# StableUnit v2: Anonymous low-volatility p2p E-Cash System

*Version 2.0.1*
*Alex Lebed (alex@stableunit.org),*
*Igor Gulamov (igor@stableunit.org)*
stableunit.org

**Abstract.** Bitcoin introduced a purely peer-to-peer version of electronic cash which allows online payments to be sent from one party to another without going through a financial institution. Despite all of its advantages, predefined finite monetary supply inevitably leads to price volatility which makes Bitcoin suboptimal as a medium of exchange, or unit of account. Public blockchain of Bitcoin also makes it completely traceable. We propose a solution which address both problems of the price stabilization and anonymity using a decentralized currency unit which evolves from being collateral-backed to being regulated by an autonomous monetary policy as the network receives wider adoption. The system defines a peg to any measurable unit of value with low-volatility purchase power, such as US dollar, purchase power of USD in 2010 or a basket of currencies. The peg maintains using a collateralized stabilization reserve and if necessary expands and contracts available money supply via on-chain borrowing, token sale and temporary parking of funds. Ownership of the network is distributed to tokenholders who form a decentralized autonomous organization DAO capable of changing some parameters of the system through a voting consensus mechanism. Both balances and transaction history are hidden using zkSnarks circuits and can be revealed only by user. This creates a new type of crypto-asset which combines the advantages of p2p nature of Bitcoin with the predictable purchase power of the US dollar combined with privacy necessary for real world adoption.

## Introduction

Bitcoin was a phenomenal innovation: for the first time, people could hold and transfer assets to anyone on the network quickly and privately. Furthermore, by its nature, this new asset could be stored and transmitted with open-source software, in a cryptographically secure manner while completely eliminating the requirement of relying on a trusted third party[1.1]. All these features plus improved durability, portability and divisibility created a new class of digital asset, a decentralized cryptocurrency[1.2].

Despite its advantages Bitcoin has failed to achieve mass adoption for a number of reasons.

### Mass adoption of cryptocurrency

Why has Bitcoin not achieved mass adoption in finance and commerce[2.1]?

**Slow and expensive to use**. With the introduction of Lightning Network[2.2, 2.3], this issue has been largely resolved. Many other cryptocurrencies appears to have also resolved this issue[2.4-2.6].

**Time for adoption.** Bitcoin has existed since 2009, and even if we start counting from 2013 (the first Bitcoin bubble and arguably the first introduction of Bitcoin into collective consciousness), it is longer than PayPal had to conquer e-commerce and online transaction market. Moreover, PayPal marketing campaigns raised less awareness than Bitcoin's news coverage[2.7-2.8].

**Transaction immutability.** Introduction of decentralized escrow services have solved this problem[2.14-2.15].

## Bitcoin price

The main functions of money include[3.1]:

- Medium of exchange.
- Unit of account.
- Store of value.

The value of Bitcoin often experienced large fluctuations, rising over 178% a month[3.2], or losing 35% a week[3.3]. We argue that these large price fluctuations make Bitcoin a suboptimal medium of exchange, unit of account or short-term store of value in comparison with fiat money therefore make it less preferable form of money for payments.

There are some companies including Microsoft[3.4], Overstock[3.5], Virgin Galactic[3.6] and others which accept Bitcoin and other cryptocurrencies directly or via payment services such as BitPay[3.7]. However, these companies **do not keep their money in cryptocurrencies** and immediately convert it back to fiat currency. This is primarily because these companies are not in the business of speculating on the price of cryptocurrencies. Companies have expenses to cover and do not want to carry any additional currency volatility risk which includes foreign exchange risk as well as Bitcoin price fluctuations. Some companies such as Steam platform, the largest digital distribution platform for PC gaming[3.8], stopped accepting Bitcoin due to volatility complications[3.9-3.10]. In other words, **Bitcoin doesn't serve as a good store of value** for these companies.

Despite these issues, consumer demand drives what businesses will to accept as payment. If consumers like including ordinary people would like to use Bitcoin or other cryptocurrencies as payment, these businesses would certainly adapt despite the additional risks and complexities. However, even with general audience, Bitcoin and other cryptocurrencies face a significant challenge in gaining widespread adoption. Even people who are aware of Bitcoin and are comfortable with the technology, most would prefer payments for everyday economic activity in fiat instead of Bitcoin because from one month to another it is unclear if some fixed amount of Bitcoin will be enough income to cover household bills. So **Bitcoin doesn't serve as a preferable unit of account** than US dollar.

Furthermore, people often see Bitcoin as a vehicle for speculation. If they held Bitcoin they would, for instance, defer a purchase of headphones worth $300 today and wait until Bitcoin rises in price relative to fiat, thereby reducing the cost of the headphones in Bitcoin. In this manner, Bitcoin's rising prices create an **incentive not to spend**. This incentive makes **Bitcoin less workable a medium of exchange** than fiat currencies with predictable purchase power.

We posit that **the price volatility is the biggest barrier to its widespread adoption of Bitcoin.** Similarly, other cryptocurrencies also suffer from the same barrier to adoption due to their volatility.

## Eventual stabilization of Bitcoin

Will Bitcoin eventually grow less volatile?

While volatility will decrease with greater adoption, it is unlikely that price fluctuations will ever be less than that which occurs in large-cap stocks such as Google[4.1] or in gold[4.2]. In case the price eventually did stabilize it would likely imply that it was performing poorly as an investestment, therefore creating a new sell-off cycle introducing renewed volatility.

Some crypto enthusiasts might have very valuable arguments as to why Bitcoin or another system with predefined supply might eventually stabilize itself. Because we can not provide all the possible counterarguments in this whitepaper let us settle on the fact that there is a non-zero probability that there will be a market demand for a completely decentralized low-volatility cryptocurrency.

Tether's 2B+ market cap[4.3] proves that market demand exists even without decentralization.

## Low-volatility cryptocurrency

What is a stable currency? It is a currency which successfully performs its functions as a means of exchange, a unit of account and a store of value because its purchasing power is stable[5.1]. Purchasing power is the value of a currency expressed in terms of the amount of goods or services that one unit of money can buy[5.2]. In other words, a currency is stable if its purchasing power for a wide variety of goods and services remains roughly the same.

This ability to buy might be direct or indirect. Direct via greater market adoption and indirect via ensured exchangeability to other assets in the same manner that gold has purchasing power despite there being very few services that will accept it as a direct payment method. This is very simplified classification of the whole spectrum of liquidity of the assets.

Based on the definitions mentioned above, a decentralized cryptocurrency is stable if it is accepted as a payment method for the same amount of goods or services or if it provides exchangeability for other assets at the market price in a decentralized way. At present cryptocurrency is arguably the only asset capable of providing exchangeability in a decentralized cryptographically secure way. Therefore for cryptocurrency to be stable **it is sufficient**, if at any given moment of time each user is able to exchange it for another cryptocurrency at the current market price.

## Example of usage

*"Stable Coins, The Holy Grail Of Cryptocurrency" - Forbes*[6.0]

There are many use-cases for a stable cryptocurrency: it offers advantages over both Bitcoin and a strong fiat currency like the US dollar:

- In the same manner as any cryptocurrency it can be freely sent to others, used as payment for goods and services
- A low-volatility crypto-asset for traders
- Gateway between fiat money and crypto
- Prevents creation of taxable events for holders/traders (in certain jurisdictions[6.1])
- Credit and debt markets
- Means of savings for people in nations with weak institutions or unstable local currencies[6.2] (Ukraine, Argentina, South Africa).
- Temporary currency for countries with extremely weak local currencies[6.3] (Ecuador, Panama, Venezuela, Zimbabwe)
- International trading for countries which might prefer cryptocurrency for political reasons[6.4] (Russia, Turkey, Jordan)

Since this fusion of fiat money and cryptocurrency has a unique set of properties that never existed before, there might be some usage which we cannot foresee. You can read more about the problems which stable cryptocurrency can solve (use cases) and examples of the products on www.stableunt.org.

## Stable Unit System

**Tokens**

**SU** - unit of currency, main token of the system. The value is equivalent of purchase power of US-dollar 2020 + 1% annual inflation. It can be created ONLY at system's reserve (onchain analog of "central bank").

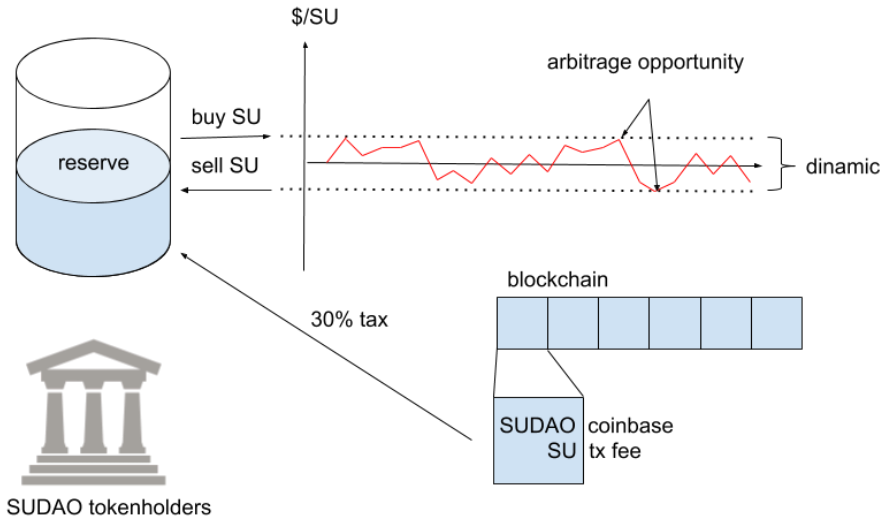
**SUDAO** - governance token with finite supply. It used for staking to become the block producer who verify transactions and provide data for oracles. Each new minted block contains freshly created SUDAO with the decreasing rate such the total supply is limited by the number 21,000,000. Half of this tokens go to the block producer as the infrastructure reward and another half goes to the System's pool.

Second usage of the token is voting to change parameters of the system, if necessary.

When the system has funds which exceed stabilization needs it buyout SUDAO and burnes it

## Overall schema

All components of the system can be schematically drawn as on the image below:




## Components

Components of the System:

The main component of the System is Stabilization Reserve, which stores Bitcoin in the smartcontact, as an analog on central banking gold reserve. This smart contract reserve has two functions:

- [**Create SU** from Bitcoin] which sends Bitcoin to the Reserve and creates equavalument value of Stable Units using current Bitcoin market price. I.e. if user sends $100 worth of BTC - he will get 100 SU.
- [**Redeem SU** for Bitcoin] which takes SU and burns them and sends the equivalent value of BTC from the reserve.

So reserve is acting as a liquidity pool which is always ready to sell and to buy SU by face $1 value.

## Lending Pool

Sometimes, when Bitcoin price is low, the system needs to recapitalize itself to provide sufficient liquidity for all users who want to redeem SU. To achieve that the system borrows money from the users via decentralized borrowing/lending pool similar to DeFi lending platforms, specifically Compound v2. When the reserve receives money, part of these funds goes to the Reward pool and distribute across lenders. Technically, each user who locks their SU receive new token pSU which proofs ownership of certain percentage of the whole lending pool. Also, it's always possible to borrow money from the pool with dynamically calculated interest rate. Each SU load have to overcollateralized by Bitcoin for at least 150% overcollateralization rate.

## Collateralisation

In other words the Stabilization reserve can be think as pool with three type of assets:

1. Liquid and immediately available Bitcoinss
2. Less liquid and available via borrowing StableUnit
3. Finite supply SUDAO pool which can be sold via auction

The system is re-balancing this assets to maximize immediately available liquid part. Overall, the sum on the funds in the reserve should always exceed the market cap of SU.

## Reserve income

There are 3 additional functional components of the System which provide complimentary service for the StableUnit as well as redirects all profit of it to reserve:

1) Crypto DEX - Decentralized exchange for cryptocurrencies. Natively supports only available pairs of SU/crypto assets, SUDAO/crypto asset and charge fee in SU/SUDAO correspondly. Technically it's just a smartcontact similar to Uniswap liquidity pool's design - which can be as well integrated in all meta-dex aggregators, such as 1inch.exchange or similar.

2) Lending pool, mentioned above. The difference between interest paid by people who borrow SU and the reward which paid to user who lended SU redirects to the reserve.

3) Tax from block-production. Block-producers (analog of miners in the Proof of Stake system's) receive reward in the form of transaction commission and coinbase (newly created SUDAO tokens for each block, similar of how miners get newly generated Bitcoin each block but decreasing amount during the time). 30% of this reward, both SU transaction fee and coin based goes to the reserve.

4) Sell/Buy SU methods of Reserve itself. These methods both create and destroy Stableunit in exchange of Bitcoin in majority of time charge minimal fee 0.01%. However, in certain circumstances when demand is too high the system adds additional fee all way up to 1%. You can think of it as Uber increase rates during rush hour. This both generates little upstream of funds to the reserve and prevents the system from draining too much liquidity assets during the marketing panic, giving to users an incentive to sell their SU a bit later so overall buy/sell dynamic doesn't have big spikes.

5) Forex DEX. Apart of SU, which is default token of the system and can be think as crypto US-dollar, the system is able to handle any amount of another fiat currencies, such as suEUR, suRUB, suCAD etc Because the whole system based of the assumption that in the long run cryptocurrencies in the reserve will outperform national fiat currencies as the store of value, dollar have similar or better value asymptotic to Bitcoin as Euro or Canadian dollar. So with close to no overhead the system can work with secondary time of StableUnti in exact same way it does work with main SU token which is syntenic USD in the blockchain. The value proposition of the Stableunit working with them all is ability to exchange it other to another with no overhead.

## Governance

The reserve pool belong to noone. There's only one way to get Bitcoin or other assets out of it - buy it out. However, the have to be mechanism, at least initially, to change ceratin parameters of the system, such as commissions size, add new currencies, adjust Safe Reserve Ratio asymptota etc. For all these operations all SUDAO token holders form the onchain organization called StableUnit decentralized autonomous organization (hance the name of the token). Everyone can proposed update of these parameters, however in order to get listed as voting proposal it has to get at least 10% of all available minted SUDAO tokens locked to get into voting list. This token get lock for the duration of the voting. As soon as the proposal gets into voting list, there's one week during which token holder can vote in favour or against the proposal. If, after one week amount of tokens staked in favour of proposal exceeded for at least 30% amount of tokens stake against - the proposal gets resolved, and it take another week until it gets into production. During this last week, everyone who can't agree with these changes - have time to move out their funds.

## Blockchain infrastructure & consensus

Consensus of the digital ledger state as well as currency market prices are achieved via Proof of Stack blockchain system. There are 100 block producers who have locked the largest amount of stake of SUDAO. Every turn they consensually agree on what's the the state of the ledger with all SU transactions and what's the market prices of the assets the system uses. To incentivise block producers to do so, they receive the reward in form of fees from all transaction which got included in the block (which is paid in SU) and newly minted SUDAO token which called coinbase. Amount of newly created SUDAO token get decrease during the time in such way that the total amount of ever create SUDAO token is limited by the number of 21.000.000. In the same way, as Bitcoin mining reward works. However, the block producers get only 70% of transaction fees and coinbase as 30% of it automatically goes to the reserve for the further stabilization of the system.

## SUDAO burn

Because the system continuously receives the funds from the adjusted services, mentioned above, there would be a point in time when the System doesn't use all liquid assets for stabilization therefore it's suboptimal to keep them locked in the smart contract. This is one of the main disadvantages of CDP based models. So when the system doesn't need this funds, they get distributed as buyout auction SUDAO tokens and burn them.

Because the system is probabilistic, this moment when the system securely doesn't need this liquid assets can be only probabilistically estimated. So instead of have one rigidly defined trigger (safe/no safe) to redistribute liquid assets - the system gradually increase the amount of funds it accumulates every block in the reward pool. Practically, when the system is small it pays close no additional reward, since all this funds potentially needed for stabilization and against potential market crash. However, with time this amount gradually grows to the level of distribution of all profit which goes to the system as the additional reward.

## Privacy

When the system receives real commerce and business adoption, it's very important to give to users at least the same level of privacy at they currently experience using conventional payment systems such as Visa or MasterCard. While these system are centralized, transaction history is generally not accessible by competitors. To achieve that in open ledge blockchain, StableUnit system uses privacy schema based on zero knowledge proofs of knowledge. Access to a wallet is build by standard

cryptographic signatures. So every user has pair public private key as their access to funds key and the address respectively. However there's another pair of keys which can get regenerated unlimited amount of times: visibility keys. One key is currently associated with wallet address is the hidden key, and other (private) is reveal key. To send funds user have to not only sign transaction by private key of sender, but also hide the transaction by public visibility key of recipenet (hidden key). This schema is very flexible, and provides not only secure anonymity of both the amount and the history of transaction but also allows to reveal the transaction history of demand. For the taxation or accounting purposes. Please see the end of the article for greater details.

## Problem formalisation

**Our assumption** that, as a payment system gains adoption, many users just keep their assets in the system and not all users want to exchange it back at the same time[7.2]. Therefore when the number of users exceeds some critical mass, the condition "always fully backed by any equal amount of collateral" is unnecessary. Similar to the present day banking system, where the majority of people keep their money: if too many clients decide to withdraw their funds - the bank will not have enough cash to cover their liabilities and this could lead to a collapse[7.3]. This users behavior is also evident in MakerDAO's Dai, Tether and other asset-backed systems (as the system grows, more and more users continue to stay in the system).

Another consequence from our assumption, that exact same principle of staying on the system creates a network effect value, similar to non-zero value of Bitcoin, and other cryptocurrencies without collateral. From this perspective, our proposal is an evolutionary system. It starts from using already popular cryptocurrencies as decentralized collateral to bootstrap network when it is small. And it uses decentralized on-chain monetary policy to regulate the price, when network receives greater adoption. This in some sense recapitulates evolution of the US dollar (which used already popular gold as collateral and moved away from it then dollar became popular by its own).

Does fluctuation of the price of assets held in reserves impacts the volatility of SU? How does the dynamic of the demand for SU on the market or global market trend affect the price of SU? To answer these questions we have to formalize the definitions and the problem we solve.

StableUnit is a system which uses multiple unpredictable inputs and complex configuration. We can define the price of SU as a random variable, which depends on time and multiple random-process inputs such as price of assets in the reserve or market demand for the SU.

We define the price as stochastic process

$$\mathbf{price(SU(init\_conf, inputs))[t]} \rightarrow \mathbb{R},$$

where:

**SU(init_conf, inputs)** - state of the StableUnit system with initial configuration and all external inputs such as price of assets in reserve and demand for SU.

**init_conf = {stabilisation_conf, oracle_conf, Peg(t)}** - initial configuration which specifies all parameters of the System,

**stabilisation_conf = {Δs, Δb, Δd, Δp, f_reserve_sell(), f_reserve_buy()}** - configuration of the multilayer stabilisation mechanism, see Multilayer Stabilization paragraph for detailed explanation of this parameters.

**oracle_conf = {oracle1, oracle2, Δt, w[t, C], max_delta_price}** - configuration of the oracle,

**oracle(crypto)[$t_i$]**$\rightarrow \mathbb{R}$ - function which determines current market price of the crypto asset,

**Δt, w[t, C], max_delta_price** - additional configuration parameters, see paragraph Oracle system to additional information,

**Peg(t)** - target for the sort peg,

**inputs = {reserve_assets, market_demand}** - dynamic inputs to the System,

**reserve_assets = $crypto_1$ … $crypto_n$, $crypto_i$** - asset held in reserve,

**market_demand(SU)[t]** - function which represents the accumulated market demand for SU at the moment of time t

Using this definition, decentralized cryptocurrency is stable **if and only if** the mathematical expectation (mean value) is equal to peg:

$$\mathbf{E(Price(SU)[t]) == Peg(t)}.$$

Our goal is to design such a system that SU price fluctuation i.e. variance (expectation of the squared deviation of a random variable from its mean) of the prices will be minimum:

$$\mathbf{Var\_max \rightarrow min}: \mathbf{Var(Price(SU(init\_conf, inputs))[t]) < Var\_max},$$

for $\forall$ **inputs** $\in$ **{acceptable_inputs}** and $\exists$ **init_conf.** Such *init_conf* is called the **optimal configuration** and *Var_max* **- expected price fluctuation** for the particular design of the System.

**Without loss of generality,** assume pegged value is equal to 1 USD. Also, for simplicity, let Stabilization Reserve store only ETH and measure all exchange prices directly in USD($), however real exchanges happen in SU/ETH and SU/BTC

Let us see the components of this system in detail.

## Oracle system

An Oracle system is on-chain smart contract which is able to determine the current market price of SU. SU is a decentralized cryptocurrency therefore is freely exchangeable on different markets so the market price is not just one numerical value but a set of values in 4 dimensional space of exchange history {price, volume, time, place}.

We define the price as "some average" value in the specific timeframe $\Delta t$: $\mathbf{price(SU)[t_i]} \rightarrow \mathbb{R}$, where $[t_i]$ defines the range $[t_0+\Delta t*i,\ t_0+\Delta t*(i+1)]$ and an average value can be the median, arithmetic mean, weighted average or other approximation function depending on the method of measurement.

Let us define two such price measurement methods (oracles) which can be implemented using smart-contracts.

### Median value from the set of trusted providers

The oracle takes the median price from the set of predefined trusted providers (forex exchanges).

This semi-decentralized approach is used by MakerDAO's DAI[8.1]. As of May 2018, they use the following set of trusted providers: [bitbay, bitfinex, bittrex, cex, coinbase, cryptocompare, etherscan, gdaxgemini, hitbtc, kraken, livecoin, poloniex, yobit][8.2].

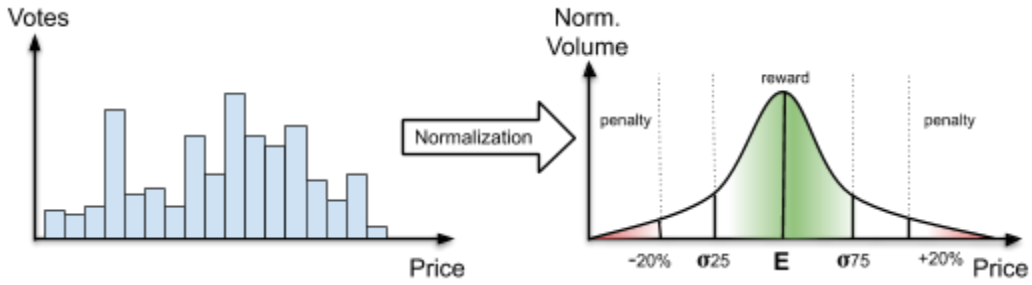
This approach is an easy way to securely bootstrap the protocol but with cost of sacrificing decentralization.

The problem with this approach is that the trusted providers might be compromised. If we have N providers, then to fake a median value the attacker should compromise at least round_up[N/2].

To prevent this, the System uses a parameter **max_delta_price** $\geq \max(\Delta price(SU))_{\Delta t} = \max\ (\ |price(SU[t_1]) - price(SU[t_2])|\ )$ for $\forall\ t_1,t_2: |t_1 - t_2| < \Delta t$, which restricts the movement of the price. If the trusted provider returns the price value beyond *max_delta_price* range, this provider is excluded from further participation. If more than 30% of providers are excluded – the **global settlement event** is triggered to investigate the reasons and mitigate the potential attack. This price movement restriction also ensures that there is enough time to trigger *the settlement event*.

### Decentralized betting mechanism

Every turn of Δt time any person is allowed to bet some SU on what was the average price in the last Δt time. At the end of the turn the System takes all bets (an array of pairs {**betted_price**, **betted_value**}: *betted_value* of SU in total was bet that the price was *betted_price* USD), after that normalizes the distribution of the values and calculates the weighted average and variation i.e. approximates with normal distribution N(mu, sigma^2), where mu is "weighted average" and sigma is "dispersion". To prevent potential vector of an attack of betting 1 SU on +/-inf, vote range is limited by current_price +/- max_delta_price. The weighted average is taken as the true price. People who voted between 25$^{th}$ and 75$^{th}$ percentiles receive reward proportional to the betted amount and the reward function **reward(betted_price)** = (1–|betted_price–actual_price|$^2$). People who get reward - receive it that from new issued SU. Total reward if relatively small to the SU in circulation and comparable to the miners' reward for the same period - so it will not be able to destabilize the System (read more in the incentives paragraph). People who made an error more than 2*max_delta_price of absolute value of the price (for example 20%, i.e. M=median(SU) is 0.97, threshold is 0.8*M..1.2*M) lose their bets:



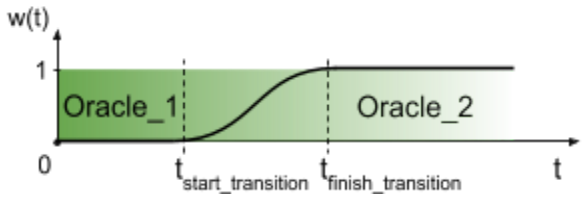

The threshold of the rewards and penalties for a reason are made independent: reward depends on the distribution of bets and penalties on absolute value. This made to motivate users to vote for correct price, the more accurately a speculators guess what was the average price – the more reward they will receive without a fear of being penalized if the distribution is very small. If the System uses only distribution – there would always be people who lose bets even if everyone tried to guess correctly.

It is easy to show that this decentralized approach of finding consensus about the price - provides the same level of security against a potential attacker as Proof of Stake (PoS). However this works only when there are enough actors in the System.

## Smooth transition

Each of the oracle mechanisms described above have their own pros and cons. Despite the median value from the list of trusted providers (Oracle_1) was proved to be reliable[8.1] for MakerDAO, it can be used only as a temporary solution because it is not truly decentralized and it is possible to take control over a majority of trusted provides. Decentralized betting mechanism (Oracle_2) does not have that issue but works only when there are enough users ready to participate. To provide an easy bootstrap solution while the system is small and without compromising the decentralisation in the long-term the System uses the **smooth transition** of these two mechanisms.

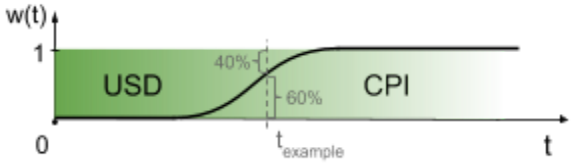
The System use **Oracle** = (1-w[t,C])*Oracle_1 + w[t,C]*Oracle_2, where **w[t,C]** is a weight function from the time and amount SU in circulation which grows from 0 to 1 as the network grows.




Despite its simplicity, the smooth transition is a very powerful mechanism which allows us to use the optimal algorithms and approaches for the **growing network** (due to very different properties of the small and large networks) and does this without compromising security or decentralization unlike one moment switch[8.3].

During the growth of the network the System also performs an implicit transition from being purely overcollateralized to being regulated by monetary policy.

Another example of the smooth transition is the movement of the soft peg from US dollar to CPI. If the System were adopted for international trading then US dollar might became suboptimal as unit of value with stable purchase power. Instead of calculating the price of USD2018 with appreciation of inflation, the direct measurement of CPI **might be** considered as an alternative solution for the peg. CPI might be calculated as CPU-U/W[8.4], an average price of popular commodities, indexes, market buckets, etc. however there are some challenges with this approach [8.4] and further investigation is needed.



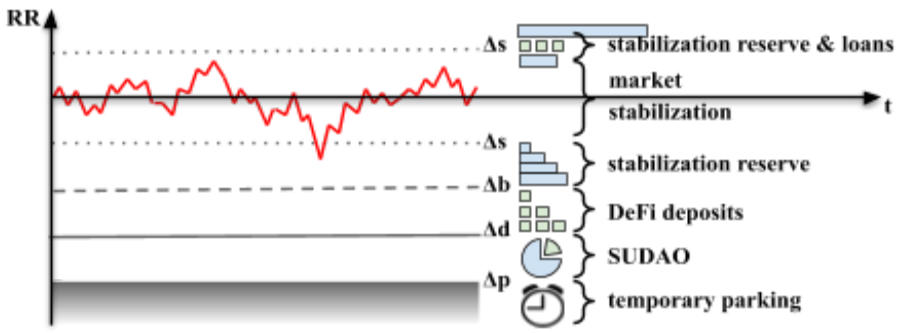

In this example, at the moment $t_{example}$ the peg value is equal to 0.4*Price(USD) + 0.6*Price(CPI).

## Decentralized Autonomous Organization

The StableUnit blockchain is controlled not by a centralized organisation such as corporation or government but by a **Decentralized Autonomous Organization (DAO)** via special governance utility token **SUDAO**. Ownership of the System is distributed among DAO tokenholders, who can vote for resolutions during a **Voting Event** (if such an event is triggered, for example by attack on the Oracle). Voters can also reach consensus about dividends. Voting for dividends (*isThisYearWillPayDifidends() => {votedYes > votedNo}*) happens once a year for the 50% of the minimum amount of the reserves during the last 12 months (for example, if during the year the reserve never held less than X USD worth of crypto assets, then people can vote for distribution of X/2 SU to the DAO tokenholders proportionally to the amount of shares they control). Such distributing should not compromise the stability of the System since that collateral was not used for a significant period of time. However, this will still reduce the theoretical resilience against potential loss of demand and future black swan events.

## Multi-layer Stabilisation Mechanism

Multi-layer stabilisation mechanism is a the stack of different stabilisation methods which are used one by one. Next mechanism is used if price fluctuation has exceeded the stabilisation capacity of the previous mechanism. The stack has several layers and best understood via this diagram




SU is a decentralized cryptocurrency therefore is freely tradable and inevitably has a volatile market price.

Let us fix some constant parameters, such $0 < \mathbf{\Delta s < \Delta b < \Delta d < \Delta p}$.

## 1. Market Stabilization

**If 1-Δs < SU < 1: (**i.e. Price(SU) ∈ (1-Δs...1) at the current time) What does it mean that the price of SU is below the nominal value $1? It means that there are not enough buyers so sellers have to offer lower bid to find buyers. But offers of $1 worth SU for lower price create an opportunity to make a profit, more accurately the **return on investment (ROI)** = $\frac{1-Price(SU)}{Price(SU)}$. For example, if you see $0.97 = SU sell offer, by buying it now you might expect to sell it back for $1 and make an ROI = 3/97 ≈ 3.09%. This profitability creates a competition among traders, which increases price back to ~$1.

**If 1 < SU < 1+Δs:** If price of SU is above $1 that means there are not enough traders who want to sell their SU for nominal i.e. $1 price. So buyers have to offer higher bid in order to buy some SU. But existence of buyers who are ready to pay more than $1 creates an opportunity for traders to make almost instant profit by selling $1 worth SU for higher, i.e. 1+(ROI) price. Therefore competition among these sellers reduces the price back to ~$1.

In other words, under normal circumstances the traders have a financial initiative to perform operations, which stabilize the price of SU. This implicit mechanism is not unique for our System and we constantly see this trading behavior on markets with Tether, Dai and others.

We define an **Elasticity of Layer (El)** - as a maximum change of demand, while current stabilisation layer is still able to keep the price on the desirable [a..b] range level. In other words, it is a cumulative demand shock the System can sustain. A very rough approximation:

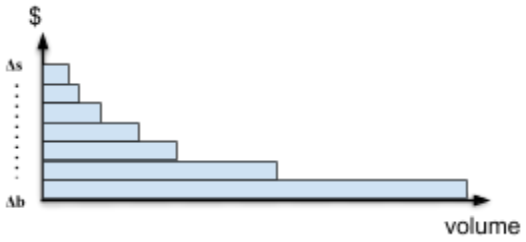
**El(L1)** = $\frac{Vol(L1)}{Cir(SU)}$, where **stabilization volume Vol(L1)** = $\sum_{traders}(risk * portfolio)$ and **Cir(SU)** - amount of SU in the circulation.

The market stabilization has limited elasticity, and when supply exceeds total demand provided by traders - the price will go down. To prevent further price fall the System uses special crypto reserves to provided necessary liquidity by offering a guaranteed supply.

## 2. Stabilization Reserve

The Stabilization Reserve (SR) is a smart contract which stores crypto asset as a collateral (for this example we will show only ETH but actual collateral could store diverse portfolio of cryptocurrencies which is possible to keep in a cryptographically secure way on the **parent blockchain** via **bridging mechanism**) and has an interface, which allows the caller to exchange their SU to ETH and back with predefined prices and limits.

**If 1-Δb < SU ≤ 1-Δs:** The System has a limited SU buy order at the prices [**$1-Δs … $1-Δb]** and gradually increased volumes, that total buy order volume per day is $SR_{daylimit}$ * Size(SR). The bids are distributed



according to the the hyperbolic function f_reserve_buy to provide a balance between a guaranteed liquidity for SU on the wide range of possible prices and maximizing the profit for the reserve from these exchange operations. This increases the maximum amount of SU users can sell to the reserve therefore increasing the set of possible market conditions when the System can guarantee stability.

**If 1+Δs ≤ SU:** Stabilization Reserve has an unlimited sell offer at $1+Δs level enforcing the upper limit for the SU price. All ETH which the System receives – it stores in the Stabilization Reserve.

Since the algorithm of the Stabilization Reserve is open source and executed on a blockchain, traders have full transparency of the predictability of the System's behavior.

Why the Stabilization Reserve cannot be manipulated? If we let ETH price be constant, then the Stabilization Reserve obviously cannot be out-traded because it buys low (1-Δb … Δs) and sells high (1+Δs). However, the ETH price (or other crypto asset held in reserve) is not constant. It is possible to exchange ETH for SU when the price of ETH is high and back when ETH has a low price, therefore, gain profit at Stabilisation Reserve's expense. However, it implies that such a trader is able to predict movements of the price of ETH but with this insight this trader is able to gain a bigger ROI on centralized exchanges due to the spread of buy/sell offers than from the Stabilisation Reserve. Stabilization volume

$$\mathbf{Vol(L2)} = \sum_{price=from\ 1-\Delta b}^{to\ 1-\Delta s} f\_reserve\_buy(price)$$

therefore the total cumulative demand shock the System can sustain is: El(L1) + El(L2). Keep in mind that we use Vol(L2) only for analysis, the actually smart contract uses a computationally simpler **reserve_ratio[t]** = $\frac{Price(reserve)}{Cir(SU)}$ which is **lower** than Vol(L2).

Hyperbolic function **f_reserve_buy** provides an optimal solution for stabilisation in the acceptable range of market conditions (see Appendix A).

This Stabilisation Reserve is sufficient to keep the price of SU stable for most of the expected market conditions. However during significant market volatility, when the portfolio of the crypto assets in the Reserve has a low price and at the same time the market demand for SU is low (which should not be a likely event because during the price fall of volatile cryptocurrency we naturally expect higher demand for the stable cryptocurrency to hedge the risks) this might be insufficient to prevent the further price fall.

## 3. DeFi borrowing and lending

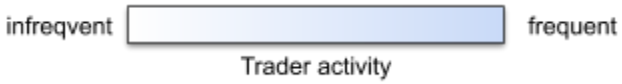
Repurchase agreements or repos have proved to be a very reliable and efficient method of monetary policy for central banks(in form of treasury bonds). If a Stabilization Reserve is unable to stop the fall of SU price, the System sells some repos which later can be redeemed at a higher price.

**If 1-Δd < RR ≤ 1-Δ:** The System unlocks repos which are enough to buy out some percentage (i.e. 5%) of the SU in circulation. Users are able to exchange SU for the resps in form of token and later to redeem it back when the Stabilisation Reserve regains its loses. More formally when **reserve_ratio > srr(t,C) safe reserve ratio.**

**If 1 < SU and SR had regained loses:** the holders are able to exchange repos back to SU for the price 1+Δs i.e. receiving ROI = (1+Δs - repo_price)/(repo_price).

Why the System does not automatically exchange REPOs for SU using a FIFO queue, when it is able, to motivate potential buyers to buy REPOs as soon as possible and get profit earlier? The reason is that we want to give holders of REPO an opportunity to freely exchange their assets on a secondary market. This market will emerge anyway. Some of the holders might want to sell their unredeemed REPOs right away and FIFO complication will only create the frustration which inevitably reduces the price for the REPOs (in FIFO model repo[i] cannot be redeemed before repo[i-1]). Instead we use special property of a REPO **unlock_time.** It is proportional to the time the REPO was bought since REPOs were unlocked. This time restricts redeemability of REPO in the System's smartcontact by the corresponded time.

Why does the System sell REPOs instead of offering some sort of savings account that people can use to park their funds for a bonus? Despite both these methods offering a mechanism to reduce SU in the circulation, in reality we can put all traders on a scale of how frequently they perform SU exchanges.




Savings accounts might be implemented into wallets therefore engage players who rarely trade to park their funds. While this does indeed reduce the money supply it does not influence market movement as much as selling REPOs to the traders who influence price faster.

### 4. Token sale

The stabilization mechanism is unlikely to initiate, since it is implied that very severe market changes have happened or that the System has been compromised. According to our principles (please see p. Foundation), the System exists solely for users, therefore DAO tokenholders should take this risk before regular users. While all current proposals of uncollateralized cryptocurrencies expected to collapse at this point, we propose to preserve the System with a token dilution mechanism.

**If (1-Δp < SU ≤ 1-Δd)**: automatically dilute the "SU_DAO" to recapitalize the System by selling it in an automatic auction until Price(SU) goes up or newly minted shares exceeds the maximum limit which is equal to some percentage of the available supply and depends on the time the System has existed. This limit is equal to

$$\sum_{t=0}^{current\_year} \left(\frac{1}{4}\right)^{(t/4+1)}$$

which is harmonic Dirichlet series and converges because (t/4+1) > 1. It leads to possible additional supply of 25% of SU_DAO in the first year, ~17% in the second, ~12.5% in the third and so on, defining the theoretical additional supply of SU_DAO by $\frac{1}{4}(1 + \sqrt{2}) \approx 60\%$.

### 5. Temporary Funds Parking

In the event of extreme or severe market instability, the Funds Parking Mechanism can be engaged. This process can be used as a last resort to cryptographically guarantee stability of the system.

Parking is in many ways comparable to lending at interest, where holder enters into an agreement to lock some portion of their funds at ($t_{starttime}$) for some period of time and unlocks their funds at an end time ($t_{endtime}$) and in the process receives the promise of some rate of return. Design of the Stabilization Reserve guarantees that reserves are never empty unless the cumulative price of all crypto assets in the reserve are zero. Therefore **∃ parking_ratio: Cir(SU)*parking_ratio*srr(t,C) ≤ Vol(L2) ⇒** it is always possible to temporarily park some percentage of available funds that rest of funds will be under conditions of guaranteed liquidity that drives SU price back to the peg value.

### Why stabilisation mechanism layers are in exactly this order

The Market Stabilisation happens only among the traders and it does not use reserves or other mechanisms so it does not affect the ability of the System to withstand against future black swans. To give a market players an initiative to perform operations which both benefit them and stabilize the System - smart contracts relating to the Stabilization Reserve should have spread/fee to motivate potential sellers to find a buyer before selling SU to the System.

Stabilization Reserve is used before REPOs because each time the System sells REPOs, it takes a responsibility to redeem it later with additional interest. With each REPO sell↔redeem cycle there is an increase in the amount of SU in circulation. The more SU is in circulation, the more REPOs the System may be required to sell in future volatility cycles to reduce the supply and stabilize the price, if necessary. The more REPOs are already in the circulation - the lower the price of the REPOs in the market so the System should sell even more REPOs to cover the exceeded supply. To prevent this positive feedback loop the System puts usage of REPOs as far in the future as possible and try to utilize Stabilisation Reserve first. Also, the condition that the REPOs will not be redeemed before

Stabilization Reserve regains its loses - prevents potential vector of attack when hostile players with sufficient funds are able to empty reserves during the low price for assets in the reserve. Because this player is able to redeem the REPOs only when reserves exceed the safe reserve ratio (i.e. regained it loss), this does not compromise stability of the System.

SU_DAO dilution does not have big demand elasticity but it motivates DAO tokenholders to behave in favor of the System's stability to preserve their stake in the DAO.

Temporary Funds Parking creates inconvenience and complications for users so it should be used only as the last resort for preserving the funds. It is better to still have funds with reduced liquidity than to lose all funds as would happen otherwise (and this can happen in all systems without a mechanism of unlimited demand elasticity). In the unlikely event of market crash of all assets in the reserves **and** demand for stable units **at the same time**, including a crash in the prices of REPOs and SU_DAO, the Temporary Funds Parking will restore price of the SU for some period of time which might be sufficient for the market to regain its value (especially if the event was a 'flash crash' only lasting for a short time). In a long-run this may be sufficient because this configuration will preserve stability longer than other configurations of a stable price cryptocurrency systems. Users knowing this fact will prefer to hedge their holdings and convert some of their assets from other systems to SU which increases demand for SU and brings the System into the stable condition without utilizing additional mechanisms of stablisations.

### Fees

Fees should be sufficiently small so that they provide little to no friction for the user.

Although it may be technically possible to eliminate transaction fees - this creates the risk of Denial of Service attack (DoS), where bad actors can send massive amounts of transactions at no or little cost. For this reason fees are necessary. Furthermore, transaction fees cover the System's expenses to support that network (miners and oracles).

### Incentives

There are several incentives to participate in the System work.

1. Oracles. Users who participate in the oracle system by providing accurate information about the market prices receive reward proportional the accuracy of the information provided. Funds they earn are cumulatively equal to the total reward that all of the miners received for the same period of time.
2. Miners. The type of mining incentives depend on the parent protocol, which may be PoW miners, Virtual mining buyers or PoS DAO tokenholders. These miners get rewards by mining a new block with a predefined rate specified by the parent protocol. In addition to that, a newly minded block sometimes includes additional transactions to the **SU_pool** - it is a special smart contract which stores SU the System uses to reward oracle participants and to pay to the sellers who would like to buy SU from the Reserve. To guarantee that *SU_pool* is sufficiently large to perform all sells for the exceeded demand, it always has at least 10% of the amount SU in the circulation. If the amount of funds in the *SU_pool* goes lower than 10% of SU in the circulation it requests *0.1*Cir(SU)* from the next mined block. This request is specified on the protocol level and is part of the consensus algorithm that all minersagree on by participating.
3. DAO tokenholders. Because SU is designed to have low-volatility, speculators should not expect any growth of the value of funds by purchasing SU relative to the peg. It is absolutely essential for wide adoption of SU as a cryptocurrency that any there are no potential profits of keeping SU as this would create incentives for users to keep it and not to spend it. However *SUDAOs* have predefined finite supply despite the possibility of inflation in the number of shares causing dilution (i.e. +60% as an additional stabilization mechanism). The possibility of dividends from holding *SUDAOs* creates a demand for them which will be higher for a growing and stable System. For example, the bigger the System , the more funds the System stores in the Reserves, the more stable and therefore the less Reserves will be used allowing

for more dividends. In an efficient market, the price of these shares should reflect the market's expectations of how the System will perform as low-volatility cryptocurrency and how large it is expected to grow over time.

## Privacy

Blockchain transactions needs more privacy. We are going to provide it via our zkSNARKs. Our zkSNARK circuit is providing following primitives:

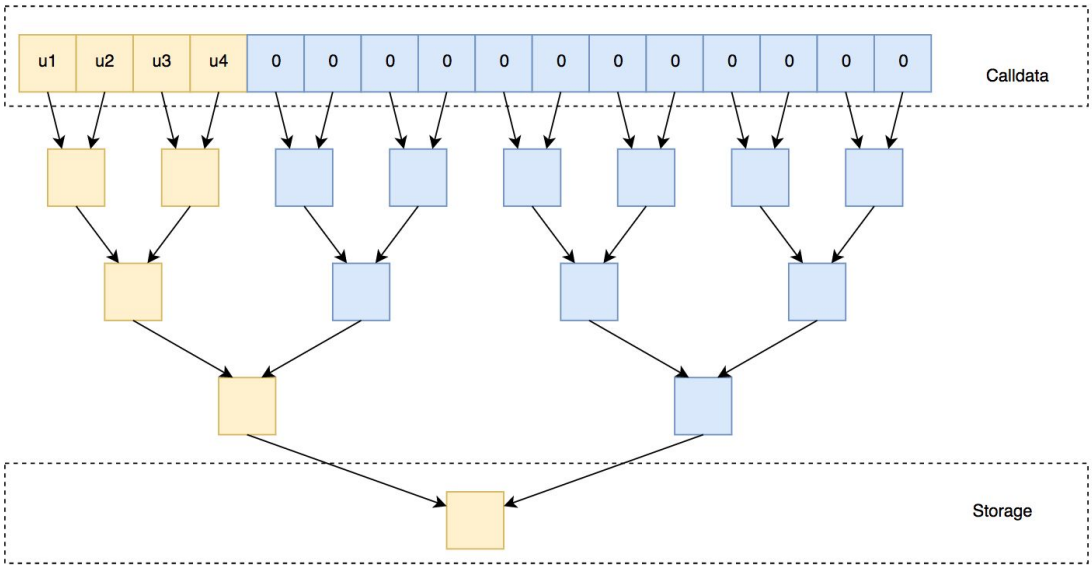
- multiple assets
- deposits
- withdrawals
- transfers
- atomic swaps

Also the user can show all his data for current time period for tax inspection or other government services as accounting report.

### State

StableUnit anonymous dApp is UTXO based solution, where hashes of UTXOs are stored in calldata, root hash of UTXO is stored in storage and UTXOs and transactions data is completely private.

All UTXO hashes are stored inside Merkle tree, and new elements may be added into this merkle tree one by one. Default values are zeroes.




### UTXO

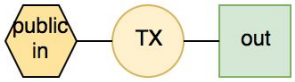
Utxo is composite of 4 parameters: assetId, amount, owner pubkey and unique utxoId. Only hashes are presented onchain without additional encryption.

To spend the utxo we need to publish onchain Nullifier - the deterministic function from UTXO. We are using ECVRF, this approach allow us to store private keys at hardware wallets separately from the prover.

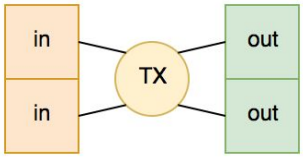
We are building chrome extension, supporting signing ECVRF and EDDSA on babyjubjub curve.

## Transaction

**Deposits** are simple and each deposit just add new element into the tree.

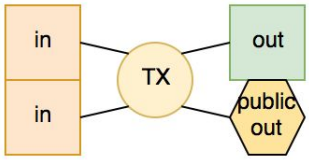



**Transfers** are based on 2to2 transfer primitive.



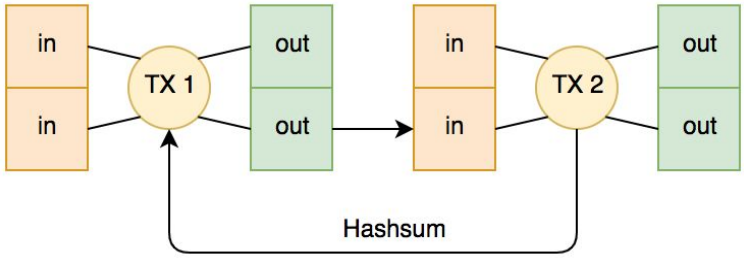

Some inputs and outputs may be zeroes. All kind of simple transfer (join, split, other transfers, excluding atomic swaps) may be presented as partial case of this construction.

## Withdrawals




Withdrawals structure is very close to transfer and allow user to withdraw arbitrary values, not strictly equal of any of any numbers of his UTXOs

## Atomic swaps

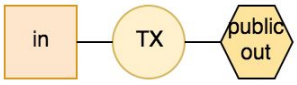



To change $n_A$ token A to $n_B$ token B maker create order at some 3rd party private platform. Taker finds the order and begin MPC procedure vs maker.

- Taker send to maker his public key
- Maker creates transaction
- $TX2 : (n_A, maker; n_B, maker) \Rightarrow (n_A, taker; n_B, maker)$ and send it to taker
- Taker creates transaction $TX1 : (\ldots;\ldots) \Rightarrow (\ldots; n_B, maker)$, add TX2 hash to the transaction and send it to maker. TX1 may be completed if TX2 completed.
- Maker joins TX1 and TX2 into one meta-transaction and show send it to dApp relayer

## Force withdrawals

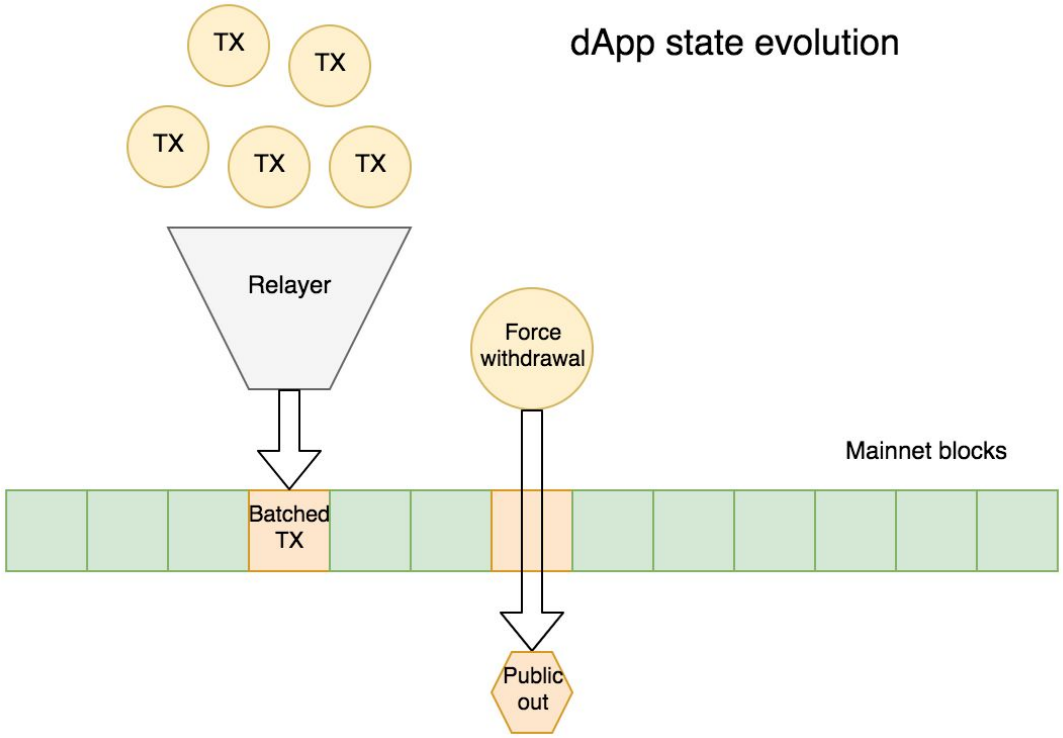
If the relayer stops working, users can process force exit procedure. It is not anonymous in single operator case. But if we have different pools/mixers onchain, it is safe. For exit user must prove that he knows UTXO preimage and publish valid nullifier.




## Relayer

There is account initialization problem in Ethereum. You cannot interact with blockchain if you have no balance, including cases when you withdraw more ether from the contract, than you pay for the gas.

That's why we need the relayer:3rd party, who process all dApp transactions with ethereum. To protect users from griefing from the relayer we allow to the relayer do only 10 interactions with the contract per 100 blocks and remember last 100 root hashes. Relayer can batch zkSNARK proofs via batch verifier. That means that instant transaction cost is asymptotic about 1 pairing for batched transaction and 4 pairings for instant transaction.




## Gas

I propose to use separate anonymous custodial pool for gas on the sidechain. User pay some ethers on main chain to the relayer. After that the relayer emit the ethers on sidechain. And user can burn that ethers on sidechain and attach such burn transactions to mainnet transactions to prove to the relayer, that he pays for the gas.

### Accounting report

Each UTXO must be encrypted via special sender's and receiver's public keys, depending from the time period of the transaction (for example, one key per day). To send report to 3rd party, user send his decryption private keys, after that 3rd party can scan the blockchain and got infromation from user's incoming and outgoing transactions. No assets can be transferred via such keys.

## Conclusion

With Satoshi's invention, Bitcoin, we stand on the shoulders of giants and for the first time in monetary history we are on the verge of completely disrupting the notion of national currencies and central banking. The keystone to this revolution is mass adoption of cryptocurrencies in which a price-stable cryptocurrency complements Bitcoin's role as digital gold. By 2022 global wealth is estimated to be over 340 trillion dollars[10.1] with a significant portion of this being held in banks or other financial institutions. This implies an incomprehensible opportunity for the first massively adopted decentralized and censorship resistant cryptocurrency as a form of money for everyday usage. Stable Unit takes the innovations made by the first price-stable cryptocurrencies to the next level because unlike the first wave of stable coins like Tether, Stable Unit is not backed by funds in an opaque bank account but rather, it is transparent and backed by top-tier cryptocurrencies like Bitcoin or Ether. Furthermore, Stable Unit provides techniques to ensure stability with a multi-layered stabilization mechanism. Finally, Stable Unit operates with a decentralized governance model in the form of a decentralized autonomous organization to effectively govern the project without a single point of failure. This approach amounts to a fusion of all the fundamental advantages of fiat and cryptocurrencies without the disadvantages like waning adoption or centralization. We believe Stable Unit has a great opportunity of becoming not just a reliable medium of exchange, but potentially the money of the future.

## References

[1.1]https://bitcoin.org/bitcoin.pdf

[1.2]https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3024330 *Economics of Networks Journal.* Social Science Research Network (SSRN).

[2.1]https://techcrunch.com/2017/04/20/whats-keeping-cryptocurrencies-from-mass-adoption

[2.2]http://lightningbitcoin.io

[2.3]https://medium.com/@rusty_lightning/bitcoin-lightning-things-to-know-e5ea8d84369f

[2.4]https://hackernoon.com/the-different-categories-of-cryptocurrencies-a57ba4d77c9a

[2.5]https://interestingengineering.com/iota-a-cryptocurrency-with-infinite-scalability-and-no-fees

[2.6]SPECTRE: A Fast and Scalable Cryptocurrency Protocol Yonatan Sompolinsky, Yoad Lewenberg, Aviv Zohar Published 2016 in IACR Cryptology ePrint Archive

[2.7]https://www.statista.com/statistics/664604/paypal-marketing-spending

[2.8]https://trends.google.com/trends/explore?cat=16&date=today%205-y&q=paypal,bitcoin

[2.9]https://arstechnica.com/tech-policy/2017/12/a-brief-history-of-bitcoin-hacks-and-frauds

[2.10]https://pando.com/2014/01/03/why-bitcoin-is-both-less-and-more-secure-than-credit-cards

[2.11]https://en.bitcoin.it/wiki/Multisignature

[2.12]https://blog.zeppelin.solutions/on-the-parity-wallet-multisig-hack-405a8c12e8f7

[2.13]https://paritytech.io/security-alert-2

[2.14]https://cointelegraph.com/explained/escrow-explained

[2.15]https://bitcointalk.org/index.php?topic=855778.0 Bitcointalk Escrows - Trade Safely

[3.1]"money : The New Palgrave Dictionary of Economics". The New Palgrave Dictionary of Economics. Retrieved 18 December 2010

[3.2-3.3]https://coinmarketcap.com/currencies/bitcoin/ (12 Nov - Dec 2017 and 17-22 Dec 2017)

[3.4]https://support.microsoft.com/en-ca/help/13942/microsoft-account-add-money-with-bitcoin

[3.5]https://www.overstock.com/blockchain

[3.6]https://www.virgin.com/richard-branson/bitcoins-space

[3.7]https://bitpay.com/

[3.8]https://www.bloomberg.com/news/articles/2013-11-04/valve-lines-up-console-partners-in-challenge-to-microsoft-sony

[3.9]https://steamcommunity.com/games/593110/announcements/detail/1464096684955433613

[3.10]http://www.bbc.com/news/technology-42264622

[4.1]https://finance.yahoo.com/quote/GOOG

[4.2]https://goldprice.org

[4.3]https://coinmarketcap.com/currencies/tether

[5.1]http://www.nbbmuseum.be/doc/chap5e.pdf

[5.2]https://www.investopedia.com/terms/p/purchasingpower.asp

[6.0]https://www.forbes.com/sites/shermanlee/2018/03/12/explaining-stable-coins-the-holy-grail-of-crytpocurrency

[6.1]https://www.coindesk.com/top-uk-tax-agency-eliminate-20-levy-bitcoin-trading/

[6.2]https://www.forbes.com/sites/nathanielparishflannery/2016/12/19/5534

[6.3]https://www.washingtonpost.com/news/worldviews/wp/2018/03/23/venezuela-hopes-to-tackle-the-worlds-worst-inflation-by-deleting-zeros-from-its-currency/

[6.4]https://www.forbes.com/sites/kenrapoza/2015/01/27/russia-to-retaliate-if-banks-given-swift-kick

[7.1]https://MakerDAO.com/whitepaper

[7.2]https://www.researchgate.net/publication/322951893_Stabilising_fractional_reserve_banking

[7.3]http://positivemoney.org/2012/06/spanish-crisis-illustrates-the-inherent-instability-of-fractional-reserve-banking

[8.1]https://www.reddit.com/r/MakerDAO/comments/7zitlv/why_should_we_trust_the_MakerDAO_oracle_system/

[8.2]https://developer.MakerDAO.com/feeds/

[8.3]https://bravenewcoin.com/news/is-bitcoin-at-risk-from-miners-leaving-when-the-block-reward-is-halved-in-july

[8.4]https://www.bls.gov/opub/mlr/2008/08/art1full.pdf

[10.1]https://www.credit-suisse.com/corporate/en/articles/news-and-expertise/global-wealth-outlook-201712.html