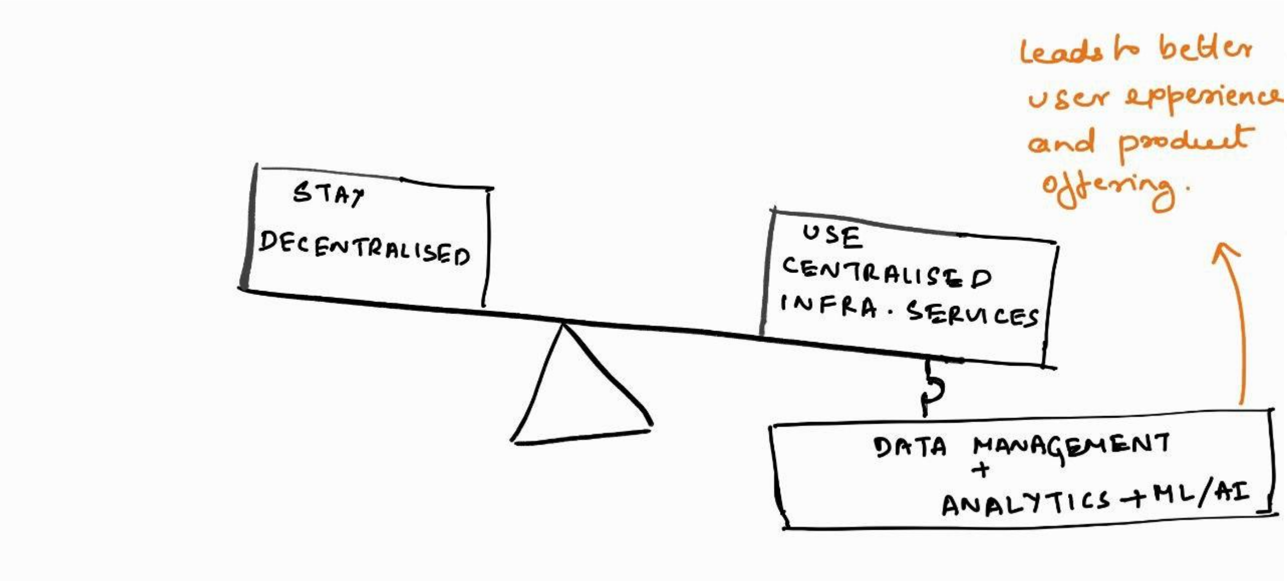
KANDOLA NETWORK

# Kandola Tokenomics

## Motivation

One of the fundamental cornerstone achievements of the decentralized ecosystem is its ability to continue operating even if some parts of it are shut down. Decentralised Smart contracts execution VMs ensure that the business logic execution will be reliably executed for ever. This is an extremely powerful construct which the centralized solution apps cannot offer to their patrons.

But there is still a gap that prevents this from becoming a **true** reality. DApps are not truly decentralized today. The reason is the application **data**. Dapps need to use their data to understand app performance, user behavior and interaction trends and customize the experience for each user. This requires analytics, rules engines, ML and AI models and data management. Without these tools, Dapps will continue to be inferior in their user experience and continue to lack widespread adoption.
Apps cannot store their data on blockchains, because it is extremely expensive to do so. 1 MB of storage on the Ethereum network costs ~$17000. To add to the misery, there are no web3 native platforms that allow data management and analytics capabilities. Applications are thereby limited in options and store data on centralized data systems compromising the whole true decentralized existence and ability to provide the Web3 promise.




Kandola Network is a decentralized database and data analytics platform that is designed to provide full database capabilities and Machine Learning at scale. Data stored in the network is available with extremely high throughput for queries and access. Unlike IPFS systems, Kandola does not store files. It is designed for storing application data which is text based structured data, mostly as JSON,XML or data tuples.

Kandola network helps enable a whole ecosystem of open, verifiable and trust based applications that are otherwise extremely difficult to build. With Kandola we hope to usher in a new age of Web3 and the next billion inclusion in Web3.

KANDOLA NETWORK

# Introduction

The objective of this document is to elucidate our token design, which has been structured to balance utility and return on investment.

The utility of the Kandola Network's token is self-evident, serving as the medium of transaction for the data storage and access services our platform provides. This utility is a fundamental contributor to the token's value.

Simultaneously, the prospect for return on investment is an essential consideration in shaping the token's value. Our token design aims to offer fair opportunities for investors, thus increasing the incentive for maintaining investments within the network.

Addressing pricing predictability and protection from price volatility emerges as a fundamental challenge that our token design confronts head-on. It is noteworthy that the crypto space has witnessed instances where unexpected price volatility led to significant fluctuations in the cost of platform services. Such scenarios have previously resulted in prohibitive service costs, undermining user experience and accessibility. Our design approach strives to mitigate this prevalent issue, ensuring that our platform remains customer-friendly and economically viable irrespective of market dynamics.

To estimate future token valuations, we utilize the monetary equation of exchange (MV=PQ), a model that has been used since the early days of Bitcoin. We employ a variant of this equation, as proposed by the INET valuation model, to conduct simulations that ascertain the value of M under various stress scenarios, thus defining the boundaries of equilibrium pricing.

# The Kandola Ecosystem

## Actors

Before delving into the specifics of the Kandola Ecosystem, it is essential to comprehend the holistic structure and the flow of the token within this system. The Kandola Network is composed of the following internal actors, each playing a distinct and crucial role in maintaining the efficiency and integrity of the platform:

- Log Nodes: These nodes form the backbone of the Kandola Consensus and are tasked with maintaining the distributed ledger known as the “commit log”. All customer write operations are facilitated via these nodes, making them a key component of the network.
- DataBase Nodes: These nodes interface with the Log Nodes, ingesting messages once the data has been finalized in the “commit log”. Subsequently, they modify the database. In essence, these nodes provide the Database Service to the end customers. All customer read operations are facilitated through these nodes directly, marking their central role in data delivery.
- Audit Nodes: The Audit Nodes are responsible for conducting regular assessments of the availability of service provided by other nodes. These assessments encompass verifying the accuracy of data served by the DataBase Nodes, ensuring adherence to service agreements, and detecting any byzantine behavior within the network.

For more details please refer to the Kandola Consensus whitepaper here.

## The Customer Flow

In the Kandola Network, customers compensate the network for the utility it provides, specifically a decentralized database and analytics service. The operation of such a network incurs costs that are broadly categorized into two components - storage cost/GB/month and query/access cost/GB/month. These two facets represent the dual services demanded from the network: storage size for data preservation and compute and bandwidth for queries and updates.
The customer experience within the Kandola network follows these steps

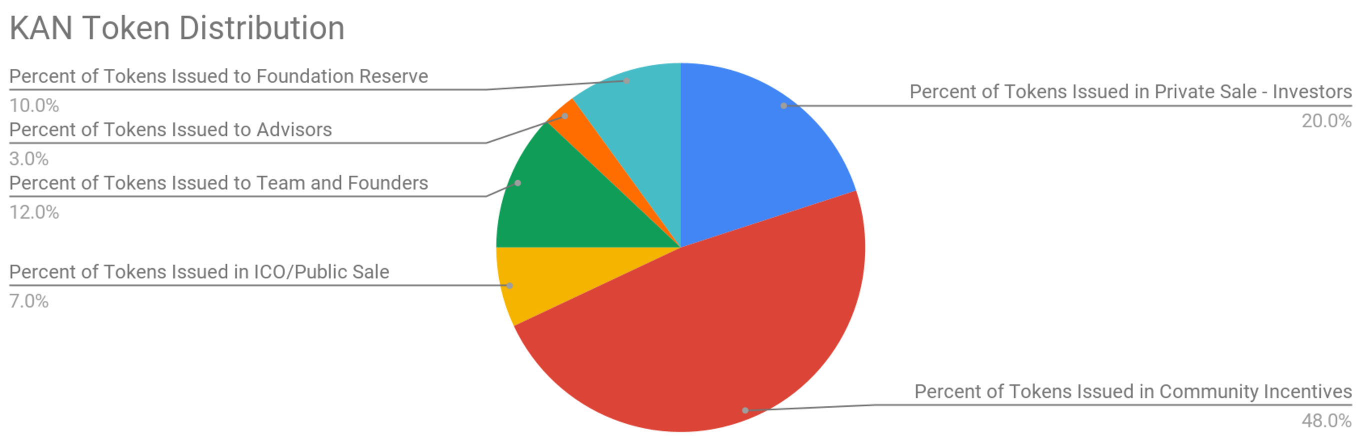
1. The customer specifies the total storage required and an estimate on the egress(query) frequency, referred to as a Request for Storage Proposal (RSP). While the RSP includes additional details such as replication factor, region, data schema, and backup requirements, these are not pertinent to the reader interested in tokenomics and have been omitted. For further details, please refer to the whitepaper. The customer also specifies the billing frequency, which could be monthly, quarterly or yearly.
2. A cost is computed for the customer based on the RSP and the chosen billing period and a minimum stake based on this RSP is calculated by the network. This would be the stake the customer would have to put in to avail the data services. This stake serves as a payment guarantee, akin to the role of credit cards in conventional systems.
3. The RSP is disseminated to all the database nodes. Nodes capable of providing the requested storage respond with an RSP Response (RSPR) to the customer. Details of this selection process are mentioned in the whitepaper.
4. Upon customer agreement to the RSPR, a storage contract is formalized between the customer and the Database nodes, and the environment is prepared for the customer. This initiates the creation of a new chain and the commissioning of endpoints and database services for the customer.
5. The execution of the above contract also includes the staking of the customer mentioned in Step 2 and locking it for the mentioned billing period.
6. Customers can now begin storing and retrieving data through the Kandola Network.
7. At the end of the billing period, the customer is invoiced based on the storage size and the amount of data requested in queries. The corresponding amount is deducted from the customer's stake. Customers are expected to top by the stake again for the subsequent billing cycles.
8. The deducted sum is then apportioned between the various nodes, based on the storage and data access provided and the efforts of each node in delivering the requested service.

# The Kandola Token and PoS

The Kandola Token ( symbol KAN ) is an ERC20 token. The token economics is a variant of PoS especially around the utility pricing mechanism. The technical whitepaper explains the stake implications for the nodes in the leader election process and how it also impacts the database nodes. The later section in this document will detail the tokenomics approach and the reasons for the choices we made.

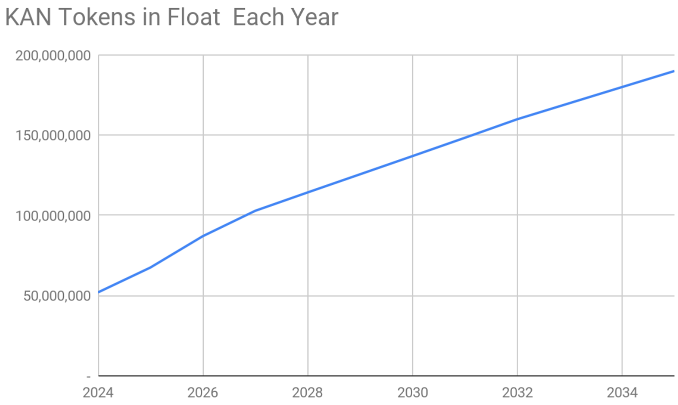
## Token Supply and Distribution

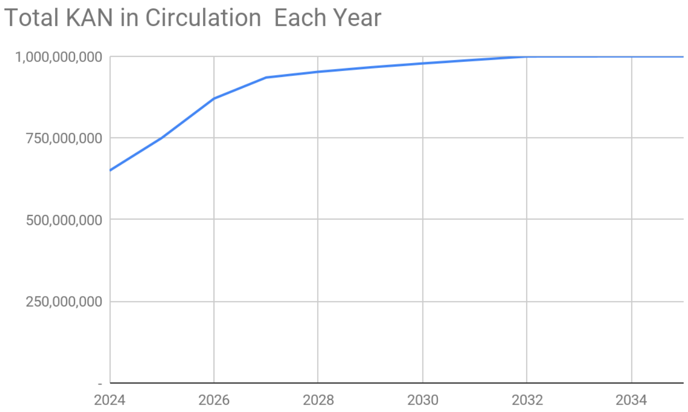
KAN token supply is capped at 1 billion tokens. Distribution of the token is as follows:



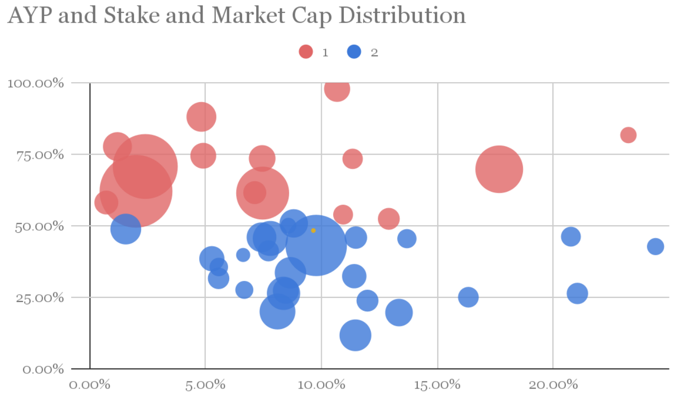

| Allocation heads | Percent Allocation | Lock up Period( Months) |
|---|---|---|
| Community Incentives | 48% | 0 |
| Founders | 12% | 36 |
| Advisors | 3% | 24 |
| Public Sale | 7% | 0 |
| Private Sale | 20% | 12 |
| Foundation Reserve | 10% | 0* |

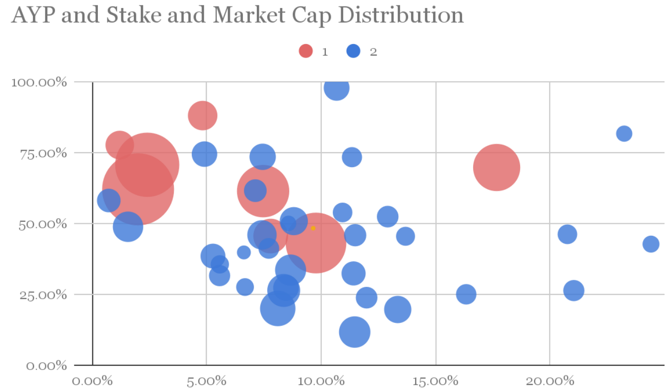
With the above allocation and an assumption of a token velocity of 20, and a total hodl of 60%, the total tokens in circulations and the token in float circulations would be the following. More details here




Assumptions of hodl ratio have been calculated on empirical evidence. Below graphs depict the top 50 crypto projects hodl ratio and AYP and market caps. More details here.




# System Design

For KAN economics we have separated the two concerns of utility and return on investment. Each has different objectives and actors involved. For example, utility pricing design has to keep the customer needs and objectives in the foreground, while staking designs which address hodl return have to keep the nodes, community and the consensus design into consideration.
One could also argue that for the nodes, these are also two distinct investments.
One is the hardware and infrastructure cost that is required to set up the node and other is the stake the node wants to invest. Till the time hardware companies and ISP providers accept payments in ERC20 tokens, it would be a fair assumption that infra investment by the nodes is in FIAT(could be equated to stablecoins). Nodes would want to protect this investment with a return that is comparable to a parallel

investment for the same amount ( like putting the same amount in bond ). If this is tied to the volatility of the token, then prediction on the return on investment for the FIAT( or stablecoin) investment cannot be predicted and this investment would not be lucrative.
The other investment is the stake the node is willing to pledge to participate in the network. Staking is a means of increasing chances of getting more rewards from the network, but the more direct impact it has on the economy is creating scarcity of tokens. This is a macroeconomic play with more variables involved. If there is sufficient scarcity, then it is reasonable to expect that the price of the token will increase.
The Kandola economics design is created to reduce the risk for every participant and provide some guaranteed reasonable returns. Hence bifurcation of the utility and staking economics was the logical choice.

## Utility Pricing Design

Central to the design of a successful token model is the thoughtful design of its utility function, a process which invariably places the customer at the heart of the system. In designing our token, we sought to prioritize ease of adoption and use of our primary utility - the data storage and access service.

Consider a scenario where the adoption and use of a particular service drives the price of its associated token upwards. While initially appearing beneficial for token holders, this trend can quickly reach an inflection point where the cost of the service becomes prohibitively expensive due to the inflated token price. The end result is a paradoxical situation where increased demand for a service leads to reduced accessibility, a dynamic that is fundamentally counterproductive to the goal of widespread adoption.

A prominent real-world example of this dynamic in action is the Ethereum network. As the popularity and usage of the Ethereum network has grown, so too has the price of Ether (ETH), its native token. While this has led to significant returns for early adopters and investors, it has simultaneously resulted in skyrocketing transaction fees (gas fees) on the network. This has raised the cost of interaction with Ethereum-based services and reduced their accessibility for many users, particularly those engaging in smaller transactions.

This also is better understood with another example. Take Uber's dynamic pricing. Uber decides the dynamic pricing based on its algorithms. It does not allow its drivers to set the prices. One could say that drivers are not best positioned to understand the supply demand mechanics in realtime and hence be the wrong choice for such a responsibility. Also it would increase the complexity of the system and would negatively impact the end user experience(there are few aggregators worldwide that allow customers to pay extra if they want a ride urgently and also negotiate prices). But one could also argue that the reason why Uber does so is for profit maximization.
But there is a more fundamental impact to dynamic pricing, and that is the impact it has for wider adoption. As Steve Waldman puts it

*"If the "ride-sharing revolution" really takes hold, a lot of us will have decisions to make about whether to own a car or rely upon the Sidecars, Lyfts, and Ubers of the world to take us to work every day. To make those calculations, we will need something like predictable pricing. Commuting to our minimum wage jobs (average is over!) by Uber may be OK at standard pricing, but not so OK on a surge."…..*
*"It's clear that in a lot of contexts, people have a strong preference for price-predictability over immediate access.The vast majority of services that we purchase and consume are not price-rationed in*

*any fine-grained way. If your hairdresser or auto mechanic is busy, you get penciled in for next week. She doesn't tell you she'll fit you in tomorrow at double her usual rate."*

The other reason which mandates a predictable pricing is the price elasticity of the data storage market.

*"With the fixed-asset model, when enterprise consumers need computing resources, the budget for the needed computing resources must be secured before placing an order. It then takes weeks, if not months, for the needed computing resources to be delivered and deployed. With the utility model, consumers have instant access to computing resources with no capital expense, zero lead time, and affordable prices. In other words, the utility model makes it possible for customers to react to price changes in a timely manner."*
*- Jiang, Qingye & Lee, Young & Zomaya, Albert. (2016). Price Elasticity in the Enterprise Computing Resource Market. IEEE Cloud Computing. 3. 24-31. 10.1109/MCC.2016.14.*

At Kandola, we have sought to learn from these and several other examples and customer interviews and designed our token model to avoid such pitfalls. As a result, we have chosen to **compute the price of the network services in stablecoins.**

By decoupling the cost of our service from the price of our token, we aim to provide our customers with a predictable and stable pricing model, irrespective of market fluctuations in the price of KAN. While the pricing is calculated and presented in stablecoins, the payment is accepted only in KAN, and hence still maintaining the buy pressure on the token while not sacrificing predictive pricing. Mechanisms like locking price for longer selected billing terms are other options to allow even smoother price experience in our network.

Let's understand this a bit more. When we say the pricing is computed in stablecoins it means that the cost of the service is represented in stablecoins. For example, let's assume that at time T0 the cost of storing 1gb of data in the network is 1 USDT /month. Also at this time, KAN is trading at 1 KAN = 2 USD. Now because the network accepts payments only in KAN, the customer buys 0.5 KAN and pays the network. 0.5 KAN = 1 USDT based on the above assumption. At a later time T1 let's assume the cost of storage remains the same, but KAN is now trading at 10 USD. That means at time T1 for paying for 1 GB of storage, the customer now needs only 1/10 KAN tokens or 0.1 KAN. So while the amount of KAN that the customer has to pay changes, the amount in stablecoins is predictable and not as volatile because the pricing of storage does not change as frequently and does not vary much. This provides the predictability to the customer and the much needed protection from token value fluctuations. Imagine if the price was done in KAN, then at time T0 the customer would be paying 2 USD and at time T1 it would be 10 USD.

## Benefits of stablecoin Pricing Model

**Simplicity and Transparency:** Pricing our data storage and access service in stablecoins simplifies the cost understanding for all stakeholders. Nodes can easily calculate their potential return on investment (ROI) based on the cost of infrastructure setup, operation, and maintenance - all of which are typically calculated in USD. This transparent pricing model gives nodes a clear picture of their potential earnings and allows them to make informed decisions about their investments.

**Maintaining Buy Pressure on KAN:** Although we express our pricing in USD for stability and predictability, it's important to underline that the **only accepted form of payment for using our services is KAN.** This means that regardless of our pricing model, there will be consistent demand for

KAN as customers will need to acquire it to pay for services. This ensures a continuous buy pressure on KAN, supporting its value in the market.

**Predictable Returns:** With the cost of services tied to USD, nodes can anticipate a predictable return on their hardware and infrastructure investment. For instance, if they expect a 15% ROI on their infrastructure investment, our pricing model ensures that this return is achievable and not affected by the price volatility of KAN.

**Investment Separation:** Pricing in USD helps segregate the investment models - infrastructure setup and token staking. Infrastructure setup costs and their returns remain stable, insulated from the price fluctuations of KAN. This separation provides nodes with two distinct investment avenues. They can earn a steady income from the infrastructure setup and simultaneously participate in the potential upside of KAN's market value through staking.

**Fairness and Accessibility:** Our pricing model is designed to be fair and accessible to all stakeholders. Customers get a clear, stable price for data storage and access services, while nodes get a fair return on their infrastructure investment. This pricing stability ensures that our services remain accessible to users, regardless of the market value of KAN.

**Ease of Onboarding for Enterprises:** Enterprises are accustomed to traditional pricing models, which are generally expressed in a stable currency like the USD. By pricing our services in USD, we align ourselves with the models they are familiar with, potentially accelerating their adoption of our services. It simplifies budgeting and cost projections for enterprises, as they can easily understand and anticipate the cost of using our services.

**Simplification of International Transactions:** Using USD as the pricing model also simplifies international transactions. USD is a universally recognized and easily convertible currency. This eliminates the complications of exchange rates and currency conversions for our global customers.

**Price Stability Encourages Usage:** Pricing stability is crucial for encouraging regular use of our platform. When customers can predict the cost of our services, they are more likely to use them regularly. This, in turn, can lead to an increase in the overall demand for KAN.

By integrating these considerations into our pricing model, we aim to foster an environment that promotes stability, transparency, and predictability for all our stakeholders. The separation of infrastructure investment from token staking allows nodes to manage their investments effectively and minimizes the impact of KAN's market volatility on their ROI from infrastructure.

## Price Determination

Hardware and infrastructure costs could be said to be a decaying cost curve due to reduced cost overtime. This is attributed to advances in technology and cheaper manufacturing of components required to compute and store data. But the hardware ecosystem should be considered as a sticky market. One argument is that the increase of cost is due to the cost of innovation . Innovation that renders older models less desirable or inefficient to current time demands, causes cost implications that are relevant to current time and does not account for cost depreciation over time.

Hence we cannot prefer one model over the other. The convergence point of both the models could be a reasonable assumption of balance over time.
In the exponential decay model, the cost of the hardware decreases exponentially over time. The formula for this model can be expressed as:

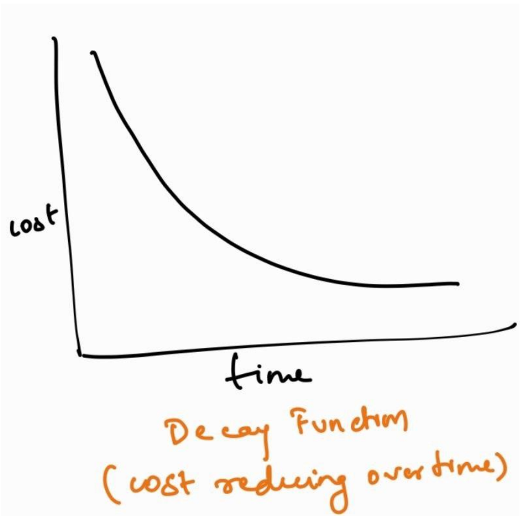
$$C_{(t)} = C_0 \times e^{(-kt)}$$

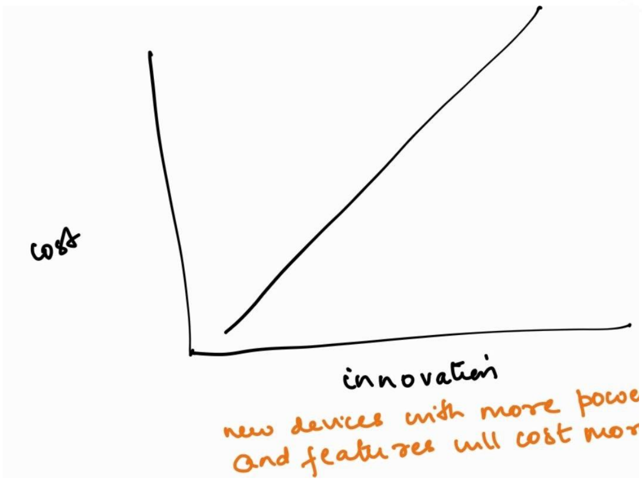
Where:
- $C_{(t)}$ represents the cost of the hardware at time t.
- $C_0$ represents the initial cost of the hardware.
- k is the decay constant, determining the rate at which the cost decreases.
- e is the base of the natural logarithm (approximately 2.71828).

In the linear growth model, the cost of the innovation causes newer device's costs to increase linearly over time. The formula for this model can be expressed as:

$$C_{(t)} = C_0 + mt$$

Where:
- $C_{(t)}$ represents the cost of the product at time t.
- $C_0$ represents the initial cost of the product.
- m represents the rate of innovation increase per unit of time.




The intersection of these two points would be the time where high cost of innovation of new devices and low cost of manufacturing cancel each other out. Let us call this balance point $C_B$ . If someone is buying hardware below this point, we can assume that it is a commodity hardware and someone buying above this point means that they are buying specialized hardware with cutting edge technology. This consideration is important because a lot depends on the consensus mechanism that drives the need for what kind of hardware is required to run the network. For example, PoW incentivised people to invest in

some of the most powerful computer grids in the world, ASICs included. That raised the investment required to participate and get meaningful rewards in PoW systems over time. Kandola consensus is designed to operate on commodity hardware and while better systems definitely help in scaling the network better, our tests show that we will hit bandwidth limits before we hit compute limits. Hence we can assume that nodes in our network will invest in commodity hardware and the cost of it will decrease over time overall. The same is true for bandwidth costs as well. We see worldwide bandwidth costs are reducing with the introduction of new technologies and ever higher bandwidths being introduced every year. Assuming the fixed costs stay the same, the overall cost of running the network reduces over time.
Let us assume this cost to be C. Total cost plotted over time would be a linear decreasing function. We assume this decrease to be 10% per year. In other words, we assume the cost of investment required for infrastructure to participate in the network to reduce 10% every year.

## Dynamic Conversion Rates

If P is the price of the service in stablecoins, and R is the current exchange rate of KAN to USD, then the number of KAN tokens (E) required to purchase the service can be calculated as:

$E = P / R$

This formula would ensure that the cost of the service remains stable in coin terms, regardless of the price of KAN.

But given that R could be changing constantly, a constant real time price oracle would only make the system complex and pricing calculations more cumbersome. Hence the value of R would be changed every 24 hrs, and would be based on the median cost of KAN over the last 24 hrs.

The rationale behind this approach is to balance the need for price stability with the need to reflect market conditions. If the conversion rate was updated more frequently, it might lead to significant price volatility due to short-term market fluctuations. This could potentially confuse or frustrate users who see the cost in KAN tokens changing rapidly.
On the other hand, if the conversion rate was updated less frequently, the prices might become out of sync with the market value of the KAN token, which could lead to either overpayment or underpayment for services, depending on market conditions.

A 24-hour period strikes a balance between these considerations. It's long enough to smooth out short-term volatility, but short enough to ensure that prices stay reasonably close to market conditions.

## Staking

Kandola Network leverages a staking mechanism reminiscent of Proof of Stake (PoS) principles to bolster network security, establish consensus, and incentivize node participation. Nodes within the network are obligated to stake KAN tokens to participate in the ecosystem. The higher the staked amount, the greater the probability of a node being chosen as a leader and reaping rewards.

Cost of entry, cost of existence and cost of exit are defenses against Sybil attacks. Because the cost of existence is fairly low, the other costs have to be high to ensure security in the system. In other words, Security in PoS systems comes from putting up economic value-of-loss.

The minimum stake required to participate in the network as a node and the lock duration of the said stake is the cost of entry. A high cost of entry ensures economic value-of-loss which drives honest behavior. In other words, the possibility of economic loss due to Byzantine behavior has to be more than the incentives from the network. But having a very high stake could mean eventual centralisation of the network by the nodes, which is detrimental to the network. Stakes can be broadly classified as pure node stakes and stake pools. Pure node stakes is the stake that nodes put in along with provisioning hardware for the network. Stake pools are for entities that do not have the hardware or do not want the overhead of managing hardware and tech or become nodes of the network yet enjoy the yield results. Obviously pure node stakes give better returns as compared to stake pools or liquid stakes because they get yield and also full participation rewards.

## Pure Node Stakes

The minimum stake required for a node to participate in the network has the following constraints

1. $E(S) > Y + R$ , indicating that the stake S should be sufficiently high that the economic value-of-loss E for Byzantine activity is greater than the yield Y and network rewards R.
2. $S < P$, indicating that the stake S should be sufficiently low to not be too prohibitive to join, i.e., it should be less than a prohibitive limit P.
3. $(N/T)*100 \leq C$, indicating that the cumulative pure node staking N should not be a high percentage of total tokens T in order to avoid leading to effective centralisation in the node network alone, i.e., it should be less than or equal to a threshold percentage C.

Where

1. S as the stake required for a node to participate in the network.
2. E(S) as the economic value-of-loss for Byzantine activity at stake level S.
3. Y as the yield.
4. R as the network rewards
5. P as the prohibitive limit to join.
6. T as total tokens in the network.
7. N as cumulative pure node staking.
8. C as the threshold percentage above which effective centralisation in the node network alone would occur.

We assume that a healthy node staking percentage should be around 33%. Given our previous assumption of a total hodl of 60% we assume that another 27% of the community would be participating in long term staking. Assumptions are made on empirical data and more details here.

Tokens staked in the Kandola Network are locked for a minimum duration of 12 months. This lock-in period is designed to stimulate enduring commitment from nodes and ensure network stability and security.
Reason for this is better security and also lesser churn. High churn networks can delay the throughput and increase the overheads in communications in the network.

Minimum Stake

The minimal stake requirement for nodes to partake in the Kandola network should strike a balance between ensuring commitment and inclusivity. Taking into account the needs of the network and the average token price, we recommend a minimum stake amount of X KAN tokens.

The formula for the minimum stake amount is as follows:

$Stake_{MIN} = BA + (Stake_{TOTAL}/N) * SF$

Where

$Stake_{MIN}$ = Minimum stake required for nodes to participate
BA = Base amount
$Stake_{TOTAL} = {}^{n}\sum_{i=0} Stake_i$
N = total number of nodes in the network
SF = Stake Factor

The base amount guarantees a certain level of commitment. The dynamic part of the equation adjusts with the network's growth, maintaining equilibrium between security and inclusivity. The stake factor is a dynamic variable governed by the Kandola Network's governance system. It could be influenced by several parameters such as network size, transaction volume, average stake amount, and overall token supply (1 billion KAN tokens in the case of Kandola Network). Stake factor adjustments aim to respond to network growth and ensure optimal security.

Slashing Algorithm

The Kandola network implements a slashing algorithm to deter malicious activities and maintain network integrity. If a node is found to be engaging in malicious practices or non-compliant behavior, a portion of its stake is slashed.

The formula for the slashing penalty is:

$SP = SF * Stake_n$

Where
SP = Slashing Penalty in KAN
SF = Slashing factor - this is a network constant
$Stake_n$ = Stake of the node in the network.

The slash factor is another dynamic variable governed by the Kandola Network's governance system. It could be adjusted based on network conditions, the frequency of malicious activities, and the overall health of the network. The goal is to set a slash factor that strongly disincentivizes malicious behavior without overly punishing honest mistakes.

The slashed amount is bifurcated: a part is burned to reduce the overall token supply, increasing the token's value, while the remainder is awarded to the nodes that reported the errant behavior, fostering vigilance within the network.

For instance, if a node with a stake of 1000 KAN is found engaging in malicious activities and the slash factor is 0.05, a total of 50 KAN will be slashed from the stake. If the burn ratio is 50%, then 25 KAN will be burned, and 25 KAN will be distributed among the reporting nodes.

Staking Rewards

The Kandola Network is designed to incentivize long-term staking and participation in the ecosystem. Long-term stakers and participating nodes should expect a fair return on investment and opportunities to partake in network governance.

The principle driving this approach is the anticipation of reaching a threshold token velocity (TTV). At TTV, the network is expected to have enough transaction volume to create constant demand (buy pressure) for the token, thereby driving its price up. Long-term stakers are rewarded by the increased value of the token.

Until TTV is reached, the network incentivizes participation by periodically introducing rewards into the ecosystem. The reward allocation system operates on a decaying curve function, with rewards being distributed every 15 days.

Decaying Curve Rewards Function

The decaying curve reward function is designed to distribute fewer rewards as time progresses. It is aimed at rewarding early adopters who take higher risks by staking and participating in the network early on. The function works in such a way that the rewards distributed during each 15-day epoch are less than the rewards distributed during the previous epoch.

The decaying function can be modeled using a negative exponential function:

$R = R_0 * e^\wedge(-\lambda t)$

where:

R is the reward at time t,
$R_0$ is the initial reward,
e is the base of natural logarithm (~2.71828),
λ is the decay constant, and
t is the time (measured in epochs of 15 days).
The decay constant λ determines how rapidly the rewards decrease over time. A higher λ means a faster decay.

The rewards are distributed proportionately among stakers based on the average stake held during the epoch, the number of leader participations, and the storage responsibilities they have fulfilled. Any negative actions (like double signing, being offline, etc.) decrease a node's share of the rewards.

Inflation and Early Adoption Incentive

The reward mechanism serves two primary functions:

It introduces controlled inflation into the ecosystem. This inflation is beneficial as it incentivizes staking and participation, which are crucial for network security and consensus.
It rewards early adopters for their higher risk. Early adopters stake and participate in the network when the token price is relatively low and the network is not fully mature. By giving them more rewards, we compensate for their higher risk.
Threshold Token Velocity and Decaying Curve Intersection

The intersection of the threshold token velocity (TTV) and the decaying reward function is an important point in the network's evolution. After this point, the network has enough transaction volume to maintain or increase the token price without the need for large rewards. The token price growth provides return on investment for stakers, fulfilling the function previously served by the rewards.

The exact timing of this intersection will depend on the network's growth and transaction volume, as well as external market factors. However, it is expected to occur when the network has matured and has a substantial number of users.

At this intersection, we reach a state of equilibrium where the returns from token price growth balance out the diminishing staking rewards, thereby sustaining the incentive for stakers and participants. This equilibrium ensures the stability of the network and the fairness of the reward distribution system.

## Staking Pools

Staking pools are an integral part of the Kandola Network, as they provide a mechanism for stakeholders to join forces and enhance their opportunities to validate transactions, secure the network, and earn rewards. This approach allows participants with smaller amounts of tokens to still have an active role and earn returns, thereby democratizing the network and supporting more robust decentralization.

Pool Formation and Participation:

Any participant in the network can create a staking pool, and others can join by staking their tokens. The pool leader is responsible for participating in consensus, and in return, they earn a fraction of the rewards earned by the pool. This system allows those who might not have enough tokens to stake individually a chance to earn returns.

Pool Rewards and Distribution:

The rewards earned by a staking pool are divided amongst the pool members based on their proportion of the total stake. This includes the pool leader, who typically earns a slightly higher portion of the rewards to compensate for their increased responsibilities.

Pool Governance and Security:

The governance of the pool, including reward distribution and decision-making, is handled through pool governance smart contracts, ensuring decentralized and fair management. Furthermore, the network's penalty mechanisms apply to staking pools as well. Any negative actions by the pool leader or members, such as attempting to validate fraudulent transactions, can result in a loss of staked tokens, thus encouraging honest and beneficial participation.

Role in Network Economics:

Staking pools play a crucial role in network economics. They help ensure a broad distribution of tokens and active participants, encouraging more robust decentralization. The availability of staking pools also helps maintain network security by ensuring a large number of validators. Moreover, pools can help stabilize the token velocity as tokens staked in pools will tend to stay put, thereby reducing token circulation and maintaining healthy buy pressure.

Staking Pools and Threshold Token Velocity:

As the Kandola Network approaches the threshold token velocity, staking pools will continue to play a significant role. Even as the staking rewards decay over time, the collective power of staking pools allows for continued participation and network validation. Thus, they support the long-term sustainability and security of the network, even as the tokenomics dynamics evolve.

## Token Price and Stability Analysis

Chris Burniske writes - *"The first thing to note with crypto valuations is these aren't companies; they don't have cash flows. Hence, using a discounted cash flow (DCF) analysis is not suitable. Instead, valuing crypto assets requires setting up models structurally similar to what a DCF would look like, with a projection for each year, but instead of revenues, margins and profits, the equation of exchange is used to derive each year's current utility value (CUV). Then, since markets price assets based on future expectations, one must discount a future utility value back to the present to derive a rational market price for any given year."*

This makes a lot of sense because, at the fundamental level it ties the equation of exchange MV = PQ and applies it to crypto assets

Where
M = size of the asset base
V = velocity of the asset
P = price of the digital resource being provisioned
Q = quantity of the digital resource being provisioned
The thing to note is P does ***not*** represent the price of the crypto asset, but instead the price of the resource being provisioned by the cryptonetwork. We should be looking to solve for M which should give us the economic strength of the network required to drive this network.

Simulations on various parameters lead us to the following outcomes. Details can be found here.