Savix

Easy Defi For Everyone

Whitepaper

https://github.com/savix_org

https://savix-org.medium.com

https://t.me/savixhq

https://twitter.com/savix_org

https://savix.org/

Feb 28, 2021

Version #2.0

Contents

1. ABSTRACT

Applications in the area of decentralized finance are one of the most important fields of use for digital currencies today. With the help of convenient platforms such as Uniswap users can easily use innovative financial products based on **D**istributed **L**edger **T**echnologies (DLT). The digital currencies that are used for this purpose must be transferred to the respective ledger in which the desired financial instrument is mapped. Tokens used this way are temporarily “locked” and therefore cannot simultaneously participate in a staking protocol. The user has to make a decision, so to speak, whether he invests his money conservatively and safely by staking or more risk-consciously with greater possible profit using a DeFi product. With Savix a virtual currency is now available for the first time, which makes it possible to profit from staking rewards while keeping the token unlocked and liquid, freely available for use in any other DeFi product at the same time.

Protocol **E**mbedded **S**taking (PES)* used here is deeply anchored in the base code of the currency and does not require any explicit control or triggering by the user (see 3). All wallets holding Savix tokens automatically and permanently take part in the staking process. Figuratively speaking, all wallets holdings Savix tokens regularly receive a kind of interest being reminiscent of the old-fashioned savings book offered in the last century, a savings book though, which concurrently may be used as a collateral for dynamic online brokerage. In Allusion to this, the currency token’s name “Savix came” into existence.

In future the already twofold rewarding possibilities of Savix will be extended to triple. By then Savix Trinary (see paragraph 5) will add another rewarding layer letting Savix holders participate in transfer fees generated by the Uniswap liquidity pool(s). All options of generating passive income using Savix will be integrated into the Savix DeFi dashboard (see paragraph 4). The DeFi dashboard will empower Savix holders to invest their Savix tokens into a multitude of DeFi products with a single click.

Savix actually is the best collateral coin for decentralized finance, offering complete investing flexibility and multiple conservative income streams at the same time.

Due to the fact that the Savix rewarding system does not required any locking of tokens, the term “interest” would be better fitting here. Since “staking” is a more popular term with the cryptocurrency community this term will be used throughout this paper.

2. VALUE PROPOSITION

Savix is the best collateral for decentralized finance because, it is

1. Multi-Beneficiary

Since Protocol Embedded Staking (PES) allows complete availability of token usage in other DeFi products, rewards can be “doubled” using Savix. In this sense staking rewards are an extra Layer of passive income. The upcoming liquidity incentive program "Trinary" (see paragraph 5) will offer Savix holders another distinctive additional income stream.

2. Convenient

Savix staking is embedded within standard ERC20 transfer functions and is fully automated and 100% passive with no need for user decisions or interactions to receive rewards therefore.

3. Flexible

Savix is compatible with any Ethereum based DeFi project. Savix coins can be used like all standard ERC20 tokens for pooling, lending, yield farming, mining and so on, all this while continuously yielding additional staking tokens to holders.

4. Fair

Savix protocol embedded staking evenly adjust all balances according to the embedded supply development curve (see 3). No preference whatsoever is given to any specific holder. All wallets are treated in the same way, independent of balances, transaction volume or other parameters.

5. Transparent

Savix sources are open. All program codes and contracts are made available through Github and can be inspected and tested by anybody. Due to single contract deployment any manipulation of the contract logic or maximum supply is impossible, no minting of additional coins. Staking rewards are fully transparent and predictable.

6. Stable

With Savix there aren’t any reward releasing events at the end of locking periods which could generate cyclical dumps. Except for market reasons selling Savix is never easier or more profitable at any specific point in time, creating a less volatility.

7. **Independent**

   With Savix you stay independent because the tokens always stay liquid while earning rewards (no locking) and can be freely moved or invested into DeFi products.

These features are made possible by the unique Protocol Embedded Staking (PES), which implies the possibility of investing in highly profitable DeFi products with a predictable backing by conservative staking at the same time.

Savix features in short:

- protocol embedded staking
- compatible with any DeFi project
- fully automated - 100% passive
- unique project (no copy/paste)
- real time rewards
- extra layer of passive income
- fully transparent working mechanism
- stable against manipulation
- working product ready for deployment

Using Savix investors can receive threefold rewards:

- ETH and Savix from the Uniswap fee share
- Savix from the ERC20 embedded staking protocol
- ETH from the Savix “Trinary” Pool

This triple income stream is possible through Savix unique staking protocol only!

3. PROTOCOL EMBEDDED STAKING (PES)

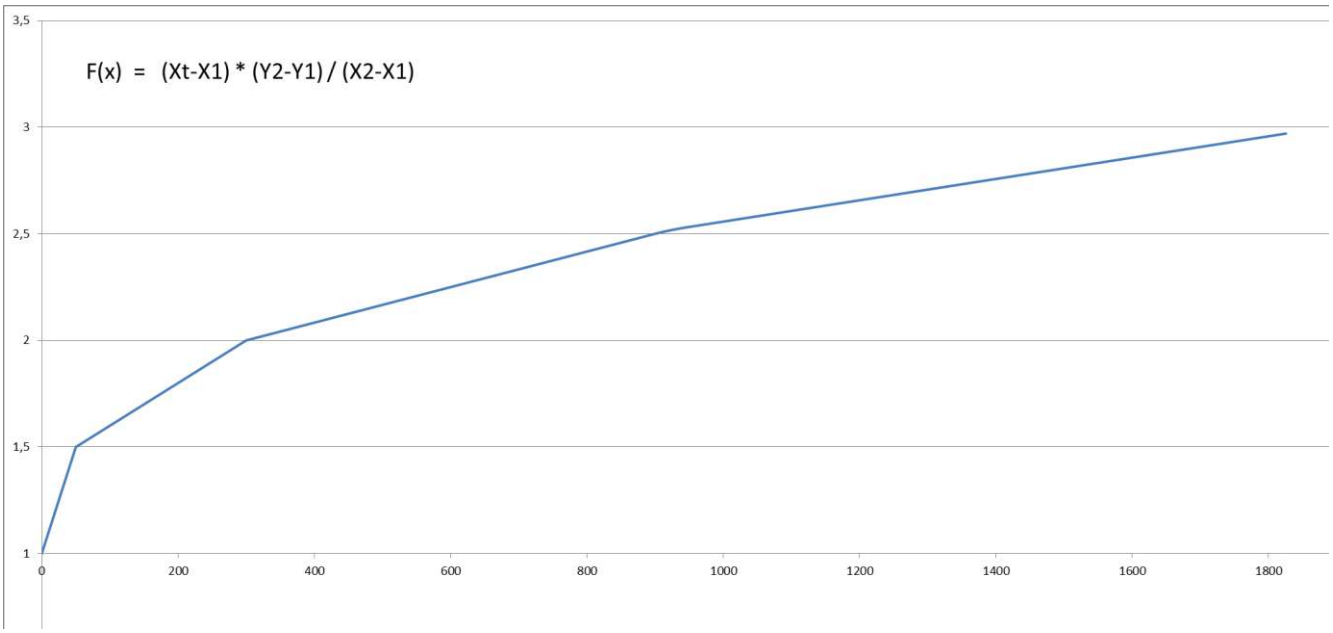
In order to realize the staking features mentioned above the staking mechanic has been embedded into the ERC20 protocol. The algorithm works by regularly inflating the total token supply according to a mathematical logic implemented into the smart contract. Account balances are defined by their individual share of the total supply thus guaranteeing a non-dilutive allocation of tokens. This way the relative staking profit is and remains equal for all accounts independent of size and user related parameters like staking duration, choice of staking pool etc.

The mathematical logic forming the basis of Savix’s protocol embedded staking follows the following characteristics:

- Transparent supply calculation predictable for investors
- Stability of the calculation towards user behavior and network effects
- Effectiveness of the calculation regarding computing power and transaction costs

The Savix supply development curve is the best combination of these characteristics. Supply development is gradually defined by a sequence of straights (gradient) which determines the interest rate at a specific point in time. Start and end points (corner points) of these straights define the global shape of the supply curve.

The following diagram shows the Savix supply curve normalized to a total supply of 1.



1: Savix supply curve by time in days

Details on the mathematics of the supply curve can be found in the appendix.

We used a specifically developed simulation program in order to determine appropriate corner points for Savix. Shortly before the presale this tool will be offered for download at Github to interested investors (see savix.org for further information)

The total supply is recalculated with each token transfer executed by the smart contract. This way the staking mechanism gets by without any external triggers.

The chronologically fixed supply curve enables exact prediction of staking rewards for any point in time.

4. VISION FOR THE FUTURE

Savix main objective is to let users profit as much as possible from the new investment opportunities created by decentralized finance with as less barriers as feasible. This is the Savix mission.

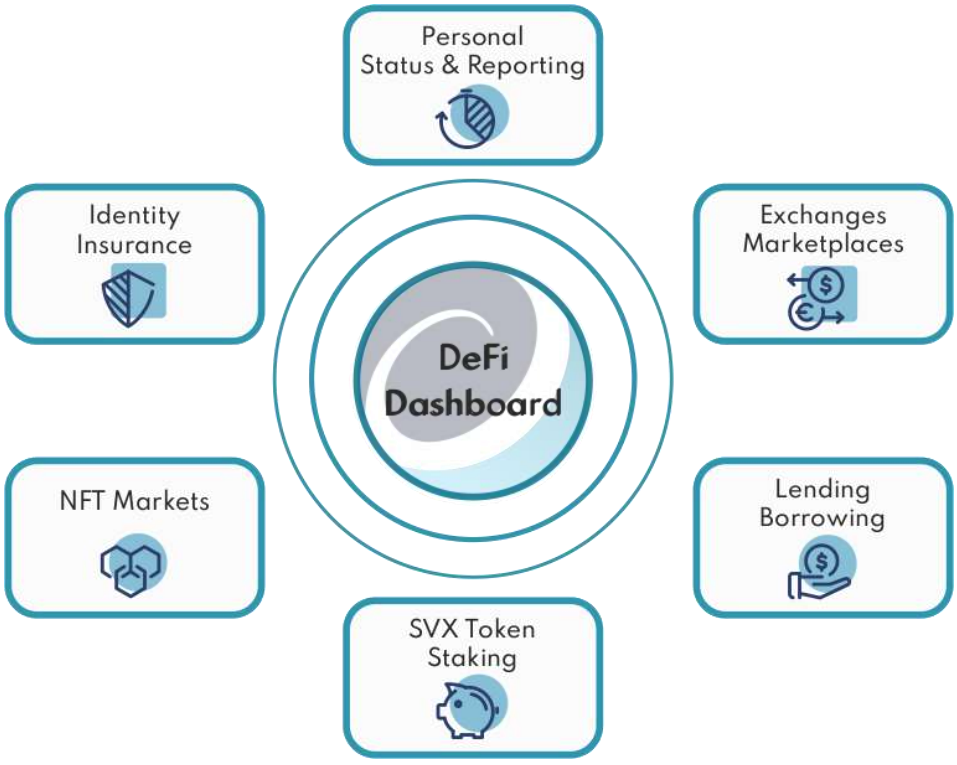
Therefore we let users combine multiple income streams while keeping full flexibility of token usage. The staking mechanism built into the Savix protocol works without any need for user actions. Users don't have to lock their tokens and don't have to claim their rewards since the staking process works completely automated.

Easy-to-Use DeFi For Everyone"

The Savix Mission Statement

In order to achieve the goal of Savix' mission statement the following elements will be built into an integrated application, the Savix DeFi dashboard:

- Personal finance overview showing balances, rewards and their historical development
- Integration of decentralized exchange for converting to / from Fiat currencies and other blockchains using existing bridging technologies
- Steering modules for different DeFi products like lending, money markets / liquidity providing, decentralized synthetics or non-fungible token markets



All of these applications will be combined into the Savix dasboard which will serve as entry point into the world of decentralized finance for Savix holders.

Most important for the design of the Savix DeFi dashboard is the maxime to reduce complexity as much as possible by concentrating all functionality to the minimum required to guarantee smooth and stable processing of all functions. This is achieved by designing predefined settings and process templates for all actions and interactions investors are able to launch and trigger through the dashboard.

In contrast to other user interfaces the Savix dashboard will stick to main features of applications integrated. Users will be offered single choices in an easy-to-use interface, handling the complex details of DeFi products in the background powered by predefined processes and workflows designed for typical usage szenarios.

Savix Trinary is the first DeFi App that will be integrated in the Savix dashboard.

5. SAVIX TRINARY – THE SAVIX DEFI APP

Combined profits from three distinct income streams.

Savix “Trinary” will be the first dapp to demonstrate the power of ERC-20 embedded staking. Users receive ETH for providing liquidity on automated market making platforms (AMMs) like Uniswap. The more liquidity you provide, and for longer, the greater share of the ETH pool you receive.

- **Uniswap Rewards**

  There is a 0.3% fee for swapping tokens on Uniswap. This fee is split by liquidity providers proportional to their contribution to liquidity reserves. It is functioning as a payout to all liquidity providers proportional to their share of the pool.

- **Staking rewards**

  In addition, you will earn Savix staking rewards from the embedded staking protocol while your tokens get used as liquidity.

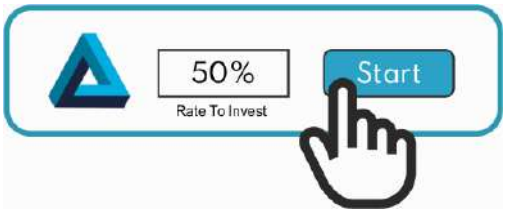
- **Trinary rewards**

  Whenever liquidity is deposited into a uniswap pool, special tokens known as liquidity tokens are minted to the provider's address, in proportion to how much liquidity they contributed to the pool. These tokens are a representation of a liquidity provider's contribution to a pool. With Savix "Trinary" it is possible to deposit Savix liquidity tokens for up to 6 month to receive yet another reward layer:

  ETH from our Ecosystem Fund. We expect a return of 7% to 15% of the initial investment directly paid in ETH over the first 6 month.

The complex interaction of minting the Uniswap specific liquidity providing tokens (LP tokens) and reinvesting them for optimizing gains will be automated by this application. Instead of having to actively observe the development of the liquidity pool, withdrawing LP tokens and deciding on whether to reinvest or withdraw them, Trinary will just offer investors to participate in the application taking care of all details by itself.

This way the complexity of this optimizing liquidity providing is reduced to a single decision, to participate or not, to click or not to click.



6. THE SAVIX TOKEN (SVX)

The Savix (SVX) token has got a double function within the Savix ecosystem:

SVX tokens create predictable regular interests generated by its protocol embedded staking mechanism. At the

core SVX tokens are the DeFi answer to traditional savings accounts since they embody the option to receive rewards which are regular and predictable as classical fiat savings once used to do.

Additionally the flexibility of letting all SVX tokens unlocked while participating in the staking system offers the potential to use SVX tokens as collateral for other DeFi investments at the same time.

This double nature of SVX tokens lead to the exceptional concept of multiple streams of passive income made possible. Or, referring to the integration and usage of the Trinary application, to the concept of

***One** Token Enabling **Three** Income Streams*

The Savix token is the backbone of the integrated DeFi dashboard application built on top of it.

6.1. What the Savix Token Isn't

Savix tokens do **NOT**

- grant access to any value other than the token itself
- grant access to service otherwise not accessible. This includes that Savix tokens do serve as a requirement to participate in the Savix ecosystem or to use any dapps created by the ecosystem

grant access to any kind of participation in decisions on related with the Savix token or ecosystem of any kind

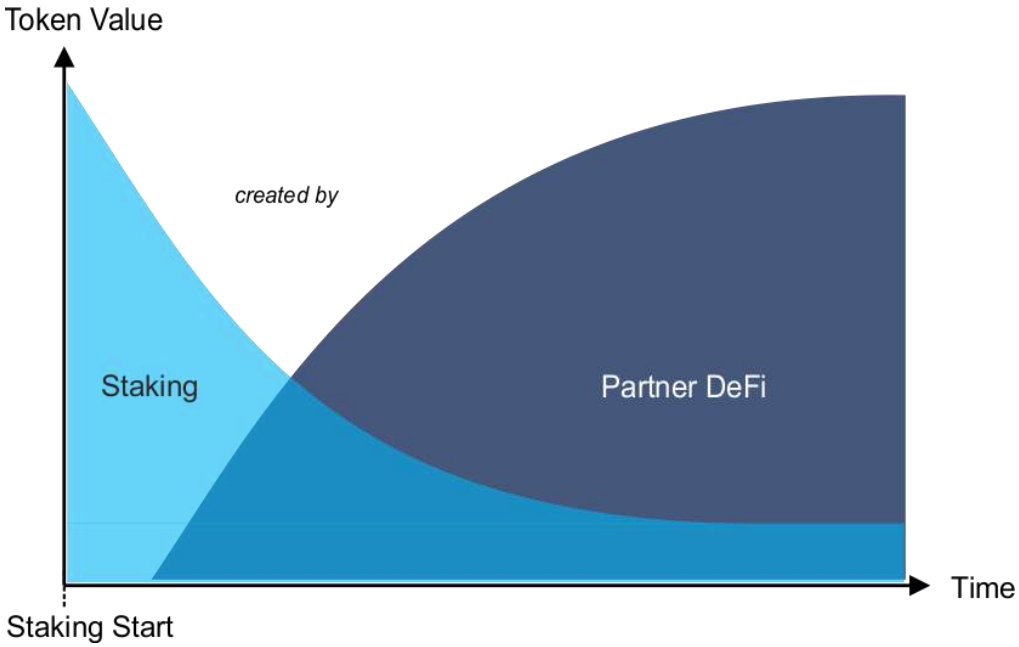
6.2. Where does the value of Savix tokens come from?

As described above there are two main use cases of the SVX token which are independent from each other and can therefore be combined. This independence allow the feature of combining multiple streams of income simultaneously.

Directly after the launch of SVX token trading there won't be many partnerships with DeFi project established yet, Therefore the range of DeFi projects Savix holders can chose from using SVX tokens as collateral will be small in the beginning. On the other hand the Savix staking mechanics are designed so that staking rewards will be highest during these early project stages.

While the height of staking rewards will slowly be reduces by time the amount of partnerships with DeFi projects for which SVX tokens can be used as collateral will increase as well as the functionalities of the Savix

DeFi dashboard will. The more time since Savix trading started will have passed the more of Savix tokens' value will come from its ability to enable easy usage of DeFi products having partnered with Savix.



7. TOKEN TECHNOLOGY

Since using DeFi applications (like participating in Uniswap liquidity pools) is a main usage scenario for the Savix token, an implementation as ERC-20 token is required. Non-fungible tokens may play an important role within future concepts and the ERC-721 or ERC-1155 standards may come into use for future developments as well.
(see https://yos.io/2019/04/14/erc-standards-you-should-know-about/ for a simple overview of Ethereum token standards)

However, **the Savix token itself will remain untouched by all future developments**. These will be designed as separate contracts interacting with the Savix original contract. The immutability of the Savix contract implementation is guaranteed and an important part of **Savix`s trust building concept**:

- Absolute transparency and liability of contract code due to single contract deployment (no unforeseen changes whatsoever can be made to the staking parameters).
- Absolute transparency and liability of staking rewards due to non-dilutive staking rewards defined by an immutable supply map (future interest rates can be exactly predicted)

- Absolute accuracy and transparency of total circulation supply. The circulating token supply is always identical to the total supply of tokens, there are no tokens held back in any way. The only exception are token remaining unsold during the presale (6 month locking time)

8. SMART CONTRACT AUDIT

The decision for a single contract deployment increases the importance of sufficient testing of the contract, since no updates of the code are possible. Scope and proceeding of smart contract testing follows standard audit procedures as described at https://github.com/knownsec/Ethereum-Smart-Contracts-Security-CheckList/blob/master/Ethereum%20Smart%20Contract%20Audit%20CheckList.pdf.

8.1. Internal Auditing

The audit for the Savix smart contract was carried out according to the following procedure:

1) **Basic Coding Bugs**: The smart contract has been checked by a static code analyzer for known coding bugs and these have been manually verified (reject or confirm). Confirmed bugs have been fixed and the static code analysis has been repeated until the check results have been satisfactory (see Audit Paper).

2) **Semantic Consistency Checks**: The logic implemented in the smart contract has been compared with the logic described in the white paper, which is limited to calculating the inflationary staking token supply over transaction timestamps. The correspondence of smart contract calculations und theoretical logic has been double checked by

    a) Running equivalent tests with identical test data a specific c# supply simulation tool and by Remix unit testing.

    b) Running simulation of real on-chain transactions with time lapse by Javacript-based test files using the truffle-ganache setup.

    Semantic consistency checks have to be conducted manually. For more details on this audit step see the audit paper which will be published through savix.org.

3) **Advanced Business / DeFi Scrutiny**: usually in this step of the audit business logics and system operations are inspected in more detail with respect to general business or specific DeFi-related aspects. Usually this is important for finding weaknesses within the business logic itself, for instance features that can be intendedly or mistakenly abused. After the start of Staking the Savix contract does not allow any actions which could alter supply or other contract parameters not even if initiated by the contract owner. Therefore this step of the audit process has been skipped.

4) **Transparency / Readability / Best Practice**: Finally the smart contract has been checked from the perspective of proven programming practices (best practice) with respect to style and commenting in order to ensure that the code and it`s logic can be comprehended by the community.

All details for the important process steps 1 and 2 of the smart contract audit can be found in the Audit Paper. All source and test files can be accessed through the Savix Github repository (https://github.com/SavixOrg). This way all audit results can be reproduced by any interested user.

Typical auditing steps have been professionally carried out and potential security issues detected have been dealt with. A focus of this audit has been put onto verifying the correct implementation of business logic and the consistent implementation of the automated supply staking mechanism respectively.

8.2. External Auditing

At the time of writing the external audit was being processed. Results will be included here as soon as they are available.

The audit is executed by:
QuillHash Technologies Pvt Ltd.
3rd Floor, Plot 448-A, EnKay Square, Opposite Cyber -Hub
Udyog Vihar Phase V, Gurugram
Haryana - 122016, India

https://audits.quillhash.com/smart-contract-audit

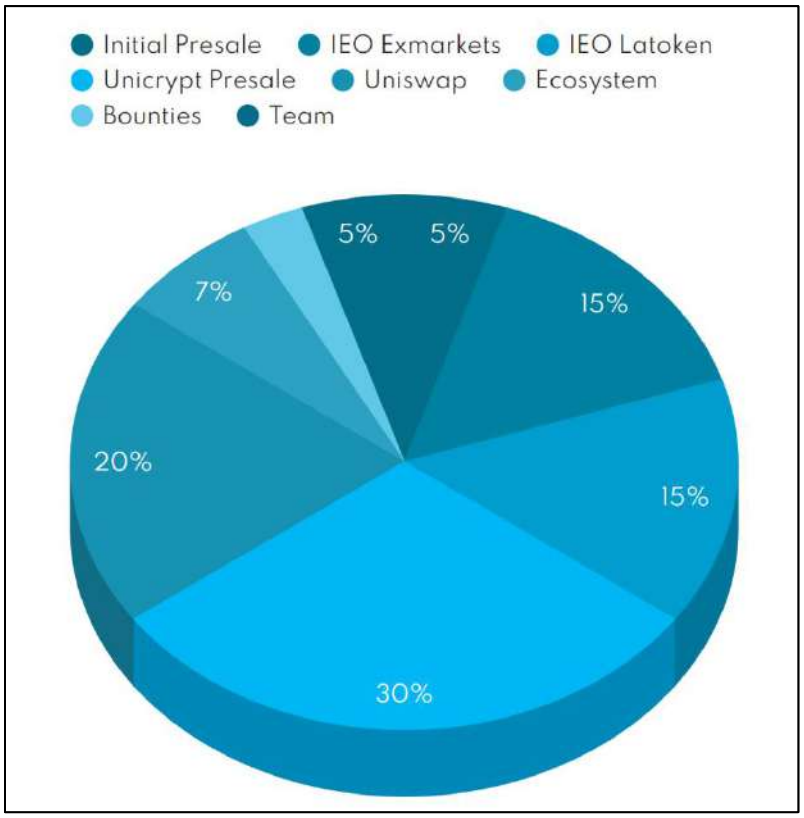
9. TOKENONMICS & TOKEN SALE

The introductory price of the SVX token at the market will be 50 SVX / ETH. A private sale will not be executed. Public presale will be spilt into three phases, each phase – and possible sub-phase - having different discounts and processors. Public trading will start at Uniswap first and then be extended to traditional exchanges for increasing the reach of the token.

9.1. Token Distribution & Budget Allocation

- 5% Initial Public Presale
- 15% IEO Exmarkets

- 15% IEO Latoken
- 30% Unicrypt Presale
- 20% Uniswap Liquidity (Public Sale)
- 7% Ecosystem Fund
- 3% Bounty Program
- 5% Team



Unsold tokens will be burned.

BUDGET ALLOCATION (ETH):

- 36% Uniswap Liquidity
- 7% Ecosystem Fund*
- 30% Further Development
- 17% Marketing
- 10% Reserve

9.2. Initial Public Presale

We do not want whales to dominate the Savix ecosystem!

You need the following pre-requisites to participate in the Savix presale:

- Ethereum Wallet used for ether contribution and to receive Savix.
- Metamask in order to use our presale dapp

In order to participate in the initial presale through the presale dapp a MetaMask Ethereum wallet is needed. Existing wallets can be easily and securely imported into Metamask.

On desktops the presale application can be used with any web browser capable of activating the MetaMask browser plugin (Chrome, Firefox, Edge etc). On mobile devices the MetaMask app can be used, which is a combination of Ethereum wallet and dapp browser.

Alternatively technically experienced users may use any Ethereum wallet to participate.
Detail information on the presale is available at the website: https:/savix.org/presale/dapp

GENERAL CHARACTERISTICS OF THE INITIAL PUBLIC PRESALE:

The presale goal is to raise a maximum of 83 Ethereum without a minimum. Savix is privately funded and already shows a working product with (internally) audited smart contract. Contributions of the initial presale will be used for next developments, external audit(s), marketing and partnerships. Unsold tokens will be moved on to the next presale phases.

- Presale Exchange Rate: 1 ETH = 60 SVX
- Adjustment on Feb 4th 2021: Additional bonus of 10% (10SVX/ETH) taken from ecosystem fund
- Token contract address: 0x8a6e8e9f7d61e97bde7e66336dbeea4fcbb388ae
- Presale will complete on Feb 8th 2021 or when the maximum amount of ETH is raised.
- Minimum Contribution: 0.1 ETH
- Maximum Contribution: 15 ETH

9.3. IEO Launchpads

The 2nd phase of the presale will be executed as Initial Exchange Offering (IEO) at an established cryptocurrency exchange. The main advantages are:

- Extend group of investors by the exchanges users base
- Trust: Ensure liability and value to new investors due to exchange's research on and verification of the Savix project and token
- Marketing cooperation: Mutual support and consultancy with the exchange's marketing team helps increase efiffiency and reach of marketing. Additionally all marketing channels of the exchange are used to promote the IEO.
- Security: The exchange is managing smart contracts of IEO. There's no direct link between the project's team and the IEO smart contracts. Thus, they are protected from theft, corruption, and any attack to investments.
- Direct exchange listing after presale phases are completed.*

**Exchange listing will be started after the Uniswap liquidity pool has been created only.*

The IEO will be executed by two exchanges: Exmarkets (https://exmarkets.com/launchpad) and LaToken (https://latoken.com/ieo/).

After the IEOs have finished, Savix will be listed for trading at both exchanges. Listing price will 50 SVX/ETH or 40 USD/SVX in case prices need to be defined in Fiat currency. Up to 3 sales rounds will offer SXV tokens with up to 20% discount.

9.4. Unicrypt Presale

The 3nd phase of the presale will be executed at the recently created decentralized Launchpad unicrypt.network.

The Unicrypt presale platform is completely decentralized and open to anyone, much like Uniswap. Rigid sales rules of the platform ensure best security for investors from a technical and financial perspective:

- Contract auditors are verified by the platform
- Sales parameters like caps and minimum liquidity requirements are managed by the platform reducing the risk of instable or failing projects

This final presale will be conducted in two rounds:

Round 1 - 2 hours:

In round 1 only UNCX or UNCL holders may participate. The current rate for admission is 4 UNCX or 50 UNCL. Round 1 lasts 2 hours.

Round 2:

If the hardcap has not yet been reached the presale moves on to round 2. This round is open to everyone.

Discounts for this sale will be up to 20% in relation to the Uniswap starting price of 50 SVX/ETH.

9.5. Uniswap Liquidity Pool

Shortly after the presale we will list Savix on Uniswap for public trading.

- Initial Price: 1 ETH = 50 SVX
- Estimated Liquidity (Presale Goal Reached): 800.000 USD
  (Depends on Ethereum volatility)
- Estimated Market Cap: 2.000.000 USD

The Uniswap Pool will be locked for 6 month, long enough to build trust with the community. Locking the pool longer than necessary represents a certain risk to the liquidity in case the Savix contract needs to be updated or migrated in the future.

9.6. Ecosystem Fund

An Ecosystem Fund in SVX and ETH will be maintained. This Fund will be used to strengthen the Savix ecosystem (see next paragraph) in general and to incentivize long-term holding of Savix in particular.

The fund will start with 7% of the total Savix supply and 7% of the amount of Ethereum raised during the presale. All fees generated with the liquidity pool will be added to the Ecosystem Fund, the majority of these fees being used as guarantee for Trinary rewards later (see paragraph 5).

10. THE SAVIX ECOSYSTEM

The Savix ecosystem consists of funds and applications which support the acceptance of cryptocurrencies and DeFi among non tech-savvy users in general and the spread and circulation of the Savix token in particular.

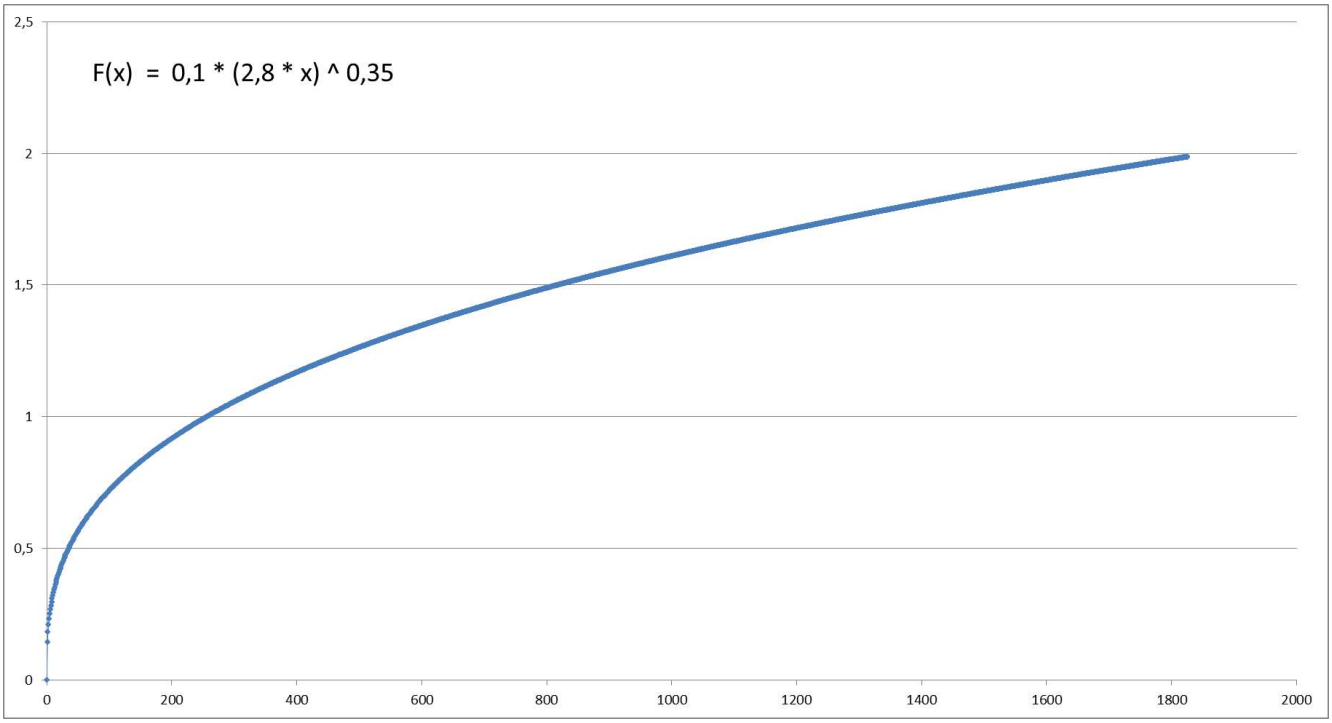
It is neither restricted to the Savix token technology or token holders nor dependent on these. Since the Savix is very easy to use there is a big opportunity of synergies between both (Savix token and Savix ecosystem) nevertheless.

Within the Savix ecosystem several decentralized applications (dapps) will be developed focusing on GUI functionality reduced to the necessary and aiming to integrate functions of currency / token exchange, trade, usage of DeFi products and wallet functions (Savix DeFi dashboard). As a first step of development a decentralized application called Trinary planned, which will collect Uniswap fees created by liquidity pools (LP tokens) and distribute those to participating investors(see paragraph 5).

APPENDIX 1: MATHEMATICAL DETAILS OF THE SAVIX SUPPLY CURVE

Theoretically logarithmic function would serve well as a basis for staking mathematics. Unfortunately complex calculations necessary to realize this within a smart contract are very expensive, leading to high gas costs. Since the supply recalculation is triggered by the execution of contract transactions it is very important to keep gas costs and computing requirements respectively low. Moreover, due to the fact that solidity works with unsigned integer calculations only, complex mathematical functions have to be approximated by recursive sequences or other mathematical approximation methods (Newton etc.)

A mathematical curve perfectly fitting the characteristics needed for the Savix supply curve would be using a cube root function.



2: Example of cube root function applied to supply normalized to 1

The corresponding recursive definition of cube roots are defined by: $X_n = (2x_{n-1}^3 + c) / (3x_{n-1}^2)$. For details on this kind of calculation see for instance http://elib.mi.sanu.ac.rs/files/journals/tm/44/tmn44p35-50.pdf, page 44 ff.

With X representing the time it is obvious that this calculation approach needs equidistant processing steps and would therefore require that the contract transactions should be executed in (more or less) equidistant

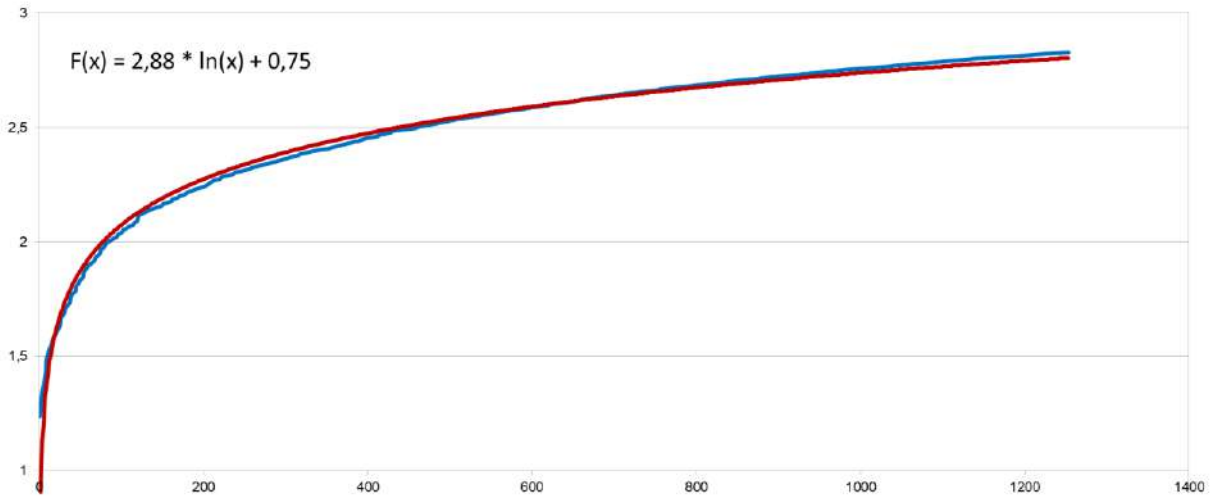
periods of time. Simulations showed that the differences resulting from different transaction timings are too great to allow this procedure to be used here.

A logarithmic staking curve could be another option to realize a staking curve fitting the requirements. Typical approximation methods like Taylor series are not suited for smart contract calculations due to the huge computational resources needed (see https://en.wikipedia.org/wiki/Taylor_series#Natural_logarithm).

We tried a calculated geometrical approach by defining an end point of the curve desired in the future. Taking the straight from the starting point (Initial supply at time of start of staking) to this target point as a basis, the supply at each point of time within the time interval can be approximated by the following computational steps:

- Calculate the additional supply of time x using the straight given: $F(x) = (X - X_0) * \{(Y_1 - Y_0) / (X_1 - X_0)\}$
- Get the mean value of this calculated supply and the initial supply: $F^1(x) = (F(x) - F^0) / 2$
- Define a new straight taking $F^1(x)$ as new arithmetical supply at the starting time and recalculate the additional supply as above: $F^2(x) = (X - X_0) * \{(Y_1 - F^1(x)) / (X_1 - X_0)\}$

For points in time after to target point defined supply values have to be approximated by a firmly defined straight with a given gradient dependent on the interest rate desired for the far future.
Indeed, this simulation leads to a rather good approximation of the logarithmic curve desired within the time span observed.

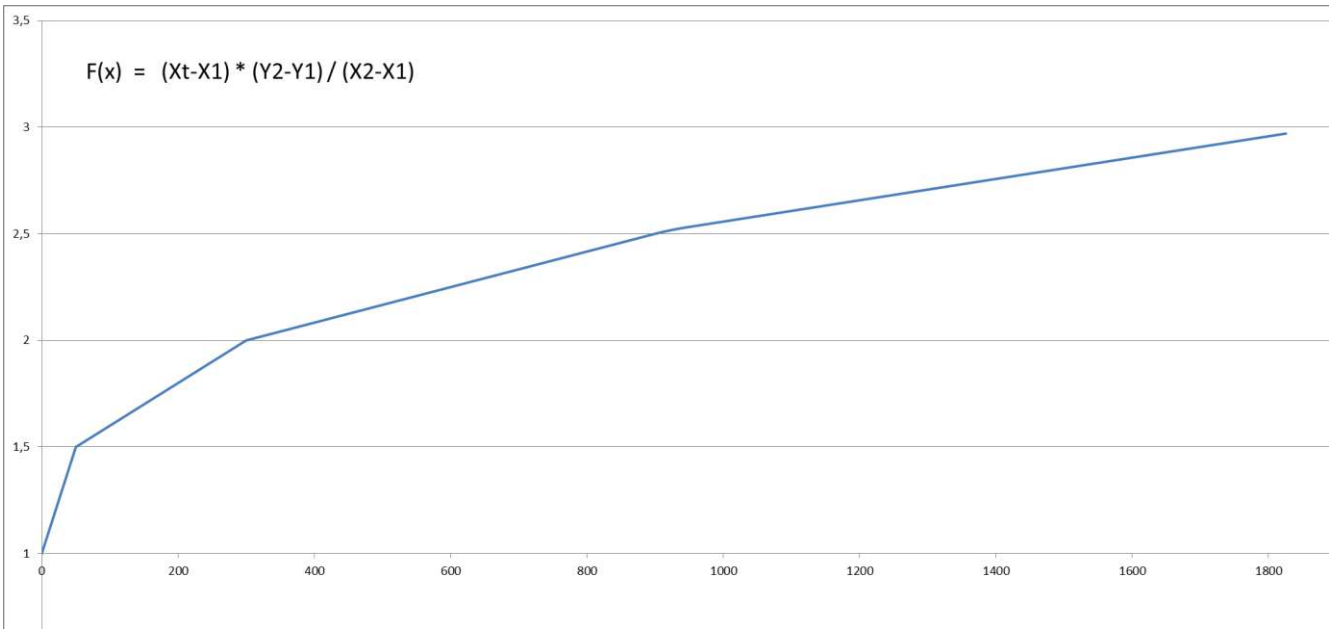


3: Geometrical approximation of logarithmic curve

Unfortunately simulations with varying smart contract transaction behavior steering the supply recalculations showed that the shape differences of the curve get too big to tolerate in certain usage scenarios, for example

when having very many transaction within certain time spans and almost none in other time spans. Therefore this approach cannot be used for our means either, since the resulting supply curve would be too volatile regarding transaction behavior.

The final solution for the Savix supply curve turned up to be a reduction of calculation complexity combined with a sequential definition of time spans. By defining more than one target point a so called supply map is created which determines important point in the supply curve desired (corner points). Using a sufficient amount of corner points any mathematical curve with always positive gradient can be approximated this way with sufficient accuracy. This is the case even if simple straight approximation is used between each set of corner points.



4: Savix supply curve normalized to initial supply of 1 (X_1/Y_1 and X_2/Y_2 being the lower and upper corner points of X_t)

The advantages of this calculation method are exactly what is needed for this purpose:

- No dependency on frequency of transactions
- Stable and “cheap” calculation (low gas fees)

APPENDIX 2: LIBRARIES AND INTERFACES

The Savix smart contract uses 3 libraries

- a short version of Zeppelin's ERC20-token interface (IERC20.sol)
- a copy of Zeppelin's SafeMath library (SafeMath.sol)
- a unique library to calculate adjustments of the total supply with some utility methods (SavixSupply.sol).

ABI DESCRIPTION

Savix.sol

Type	Method	Parameters	Output	Description
External view	supplyMap	-	uint256[2][]	Returns the corner points needed to define the mathematical curve used to calculate the supply inflation
External pure	initialSupply	-	uint256	Returns the initial supply
External view	finalGradient	-	uint	Returns thegradient of the supply curve used after the last corner point of the supply map has been passed
External view	lastAdjustTime	-	uint	Returns the timestamp of last change of supply
External view	lastTotalSupply	-	uint	Returns the total supply before the last change occurred (for comparison purposes)
external onlyOwner	startStaking	-	-	Starts the automatic supply inflation (
External pure	name	-	string	Returns the contract's name
External pure	symbol	-	string	Returns the token's symbol
External pure	decimals	-	unit8	Returns the token's decimals
External view	stakingActive	-	bool	Returns if supply inflation has been activated
External view	stakingSince	-	uint256	Returns timestamp when supply inflation has been activated
External pure	stakingFrequence	-	uint	Returns minimum timespan in seconds that has to pass between two recalculations of supply (to avoid too frequent recalculations and save gas costs)
External	totalSupply	-	uint256	Returns the current total supply

view				
External view	dailyInterest	-	uint	Returns the current theoretical daily interest rate
External view	yearlyInterest	-	uint	Returns the current theoretical yearly interest rate
public view	balanceOf	address	uint256	Returns the current balance of an address in SVX
internal	_calculateFragments	uint256	uint256	Calculates the fragment of total supply corresponding to a given amount of tokens
external	transfer	address, unit256	bool	Transfers an amount of tokens given as unit256 to the address given and returns true if successful
external	transferFrom	address, address, uint256	bool	Executes an externally triggered transfer from the first given address to the second given address of amount given as unit256 if allowed to do so. Returns true in case of success.
External view	allowance	address, address	unit256	Returns the amount of tokens which is allowed for externally triggered transfer from the frist to the second given address.
external	increaseAllowance	address, unit256	bool	Increase the amount of tokens allowed to transfer from message sender to the address given by the amount given as unit256.
external	decreaseAllowance	address, unit256	bool	Decrease the amount of tokens allowed to transfer from message sender to the address given by the amount given as unit256.
external	approve	address, unit256	bool	Set the amount of tokens allowed to transfer from message sender to the address given to the amount given as unit256.
internal	_approve	address, address, uint256	-	Internal function used to set the amount of tokens allowed to transfer from first address given to the address given to the amount given as unit256. Used by increaseAllowance, decreaseAllowance, approve.
external onlyOwner	distributeTokens	address[], uint256	-	Executes token transfers of amount given in unit256 to all addresses given in the address array.
external onlyOwner	distributeTokensFlexSum	address[], uint256[]	-	Executes token transfers of amounts given in unit256 array to all addresses given in the address array. The position in the array map the amounts to the addresses.
External view	getOwner	-	address	Returns the address of the contract owner

External	burn	uint256	bool	Burn the number of tokens given from the balance of the sender.
Public View	getBurnAmount	-	uint256	Returns the amount of tokens burnt in total (internal burns plus tokens sent to general burn address)

SavixSupply.sol

Type	Method	Parameters	Output	Description
internal pure	getSupplyWindow	uint256[2][], unint256	Struct SupplyWinBoundery	The supply map is defines points (x-value: time, y-value: supply) describing the inflation curve of the total supply. Between two such points the supply is defined by a straight through these two points. This methods returns the two points of the curve defining the straight relevant for a timestamp given. The map of all such points is given by the first parameter (2-dim array).
internal pure	getAdjustedSupply	uint256[2][] memory map, uint256 transactionTime, uint256 lastAdjustTime, uint256 currentSupply, uint constGradient	Struct AdjustedSupplyData	Using the input parameters the total supply is inflated according to the straight (see getSupplyWindow= relevant for the given transaction time.
internal pure	getDailyInterest	uint256 currentTime, uint256 lastAdjustTime, uint256 currentSupply, uint256 lastSupply	unit	Calculates the theoretical daily interest rate for a given timetamp
internal pure	getYearlyInterest	uint256 currentTime, uint256 lastAdjustTime, uint256 currentSupply, uint256 lastSupply	unit	Calculates the theoretical yearly interest rate for a given timetamp