Shardeum

Shardeum Discussion Paper v1.0

This paper is not a whitepaper nor are any of the points final, they are purely topics for discussion.
The intention of this document is to be a starting point for the community to collect its thoughts.

Some or all of the details in this document will change

Version history

Version	Date	Comment	Author
1.0	21/03/2022	Document Created	Daniel Ingamells

Introduction

The economic model of any new digital currency / asset is one of the most critical components of the platform that the asset resides on. This is especially true for the native coin of a self-sovereign, permission-less platform. In this paper, we discuss the economic design of the native token, called $SHM.

Coin Distribution

The Shardeum network will have a native coin called Shard with a ticker symbol of $SHM. The coin will be mined by validator, archive and standby nodes as reward for providing resources to the network. The coin will be used for paying transaction fees (tx) associated with executing transfer transactions as well as smart contract execution on the Shardeum network.

Max supply: 508 million SHM
Burn mechanic: N/A (explained below)
Smallest divisible unit: 0.000000000000000001 SHM = 1 XXX (community to decide)

Distribution:

- 51% Community - reward to nodes; validators, archive and standby servers
- 18% Sale - 3 month cliff then 2 years linear vesting
- 15% Team - 3 month cliff then 2 years linear vesting
- 11% Foundation - unlocked at Token Generation Event (TGE)
- 5% Ecosystem - unlocked at TGE

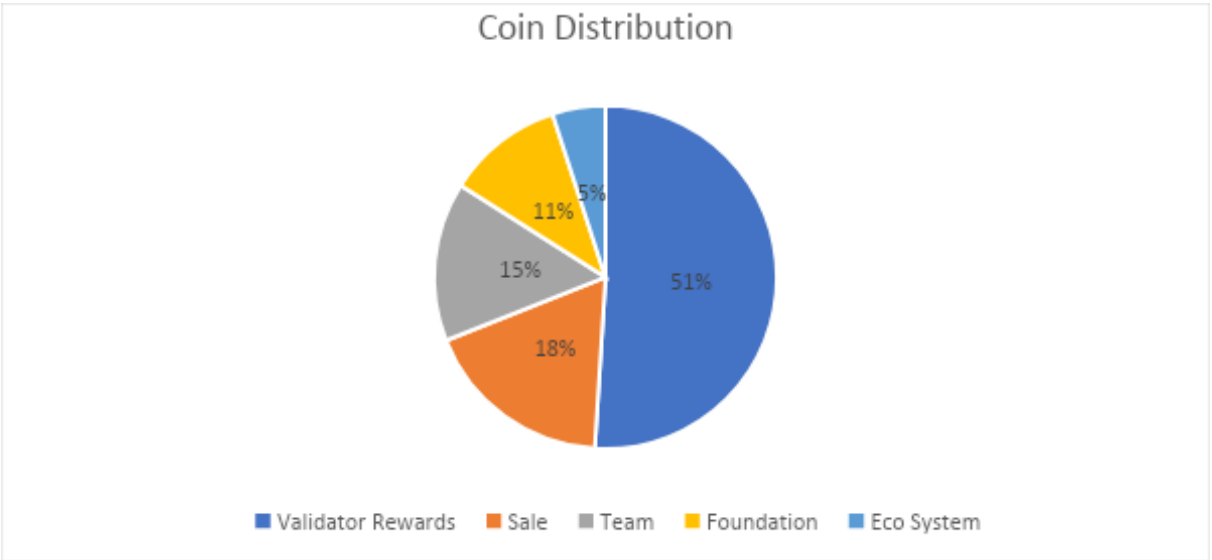

Figure 1.1 *Shardeum coin distribution*

Why is simplicity important?

Bitcoin demonstrates the power of simplicity. The economic model is so simple it can be expressed in one sentence: “Issuance halving every 4 years until a maximum supply of 21 million.” The simplicity of Bitcoin’s economic model makes it quick for people to understand, meaning that it is easy to onboard new community members. The more understandable the model is, the easier it is to spread. Conversely the more complicated the economics are, the less likely the protocol is to reach mass adoption.

An advantage of this simple model is that it becomes easy for validators to project their future rewards and it becomes easy for coin holders to project future circulating supply. This predictability gives the protocol a stable economic base for our stakeholders to rely upon.

Why limited supply?

The purpose of having a limited supply is to give clarity from genesis on the Shardeum monetary policy. One of the key factors in Bitcoin’s success is its predictability; the halving and difficulty calculation make the inflation rate highly predictable (figure 2.1)**.**

Year	#bitcoins	Inflation per annum	
2009	1,624,250	-	
2010	5,020,250	209.1%	
2011	8,001,400	59.4%	
2012	10,733,825	34.1%	
2013	12,199,725	13.7%	
2014	13,671,200	12.1%	
2015	15,029,525	9.9%	
2016	16,075,400	7.0%	
2017	16,774,500	4.3%	
2018	17,455,725	4.1%	
2019	18,133,625	3.9%	
2020	18,591,806	2.5%	(estimate, halvening)
2021	18,929,306	1.8%	(estimate)
2022	19,266,806	1.8%	(estimate)
2023	19,604,306	1.8%	(estimate)
2024	19,814,647	1.1%	(estimate, halvening)
2025	19,983,397	0.9%	(estimate)
2026	20,152,147	0.8%	(estimate)
2027	20,320,897	0.8%	(estimate)
2028	20,416,692	0.5%	(estimate, halvening)
2029	20,501,067	0.4%	(estimate)
2030	20,585,442	0.4%	(estimate)

Figure 2.1 *Bitcoin’s predictable inflation rate*

Conversely Ethereum has an uncapped supply of eth and it has been well documented each time the project generates a new hard fork such as EIP 1559. This type of hard fork makes it difficult to assess Ethereum as a suitable long term store of value, because the key parameters that govern the monetary policy can be changed at any time.

Why no burn mechanism?

Coin burn mechanisms seem to be the new craze in cryptocurrency. The most common way coins are burned is sending a proportion of the transaction fees (tx) into a frozen private address called a burn address. It is a one-way address with no ability to reverse the transaction or withdraw the coins, and the burn address to which the tokens are sent can never be recovered because there is no private key corresponding to that address. For all practical purposes, the asset no longer exists - it has been “burned”.

Generally speaking there are two reasons why burn mechanics are used / introduced. The first is to reduce the supply if there is an abundance of coins in circulation. Ethereum 2.0 is a good example of this - according to the proposal’s description, this was done to counterbalance Ethereum inflation while still giving the block reward and priority fee (the maximum fee users are willing to spend to include their transaction in a block) to miners. This makes logical sense as Ethereum has an uncapped supply of eth, the burn is used to offset inflation and, in some circumstances, create deflationary pressure. The example below (figure 3.1) shows Ethereum currently has a 3.8% yearly inflation rate, this is expected to become deflationary after the merge (figure 3.2) falling to -0.3%.

Figure 3.1 *Ethereum pre-merge (burn) / yearly inflation*

Figure 3.2 *Ethereum post-merge (burn) / yearly deflation (simulated)*

The second reason is that token burns may result in the value of the coin increasing. Stellar, another cryptocurrency company, proceeded with a token burn of 55 billion XLM to increase the coin’s value. This burn effectively reduced XLM supply by over 50%. The price effect on XLM was quickly noticeable in the short term, moving from $0.069 to $0.088 in a day (around 25% shown in figure 3.3). This type of benefit has been proven to have very little

long term effect on the price and only really benefits coin holders that short their own stake after the market bounce.

Figure 3.3 *XLM price spikes following burn announcement*

In terms of the Shardeum project, neither of the reasons for having a burn mechanic explained above would significantly benefit $SHM in the long term. $SHM is a limited supply coin (508m) so there is no reason to control the supply because it will reach maximum issuance and effectively have 0% inflation. The second reason around increasing the token value in the short term is against the OCC guidelines that Shardeum follows (open, collaborative and community driven).

How should $SHM be issued?

Linear issuance

Monetary statement: $SHM will be distributed for **XXX** years at **XXX** coins per year until maximum issuance of 508 million is reached.

This form of distribution would set a number of years from genesis until the capped supply level is reached. Each year the same amount of coins (S$HM) would be minted and added to the overall supply. This model is the most predictable as inflation rates can be determined even before the Mainnet has been launched as per the Bitcoin example above but without halving (figure 2.1).

The only variable that would impact inflation rate would be the number of years it would take to reach maximum issuance (community to decide). The longer the issuance years to maximum supply, the lower the inflation rate will be (see figures 4.1-4.5). The drawback of this approach is if the issuance period is too long, the amount of liquid $SHM could be low. Other considerations with this approach are around encouraging early validator adoption; this approach gives less incentive for early adopters as the reward scale is the same from the first to last day of issuance (offset by loyalty reward explained below).

20 years

Totally Supply	508000000
Day 0 Supply	248920000
Staking Reward Supply (51%)	259080000
Years to Cap	20
Days to Cap	7300
Yearly Coin Reward	12954000
Daily Coin Reward	35490.4

50 years

Totally Supply	508000000
Day 0 Supply	248920000
Staking Reward Supply (51%)	259080000
Years to Cap	50
Days to Cap	18250
Yearly Coin Reward	5181600
Daily Coin Reward	14196.2

100 years

Totally Supply	508000000
Day 0 Supply	248920000
Staking Reward Supply (51%)	259080000
Years to Cap	100
Days to Cap	36500
Yearly Coin Reward	2590800
Daily Coin Reward	7098.1

Figure 4.1 *20, 50 and 100 year linear issuance details*

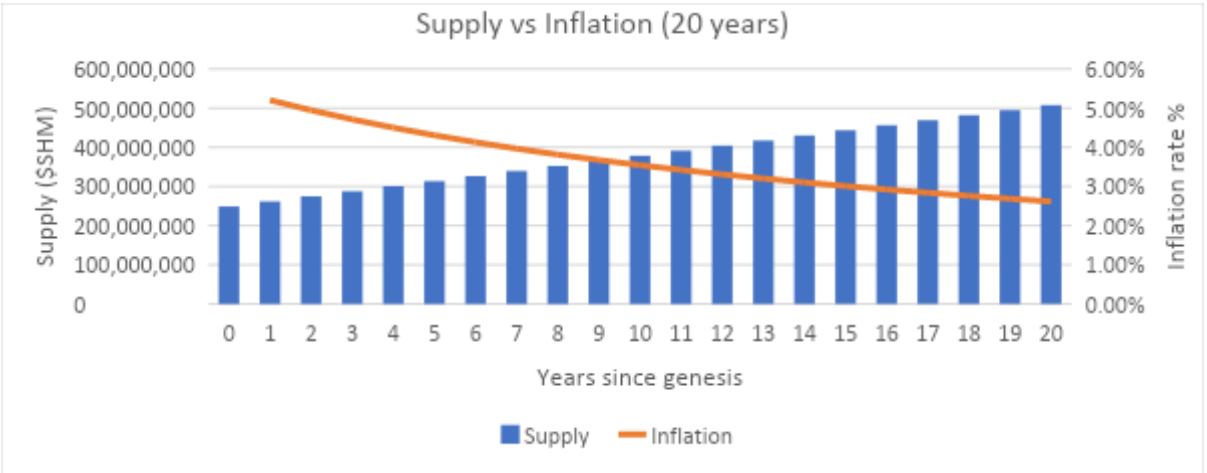

Figure 4.2 *20 year linear issuance supply vs inflation*

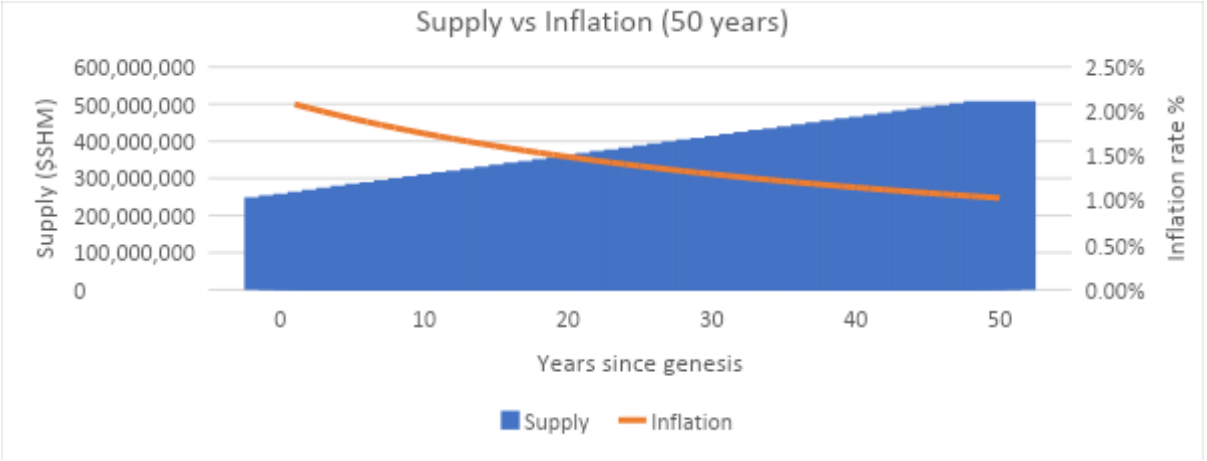

Figure 4.3 *50 year linear issuance supply vs inflation*

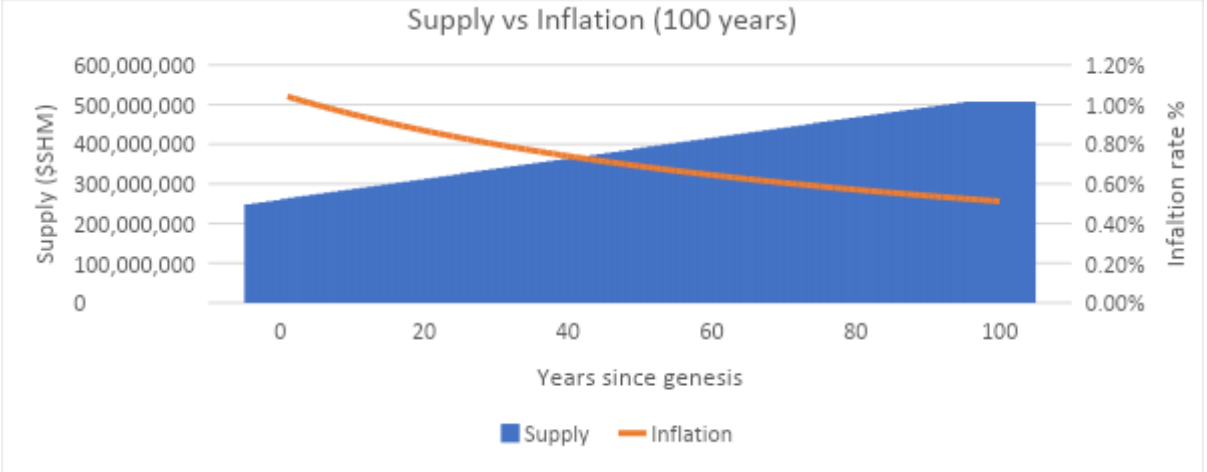

Figure 4.4 *100 year linear issuance supply vs inflation*

Figure 4.5 *20 year linear supply per year (until maximum issuance)*

This approach would make Shardeum an excellent store of value and would make it an easier sell to investors who will not see hyper-inflation reduce the value of their early investment.

Scaling distribution

Monetary statement: $SHM will be distributed for **XXX** years halving every **XXX** years until maximum issuance of 508 million is reached.

This approach is identical to Bitcoin, essentially a number of issuance years is selected and at regular intervals a halving will occur that will reduce the reward for helping to secure the network.

One major benefit of this approach is that early adoption in terms of attracting nodes should be more appealing because the vast majority of coins are given out in the early years as a reward for early adoption (Bitcoin example figure 5.2). The issue this causes is in the early years, inflation of the coins supply is very high. Bitcoin for example had a 209.1% inflation rate in its first full year of issuance (figure 5.1). Bitcoin however did not have a funding round so this was less of an issue, but for Shardeum it is a harder sell to venture capitalists when the inflation rate will effectively reduce the value of their $SHM coins heavily in the early years.

Another issue with this approach is that it is possible that the halving events could make it not profitable for nodes to secure the network. This could happen if the coin price does not increase in-line with the reduction in validator rewards after the halving (more information below in "Ability to remain secure based on transaction fees (tx) alone" section).

Year	#bitcoins	Inflation per annum
2009	1,624,250	-
2010	5,020,250	209.1%
2011	8,001,400	59.4%
2012	10,733,825	34.1%
2013	12,199,725	13.7%
2014	13,671,200	12.1%

Figure 5.1 *Bitcoin's high inflation rate 2009-2014*

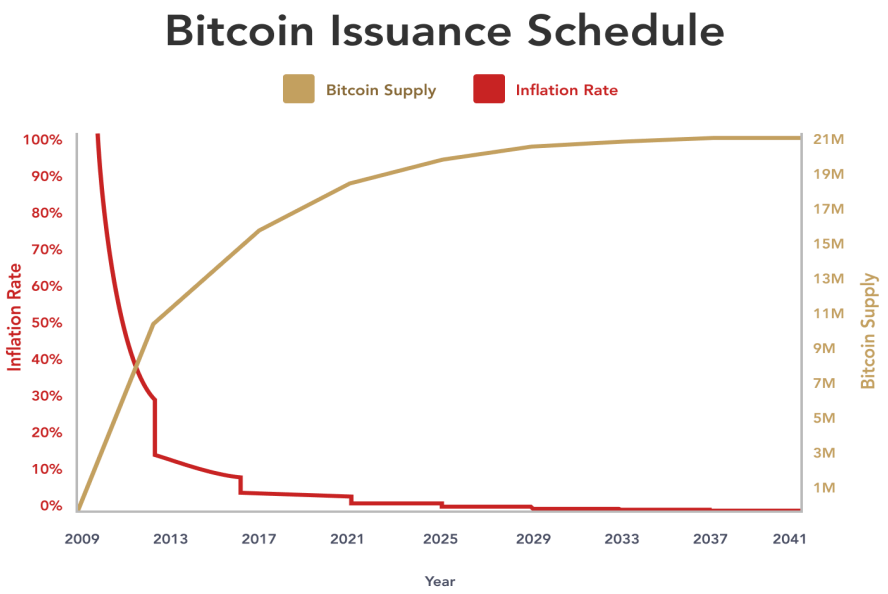

Figure 5.2 *Bitcoin supply curve to maximum supply*

Below is an example of how the $SHM issuance may look following a 20 year scaling model. Due to 49% of the $SHM supply being distributed before Mainnet launch, we do not see hyper-inflation as shown with Bitcoin (figure 5.1). The inflation rate and issuance are highest in the first 5 years, scaling down significantly after each 5 year halving (figure 5.3 and 5.4).

Year	Reward	Supply	Coins Minted	inflation
0	0.00%	248920000		
1	53.33%	276555200	27635200	11.10%
2	53.33%	304190400	27635200	9.99%
3	53.33%	331825600	27635200	9.08%
4	53.33%	359460800	27635200	8.33%
5	53.33%	387096000	27635200	7.69%
6	26.67%	400913600	13817600	3.57%
7	26.67%	414731200	13817600	3.45%
8	26.67%	428548800	13817600	3.33%
9	26.67%	442366400	13817600	3.22%
10	26.67%	456184000	13817600	3.12%
11	13.33%	463092800	6908800	1.51%
12	13.33%	470001600	6908800	1.49%
13	13.33%	476910400	6908800	1.47%
14	13.33%	483819200	6908800	1.45%
15	13.33%	490728000	6908800	1.43%
16	6.67%	494182400	3454400	0.70%
17	6.67%	497636800	3454400	0.70%
18	6.67%	501091200	3454400	0.69%
19	6.67%	504545600	3454400	0.69%
20	6.67%	508000000	3454400	0.68%

Figure 5.3 *Example $SHM issuance following scaling model*

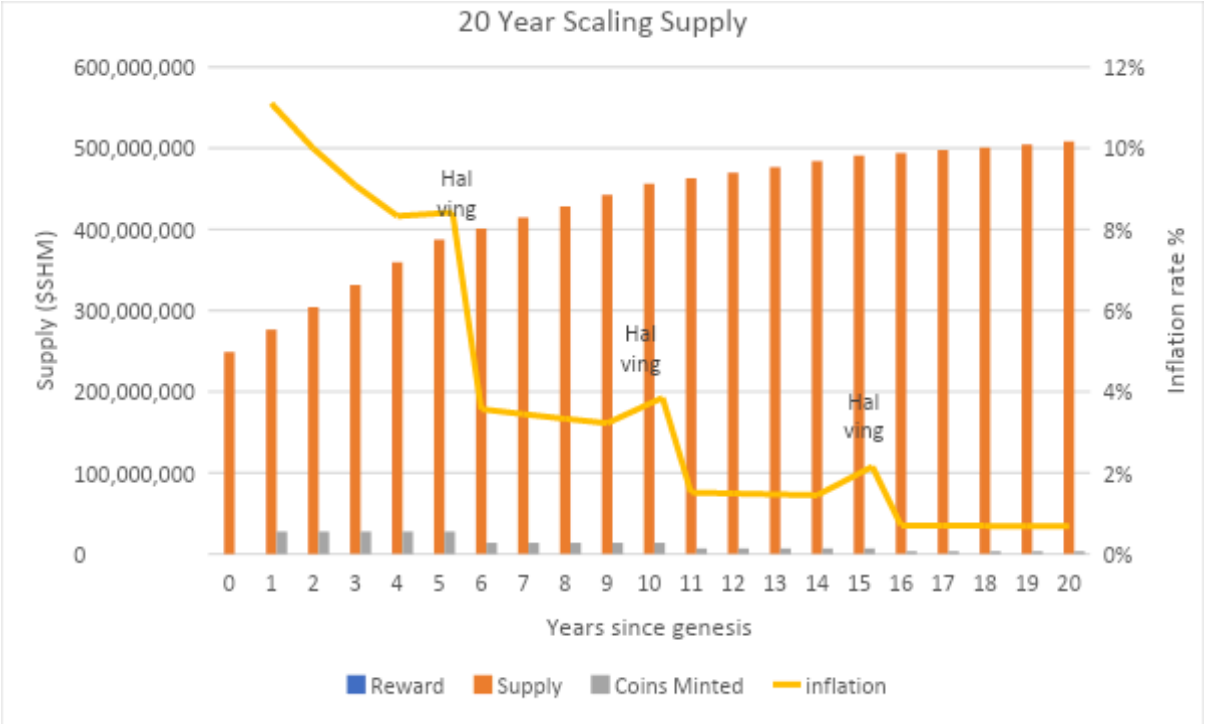

Figure 5.4 *Example graph of $SHM issuance following scaling model*

Variable distribution

Monetary statement: $SHM will be distributed at a variable rate-based on transaction volume for an unknown number of years until maximum issuance of 508 million is reached.

This approach is the least predictable as it has no set number of years until maximum supply cap is reached. It involves using some form of variable to control the rate of issuance and therefore determining the length of time until maximum supply.

This is the approach taken by Avalanche with its native token $AVAX, where it is the percentage of the supply that is staked and the duration of the stake that is variable. In the example below (figure 6.1), the red line represents the shortest issuance duration of around 20 years with 100% of the supply staked for the maximum duration (1 year). The blue line is representative of 50% of the supply being staked for the shortest duration (2 weeks), and full supply is not reached even at the end of the 100 year scale. In reality the time to maximum supply is likely to fall somewhere between these two scenarios.

The benefit with this approach is that stakers are rewarded for staking their tokens for a longer period of time, increasing the staking ratio, adding to the scarcity of the token and most likely increasing its value (more information in the "Should higher node stake be rewarded" section below).

The drawback of this approach is the unpredictability. It would be impossible to identify exactly when full issuance would be reached and so this type of monetary policy makes investors less confident in a project as a long term store of value. This lack of confidence comes from the total unknown value of inflation rate over time.

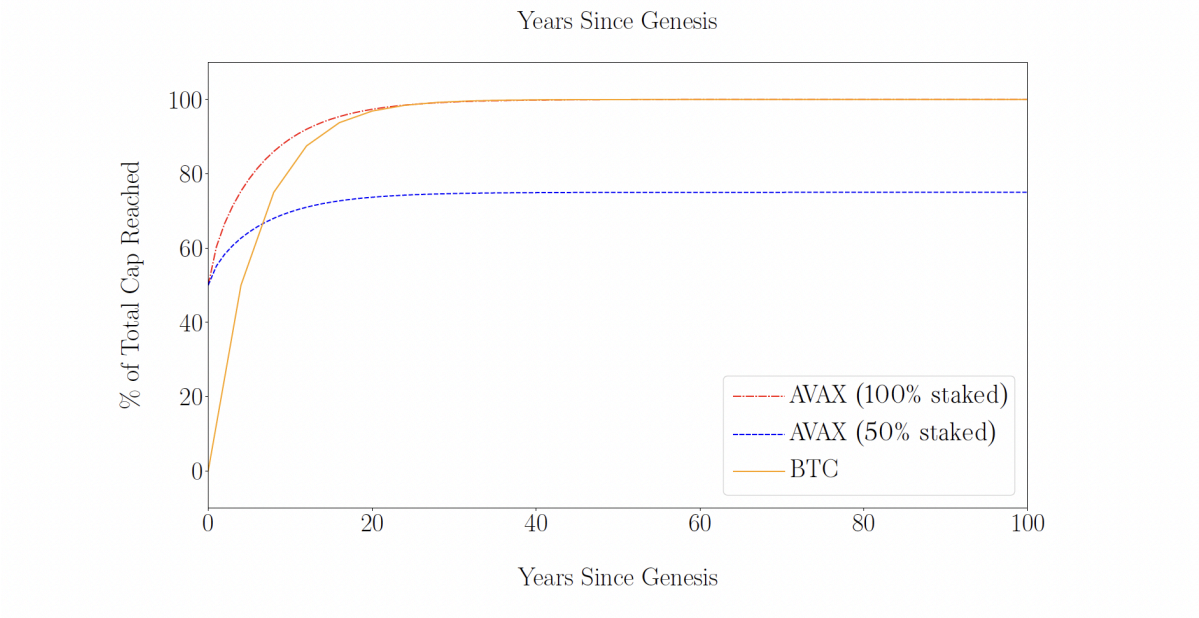

Figure 6.1 *$AVAX variable issuance schedule*

If Shardeum were to adopt this approach, it would need to base its issuance on either the number of transactions or number of validator nodes on the network (see figure 6.2 for example reward scale based on transactions). This would mean if adoption of the project was very high and on-chain transactions increased rapidly, issuance would also increase and therefore maximum supply would be achieved quickly (figures 6.3 and 6.4). If adoption was slower then issuance would be lower, resulting in a longer time period until full issuance was distributed (figures 6.5 and 6.6).

Transaction per second	Reward scale
15k+	53.33333%
10-15k	26.66667%
5-10k	13.33333%
1-5k	6.66667%

Figure 6.2 *Example reward scale based on transactions*

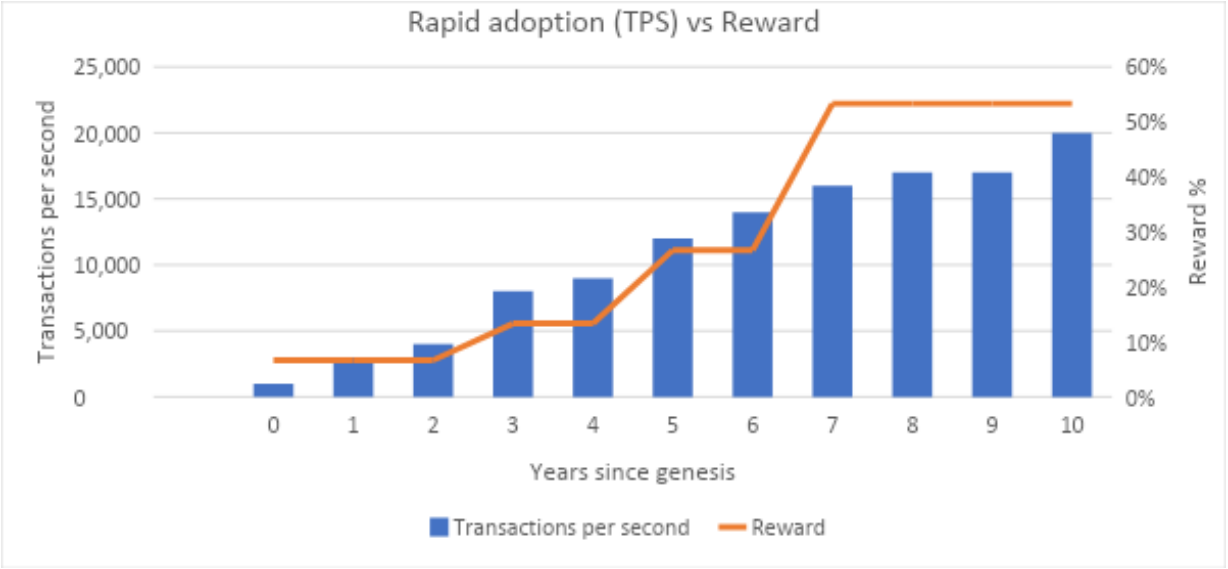

Figure 6.3 *Rapid adoption increasing reward example*

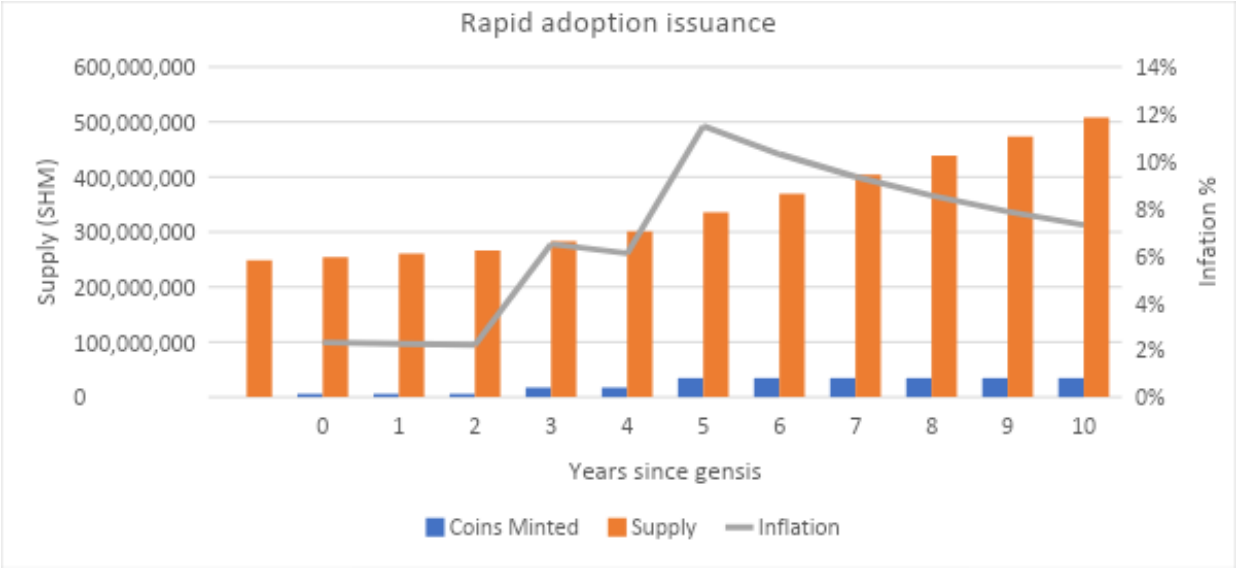

Figure 6.4 *Rapid adoption issuance / inflation example*

The graphs above demonstrate rapid adoption of the Shardeum network. This would mean high transactions leading to high reward and thus increasing the speed of issuance; in this example the entire $SHM supply would be issued in around 10 years.

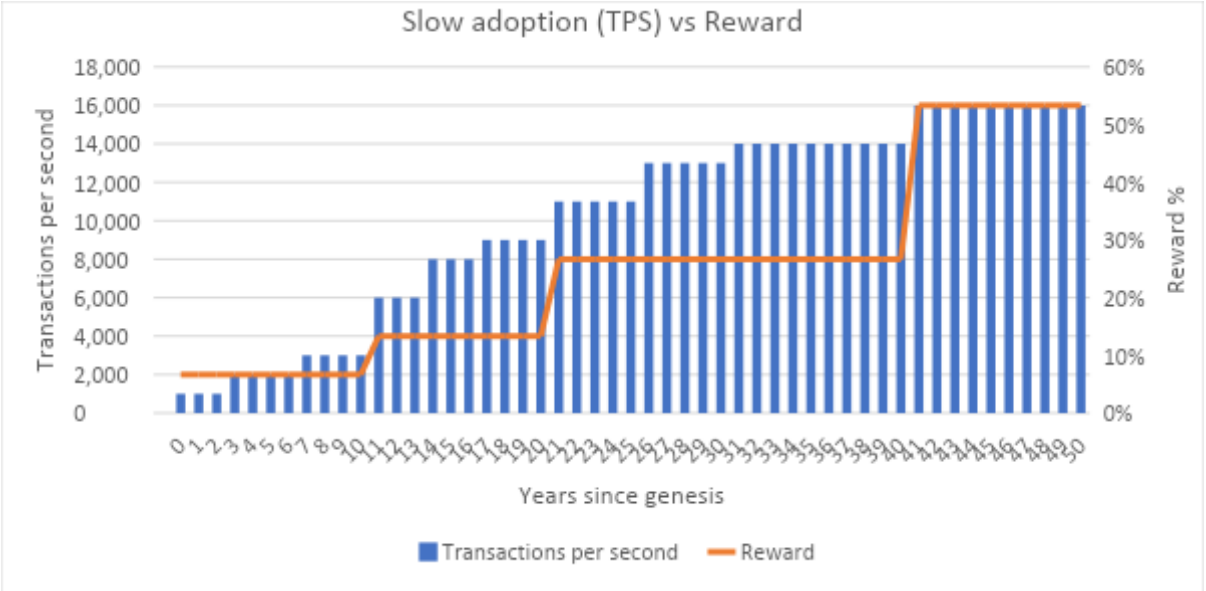

Figure 6.5 *Slow adoption increasing reward example*

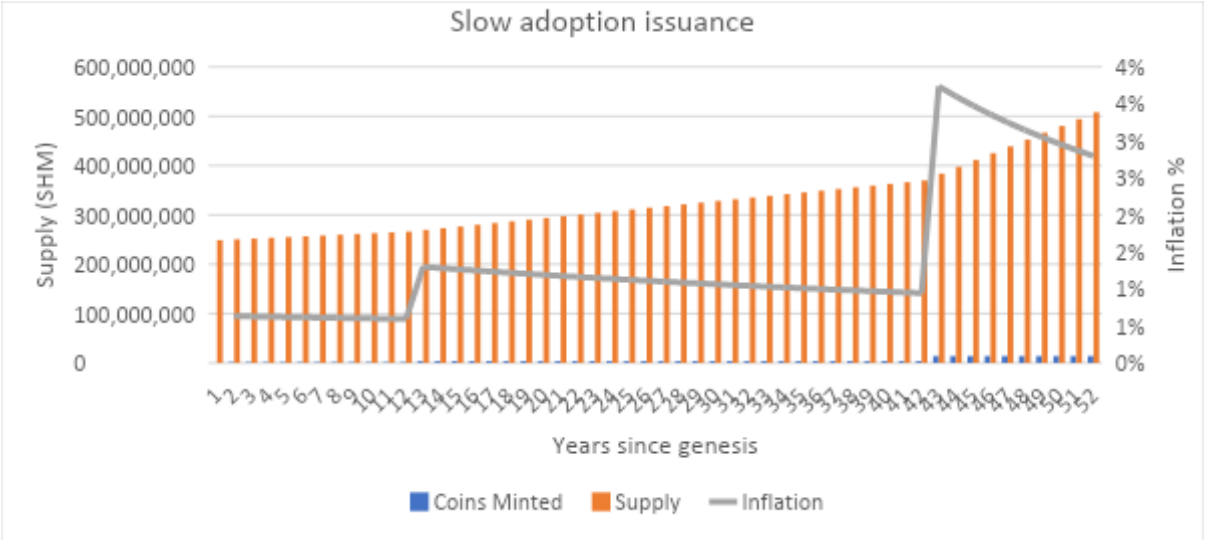

Figure 6.6 *Slow adoption issuance / inflation example*

The graphs above demonstrate slow adoption of the Shardeum network. This would mean a low number of transactions leading to low reward and thus slowing down the speed of issuance; in this example the entire $SHM supply would be issued in around 50 years.

What monetary approach should we take?

Based on the findings above either linear or scaling issuance options would best suit the Shardeum model. The variable approach adds complexity and uncertainty and is not in-line with the OCC guidelines that Shardeum follows (open, collaborative and community driven).

What about rewarding the early adopters?

With most new cryptocurrencies mining / staking yields (in terms of coins) are much higher in the early months / years of the new currency. This is due to the inflation rate (amount of coins minted) being very high initially and tapering down over a number of years via halving or reducing block rewards**.**

The reason this is necessary is because the coins will have a relatively low initial value, so large numbers of coins are given out to incentivise miners / stakers to join a network with limited financial benefit at time of issuance but with the potential for massive financial gain in the future (as the value of the coins increase).

With a linear / scaling issuance of coins for validator nodes, early adopters may not be fairly rewarded for helping the network grow. To counter this a wallet variable (explained below) could be used to reward validator loyalty. This means rather than a validator being rewarded initially for being an early adopter, they would be rewarded in the longer term as they will be entitled to a larger proportion of the coins for being a loyal node (figure 7.1).

Node loyalty example:

Loyal Node			**New Node**	
Stake	10		Stake	10
Node Loyalty	**10**		**Node Loyalty**	**3**
Patience Score	10		Patience Score	10
Daily Time as Validator	10		Daily Time as Validator	10
Total Wallet Score	200		Total Wallet Score	60
Total Day Reward	9668		Total Day Reward	2900

Figure 7.1 *Examples of loyalty being rewarded with higher issuance*

Total wallet score =
SUM((Stake*1)*(Loyalty*0.5)*(Patience*0.2)*(DailyValidatorTime*0.2))

*weightings to be determined by community

This approach not only allows early adopters to benefit in the early months but in fact they will benefit for the entire time they remain part of the network (figure 7.2). In theory this should encourage validator loyalty as well as avoiding validators leaving and re-joining the network when it is most profitable.

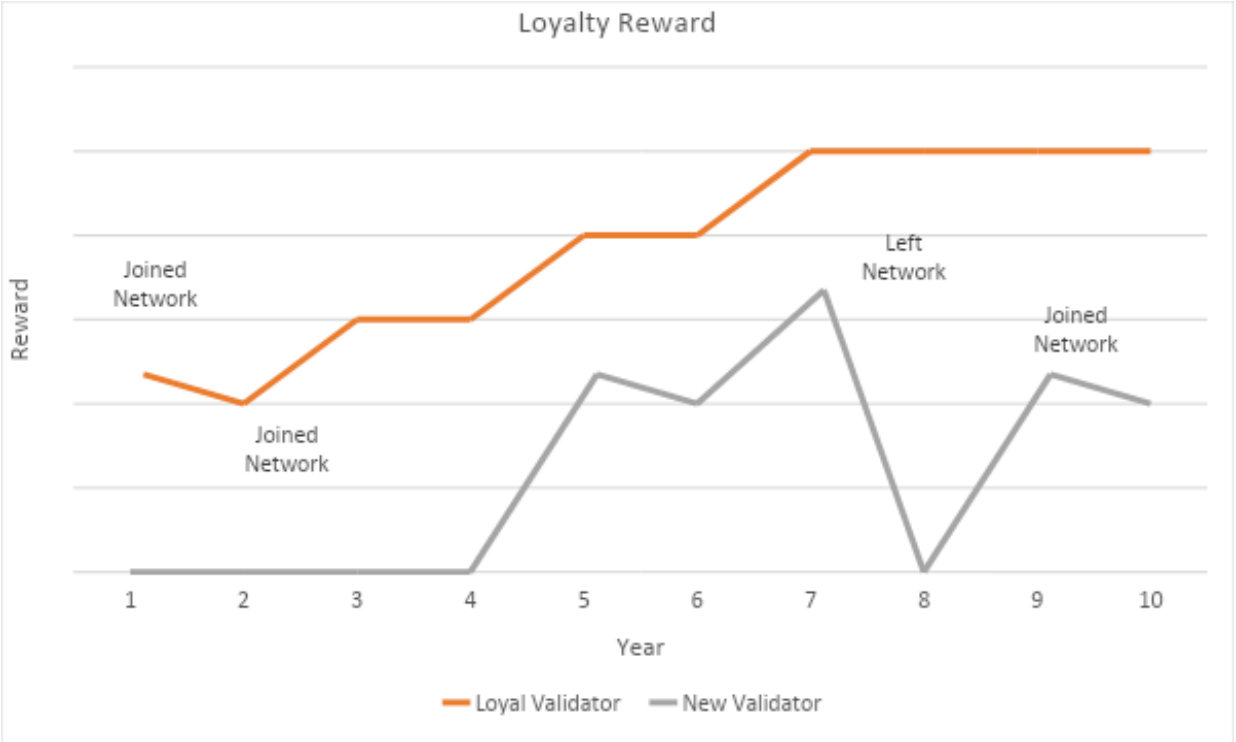

Figure 7.2 Demonstrates how staying part of the network would lead to higher rewards, whilst leaving the network and re-joining will reset the loyalty score and reduce rewards

Potential reward structure

How could archive nodes be rewarded?

The reward scheme for archive nodes could be very simple. These nodes would be awarded a percentage of the total reward (transaction fees (tx) + issuance) that would be equally divided between all the nodes that meet a 99.9% uptime threshold (daily, weekly or monthly) and the minimum hardware requirements (shown below). The validator / archive reward ratio will dictate the percentage of the total reward that the archive nodes receive. The validator / archive reward ratio will be a community variable that could change over time (changeable by User-Activated Software Forks) to ensure fair distribution of rewards among different node types.

How could standby nodes be rewarded?

Standby nodes should not be rewarded with coins, instead the longer a node remains as a standby it will increase its “Patience Score”. This score is a wallet variable (explained below) that will allow the node to claim a larger proportion of the distributed coins if it eventually becomes a validator node.

One of the main reasons behind not rewarding standby nodes with coins is to avoid fake / spoofed nodes from spamming the network and attempting to collect standby rewards.

How could validator nodes be rewarded?

Validator nodes could be rewarded based on the amount of daily issuance controlled by the consensus rules (set at genesis), network variables (controlled by the community) and wallet variables (dynamic based on node behaviour).

Consensus rules (set at genesis, cannot be changed)

Supply: 508,000,000 (community decision)
Burn mechanic: n/a
End of issuance: community to decide
Issuance type: linear, scaling or variable (as described above)
Yearly issuance: depending on above decision
Daily issuance: depending on above decision

Network variables (dynamic or changeable by User-Activated Software Forks (UASF))

Transaction fees (tx): dynamic (depending on network strength (explanation below))
Ideal standby to validator node ratio: community decision - example 10:1
Network strength: dynamic (depending on standby to validator ratio)
Minimum staking requirement: community decision – example 10,000 $SHM
Maximum staking requirement: community decision – example 10x min stake or unlimited
Validator/archive reward ratio: community decision – example 99:1

Wallet variables (dynamic based on validator behaviour, impact validator reward)

Stake – This is the amount of $SHM a node is staking. The minimum and maximum amounts for this will be decided by the community and added to the network rules, and will change over time as the network develops and the coin value increases (User-Activated Software Fork). The higher the stake a node has, the higher reward it will receive when becoming a validator node (explained in "Should higher node stake be rewarded" below).

Node Loyalty – This is a score based on how long a node has been contributing consistently to the network (minor downtime does not affect this). This variable is designed to reward loyal nodes (explained above In "What about rewarding the early adopters" section) as the higher the score, the higher the reward the node will receive when becoming a validator.

Patience Score – This is a score based on how long a node has been on standby and available to join the network as a validator – the higher the score, the more a node will be rewarded when it eventually joins the network as a validator. This score can be slashed if the node is down or does not meet minimum hardware requirements when tested by the network (explained in "How could slashing work?" section below).

Daily time as validator (hours) – This variable is based on how many hours in the current reward day the validator node was available and meeting minimum hardware requirements. The more hours the node was a validator available to process transactions (number of transactions processed has no bearing on reward, all validator nodes are rewarded), the higher reward it will receive at the end of that settlement day (23:59 UTC). The node will lose any hour it was not available to process transactions as a form of slashing (explained in "How could slashing work?" section below).

Example on next page:

Basic Example:

Total Reward (SHM)	
Daily Issuance	35490
Transactions	100000
Transaction Fees (0.1 SHM)	10000
Total Reward (SHM) / Day	145490

Wallet Variables	
Stake	1 - 10
Node Loyalty	1 - 10
Patience Score	1 - 10
Daily Time as Validator	1 - 10

Total Day Validators	10

Validator 1	
Stake	10
Node Loyalty	6
Patience Score	4
Daily Time as Validator	10
Total Wallet Score	48
Total Day Reward	31720

Validator 2	
Stake	10
Node Loyalty	3
Patience Score	10
Daily Time as Validator	10
Total Wallet Score	60
Total Day Reward	39650

Validator 3	
Stake	10
Node Loyalty	4
Patience Score	4
Daily Time as Validator	7
Total Wallet Score	22.4
Total Day Reward	14803

Validator 4	
Stake	7
Node Loyalty	7
Patience Score	6
Daily Time as Validator	7
Total Wallet Score	41.16
Total Day Reward	27200

Validator 5	
Stake	6
Node Loyalty	3
Patience Score	6
Daily Time as Validator	6
Total Wallet Score	12.96
Total Day Reward	8564

Validator 6	
Stake	5
Node Loyalty	5
Patience Score	5
Daily Time as Validator	5
Total Wallet Score	12.5
Total Day Reward	8260

Validator 7	
Stake	4
Node Loyalty	4
Patience Score	4
Daily Time as Validator	4
Total Wallet Score	5.12
Total Day Reward	3383

Validator 8	
Stake	3
Node Loyalty	3
Patience Score	3
Daily Time as Validator	3
Total Wallet Score	1.62
Total Day Reward	1071

Validator 9	
Stake	8
Node Loyalty	5
Patience Score	5
Daily Time as Validator	4
Total Wallet Score	16
Total Day Reward	10573

Validator 10	
Stake	1
Node Loyalty	2
Patience Score	2
Daily Time as Validator	5
Total Wallet Score	0.4
Total Day Reward	264

Total Wallet Scores	220
Total Reward	145490
Reward per Wallet Score	661
Total Rewarded	145490

Figure 8.1 Example validator rewards based on wallet variables

Reward payment

Node rewards could be calculated on a daily basis (could be hourly) at 23:59 UTC based on the wallet variables described above and total daily reward.

Reward Calculation

Total reward = Daily validator reward (issuance) + Transaction fees (tx)

Total wallet score (per node) =
SUM((Stake*1)*(Loyalty*0.5)*(Patience*0.2)*(DailyValidatorTime*0.2))

*weightings to be determined by community

Total wallet scores (all nodes) = All nodes total wallet scores combined

Reward per wallet score = Total wallet scores / total reward

Total day reward (per node) = Reward per wallet score * Total wallet score (per node)

Should higher node stake be rewarded?

Shardeum is a community built on fairness and follows OCC guidelines (open, collaborative and community driven). Initial thoughts suggested all nodes should be rewarded the same regardless of the amount of $SHM staked.

On deeper examination, this approach is flawed and it is likely the reward will be in some part proportional to the amount of $SHM staked. This approach has two key benefits:

Higher staking ratio

Rewarded nodes for staking more $SHM will encourage the network to stake higher amounts, this in-turn will increase the staking ratio.

A higher staking ratio is beneficial for two reasons. First, the staking ratio is a barometer for the health of a PoS chain. Just as important, a high staking percentage signals a large and loyal community that is committed to the project for the long term. If 60% or more tokens are bonded in nodes, you know that a majority of tokens belong to HODLers.

The second benefit of a higher staking ratio is that it creates organic demand for $SHM. In the long run adoption of on-chain applications will drive network usage and demand, but the first major use case and demand driver for the token will be staking.

Improved security

Nodes that stake large amounts of $SHM will be taking a bigger risk as they have more capital that could be slashed if the node misbehaves. Logically nodes with higher capital staked have more incentive to behave and help the network grow than nodes that are just meeting the minimum staking requirements.

Note, just because a node has staked more $SHM it does not get any more voting rights than a node that only meets the minimum staking requirements. The User-Activated Software Fork should entitle each node to one vote irrespective of stake.

How could slashing work?

Downtime slashing

Slashing for downtime within Shardeum could be done within the wallet variables. If a validator node goes down it will be punished by a reduction in the “Daily time as validator (hours)” variable. Any downtime within a specific hour would result in the validator losing that hour from the variable and the coins the validator would have received for that hour will be distributed proportionally to the online validators.

Availability slashing

Standby nodes could be slashed if they are not available (meeting hardware requirements and connected to the network) when checked by the Shardeum network. If the standby node is not available when checked, that nodes “Patience Score” will be reset. This means when that node does eventually become a validator it will be rewarded proportionately less compared to nodes with a higher “Patience Score”.

Double signing slashing

The punishment for double signing for either lazy or malicious behaviour should be severe, resulting in a node losing some or all of its stake and the confiscated coins could then be used for a community DAO (see details below) or for network advancements (maintenance fund).

Could we have a community DAO?

Generally speaking when stake is confiscated, it is burned (explained above) or used to fund network maintenance (maintenance fund) however instead of burning the coins they could be transferred into a community DAO (“Decentralized Autonomous Organization”). A DAO is a community-led entity with no central authority. It is fully autonomous and transparent; smart contracts lay the foundational rules, execute the agreed upon decisions and at any point proposals, voting and even the very code itself can be publicly audited.

Ultimately, a DAO is governed entirely by its individual members who collectively make critical decisions about the future of the project, such as technical upgrades and treasury allocations. The members of the Shardeum community DAO could be nominated and elected by the Shardeum community.

Community members create proposals about the future operations of the protocol and then come together to vote on each proposal. Proposals that achieve some predefined level of consensus are then accepted and enforced by the rules instantiated within the smart contract.

The purpose of this DAO would need to be properly defined but could have foundational rules such as the following:

Proposal for allocation of Shardeum community DAO treasury funds must be any of the following:

1) a project based in a country on the UN list of least developed countries
2) a project based in any country currently suffering from / recovering from geopolitical or humanitarian crisis
3) a project in any country currently suffering from / recovering from natural disaster
4) etc……

Could transaction fees (tx) be dynamic?

Transaction fees could be totally driven by the strength of the network, therefore when the rewards are high (issuance + tx) the network of nodes will grow. Once the amount of surplus nodes reaches a certain limit (and becomes wasteful), the transaction fees (tx) will automatically start to reduce (wasteful based on ideal standby / validator ratio network variable).

The reverse process would happen when the network is weaker in terms of surplus nodes (the transaction fees (tx) will automatically increase). This will make becoming a validator node more financially rewarding, attracting new nodes and increasing the strength of the network.

This can be best explained using the demand curve below (figure 9.1). Equilibrium is important to create both a balanced and efficient market. If a market is at its equilibrium price and quantity, then it has no reason to move away from that point because it is balancing the quantity supplied and the quantity demanded. However, if a market is not at equilibrium, then economic pressures arise to move the market toward the equilibrium price and equilibrium quantity. This happens either because there is more supply than what the market is demanding, or because there is more demand than the market is supplying. This balance is a natural function of a free-market economy.

In terms of Shardeum, the majority of the time the network should be able to reach its own equilibrium point. If the supply of nodes is very high then the reward per node will reduce, eventually reducing the amount of nodes on the network. If the supply of nodes is low then the rewards will be very high, eventually this will lead to additional nodes joining the network and once again finding equilibrium.

The change in transaction fees will only happen if the amount of nodes on the network is wastefully high or critically low as this will force the network to more quickly regain its equilibrium level.

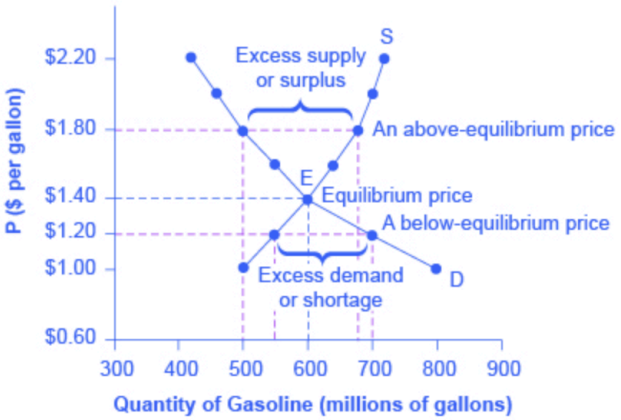

Figure 9.1 *A demand curve for Gasoline*

Ability to remain secure based on transaction fees (tx) alone

One of the potential problems of Bitcoin's economic model is that it is unclear if or when transaction fees will be able to compensate for the decreasing block reward issuance, presenting a potential time bomb within the protocol. For any protocol to survive in the long term, it will need to bring in enough transaction fees to at least sustain the cost of operating and securing the network.

The optimum scenario is that when the issuance period ends, the value of the coin ($SHM) will have appreciated enough in value for the network to be rewarding enough for nodes to ensure its security. If this is not the case the dynamic transaction fees (tx) mentioned above would compensate and increase in order to secure the network. If the reverse happens and the coin value is so high that the network is overrun by surplus (wasteful) nodes, the transaction fees (tx) will decrease and eventually lower the number of nodes on the network.

Considerations

1. With low issuance in early years and vesting periods on seed and public sale coins, the amount of liquid $SHM on exchanges could be low.

2. With this approach it would be impossible to give an accurate ROI % on staking as it would totally depend on wallet variables. It would however be possible to give averages for ROI from previous cycles.

What are the nodes types and hardware requirements?

Archive nodes

The archive nodes would be high powered servers hosted by data centres. These nodes are responsible for holding the entire ledger information and do not complete any transaction or consensus work.

Amount: (around 20)

Estimated hardware requirements

CPU: Dual Intel Xeon E5-2680 2.70 GHZ Octa Core (16 cores)
RAM: 512 GB
Storage: 10 TB drive
Other requirements:

- Hardware firewall
- Premium DDOS protection
- Multiple network cards

Estimated monthly cost (cloud) – 1000usd

Validator nodes

The validator nodes are responsible for securing the network, completing and verifying transactions. They are lower spec machines capable of joining other nodes to form a shard.

Amount: (unlimited based on transaction activity)

Estimated hardware requirements

CPU: 8 core CPU
RAM: 4 GB
Storage: 64 GB drive
Network: IPv4 or IPv6 network connection, with an open public port, capable of at least 30Mbps throughput.

Amount: (unlimited)

Standby nodes

The standby nodes are identical to validator nodes in terms of hardware and purpose but they remain idle waiting to join the network as a validator node as transaction activity increases or old validator nodes are cycled out. The number of standby nodes is unlimited and should not be capped (explanation below).

How are nodes replaced / cycled?

At time of writing, one node is replaced every one minute, meaning the network will cycle through 1440 nodes in a day. The replacement criteria is currently based on node age (oldest first) and replaced by a standby node at random (lottery).

Should the amount of standby nodes be unlimited?

Having unlimited standby nodes has a few advantages:

Better security

The more standby nodes the network has the less probable the lottery system is to pick a malicious node at random (providing the number of malicious nodes remains the same). As the amount of surplus nodes increases, the probability of a 51% attack is reduced dramatically (see figure 10.1 for calculation).

$$P(A) = \frac{n}{N} = \frac{\#\,outcomes\ in\ A}{\#\,outcomes\ in\ Sample\ Space}$$

Figure 10.1 *Probability formula*

Example 1:1 vs 6:1 standby to validator node ratio 51% attack probability (based on 10 validator node network)

1:1 Ratio (10 standby : 10 validator)

P(A) = n/N = 6 (malicious standby nodes) / 10 (total standby nodes) ^6 (number of replaced nodes)

Probability of next 6 nodes selected to be validators all being malicious resulting in 51% attack = 4.6656%

6:1 Ratio (60 standby : 10 validator)

P(A) = n/N = 6 (malicious standby nodes) / 60 (total standby nodes) ^6 (number of replaced nodes)

Probability of next 6 nodes selected to be validators all being malicious resulting in 51% attack = 0.0001%

Note, this is not how Shardeum works, this is a basic example to demonstrate the benefit of increased standby nodes.

Higher staking ratio

Allowing unlimited standby nodes on the network will increase the total amount of coins ($SHM) that are staked, thus creating a higher staking ratio (more information above in "Should higher node stake be rewarded?" section).

What other questions do we need to answer?

1. Will Archive nodes need to stake?
    - If so, how much?
    - How do we ensure decentralisation with Archive nodes?
    - What happens if all Achieve nodes go down?
    - Will they be slashed?

2. Does the project need a maintenance fund?

3. Could maximum node stake be unlimited?