves as an especially important metric for validators with low stake-weights who are already unlikely to land a leader slot, since a high skip rate can lead to problematic variance in rewards.

Root and vote distance also reflect validator performance, measuring how closely synchronized the validator is with the latest state of the blockchain (for instance, a greater distance can suggest delays in transaction processing and vote submission).

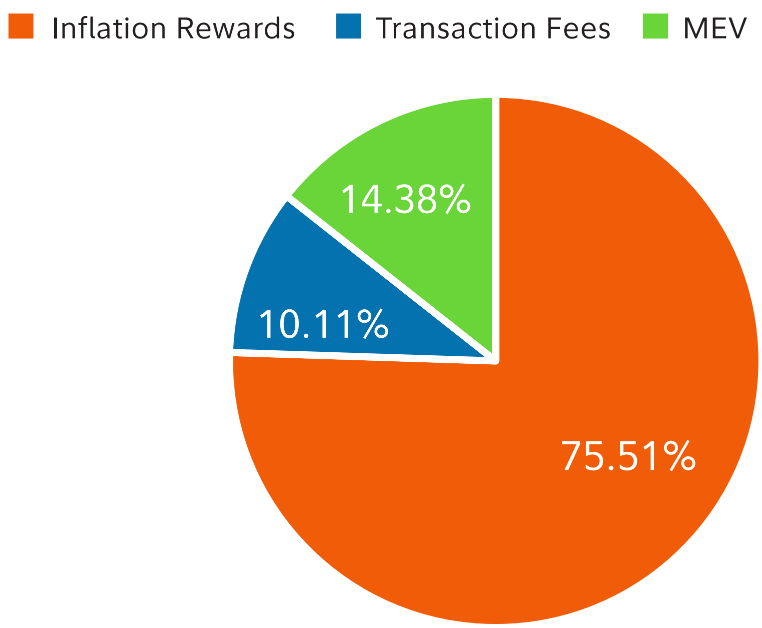
Financial Projections

While the **network APY usually remains consistently between 6% and 8%**, Solana's financial projections are quite sensitive to the volatility of the larger crypto ecosystem and constant changes to network protocols. Modeling different scenarios requires a comprehensive understanding of staking dynamics and reward structures, and annual operating costs—including hardware, maintenance, and (primarily) vote fees—can exceed $100,000.

Many validators also stake their own SOL and collect the full rewards, achieving higher profitability and possibly operating at lower commission rates than fully delegation-based operations. Currently, **a validator with 10,000 SOL in self-stake or 50,000 SOL in delegated stake would likely break even.** That said, once priority fees are distributed in-protocol at a standard commission rate, validators relying solely on delegated stake could need up to 150,000 SOL to remain profitable.

For context, the financial model leveraged for this report used historical stake ratio, price, and reward averages, along with the fixed token issuance schedule, to make key assumptions and forecast future yields (as shown in Figure 4).

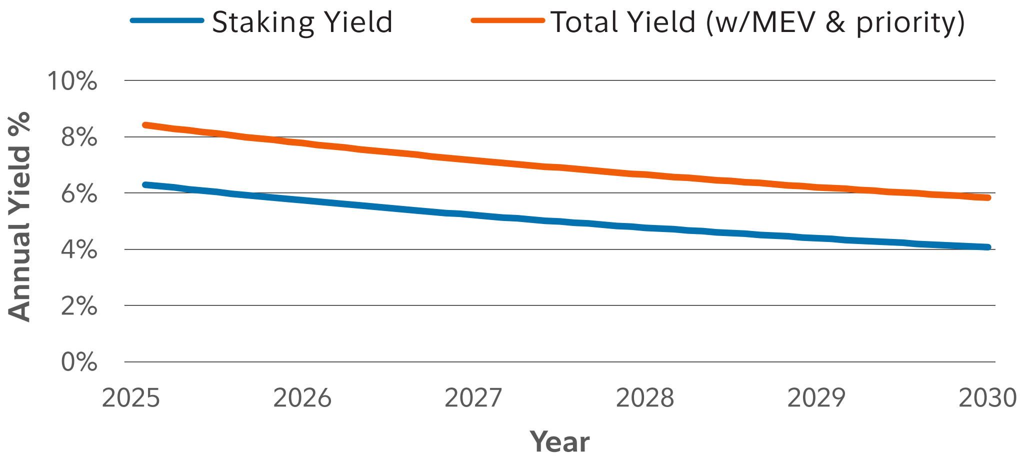
Figure 4. Validator Revenue Sources



Source: FCAT Research, 08/08/25

Fueled by a fast-moving developer community and ongoing protocol upgrades, Solana's revenue model has shifted noticeably. While inflation rewards still dominate at 75% of validator revenue, MEV and priority fees have emerged as key contributors.

Figure 5. Five-year Yield Projections



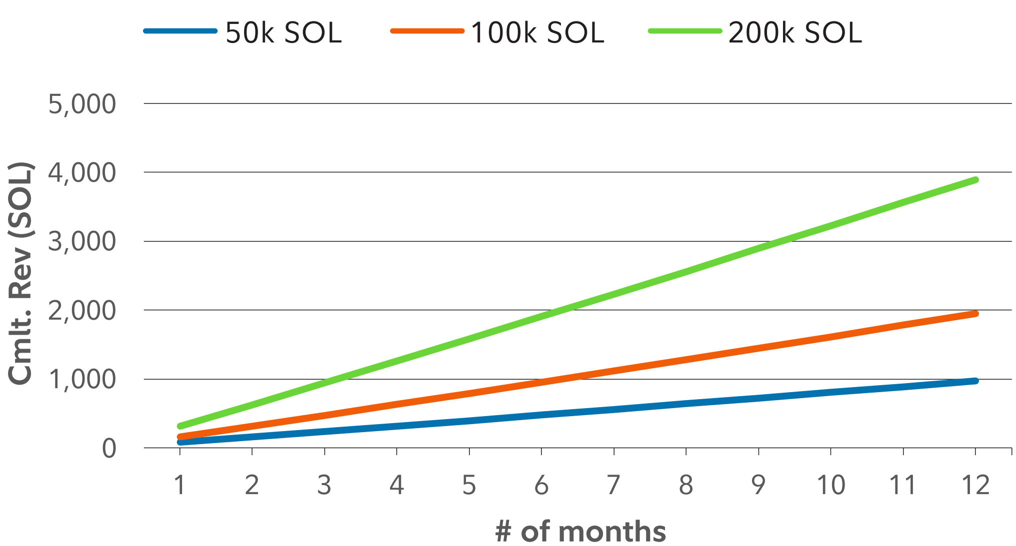
Note: MEV and priority fee yields are calculated using historical block reward averages and modeled as a fixed percentage premium above base staking yield.

Source: FCAT Research, 08/08/25.

Due to Solana's predetermined mechanisms for inflation reduction, base staking yields decline as inflation drops 15% annually to its 1.5% target. Validators capturing priority fees—and especially MEV—maintain higher yields, essential for a low-inflation future.

A validator with 100,000 SOL in managed stake (10% self-stake, 90% delegated) and a 5% commission rate for both inflation and MEV rewards can serve as a baseline case to project expected rewards. As shown below in Figure 6, rewards scale linearly with a validator's active stake balance—the base case generating around 2,000 SOL in annual revenue.

Figure 6. Annual Validator Revenue by Staked Balance



Note: Revenue projections assume constant network conditions and validator performance.

Source: FCAT Research, 08/08/25.

Solana validator profitability is primarily shaped by staked amount, commission, performance, and operating costs. While baseline projections show positive returns (especially for validators with significant self-stake), high fixed costs mean smaller operators require substantial delegator support to break even. Attracting delegators through consistent performance and a strong reputation increases stake-weight and revenue opportunities.

Upcoming Network Changes

The Solana ecosystem is constantly maturing, as Solana Improvement Documents (SIMDs)[13] and Jito Improvement Proposals (JIPs)[14] continue to be drafted, debated, and ultimately decided on via on-chain voting.

The following upcoming SIMDs are primed to alter how staking works on Solana:

- **SIMD-0123** is an on-chain protocol upgrade designed to distribute block rewards from validators to delegators; the document has been approved but has yet to hit the Feature Gate Activation Schedule, which pushes SIMDs through development, test, and mainnet.
- **SIMD-0286** was pitched by Jito Labs in May 2025 to increase the compute block limit from 60 million to 100 million CU. Four months later, Firedancer builder Jump Crypto proposed **SIMD-0370**[15] to remove that compute limit altogether following the deployment of Alpenglow. With this change, higher-performing validators with more powerful hardware could produce larger blocks, packing in more transactions and earning higher fees.
- **SIMD-0204** is currently pending testnet activation and is designed to create an on-chain program that reports when a validator commits a slashable infraction. While Ethereum incorporates a penalty system into the network, Solana was originally built without any slashing mechanisms and is now exploring implementation.
- Future proposals, such as **SIMD-0212**, are expected to introduce the mechanisms for enforcing penalties to further incentivize honesty and accountability among validators.

For Jito-related changes, proposals like JIP-4, 15, and 22 illustrate the community's prevailing emphasis on filtering and preventing malicious MEV activity, creating blacklists, and establishing a dedicated blacklist operations committee.

Additionally, the Jito client is facing a significant system upgrade, replacing its Relayer and Block Engine with specialized Block Assembly Marketplace (BAM) nodes.[16] These nodes operate within Trusted Execution Environments (TEEs) and aim to improve how blocks are built and transactions are sequenced, increasing priority fees and Jito tips per block. As of September 2025, select validators are being onboarded to BAM mainnet, with the following phases aimed to increase adoption and stake. This upgraded infrastructure is designed to reduce the harmful effects of MEV through encrypted mempools within the TEEs, while allowing the more beneficial forms of MEV to operate efficiently.

Alongside the current Solana validator clients undergoing updates, new ones are being created and tested to improve network efficiency and client diversity. **Frankendancer-Jito** is a promising hybrid solution, running Firedancer and Agave side-by-side with integration into Jito's block engine. So far, this client has been adopted by about 20% of Solana validators.

Firedancer is an independent validator client for Solana currently on testnet, enabling higher transaction throughput for the network at lower operating costs. In terms of volume, the client has a substantially higher theoretical throughput compared to Agave's current ~4,000 tps.[17]

A new modified version of the Solana client, **Paladin**, has captured approximately 6% of stake. Paladin protocols flag risky validators to improve accountability and transparency, while simultaneously upgrading security practices and optimizing block rewards.

Finally, the upcoming **Alpenglow** consensus algorithm will revolutionize Solana's protocol by enabling faster transaction finality and the removal of vote fees through off-chain consensus activity. The protocol is intended to significantly reduce validator costs and replace existing mechanisms like TowerBFT and Proof-of-History. Alpenglow also introduces new components like Votor (a finality engine) and Rotor (a mechanism for shredding and broadcasting blocks). Alpenglow's launch is targeted for sometime between late 2025 and early 2026, benefitting both validators and delegators alike through improved efficiency and flexibility, as well as laying the foundation for future innovations.

Conclusion

The Solana network is powered by validator nodes across the world, yet steep hardware requirements and high operational costs remain barriers to broader participation. For most users, staking exposure comes through delegation to trusted validators—an approach that earns rewards and strengthens decentralization.

However, staking is far more than passive yield; it underpins consensus, transaction processing, and the resilience of the entire ecosystem. As Solana continues to innovate and strengthen its staking mechanics, the role of validators and delegators will keep evolving—making this an exciting chapter for those with a passion for speed, security, and shaping the future of decentralized networks.

Works Cited

1 "Validators." Solana, https://solana.com/validators. Accessed 4 Sep. 2025.

2 "Solana: SOL Staking." Blockworks, https://blockworks.com/analytics/solana/solana-supply-staking-and-validators. Accessed 4 Sep. 2025

3 "Transaction Confirmation and Expiration." Solana, www.solana.com/developers/guides/advanced/confirmation. Accessed 4 Sep. 2025.

4 Chern, Ryan. "Turbine: Block Propagation on Solana." Helius, 7 Feb. 2025, www.helius.dev/blog/turbine-block-propagation-on-solana. Accessed 4 Sep. 2025.

5 "Jito." Jito, www.jito.wtf. Accessed 4 Sep. 2025.

6 "Staking." Solana, www.solana.com/staking. Accessed 4 Sep. 2025.

7 Solanahcl. "Hardware Compatibility List for Running a Solana Mainnet Validator." GitHub, https://github.com/solanahcl/solanahcl. Accessed 6 Oct. 2025.

8 EthStaker. "Hardware Requirements - EthStaker Knowledge Base." EthStaker, https://docs.ethstaker.org/hardware/hardware-requirements/. Accessed 6 Oct. 2025.

9 "Staking." Solana, www.solana.com/staking. Accessed 4 Sep. 2025.

10 Solana Foundation. "0033: Timely Vote Credits." GitHub, github.com/solana-foundation/solana-improvement-documents/blob/main/proposals/0033-timely-vote-credits.md. Accessed 4 Sep. 2025.

11 "Transaction Fees." Solana, www.solana.com/docs/core/fees. Accessed 4 Sep. 2025.

12 Yan, Hwee. "What Are Sandwich Attacks in Crypto and How to Prevent Them?" CoinGecko, 20 Dec. 2023, https://www.coingecko.com/learn/sandwich-attacks-prevention-crypto. Accessed 30 Sep, 2025.

13 "Solana Improvement Documents." GitHub, github.com/solana-foundation/solana-improvement-documents. Accessed 22 Oct. 2025.

14 Jito Network. "Latest Discussions." Jito Network Forum, forum.jito.network/latest. Accessed 22 Oct. 2025.

15 "Firedancer devs' new proposal aims to make Solana even faster." Coin Telegraph, 28 Sep. 2025, https://cointelegraph.com/news/firedancer-devs-remove-solana-block-limit-to-speed-up-network. Accessed 30 Sep, 2025.

16 Team, Bam. "Introducing BAM: The Future of Block Building on Solana." BAM, 21 July 2025, www.bam.dev/blog/introducing-bam. Accessed 4 Sep. 2025.

17 "Firedancer: Solana's New Validator Client Built for Scale and Security." CryptoEQ. 3 Apr. 2025, www.cryptoeq.io/articles/firedancer-client-solana. Accessed 4 Sep. 2025.

An FCAT Publication