# Mission 70 and Accelerating the Internet Computer Economy

Dominic Williams* Björn Assmann†

February 6, 2026 (v1.1.1)

**Abstract**

This paper proposes updates to the Internet Computer (ICP) tokenomics aimed at reducing inflation and pushing ICP toward deflation. The proposed measures support Mission 70, which targets a reduction of ICP inflation by at least 70% by the end of 2026 through a combination of supply-side reductions and demand growth.

For demand acceleration, we explain how forthcoming onchain cloud engines and the growing self-writing cloud paradigm (where for example platforms such as Caffeine enable the creation of onchain applications simply by interacting with AI using natural language chat and documents) increase ICP burn. In addition, we suggest revisiting cycle pricing for onchain compute, storage, and bandwidth to increase ICP burn in proportion to platform usage.

For voting rewards, we propose shortening dissolve delays and proportionally lowering reward levels, replacing the linear dissolve-delay bonus with a convex curve, capping the total reward pool after the initial bootstrapping phase, and introducing a simpler maturity modulation mechanism based on long-term price deviations.

For node provider rewards, we propose reducing rewards for legacy Gen-1 nodes and relying on SEV-capable hardware to operate smaller but more secure subnets.

We estimate that the supply-side measures reduce ICP minting from 9.72% (January 2026) to 5.42% (January 2027), a 44% reduction. DFINITY believes that the Mission 70 target of a 70% inflation reduction will be exceeded through a combination of these supply-side measures and demand acceleration.

DFINITY believes that, even if demand acceleration measures can achieve the Mission 70 target on their own, aggressive supply-side measures are essential to position the Internet Computer network, its ecosystem, and stakeholders for long-term success.

*DFINITY Foundation; dominic@dfinity.org
†DFINITY Foundation; bjoern.assmann@dfinity.org

# 1 Introduction

ICP is the native utility token of the Internet Computer. New ICP is minted through two channels: disbursed voting rewards of participants who stake ICP in neurons and take part in Network Nervous System (NNS) governance, and remuneration paid to node providers who operate the physical infrastructure of the network. ICP is burned when it is converted into cycles, which are used to pay for onchain compute and storage, and bandwidth on the Internet Computer. The long-term evolution of the ICP supply is therefore determined by the balance between these minting streams and the burn generated by platform usage.

Since genesis in May 2021, the Internet Computer ecosystem has grown rapidly, with the cycle burn rate increasing by roughly a factor of three per year. At the same time, the current tokenomics still reflect an early bootstrapping phase: voting rewards are high to compensate for long lockup periods, node provider remuneration is generous, and the minting of ICP is only partially offset by burning through cycles. As the network matures, it becomes increasingly important to adjust these parameters in order to keep inflation predictable and to preserve an attractive risk–return profile for stakers and governance participants.

Against this backdrop, the DFINITY Foundation has articulated the goal of *Mission 70*, which aims to reduce ICP inflation by at least 70% by the end of 2026. Achieving this objective requires a two-fold approach: reducing supply by adjusting reward structures, and increasing demand by accelerating ICP burn through platform usage.

This paper presents a set of proposed enhancements to ICP tokenomics based on quantitative analysis. The proposals address both supply reduction and demand acceleration to achieve sustainable long-term inflation control while maintaining strong incentives for secure and decentralized operation of the Internet Computer.

First, we describe demand-side mechanisms that increase ICP burn. We explain how cloud engines productize enterprise workloads on the Internet Computer and create a direct link between mass market enterprise cloud usage and ICP burn through a deflationary payment model. We also explain how Caffeine.ai and other self-writing development platforms (alongside the broader expansion of ICP-focused vibe coding) expand platform adoption by enabling non-technical users to build applications through natural language interaction.

Second, we suggest revisiting cycle pricing for onchain compute, storage, and bandwidth to increase ICP burn in proportion to platform usage.

Third, we revisit the design of voting rewards. We begin by reviewing the current reward determination and allocation mechanism. On this basis, we propose several changes. First, we suggest reducing the maximum and minimum dissolve delay and proportionally lowering reward levels. Second, we replace the linear dissolve-delay bonus with a convex curve that offers modest rewards for short commitments and strong incentives for multi-year staking. Third, we introduce an explicit cap on the voting reward pool so that governance-related minting remains bounded once the initial eight-year bootstrapping phase is complete. Fourth, we propose a simpler maturity modulation mechanism that links the generation of ICP from maturity to deviations from long-term price levels.

Fourth, we analyze node provider rewards. We show that rewards are high relative to underlying costs. We also highlight that many nodes are currently unused, while forthcoming cloud engine functionality involves a new kind of node provisioning model. Building on this analysis, we suggest transferring some node capacity to cloud engines and reducing rewards for legacy Gen-1 nodes. In addition, we recommend making greater use of SEV-capable hardware to operate smaller but more secure subnets. Taken together, these measures are intended to reduce inflation from node rewards and better align payments with actual infrastructure needs.

The combined impact of the proposed tokenomics measures for supply reduction is shown in Figure 1. Total ICP minting decreases from 9.72% (January 2026) to 5.42% (January 2027), representing an absolute reduction of 4.30 percentage points (relative reduction of 44%). This refers to gross reward issuance under full maturity disbursement; realized inflation is lower in practice. Both minting components contribute significantly to this reduction: voting rewards decrease from 5.88% to 3.45% (-41%) and node rewards from 3.84% to 1.97% (-49%).

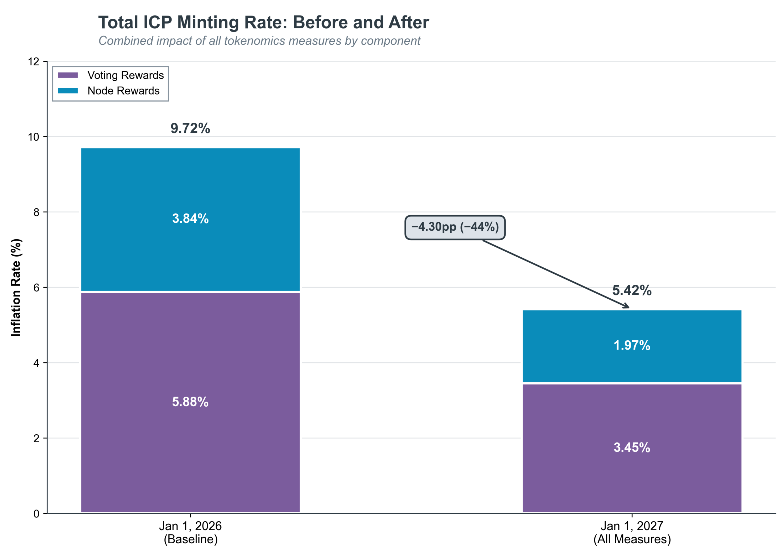



Figure 1: Total ICP minting: before and after tokenomics measures for supply reduction

To achieve the overall Mission 70 target of a 70% inflation reduction (from 9.72% to 2.92%), an additional demand impact of 26% is required beyond the 44% reduction from supply-side measures. At current price levels, this requires increasing the cycle burn rate from the current 0.05 XDR per second to 0.77 XDR per second. While this represents a significant acceleration of platform usage, DFINITY believes that forthcoming onchain cloud engines and growth in the self-writing cloud paradigm will push the burn rate well beyond this threshold in 2026. This does not obviate the need for supply-side measures. Notably, the burn rate already exceeded this level for several months in 2025, so this target is clearly achievable.

# 2 Accelerating the ICP Economy

This section describes demand-side mechanisms that connect economic activity on the Internet Computer to sustained ICP demand. In contrast to protocol-internal mechanisms that reduce ICP generation and otherwise constrain supply, these initiatives drive usage of the network by users and enterprises, thereby increasing the amount of ICP that is burned. Two key elements are *cloud engines* and *Caffeine.ai*, both described as part of the Internet Computer 2.0 vision [6]. Note that, as the Internet Computer pivots toward mass-market adoption, network tokenization will usually be hidden from those building on its onchain cloud functionality.

## 2.1 Cloud Engines

*Cloud engines* are configurable, application-specific execution environments on the Internet Computer. Conceptually, a cloud engine corresponds to a private subnet assembled and configured under the auspices of the fully autonomous Network Nervous System and its rules for safety. Cloud engines allow enterprises and developers to deploy workloads with customized security, performance, and resilience characteristics, while preserving the core properties of the Internet Computer, namely decentralization, onchain tamperproof operation, verifiability, and fault tolerance.

Cloud engines use an economic model inspired by the growth of the early internet. During the 1990s, there was an Internet Service Provider (ISP) boom. Across the world, many thousands of businesses created their own internet subnets, to which they added banks of modems. This enabled them to sell internet access using their own infrastructure and profit. Cloud engines bring this model to the Internet Computer.

Now associations of independent node providers can provide enterprise customers with an easy means to create cloud engines that run over their nodes. As with the internet, these node providers will invest in their own hardware and its hosting, but will receive all revenue generated by the cloud engines that run on their nodes, minus 20%, which the network will use to buy and burn ICP.

DFINITY has already spoken to associations of node providers that wish to sell cloud engine functionality en masse into the enterprise market, and is working with them to make this a reality as soon as possible. Notably, this model will also enable such associations of node providers to create their own new node machine specifications, optimized for the workloads that their enterprise customers will run on cloud engines, and to offer these as part of their cloud engine packages. Node providers will be able to use either virtualized nodes that run on traditional cloud infrastructure or sovereign nodes that they own and operate themselves.

Just as the internet runs over many kinds of subnets, from home WiFi routers, to cellular data networks, to international traffic routes, and those in large data centers and existing cloud services, so will cloud engines run over many kinds of node infrastructure, from sovereign nodes owned and operated by independent node providers, to virtualized nodes running on traditional cloud infrastructure.

However, the Network Nervous System will continue to enforce node provider registration and transparency, and the combination of nodes according to its deterministic decentralization rules, which ensure that cloud engines deliver the key advantages of Internet Computer sovereign cloud functionality, such as tamperproof operation (e.g., so that hosted apps can run securely without the need to be protected by traditional cybersecurity measures).

With this change, node providers participating in the cloud engine ecosystem will gain agency similar to that ISPs had in the early days of the internet. They will be able to offer one-click access to cloud engines running over their nodes via their websites, and market the benefits of onchain cloud functionality to enterprises and developers in order to profit directly.

Cloud engines will provide enterprises with configurable sovereign cloud platform functionality that is tamperproof, unstoppable, and serverless, with features that directly force-multiply AI working as autonomous tech teams and developers who are vibe coding, while delivering unique safety features (e.g., which prevent data loss during app upgrades).

The aforementioned functionality is tremendously valuable, and derives from the way that ICP creates cloud functionality using a mathematically secure and resilient network protocol. Onchain cloud is fundamentally different from traditional cloud services, which are built on top of centralized infrastructure and rely on trust in the cloud provider, and, furthermore, host applications that can be hacked. Replicating onchain cloud functionality requires solving deep research challenges in distributed systems, execution environments, cryptography, and programming languages, providing a strong competitive moat, and therefore a unique market opportunity. We expect the successful productization of ICP functionality to unlock extraordinary demand.

During 2025, the cloud market overall generated approximately $1 trillion in revenue, which is predicted to grow to over $2 trillion by 2030. Of this revenue in 2025, approximately $400 billion came from infrastructure and platform services such as Amazon Web Services, which alone accounted for approximately $140 billion. The rest comes from sectors such as SaaS (e.g. Salesforce) and AI. Cloud engines, and more specifically self-writing cloud, which will increasingly leverage cloud engines in the future, address the entire market.

We believe that the successful launch of cloud engines, and the decentralization of commercial incentives for node providers, will be a key milestone in the evolution of the Internet Computer. Furthermore, we believe that cloud engine growth driven by self-writing cloud functionality and next-generation vibe coding will drive extraordinary growth in Internet Computer usage, with a commensurate increase in ICP burn in the years to come. The target of the entire ecosystem should be to see ICP become heavily deflationary.

### 2.2 Caffeine.ai

*Caffeine.ai* is an AI-native application layer that dramatically lowers the barrier to building and deploying software on the Internet Computer. It enables users to create applications through natural language interactions via chat and documents, with the system generating, deploying, and iterating on code automatically. From a user perspective, Caffeine.ai completely abstracts away traditional software development workflows and infrastructure management. In effect, the AI plays the role of a wish machine that delivers and updates requested online functionality on demand. This approach is referred to as *self-writing cloud.*

Caffeine.ai will create a continuum with respect to vibe coding. It will be possible to export applications created in Caffeine into a next-generation vibe coding environment, and further to import vibe-coded applications into Caffeine for AI-powered extension and maintenance, enabling a seamless transition between AI-generated and human-generated code.

The self-writing cloud paradigm can deliver unprecedented velocity in application development and deployment, and commensurate cost savings. The addressable market for self-writing cloud, seen as a subset of the overall cloud market, is only limited by the ability of AI and the capabilities of the framework within which it delivers online functionality according to user wishes.

We also believe that self-writing cloud will unlock entirely new categories of applications that are not feasible using traditional software development. For example, we believe that eventually there will be a boom in hyperlocal social media, where families, friend groups, and communities create their own services that are private and secure, free from advertising and have their own unique features (which will co-exist with traditional social media platforms). This will further catalyze self-writing cloud's rapid expansion as a segment of the overall cloud market.

DFINITY is directing its efforts toward ensuring that the Internet Computer becomes the ultimate platform for self-writing cloud, along with partners such as Caffeine.ai, and partners in the enterprise sector who are already applying the technology. The ambition is for Internet Computer self-writing cloud to eventually represent a major portion of the overall cloud market, driving extraordinary ICP burn in the process.

# 3 Increasing Cycle Costs

## 3.1 Background

All computation, storage, and network activity on the Internet Computer is paid for in cycles, with the protocol specifying the cost of concrete actions, such as instruction execution, data storage, and message transmission. XDR (the IMF's Special Drawing Right, a basket-based unit of account) is used to define the conversion rate from ICP to cycles, ensuring that the cost of cycles remains fiat-stable over time. By definition, 1 XDR corresponds to 1 trillion cycles. When a developer or user tops up a canister, ICP is converted into cycles at the prevailing ICP/XDR exchange rate, the corresponding ICP is burned, and the canister is credited with the resulting cycle balance. As the canister executes code, stores data, or sends and receives messages, it continuously consumes these cycles.

Increasing cycle prices in XDR terms therefore increases the amount of ICP that must be burned for a given amount of compute, storage, or bandwidth consumption. This mechanism provides a direct and usage-proportional demand sink for ICP that scales naturally with network activity. We propose revising whether the pricing of cycles can be substantially increased to improve network tokenomics and help push ICP towards deflation.

## 3.2 Replicated canister memory vs. blob storage

Storage on the Internet Computer has historically been provided through fully replicated *canister memory*. In this model, canister state is part of the replicated state of the subnet, and each node in the subnet stores a full copy. This storage is powerful because it can be read and updated directly during replicated execution, but it is also inherently costly because the replication factor is equal to the subnet size.

A complementary model is *distributed blob storage*, designed for large, mostly static files (e.g., photos, videos, archives, document datasets). Blobs are chunked and hashed, forming a Merkle tree over their contents, and immutable replicated copies are stored on an associated network specifically designed to maintain them inexpensively. Meanwhile, the tree root is stored onchain, so any modification to a blob that is uploaded or downloaded is cryptographically detectable. This storage class can offer a greatly reduced cost per GB for storing blobs.

## 3.3 Current storage pricing and comparison

For a 13-node "application" subnet (where the nodes are owned and operated by different node providers, from different data centers, in disbursed geographies and jurisdictions), storing 1 GiB for 1 second in canister memory costs 127,000 cycles. Over a year, this amounts to approximately $5 per GiB.

For forthcoming blob storage, a plausible target price is $0.05 per GB-month for storage (corresponding to $0.60 per GB-year) and $0.10 per GB of egress traffic. Hence, canisters that require large volumes of storage for static files will soon have a significantly cheaper alternative than keeping such data in replicated canister memory.

### 3.4 Suggested changes

**Increase the cost of replicated canister memory.** We suggest substantially increasing the cycles charged for canister memory, for example by a factor of 5 to 10, to better align storage costs with the underlying economics of replicated storage. Higher storage prices also increase ICP burn in direct proportion to sustained onchain usage, thereby supporting Mission 70 objectives.

Crucially, the availability of lower-cost blob storage for static data allows developers to keep replicated state small, while still storing large datasets onchain, and thus reduces the risk that higher replicated memory prices discourage legitimate high-storage use cases.

**Rebalance the base cost of canisters.** We also suggest revisiting the base cost of operating a canister. A subnet can currently host up to roughly 100,000 canisters before performance begins to degrade. As a result, architectures that rely on a very large number of canisters can deplete a subnet's capacity and cause performance issues for all applications running on it.

Currently, the protocol charges a one time installation fee of 0.5 trillion cycles per canister, but does not impose any recurring base cost thereafter. This pricing structure undercharges the ongoing protocol overhead that each canister induces, including scheduling and the maintenance of canister metadata.

Introducing a modest recurring base cost per canister, potentially coupled with a reduction in the one time installation fee, would better align cycle pricing with actual resource consumption and reduce distortions in application architecture that arise purely from pricing asymmetries.

**Review compute and network cycle pricing.** In addition to storage and canister base fees, the cycle prices for compute and networking should be reviewed in detail. A blanket increase of all compute and network charges by a uniform factor does not seem reasonable. Instead, pricing should be assessed on a per resource basis to identify where the protocol currently undercharges relative to actual resource consumption and operational overhead.

That said, we believe that for several categories, increasing pricing by a factor of up to 5 is plausible without endangering the competitiveness of the network in its role as a sovereign onchain cloud platform. This is because the network hosts tamperproof and unstoppable serverless compute that delivers enormous value unavailable from traditional cloud platforms. These capabilities substantially reduce costs for security, systems administration, and R&D, categories that typically far exceed raw compute costs.

# 4 Voting Rewards

## 4.1 Determination and Allocation of Rewards

### 4.1.1 Background

The Network Nervous System (NNS) governs the Internet Computer and distributes voting rewards to participants who stake ICP tokens in so-called neurons. A neuron earns rewards when it casts votes on governance proposals or follows other neurons that do so. Rewards are tracked in the form of maturity, which can be used to generate new ICP by triggering a disbursement process. The NNS reward mechanism is designed to incentivize long-term participation in Internet Computer governance.

Voting rewards are allocated on a regular basis (primarily daily) based on an overall reward pool. Each neuron receives a pro-rata share of that pool according to its voting power and the number of proposals in which it participated. More precisely, this works as follows:

**Determination of the Total Reward Pool** For times $t$ between ICP genesis $G$ and $G+8$ years, the annualized voting reward rate, expressed as a percentage of total supply, is defined as

$$R(t) = 5\% + 5\% \left( \frac{G + 8y - t}{8y} \right)^2 .$$

For $t > G + 8$ years, the reward rate becomes constant: $R(t) = 5\%$. The daily voting reward pool is computed as

$$\text{VotingRewardsPool}(t) = \frac{\text{TotalSupply}(t) \times R(t)}{365.25}.$$

This declining reward function reflects the transition from early-network bootstrapping to long-term steady-state operation.

**Voting Power of Neurons** A neuron's voting power is determined by its stake, dissolve delay, and age. Only neurons with a dissolve delay of at least six months are eligible to vote. The voting power is computed at the moment a proposal is created, not when votes are cast.

The voting power is given by:

$$\text{Voting Power} = \text{Stake} \times \text{Dissolve Delay Bonus} \times \text{Age Bonus}.$$

The dissolve-delay bonus increases linearly from 1 to 2 as the dissolve delay grows from the minimum voting threshold (6 months) to the maximum (8 years). The age bonus increases linearly from 1 to 1.25 as the neuron ages over four years, and resets when the neuron enters dissolving.

**Allocation of the Reward Pool** For each reward period (typically one day), the NNS identifies all proposals that are no longer open for voting and have not yet been settled with respect to voting rewards. For this set of proposals, the total voting power contributed by all eligible neurons is aggregated. Each neuron then receives a portion of the daily reward pool proportional to:

- the voting power it contributed to each proposal, and
- the reward weight assigned to the corresponding proposal category.

Rewards are credited to the neuron's maturity, a non-tradable internal accounting quantity. The neuron owner can disburse maturity, a process that burns the maturity and mints new ICP, subject to maturity modulation.

**Simplified Allocation Formula (for Modeling).** For simulations in which every neuron is assumed to vote on all proposals, the reward allocated to neuron $i$ on day $t$ can be approximated by:

$$\text{VotingRewards}_i(t) = \frac{\text{VotingPower}_i(t)}{\text{TotalVotingPower}(t)} \times \frac{\text{TotalSupply}(t) \times R(t)}{365.25}.$$

#### 4.1.2 Analysis

The current reward determination and allocation mechanism has several limitations that are relevant when evaluating possible changes:

The eight-year maximum dissolve delay encourages long-term commitment to the network, which is desirable from a governance perspective. However, such long lockups require correspondingly high rewards to compensate participants for the extended illiquidity. This contributes significantly to overall inflation. Reducing the maximum dissolve delay, together with a proportional reduction in reward levels, could lower inflation while still preserving incentives for meaningful long-term participation.

The current minimum dissolve delay of six months is long compared to lock-up periods on many other blockchains. This creates a barrier to entry for new participants who are willing to stake but prefer shorter commitment horizons. Furthermore, the final six months of the dissolve period do not accrue rewards, making short-term staking even more economically unattractive.

#### 4.1.3 Suggested Changes

**Reducing Dissolve Delays.** We propose to reduce the maximum dissolve delay from 8 years to 2 years. At the same time, we propose to reduce the minimum dissolve delay for voting from 6 months to 2 weeks. All existing neurons would undergo a one-time migration in which their dissolve delays are capped at 2 years.

Please note that this one-time reduction of dissolve delays would not affect the age bonus of non-dissolving neurons. All such neurons would retain their existing age bonus.

After the migration, all existing dissolving neurons with remaining dissolve delays above 2 years would have their dissolve delays set to 2 years, causing them to complete dissolution simultaneously. To mitigate the resulting surge in neuron dissolutions, one could implement a dissolution queue similar to the Ethereum Validator Queue. When the total ICP amount scheduled to dissolve on a given day exceeds a governance-defined threshold, the queue would automatically stagger dissolutions over subsequent days.

**Convex Reward Curve.** The aforementioned proposal to lower the minimum dissolve delay to two weeks requires adjusting the dissolve-delay bonus curve to be more convex and steeper, so that short-term stakers receive meaningful but modest rewards while long-term commitments remain significantly better incentivized, even with a lower minimum dissolve delay.

More concretely, the dissolve-delay bonus may be defined by a convex function of the form

$$f(x) = ax^n + b,$$

where $x$ denotes the dissolve delay in years, $b = 1$ is the minimum dissolve-delay bonus, $n$ controls the degree of convexity, and

$$a = \frac{\text{maximum bonus} - \text{minimum bonus}}{(\text{maximum dissolve delay})^n}.$$

This formulation generalizes the current scheme. Choosing $n = 1$ yields a linear function, which corresponds to the existing dissolve-delay bonus curve. We suggest using $n = 2$, which yields a quadratic function that represents the simplest form of convex behavior. We also propose setting the maximum dissolve-delay bonus to 3 (up from the current value of 2), to provide stronger incentives for long-term staking. Figure 2 shows the suggested new dissolve delay bonus function.

**Special Treatment of 8-Year Gang.** The *8-year gang* refers to long-term Internet Computer supporters who set their dissolve delay to the current maximum of eight years. Neurons that have historically maintained an 8-year dissolve delay should be given a special flag and receive a dedicated reward boost of 10% in recognition of their long-term alignment with the network. The reward boost would remain in effect until the end of 2030. This special flag, and the corresponding reward boost, would be lost once the neuron starts to dissolve (similar to the age bonus).

**Reducing Reward Levels.** As mentioned above, the idea is to lower dissolve delays together with a proportional reduction in reward levels. The target is that a neuron which today has the maximum dissolve delay of 8 years, and

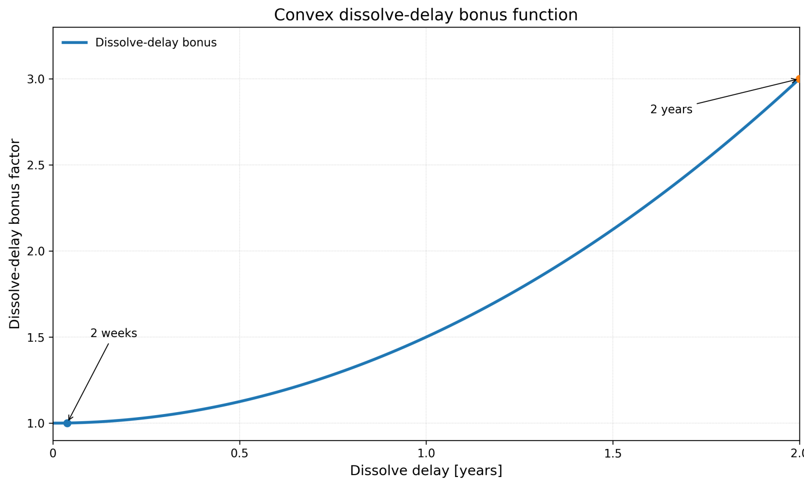



Figure 2: Suggested new dissolve delay bonus function.

which would be mapped to the new maximum dissolve delay of 2 years, receives the same APY as a neuron with a 2-year dissolve delay today.

As a first-order approximation, the impact can be estimated as follows. At present, an 8-year neuron earns about 12.3% APY, and the majority of voting power is concentrated at this maximum dissolve delay. Replacing the 12.3% rate for 8-year neurons by the 7.7% rate for 2-year neurons implies a proportional reduction in rewards of $1 - \frac{7.7\%}{12.3\%} \approx 0.37$.

The precise impact can be determined by simulating the full neuron set, applying all proposed changes to dissolve delays and reward levels, and calculating voting rewards using the allocation formula from the background section. Based on the neuron population from early December 2025, this analysis yields a reduction in voting rewards of approximately 36.4%. The corresponding reduction of the voting reward pool can be implemented in a simple and transparent way by multiplying the current voting reward function $R(t)$ by a constant scaling factor (e.g., 1 - 0.364 = 0.636).

These changes would result in the following APY levels for various dissolve delays (all figures assuming no age bonus). Short-term APYs are in line with staking options on other blockchains, as reported, for example, on DeFiLlama [1], while long-term staking remains quite attractive compared to other ecosystems.

| **Dissolve Delay** | **APY** |
|---|---|
| 2 weeks | 2.3% |
| 1 year | 3.5% |
| 2 years | 7.0% |
| 2 years with 8-year gang flag | 7.7% |

**Capping the Voting Reward Pool.** So far, the size of the voting reward pool has been declining over time because the voting reward function is designed to decrease quadratically from an initial maximum of 10% APY at genesis to 5% APY eight years after genesis. Once this transition is complete, the reward schedule flattens and the voting reward rate becomes constant. From that point onward, total voting rewards grow proportionally with the ICP supply, so any increase in supply leads to higher absolute minting from voting rewards.

We propose introducing a cap on the daily voting reward pool, effective eight years after genesis. Under this design, the daily voting reward pool is defined as

$$\text{VotingRewardsPool}(t) = \min\left\{\text{cap}, \frac{\text{TotalSupply}(t) \times R(t)}{365.25}\right\}.$$

We suggest setting the cap to the daily voting reward pool eight years after genesis, that is, $\text{cap} = \text{VotingRewardsPool}(t_{8\text{y}})$. The cap could remain a governance parameter that may be adjusted over time to reflect the size and activity of the ICP network. This ensures that inflation from voting rewards remains bounded and predictable, and prevents increases in total supply from automatically translating into ever larger governance-related minting.

**Expected Impact.** The combined effect of the proposed measures on voting rewards inflation is illustrated in Figure 3. Starting from a baseline of 5.88% inflation from voting rewards (including undisbursed and disbursed maturity), the combination of the existing quadratic decrease in the voting rewards function over the course of 2026 and the suggested measures reduces inflation to 3.45%, representing an absolute decline of 2.43 percentage points (a 41% relative reduction).

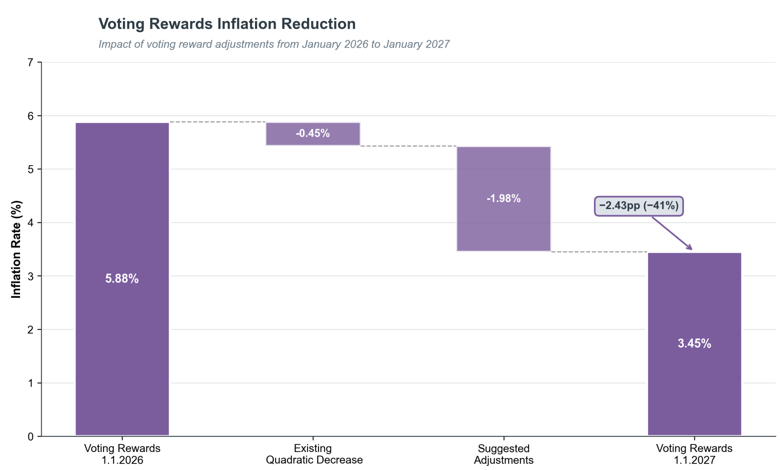



Figure 3: Impact of voting rewards adjustments

## 4.2 Disbursement of Voting Rewards

### 4.2.1 Background

Voting rewards earned by neurons accumulate as *maturity*, an internal attribute that is not a tradable asset. Neuron holders may use this maturity to generate new ICP at any time by initiating a seven-day disbursement process. When the process completes, the maturity is burned and newly minted ICP is generated, with the final amount adjusted by a modulation factor. For further background on the design rationale behind maturity and its disbursement, please refer to [5].

The modulation mechanism introduces uncertainty into the amount of ICP generated and depends on recent movements in the ICP/XDR conversion rate. The modulation factor is updated once per day. Its computation proceeds as follows:

1. For each of the last 29 days, determine the 30-day moving average ICP/cycles conversion rate. Label these values $a_1, a_2, \ldots, a_{29}$, where $a_1$ is yesterday's 30-day average, $a_2$ is the value from two days ago, and so on.

2. Compute the relative 7-day returns for the past four weeks:

$$w_1 = \frac{a_1 - a_8}{a_8}, \quad w_2 = \frac{a_8 - a_{15}}{a_{15}}, \quad w_3 = \frac{a_{15} - a_{22}}{a_{22}}, \quad w_4 = \frac{a_{22} - a_{29}}{a_{29}}.$$

3. Clip each of the weekly returns to the range $[-0.05, 0.05]$.

4. Compute the average

$$w = \frac{w_1 + w_2 + w_3 + w_4}{4}.$$

5. A maturity amount $x$ is used to generate

$$x \times (1 + w)$$

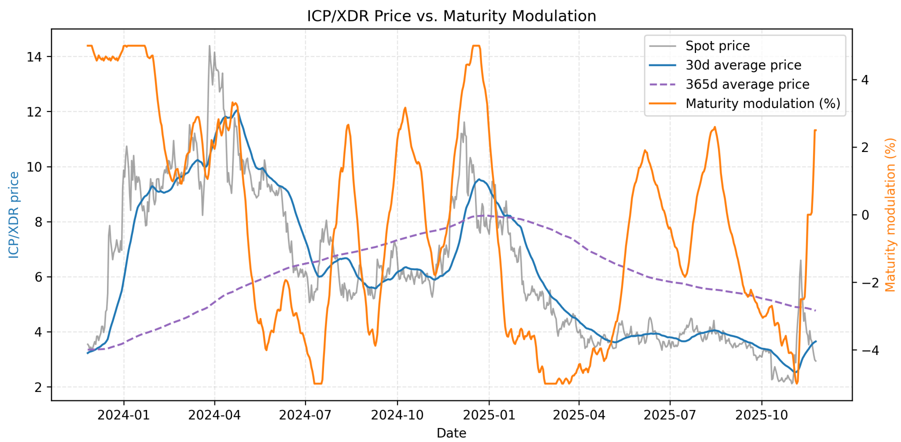
   units of ICP at the end of the 7-day period.

The maturity modulation mechanism introduces uncertainty into the amount of ICP generated, but this uncertainty is bounded: The modulation factor $w$ always lies between $-5\%$ and $+5\%$, meaning 100 maturity generates between 95 and 105 ICP. When a user initiates a disbursement, three of the four weekly return values are already known. The unknown value $w_1$ can change the final modulation factor by at most 1.25% (since 5%/4).

The mechanism incentivizes disbursing when the ICP/cycles conversion rate has been increasing, since positive weekly returns lead to a positive modulation factor $w$. Conversely, recent declines in the exchange rate tend to produce negative values of $w$, making users less inclined to disburse at that time. For further details on the maturity modulation function, please refer to [3].

### 4.2.2 Analysis

The following graph displays the maturity modulation values over the past two years:




The historical data reveals several noteworthy properties of the current mechanism:

Because the modulation factor is based on weekly returns of the *30-day moving average* of the ICP/cycles conversion rate, the response of the mechanism is inherently delayed. The 30-day average greatly stabilizes the input values, making manipulation more difficult and reducing short-term noise, but it also reduces responsiveness to genuine price movements. This is visible, for example, during the large increase and subsequent decline in November 2025. The modulation factor began to react only after the underlying exchange rate had already returned close to its previous level. In such cases, the mechanism does not provide timely signals to users and may adjust only after the relevant market movement has largely passed.

The modulation factor is constrained to the symmetric interval $[-5\%, +5\%]$. While symmetry is natural from a design perspective, user behavior may not be symmetric. In practice, many users may prefer to disburse maturity when $w > 0$, since positive modulation increases the amount of ICP received. This creates a behavioral bias toward disbursing in positive conditions. As a result, additional ICP inflation can be generated, even though the modulation range itself is symmetric. This is not the intended behavior of the mechanism.

The modulation factor is computed as the average of four weekly returns of the 30-day moving average. This structure originated from the fact that the 30-day average was already available as a data source, but the resulting formula is somewhat complex and not immediately intuitive. It combines multiple layers of smoothing (30-day averaging, weekly differences, and a four-week mean), each of which contributes to stability but at the cost of transparency and interpretability.

The mechanism is driven entirely by weekly returns of the 30-day moving average price. It therefore detects only *changes* in price, not whether the ICP price is high or low (e.g., as measured by its difference from a 365-day moving average). A stable but high price, or a stable but low price, both produce the same modulation value ($w = 0$), even though the economic implications for minting may differ.

#### 4.2.3 Suggested Changes

To address the shortcomings identified above, we propose replacing the current trend-based maturity modulation with a simpler mechanism. The revised design focuses on long-term price deviations rather than short-term returns and incorporates explicit safeguards that limit uncertainty for stakers.

**Level-Based Modulation.** Instead of computing weekly returns of the 30-day moving average, the proposed mechanism measures how far the short-term price is from the long-term price level. Concretely, it compares the 7-day moving average $P_7(t)$ to the 365-day moving average $P_{365}(t)$ and computes the relative deviation

$$z_t = \frac{P_7(t) - P_{365}(t)}{P_{365}(t)}.$$

This quantity captures whether the current price is high or low relative to its one-year trend, addressing a key limitation of the existing mechanism, which is insensitive to absolute price levels. The modulation signal is then defined as

$$w_t = k \cdot z_t,$$

where $k > 0$ is a sensitivity parameter. When the ICP price trades below its long-term average, the modulation becomes negative and discourages minting; when the price is above the long-term average, the modulation becomes mildly positive. We suggest a value of $k = 0.25$, which provides a proportional but moderate response to deviations from the long-term price.

**Daily Speed Limit and User Certainty.** To limit the uncertainty faced by stakers initiating a disbursement with a 7-day waiting period, we introduce a global speed limit on how quickly the modulation may change from day to day:

$$|w_t - w_{t-1}| \leq \Delta_{\max}.$$

A value of $\Delta_{\max} = 0.3\%$ per day restricts the 7-day variation of the modulation to at most 2.1%. This ensures that the modulation used at the end of the 7-day disbursement period cannot differ substantially from the value observed at the moment the user initiated the operation.

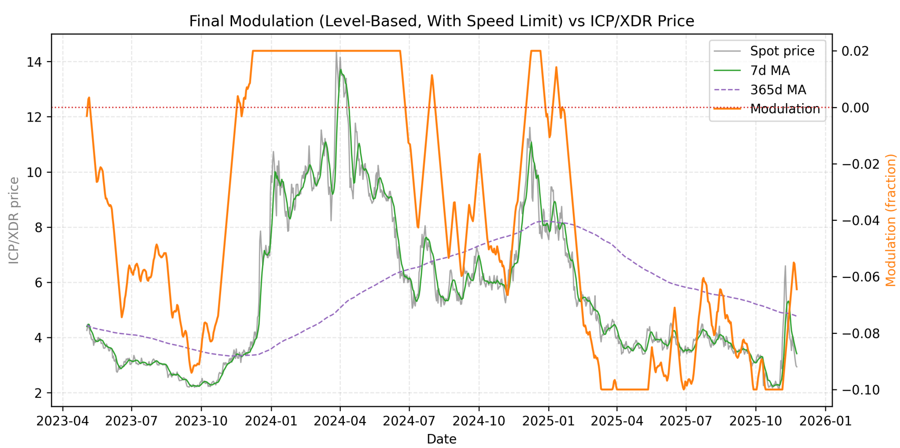
**Asymmetric Global Bounds.** To ensure bounded behavior, the modulation factor is restricted to an asymmetric interval

$$W_{\min} \leq w_t \leq W_{\max},$$

with a recommended choice of $W_{\min} = -10\%$ and $W_{\max} = +2\%$. This asymmetry reflects the intended economic effect: discouraging ICP minting during periods of market stress (strong negative modulation), while allowing only limited positive modulation in favorable conditions. This corrects the inflationary bias that may arise from a symmetric interval when user behavior is asymmetric.

**Simplicity and Transparency.** The proposed mechanism eliminates the multi-layered structure of the current design (30-day average, weekly returns, four-week averaging) and replaces it with a single, intuitive economic signal based on deviation from the long-term price trend. This improves transparency for governance participants.

In the following graph, we illustrate the behavior of the revised mechanism, showing how the modulation responds smoothly to long-term price deviations while remaining bounded and predictable due to the daily speed limit.




# 5 Node Provider Rewards

## 5.1 Background

The Internet Computer is operated by a geographically distributed set of node machines that form the subnets powering the network. These nodes are owned and maintained by independent *node providers*. The Network Nervous System (NNS) remunerates node providers for operating compliant hardware and sustaining stable network connectivity.

The network currently supports multiple generations of node hardware, each with its own remuneration model defined and governed by the NNS. Reward structures reflect differences in capital expenditure, operating expenditure, and the decentralization value contributed by specific geographies.

Monthly node rewards are denominated in XDR (a basket of major fiat currencies) and are converted into ICP by dividing the XDR amount by the 30-day average ICP/XDR price. The motivation for the XDR-linked design is that node provider expenses are largely fiat-denominated; linking rewards to XDR therefore stabilizes income for providers. This mirrors the approach used for compute and storage costs on the Internet Computer, which are also based on XDR.

At present, two active hardware generations are recognized: *Gen-1* and *Gen-2*.

**Gen-1 Hardware.** Gen-1 nodes correspond to the servers purchased prior to network genesis. These machines were deployed during the bootstrap phase and were provisioned according to the original hardware specification.

Gen-1 remuneration varies by country and was derived using a simple cost-based approach. For each region, estimates of capital expenditure and operating expenditure were collected and aggregated into a total 48-month cost. This cost was then multiplied by a generous factor of 2.5 to create a strong incentive for early node providers to acquire and operate the initial hardware. Dividing this amount by 48 yielded the monthly remuneration per node.

After 48 months of operation, Gen-1 nodes transition to the *Gen-1.1* remuneration model. This model applies a lower reward schedule for Gen-1 nodes, with a reduction of 33%. Examples include:

| **Region** | **Reward / Month (XDR)** |
|---|---:|
| US (FL/GA/CA) | 1072 |
| US (other) | 1004 |
| Canada | 1088 |
| Slovenia | 1152 |
| Switzerland | 1136 |
| Singapore | 1234 |
| Japan | 1188 |

**Gen-2 Hardware.** Gen-2 nodes represent the current recommended node machine specification. A key feature of Gen-2 nodes is the use of AMD CPUs that support SEV SNP, a hardware technology that encrypts virtual machine memory and provides attestation of the software running inside the virtual machine. This enables SEV SNP-protected replica nodes, where memory is protected from the node provider and only NNS-approved GuestOS images are permitted to run.

Gen-2 remuneration follows a formula that depends on the country and the number of nodes the provider operates. The underlying cost model uses the total cost over four years (capital plus operational expenditure) for operating a node in a specific geography. This cost is adjusted by a multiplier of two for the margin. To discourage excessive concentration, a reduction coefficient is applied so that the reward for each additional node operated by the same provider in the same country decreases progressively. For further details on the current node provider remuneration, please refer to [4].

### 5.2 Analysis

**Number of Used Nodes.** As per the Internet Computer Dashboard [2], 701 out of 1,424 nodes are in active use, that is, assigned to subnets. This corresponds to an assignment ratio of 49%. It is reasonable for the network to maintain spare capacity, although a smaller buffer would likely be sufficient. The assignment ratio for several large Gen-1 node providers, mainly in the United States and Europe, is well below 30%. Many of their nodes remain unused because of regional oversupply and therefore do not contribute to improving the decentralization profile of the subnets. On the other hand, the assignment ratio for Gen-2 SEV-capable nodes is very high, due to their high regional diversity. Several Gen-2 node providers operate close to full utilization.

**Profitability of Node Provider Rewards.** The Gen-1 hardware is fully amortized, and Gen-1.1 rewards still equal 66% of the original reward level (which included a 2.5 multiplier). Based on observed operating costs from DFINITY's own data center deployments across several countries, and including an uncertainty margin to reflect differences in data center contracts, energy prices, hardware servicing, remote hands, and operational overhead, total monthly expenses for Gen-1 servers are estimated to lie in the range of 300 to 800 USD, while average rewards are roughly 1,500 USD per month. For example, the monthly cost of running DFINITY nodes in Seattle is around 300 USD. This implies a reward-to-cost ratio between 2 and 5. Hence, Gen-1.1 rewards substantially exceed underlying operating costs.

### 5.3 Suggested Changes

To achieve inflation reduction comparable to that proposed for voting rewards, we propose reducing rewards for legacy nodes. In addition, we propose enabling smaller application subnets that make use of SEV-capable hardware. In more detail, we suggest the following specific measures.

**Reduction of Gen-1 Node Rewards.** As discussed above, Gen-1.1 rewards currently correspond to a reward multiplier of roughly 2 to 5 relative to estimated operating costs. We propose reducing Gen-1 rewards by 40%. This level would still allow most node providers to cover their costs and, in most cases, earn a positive margin, while reducing inflationary pressure from legacy nodes, many of which are currently unused.

Some node providers may shift their offering to cloud engines (see Section 2) or choose to discontinue operations under the new terms. In addition, several Gen-1.1 rewards are running out by the end of 2026, leading to a natural exit of some node providers. If 66% of Gen-1 node providers were to transfer or exit, the combined effect of the lower reward level and the reduced node count would reduce total Gen-1.1 rewards by 80% (because $1 - (1 - 0.4) \times (1 - 0.66) \approx 0.8$).

As an alternative to a plain reduction of Gen-1 rewards, one could consider reducing the number of rewardable nodes per Gen-1 node provider. This would allow providers to redeploy unrewarded nodes to cloud engines or decommission/sell them. We suggest targeting a similar overall reduction of 80% in Gen-1 rewards through this approach, including the natural expiry of Gen-1.1 reward periods in 2026 for some providers.

As a further alternative, one could consider lowering rewards for unassigned nodes. Given the low assignment ratio of many Gen-1 node providers, this would lead to a similar reduction in overall Gen-1 rewards. This approach would have the advantage of incentivizing node providers to actively promote the use of their nodes. However, it would add complexity to the remuneration scheme and could lead to unintended consequences, such as node providers repeatedly submitting proposals to assign their own nodes while removing others. For these reasons, this alternative would require further analysis.

In addition to reducing node reward levels, we recommend that monthly payouts be distributed throughout the month rather than on a single fixed date. This would smooth node provider ICP sales and contribute to more stable market conditions.

**Smaller SEV Subnets.** The introduction of SEV SNP-enabled hardware provides stronger security guarantees at the subnet level. The network already includes 442 SEV-capable nodes. To increase the effective capacity of the Internet Computer while relying on fewer nodes, we propose creating dedicated subnets composed exclusively of SEV nodes and reducing their size while keeping cycle costs unchanged. For example, reducing the size of application subnets that use only SEV nodes from 13 to 7 would almost double the available subnet capacity. This improvement in capacity supports the proposed reduction in the overall number of nodes mentioned in the paragraph above.

**Expected Impact.** The combined effect of the proposed measures on node rewards inflation is illustrated in Figure 4. Starting from a baseline of 3.84% inflation from node rewards, the suggested measures lead to a reduction to 1.97%, representing an absolute decline of 1.87 percentage points (a 49% relative reduction). Please note that while the actual node reward inflation levels depend on the ICP/XDR conversion rate, the relative reduction percentage does not depend on it.

# References

[1] DeFi Llama - Yields. `https://defillama.com/yields`. Accessed: 2026-01-08.

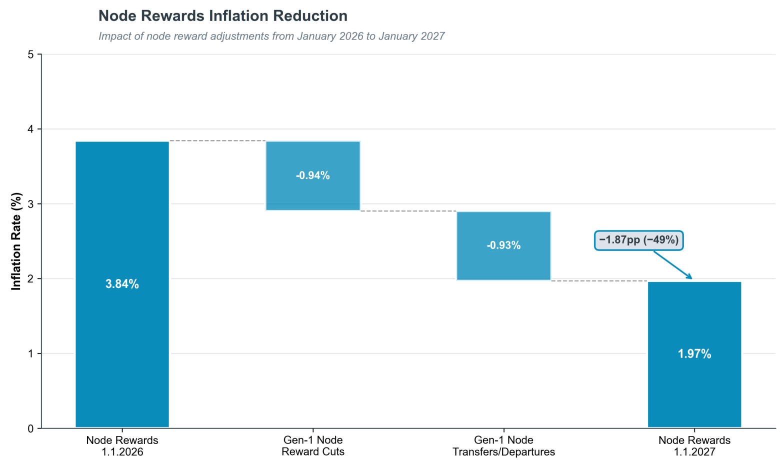



Figure 4: Impact of node rewards adjustments


[2] Internet Computer Dashboard, Network Nodes. `https://dashboard.internetcomputer.org/network/nodes`. Accessed: 2026-01-08.

[3] Maturity Modulation Wiki. `https://wiki.internetcomputer.org/wiki/Maturity_modulation`. Accessed: 2026-01-08.

[4] Node Provider Remuneration Wiki. `https://wiki.internetcomputer.org/wiki/Node_Provider_Remuneration`. Accessed: 2026-01-08.

[5] Dominic Williams. Proposal to Improve ICP Governance Staking Re: Tax and Tokenomics. `https://medium.com/dfinity/proposal-to-improve-icp-governance-staking-re-tax-and-tokenomics-df09903a2e0b`, March 2022. Accessed: 2026-01-08.

[6] Dominic Williams. Internet Computer 2.0, DFINITY 2.0 and Caffeine (self-writing apps). `https://medium.com/dfinity/internet-computer-2-0-dfnity-2-0-and-caffeine-self-writing-7a793e1b1831`, 2024. Accessed: 2026-01-08.