YggChain - White Paper

A Price-Stable Cryptocurrency for Everyday Transaction.

PHAM Tuan Anh (Zergity@gmail.com)

DRAFT 0.01.8 (2018/07/19)

Abstract

"*One of the main problems with Bitcoin for ordinary users is that, while the network may be a great way of sending payments, (...) Bitcoin the currency is a very volatile means of storing value.*" -- Vitalik Buterin on The Search for a Stable Cryptocurrency.

Price volatility and scalability keep hindering all cryptocurrencies to be widely adapted, far from the level of everyday transaction. Stability of value is one of the 5 must-have properties of money, and the lack of a proper scaling solution are the reasons why transaction throughput is extremely limited and the fee is getting higher and higher everyday. Until these curses are lifted, cryptocurrency will forever stuck in the basement of blockchain technology. And lifting those curses, is no other than YggChain's purpose: to stabilize the currency price and to scale the network for everyone, everyday and every transactions.

Table of Contents

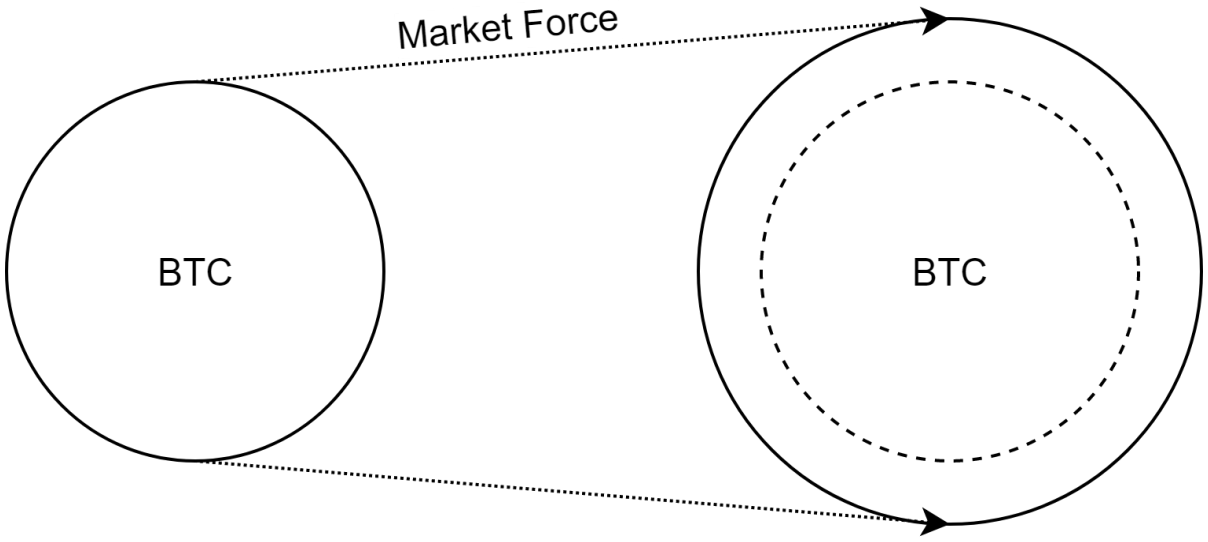
Introduction

YDR is a price-stable cryptocurrency for everyday usage, from paying for a beer to billions dollar business transaction without worrying about losing its value every other seconds. Powered by YggChain - a public, permissionless blockchain with instant confirmation and Staking Service Network, where many kinds of decentralized service are built and served. Beneath all, Yggdrasil - an economic-driven sharding protocol scales the blockchain to any level of adaptation, remove all the network bottleneck for an unlimited transaction throughput, and the lowest transaction fee ever possible.

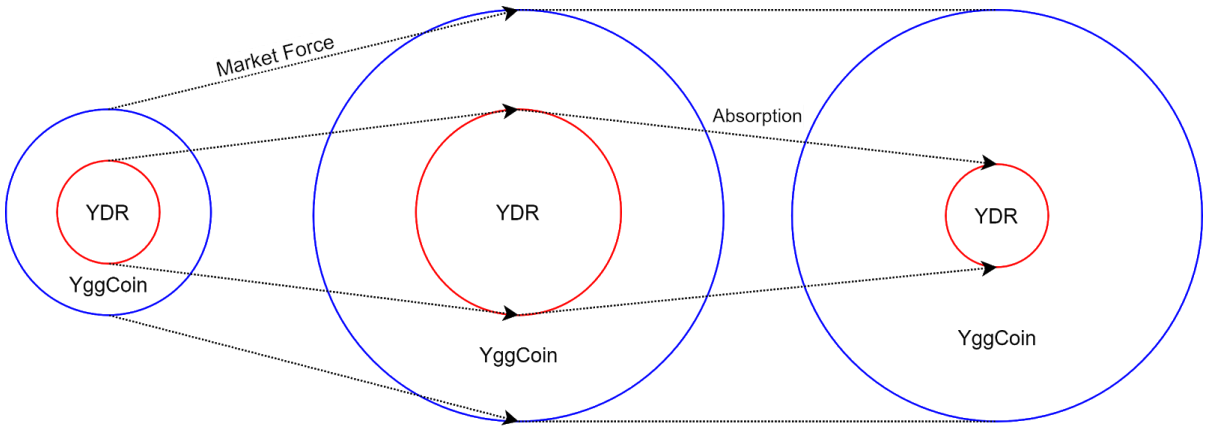
Economic

How can the market price of a free-floating asset can be stabilized? Robert Sams' paper proposes a solution to use another asset to absorb all the price volatility of the stabilized token.

In a fixed or deterministic supply currencies (gold, stock, Bitcoin, etc.), token price is completely driven by the market force. When the market is demanding or declining, the market price of each currency unit is increasing or decreasing with the same rate.



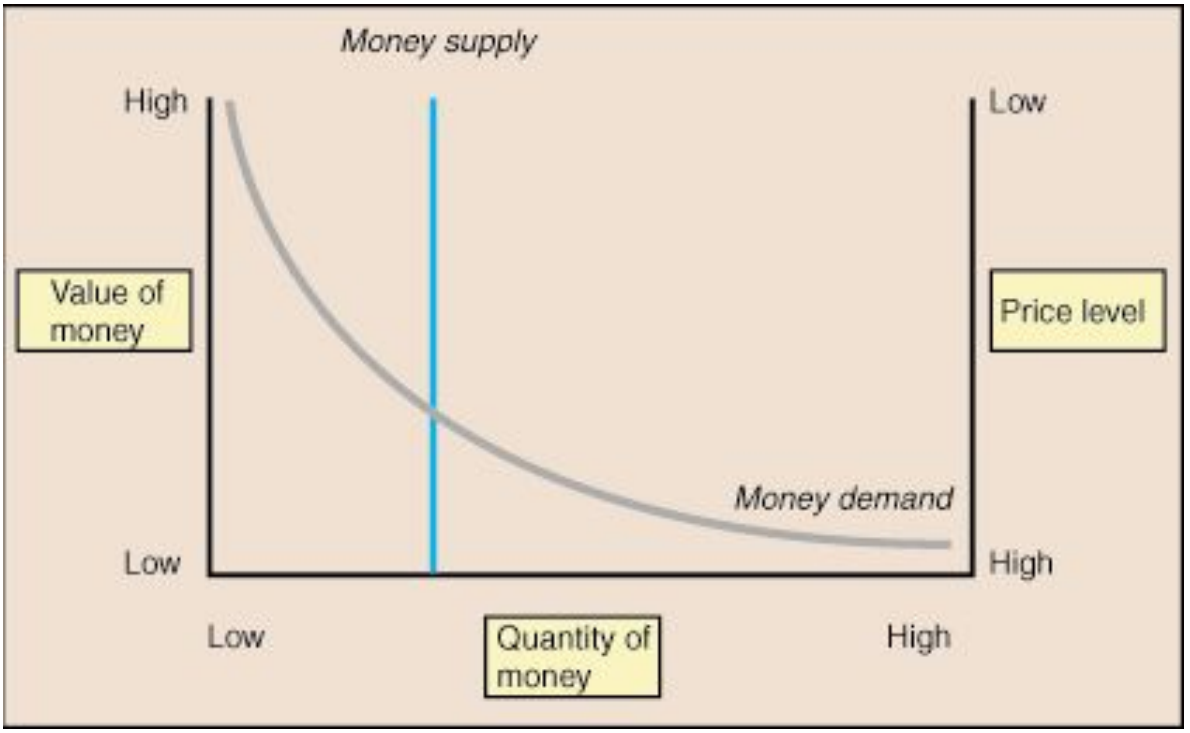
In YggChain - an elastic supply currency, there's a dedicated token (YggCoin) to absorb all the price volatility of the main price-stable token - YDR.



When the market is demanding, both YDR and YggCoin price will be increased, but the internal mechanism of YggChain will push the price of YggCoin even higher to lower YDR price back to the previous value, effectively stabilizes the YDR price around a desirable value. The same mechanism will work in revert when the market is declining.

The Quantity Theory of Money

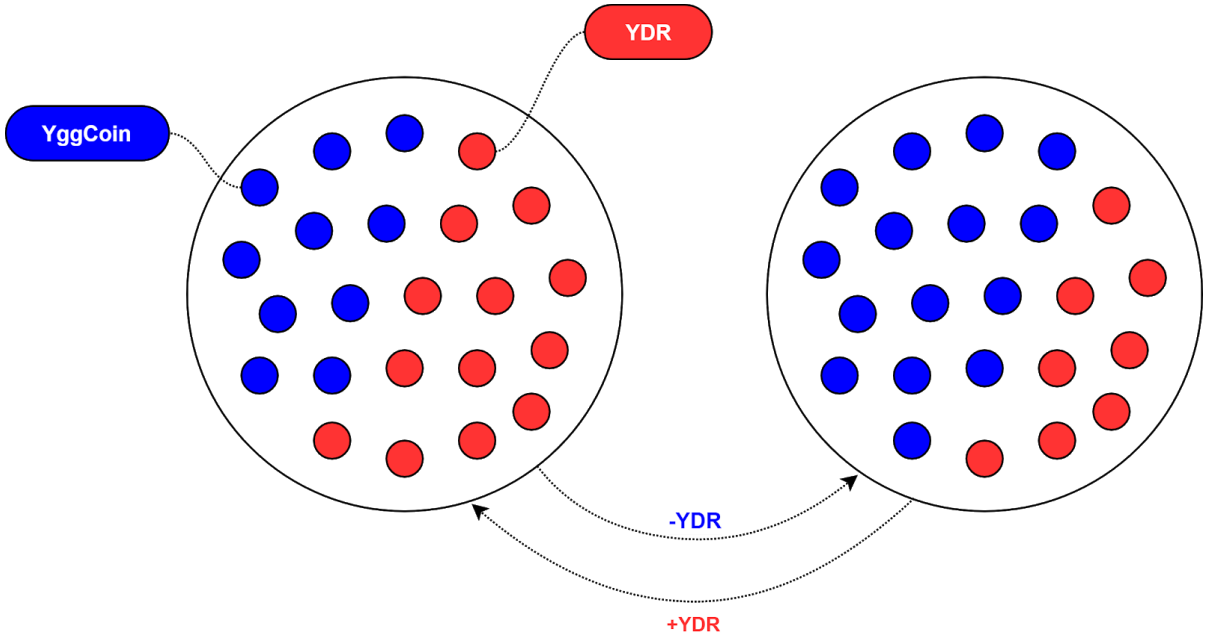
In monetary economics, the Quantity theory of money states that the general price level of goods and services is directly proportional to the amount of money in circulation, or money supply.



YggChain leverages this theory to stabilizing the price of its currency - YDR.

1. At time t(0), there is **N(YDR)** number of YDR, each worth 1.00 XDR.
2. By time t(1), each YDR has increased in value by 10% to be worth of 1.10 XDR.

3. Now, if somehow, we can increase the circulating supply of YDR by 10%, its price will be decreased to 1.00 XDR, since the market now has 10% more supply for the same demand. This process is called **10% expansion**.
4. At time t(2), each YDR has decreased in value by 5% to be worth 0.95 XDR.
5. If we can decrease the circulating supply of YDR by 5%, the price will be increased to 1.00 XDR, since the market now has 5% less supply for the same demand. This process is called **5% contraction**.



The supply of YDR is changed by converting them from and to another token, called YggCoin. YggCoin supply is also changing along with YDR supply in revert direction, effectively swings the price of YggCoin up and down to absorb YDR's price volatility.

Stablecoin Protocol

Two-tokens system

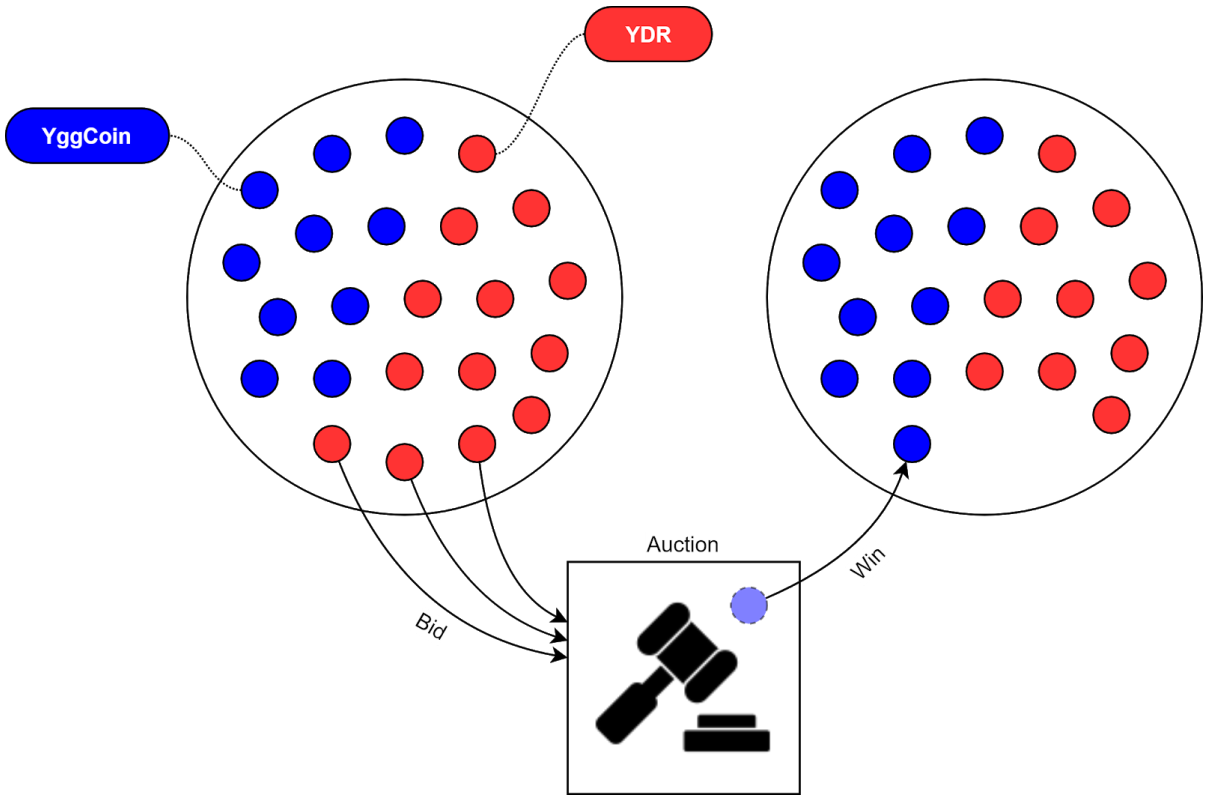
The YggChain Stablecoin Protocol consists of 2 types of token:

- YggCoin (YGC) - represents the value that is contributed to the network from outside (by capital investment, mining or providing service to the network). The holder of YggCoin take the role of stock/shares holder, which they bear the responsibility to maintain and stabilize the network; they are rewarded with transaction fee, exchange fee and all the capital value gain when the network grows. Obviously, like any other investment, their risk is losing capital value when the network shrinks.
- YDR (YggChain Drawing Right) - represents the service provided by the network. The holders of YDR take the role of customer, for currency service YggChain provides. Their benefit will be protected by the protocol with the highest priority.

To keep the price of YDR stable, the following process is continuously repeated:

1. YDR price is fed from the outside, or calculated using internal values of the network. (See Exchange Rate Feed)
2. If the YDR price is **c%** higher than **1.00 XRD**, an **c% contraction** is taken place in the next phase.
3. If the YDR price is **x%** lower than **1.00 XRD**, an **x% expansion** is taken place in the next phase.

The conversion between YggCoin and YDR is the main mechanism for YggChain stablecoin economic.



Expansion

In the event of **x% expansion**, when there is total **N(YDR)** of YDR in circulation, a total of **X = N(YDR) * x%** is created (out of thin air) and sold in an off-chain public auction (see YggAuction) for YggCoin. The price of YDR/YggCoin is completely market driven, which is usually a little less than the current market price, effectively drive the market price of YggCoin higher. The YggCoin used to buy auctioned YDR, will be destroyed, taken out of circulation.

The auction ends with new transactions included in the chain. The result are:

- The total supply of YDR is increased by **x%**, thus decrease its price by **x%** to exactly **1.00 XDR**. Newly created YDR is given to the highest bidders of the auction.
- The total supply of YggCoin is decreased, thus increase its price, benefits all current YggCoin holders. The highest bidders will be the most benefit, because not only they

have their remains YggCoin price increased, they also sold their YggCoin for a higher price than the market through the auction of the newly created YDR.

How much percent of YggCoin will be sold? [TODO: insert math proof here]

$$\%N_{YggCoin} = \%N_{YDR} \times MC_{YDR} \div MC_{YggCoin}$$

Contraction

Contraction process is exactly the opposite of Expansion.

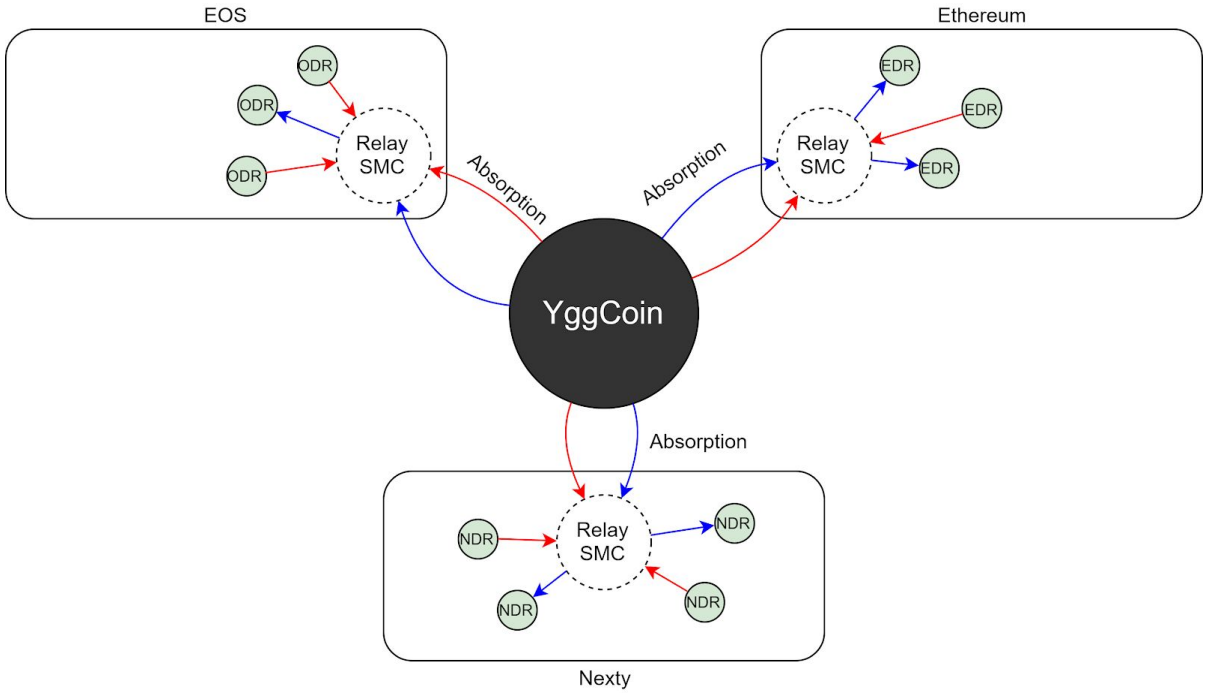
In the event of **c% contraction**, when there is total **N(YDR)** of YDR in circulation, a total of **C = N(YDR) * c%** is needed to be taken out of the circulation. New YggCoin is created (out of thin air) and sold in IggAuction, enough to cover all total **C** number of YDR. The price of YggCoin is obviously market driven, which is usually a little less than the current market price, effectively drive the market price of YggCoin a little lower. The YDR used to buy auctioned YggCoin, will be destroyed, taken out of circulation.

The auction ends with new transactions included in the chain. The result are:

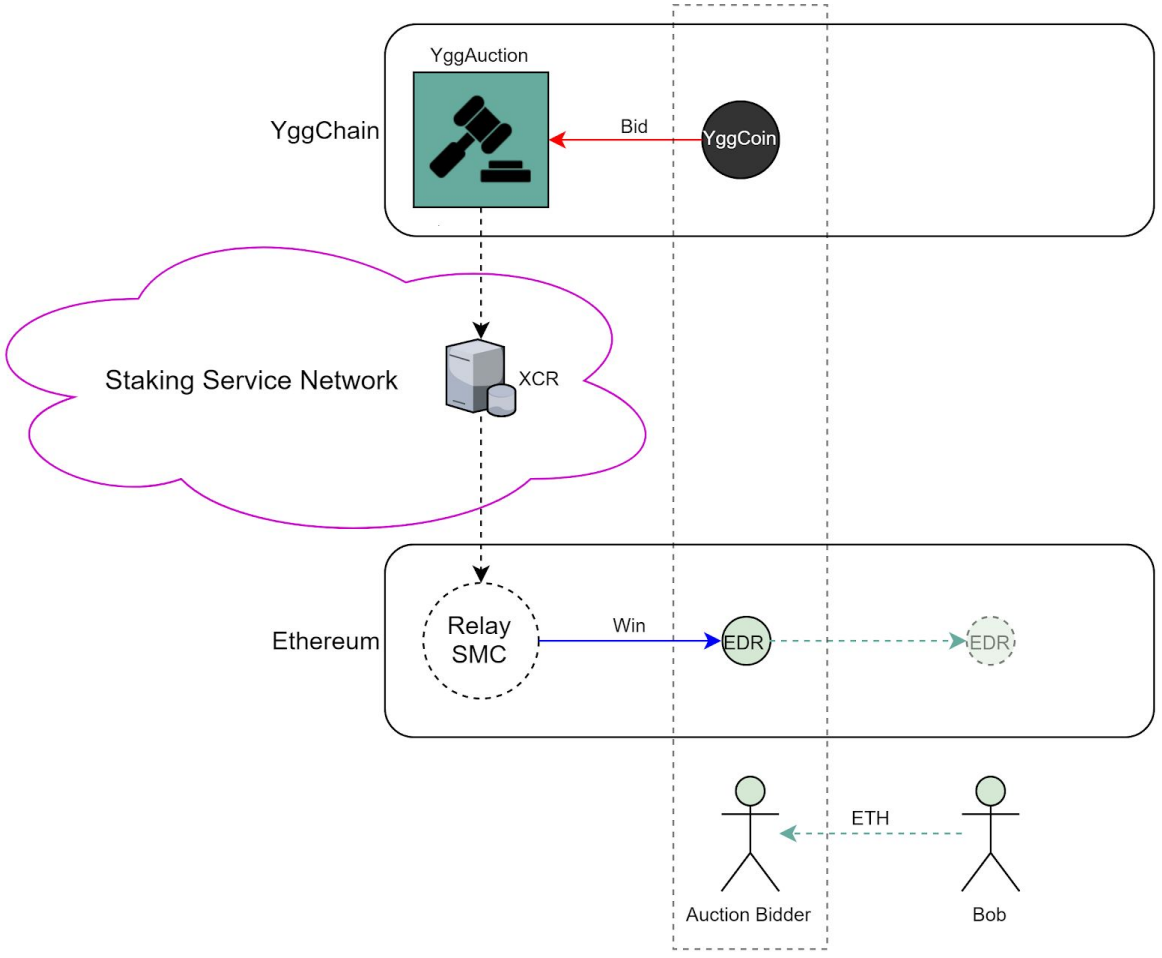
- The total supply of YDR is decreased by **c%**, thus increase its price by **c%** to exactly **1.00 XDR**. Newly created YggCoin is given to the highest bidders of the auction.
- The total supply of YggCoin is increased, thus decrease its price, hurting all current YggCoin holders. The highest bidders lose the least, because despite they have their remains YggCoin price decreased, they can buy some new YggCoin with a lower price than the market through the auction.

Cross-Chain ERC20 Token Absorption

[TODO]


EOS
Ethereum
ODR
ODR
ODR
Relay
SMC
EDR
EDR
EDR
Relay
SMC
Absorption
Absorption
YggCoin
Absorption
NDR
NDR
NDR
NDR
Relay
SMC
Nexty

[TODO]


YggAuction
YggChain
Bid
YggCoin
Staking Service Network
XCR
Ethereum
Relay
SMC
Win
EDR
EDR
ETH
Auction Bidder
Bob

Unit of Currency

YDR will always be pegged to the most stable currency of human (and aliens that we known of), which currently is XDR, a basket of fiat money.

In the future, YDR might eventually be no longer pegged to fiat money basket, but something else, be it a market basket (CPI) or anything considered having the most stable purchasing power by humanity (and alienity).

In the event of YDR unit change, the price of YDR will not be affected, only the unit reference is changed.

E.g.

Before	YDR = XDR
Unit change event	Pegged unit will be changed from XDR to EUR. Exchange rate at the event: 1 XDR = 1.18151 EUR.
After	YDR = 1.18151 EUR. The protocol will now stabilize the YDR price to 1.18151 EUR instead of 1 XDR, until the next unit change event in the future.

Because the YggChain itself does not have to know about its reference unit and exchange ratio, the system is not affected. The only affected parts are price feeding oracle services, where they feed the data to the system using the percentage of YDR itself, not of the reference unit.

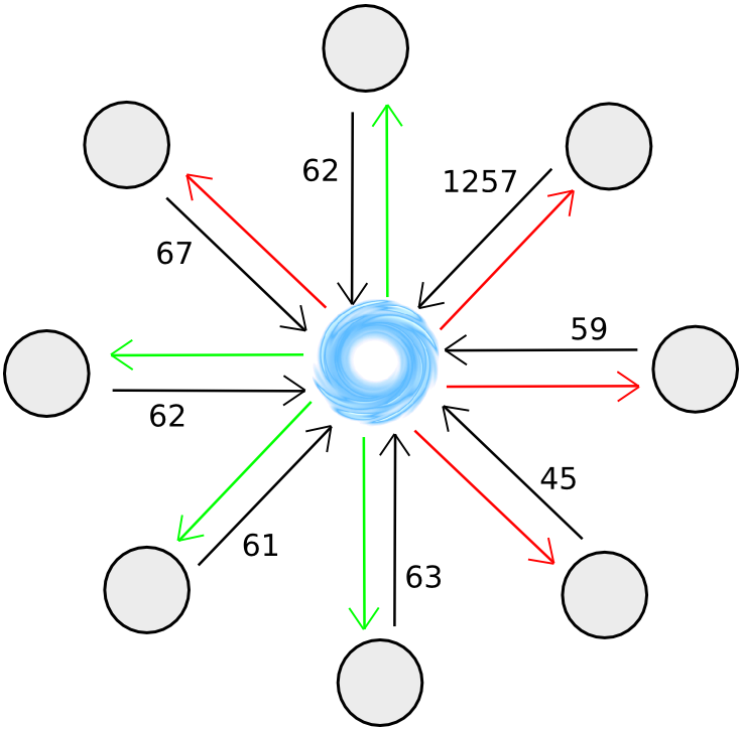
E.g.

After the YDR switch to EUR with the exchange rate of 1.18151, at 1 point of time, where YDR price is dropped to 1.1696949 EUR, oracles will feed the value of 0.99 (1.1696949/1.18151) to the system, represent the 1% price drop, and trigger an **1% contraction** event.

Exchange Rate Feed

For a blockchain to have knowledge of the exchange rate of its own crypto tokens, there are two solutions: exogenous (the price is fed from outside of the network) and endogenous (the price is measured using internal variables inside the network).

YggChain initially uses an exogenous method, which have the price fed from all oracles using SchellingCoin data feed scheme.



SchellingCoin: A Minimal-Trust Universal Data Feed

This introduces a certain degree of centralization, (in which the source for price is fed from multiple centralized exchange services), but currently is inevitable due to the lack of a complete endogenous method.

An endogenous solution is being researched and developed by YggChain team, since it requires more data collected by the the network itself in the public performance. YggChain price evaluation will eventually switch to its endogenous method, once the research is completed and fully tested.

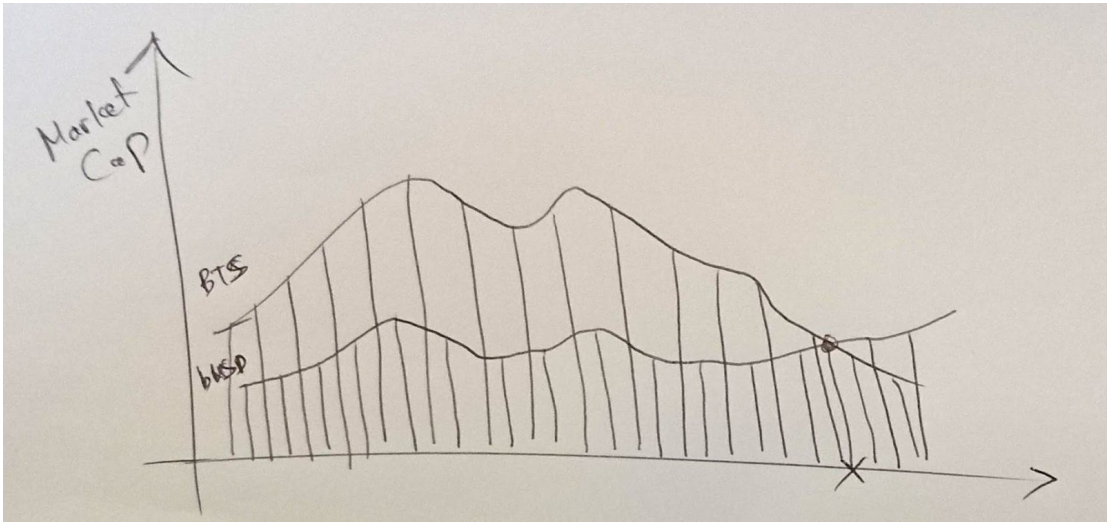
Black Swan Event

What happen when when the coin price drop too low? This is the most frequently asked question for every collateral stablecoin projects, and such event does happen to them every once in a while. YggChain is not a collateral stablecoin, but an elastic supply stablecoin. That doesn't mean it is immune to all black swan events, it means that the chance are extremely unlikely, and how the system handle it is much more elegant.

Collateral stablecoin, is essentially a collateral loans system, where you lock your asset (BTS, ETH) up into a contract, to borrow money (BitUSD, DAI) from people who need a stable-price asset. The problem is that, the borrowers always have 2 choices:

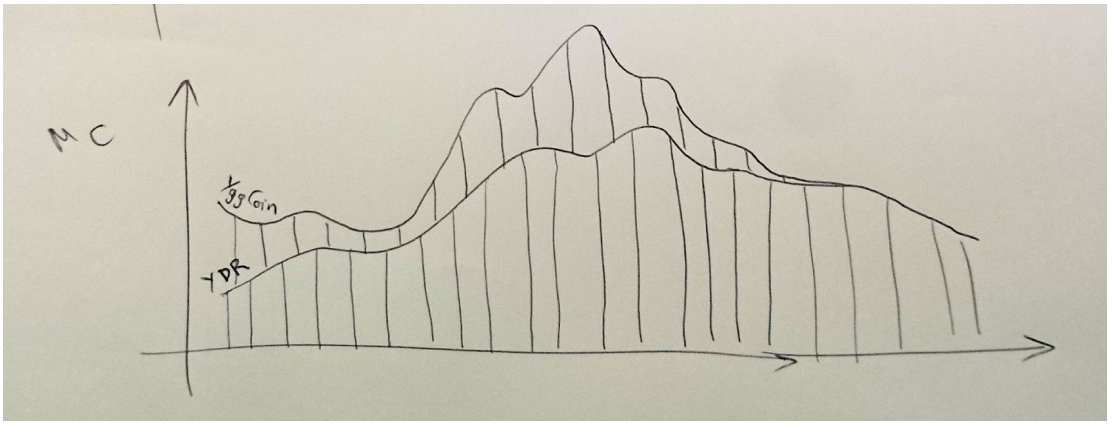
- to repay the debt (BitUSD, DAI) and get their locked up asset (BTS, ETH) back, or
- to abandon their collateral asset, and run away with their debt.

To repay or not to repay? That is the question any rational borrowers can easily answer, it depends on "*Which is worth more?*" or to be precise, "*Which will be worth more?*"



This is a mock up chart of BTS and BitUSD (unstacked). The system is healthy when the market cap of collateralized BTS is much higher than the market cap of borrowed BitUSD. BitShare requires borrowers to over-collateralize their BTS with at least 200% value of borrowed BitUSD, (MakerDAO requires 150%.) The system will break when the BTS price (along with its market cap) is dropped by 50%, (and ETH by 34%.) That the point where their locked up asset is no longer worth more than the debt they owe.

In YggChain, there is no loan nor debt, no borrower nor lender; nobody owes anyone anything. No one has such easy choice to abandon an asset for the other. YggCoin is used to absorb the price volatility of YDR, not collateralized it.



This is a mock up chart of YggCoin and YDR market cap (stacked). All the circulating YggCoin is used to stabilize YDR price, and as long as the YggCoin market cap is not zero, the YDR price can be stabilized.

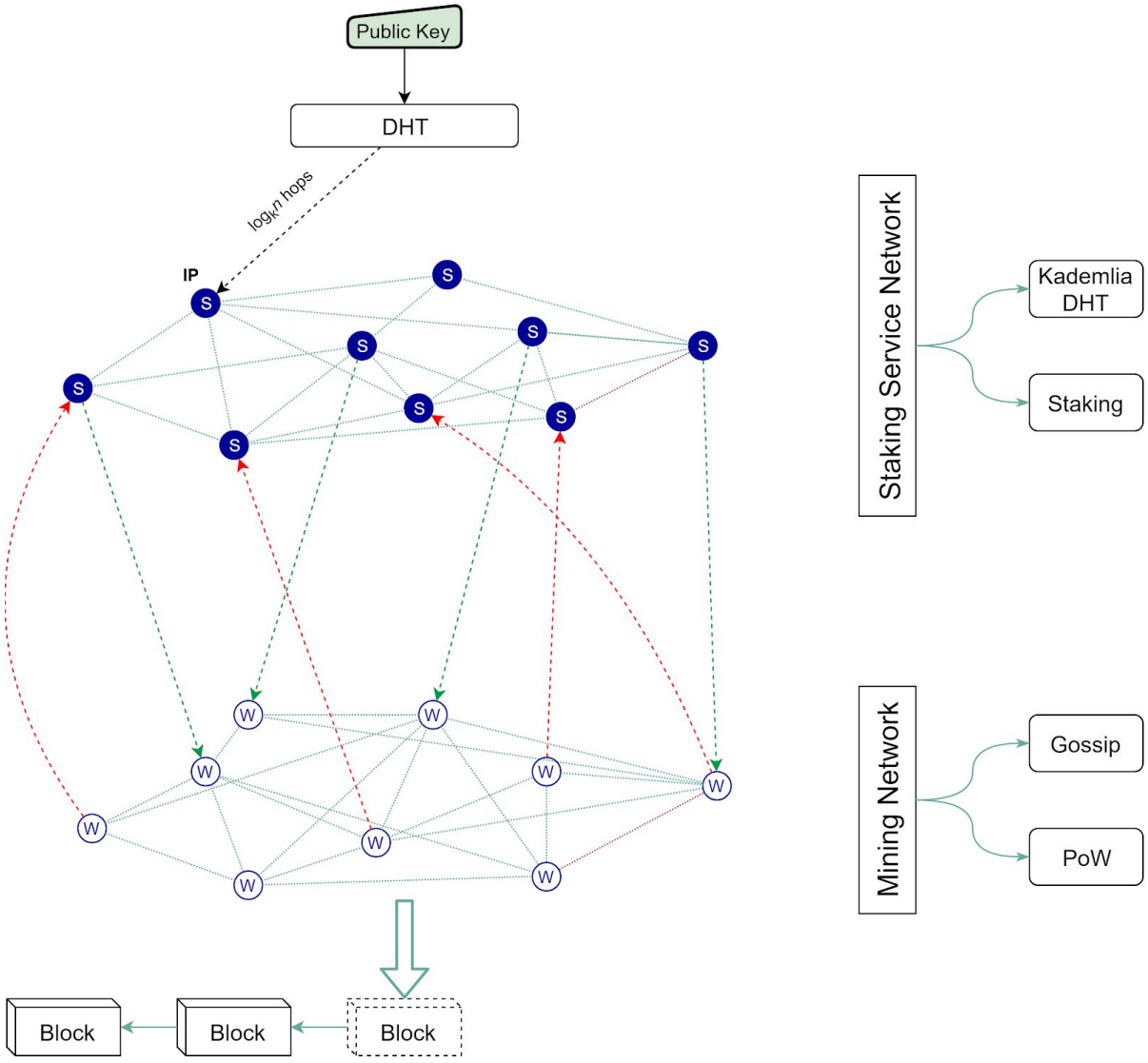
To recap, BitShare stablecoin will be unstable when the price of BitShare is quickly dropped by 50%, same with MakerDAO when ETH price dropped by 34%. While YDR price is unstablizable only when YggCoin lost all of its value, a.k.a. 100% price drop.

No financial system, traditional or digital, can assume that it is unaffected by or immune to Black Swan Events, but some system is more robust against them than others. The Quantity Theory of Money is well researched by economists and the elastic supply mechanism is what all fiat money now are made of. Thus, we believe YggChain's stablecoin will be the last one to fall, if any critical crisis ever happens to the blockchain ecosystem, or specifically stablecoins.

Consensus

There are a lot of consensus algorithms is being developed in the blockchain world, only one is well tested by time until now - the Proof of Work. YggChain's consensus at it core, is PoW, and to provide more desirable features, a Staking Service Network is implemented as the second layer, and some of the service drastically improves the consensus properties. Most notable type of Staking Service is Validator, along with PoW, introducing the Lock & Block protocol - the main consensus of YggChain.

Staking Service Network



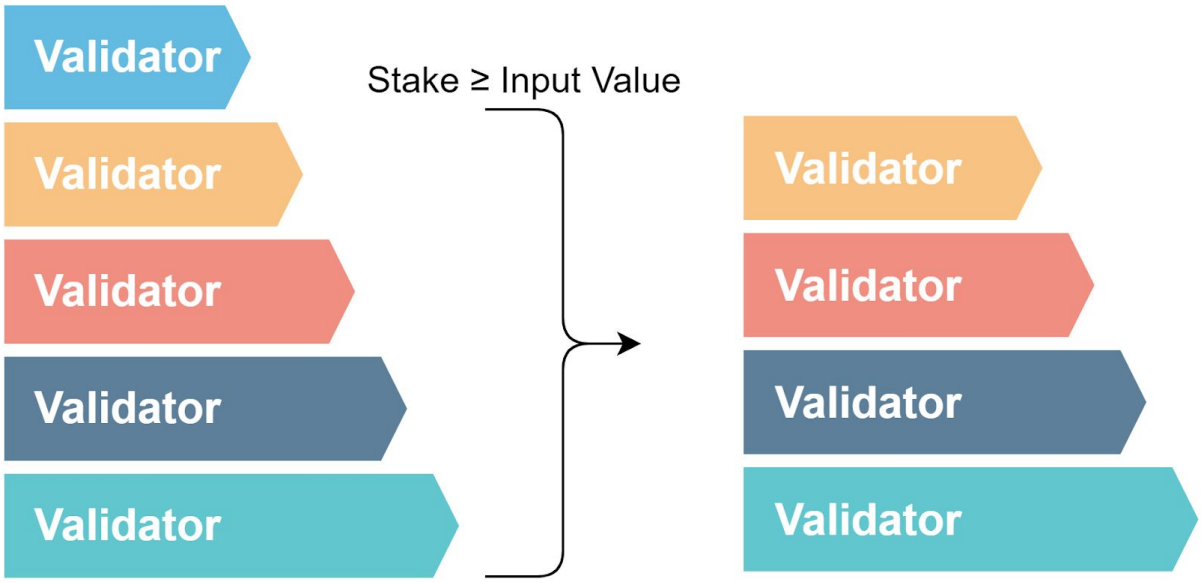
Staking Service Network (SSN) is a P2P network where nodes stake their YggCoin to provide many services to the chain, for 30% of block reward. YggChain currently has the following types of Staking Service:

- Validator: lock the transaction input to provide consistency and instant confirmation for the network.
- Oracle: feed data from outside of the network.
- Market: process the market order, including the YggAuction.
- Cross-chain Relay: relay states from and carry actions to other ledgers.

Staking Service Network provides many critical roles, in the hand of adversaries, it can sabotage the whole system for its own benefit if there's no consequence. That is why one of the condition for a Staking Node to join the network, is an amount of YggCoin has to be freezed (or staked). This freezed YggCoin serves 2 purposes:

1. Prevent Sybil attack on the SSN.
2. Will be destroyed (or slashed) when a node is detected with bad behaviour.

There is no hard limit of YggCoin a node has to stake, but each kind of service has its own limiting mechanism to protect the system. Most notably, the Validator Node can only be selected to lock an input, if its stake is no less than the input UTXO value.



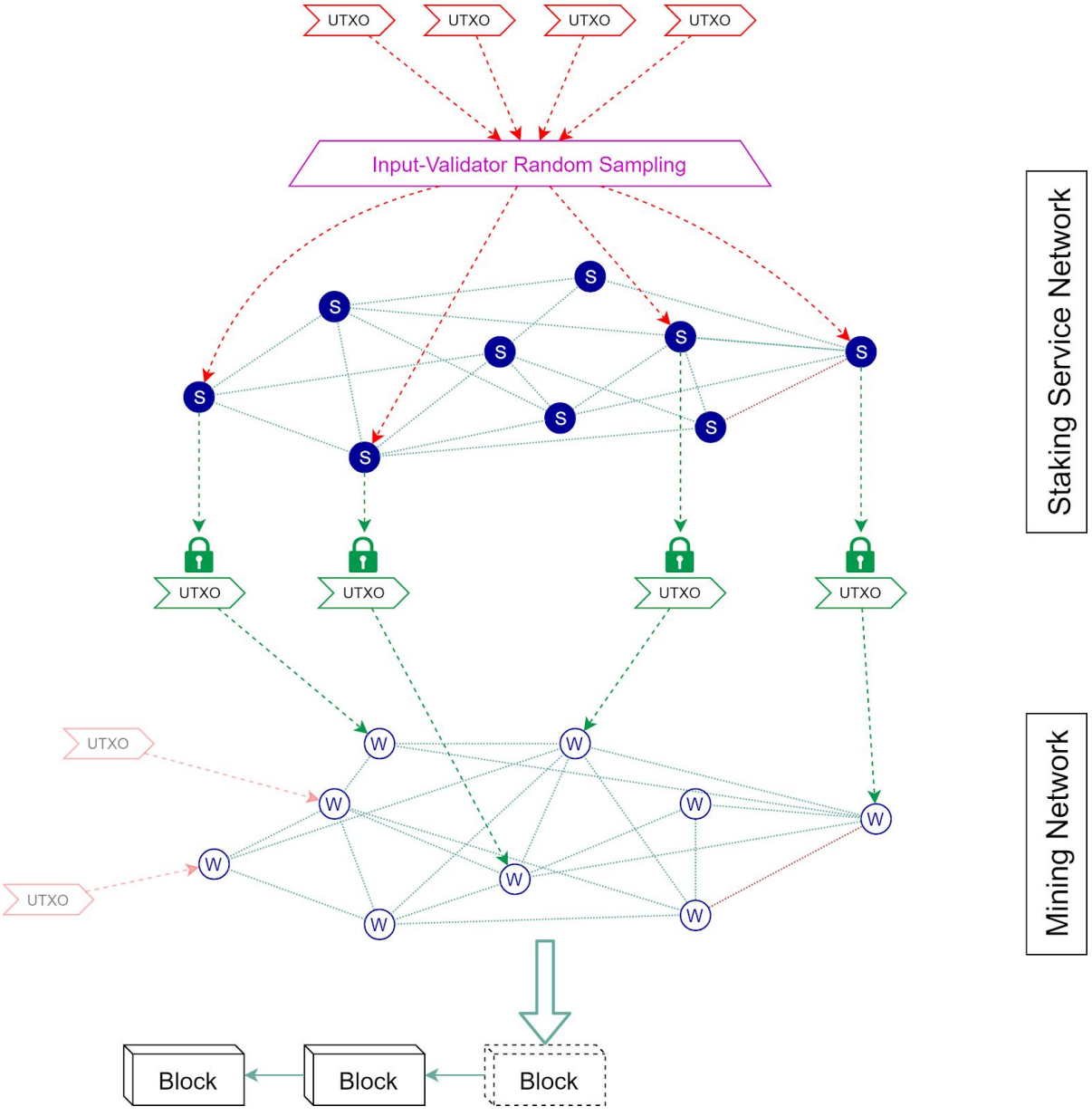
Lock & Block

Proof of Work provides the security and immutability for the blockchain, but the confirmation time is often very long, and even unpredictable. In the sharding context, a pure PoW is also extremely vulnerable to 51% attack and shard-hopping attack.

In YggChain, when transactions are propagated to the network, their inputs can be validated and locked by Validators first using validators' signature, then packed in a block by PoW miners. When a transaction is Locked for Block #B, then:

1. There will be no conflict lock in 3 blocks: #B-1, #B and #B+1, *none that cannot be punished*.
2. When properly propagated, it will override any conflicted Open Transaction in #B and #B+1.

In short, when the recipient sees a Locked Transaction propagated to his node, and latest block is #B-1 or #B, it's almost certainly that the transaction will end up in the chain.



This secondary mechanism provides the following properties to the network:

- Instant confirmation: A locked transaction will almost certainly be included in a block, and the locking time is only limited by the message traveling speed of the network, which is almost instantly.

- The possibility of double spent attack, is now not of PoW miner, but belong to Validator Nodes, which can be punished (or slashed). See Double-spent Attack Penalty.
- The possibility of rewriting history, is now extremely hard even with 51% mining power and 51% of Validator Nodes due to the combination of Lock & Block consensus. To have the same chance of successfully perform a 51% attack in Bitcoin, an YggChain adversary must have 51% mining power and 100% Staking Nodes of the target shard. [TODO: insert math proof here]

Regarding the CAP Theorem, YggChain Input Locking mechanism prioritizes Consistency over Availability. That means, sometimes, there will be not enough online Validators to lock a specific input, these cases including:

- Input value is too large, no Validator in the network has enough stake to be selected.
- Half of the validators selected for an input are offline. And user cannot find another lockable input to replace it.

In those cases, luckily, Input Locking is a secondary mechanism, transaction can still be propagated and mined by the good old Proof of Work miners as Open Transactions. Open Transactions need to wait for at least 2 confirmations in order to be spent, because they can be overridden by conflicting Locked Transactions in the very next block they were included.

The success rate of Input Locking process is directly proportional with the availability of Staking Validators Nodes. To increase this rate along with the availability of the service, one of the most effective strategy is to punish the offline Validator nodes by burning their stake. This can simply be done by burning all staking YggCoin, and compensate the active nodes by increasing the reward.

Let's say, for all staking YggCoin, 0.01% of them will be burnt every block, and currently there are total <ToS> of YggCoins are staked. So each block, a total (ToS x 0.01%) of YggCoins will be burnt, then the total Staking Service reward should be (ToS x 0.01%) of YggCoins more than when no coin is staked. This compensation in long term will negate the burning rate of active Staking Nodes, while at the same time, punishes the inactive nodes, incentivize them to only stake their coins when they are active. Sharding will allow transaction to be so fast and cheap, that Nodes will actively staking in and out of the system, as their Service Node go on and off.

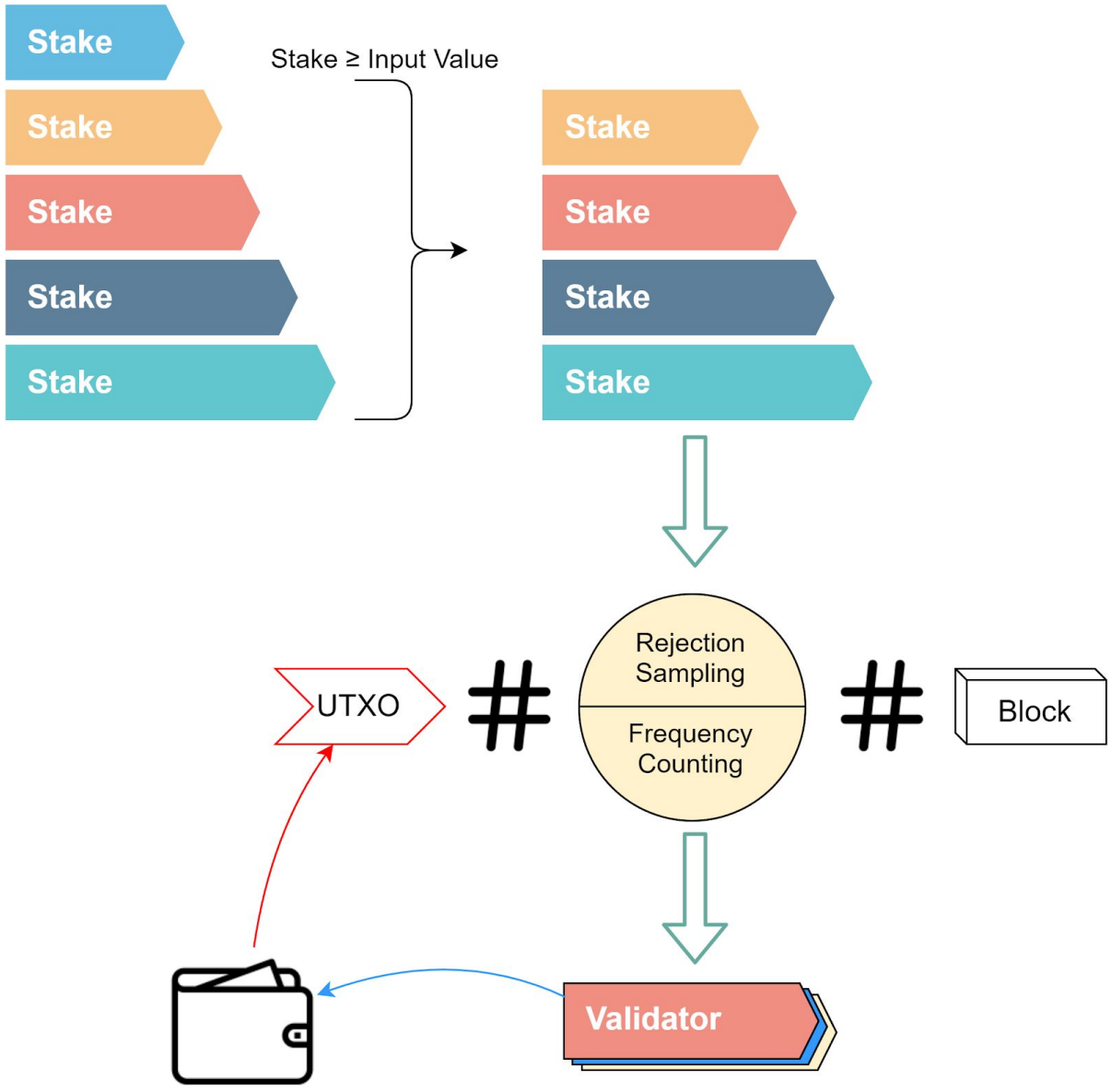
Transaction Input Locking

Each UTXO can only be locked by a specific set of Validators, and this set can be calculated by every participants of the network. For security and decentralization, an Input-Validator Random Sampling mechanism is introduced with the following desirable properties:

- Consistency: each UTXO input should map to its own Validators set, and this set is publicity accessible by anyone.
- Non-predeterministic: each newly mined block will randomize the set, so no one can pre-calculate them to coordinate an attack.

- Persistence (or Auditability): data required to calculate the set should be found on-chain, so every UTXO in the block history can always be re-validated to prevent history forging.

Input-Validator Random Sampling



The random sampling process works as follow:

1. All validators' Public-Key are fetched from the chain, (since each Staking Node has to stake a portion of YggCoin on-chain with a special transaction).
2. Only validators with stake no less than input value are qualified.
3. Using Rejection Sampling or Frequency Counting algorithm on the Frequency list, with the input UTXO and current Block Hash as the seed, a Quorum of 5 Validators are selected for the job.

The Number of Locking Validators (5) in a Quorum is subject to change in the development process, depend of the availability of the Staking Validators. Larger Quorum provides higher level of Fault Tolerant with the cost of higher network footprint.

This random sampling process is repeated 1 more times, with the previous block's hash replaced in step #3, to make sure the input UTXO cannot be double-spent in 3 blocks time around the current block. The Number of Locking Blocks (3) is also subject to change in the development process, depend on the security demand of the system. Larger value provides more security level with the cost of higher network footprint.

Locking Process

Since not every Validator is always online to serve, and some of them might even be not honest, Input Locking should be Byzantine Fault Tolerant and working under a certain level of availability. The locking process works as follow:

1. Up to 5 lock requests are sent to the selected Quorum.
2. If there are at least 3 Positive Locks are returned, the input is successfully locked.
3. Client will collect all Positive Lock signatures, along with Positive Lock signatures of other inputs, create a Locked Transaction and send it to the Mining Network.

Each Validators with a Positive Lock will assume the Lock is success, and reject any subsequence Lock attempt on the same UTXO. The Lock can only be released after 2 blocks time.

Because the hash of the latest 2 blocks are used to select 2 sets of Validators, each Validator set is responsible for this particular UTXO in 3 blocks time. This makes sure no double-spend attack can happen in the time window of 3 blocks, *none that cannot be punished*.

Double-spent Attack Penalty

This part only briefly discusses about one of the most common attack vector to demonstrate how the collateral YggCoin of Staking Service system works to punish the node with the bad behaviour detected. Full list of attack vectors and YggChain protection against them will be detailed in a separated section. [TODO: insert ref here]

In case of double spent attack detected, where two or more conflicting transactions are signed by the same validators. The following actions will be taken place:

1. Proof of Bad Conduct (all conflicting signatures) will be included on-chain.
2. All conflicting outputs will be paid up to full value, to the best effort. Inputs will be taken from: all available input of the transaction remitters, then all freezed collateral YggCoin (after an auction for YDR if necessary). The rest of the freezed collateral will be destroyed. [TBD: should they be kept for later victims?]

Because an input can only be locked by Validators with stake no less than the input value, there will always be more collateralized stake to pay for the double-spend punishment. This is not true for triple-spend attacks or more, if they can be pulled off.

Transaction Overriding

Unlike in Bitcoin, transactions in YggChain's block has different levels of finality. The following list provides different transaction types and their finality:

- Locked Transaction (LTx): highest finality, instant confirmation, pledged by it's Validators. Cannot be double-spent without Validator being punished, the stake of cheating Validators will be used to pay the victims, up to full transaction value.
- Unlockable Transaction (ULTx): transaction with input so large, no Validator in the chain can lock it. Medium finality, same level with Bitcoin transaction, no different.
- Open Transaction (OTx): low finality, requires at least 2 confirmations before its output can be spent. OTx can be overridden by a conflicting LTx before its 3th confirmation, effectively nullify it.
- Cross-chain and cross-shard Transaction (XTx): very low finality, can be overridden when the counterparty chain/shard get reverted. XTx has to wait for several confirmations before its output can be spent.

Instead of letting higher finality events revert the whole chain, (wasting time and effort of many participants and disrupting the service), transaction overriding allows low-finality transactions to gain more confirmations while waiting for its spending condition. Until the spending condition is reached, OTx and XTx's output cannot be spent, and can always be overridden by other higher finality events.

The main chain is the heaviest locked chain, which difficulty weight is less favorable for OTx. This prevent the pure 51% PoW attack, because the weight count less for OTx.

Open Transaction (OTx)

Input locking mechanism provides instant confirmation and consistency to the chain, not without a cost. The feature prioritizes Consistency and Decentralization over Availability, since there is always chance that half of the selected Validator Quorum are offline, or not being honest. Then the input UTXO cannot be locked for at least 2 more blocks. In that case, users have 2 options:

- Try another input, if they're not in extremely bad luck, they will eventually find other lockable input for replacement.
- If no other input can be locked, he can send the transaction with partial lock or none at all. That transaction when included in a block is called "Open Transaction" or OTx.

OTx is low-finality, requires at least 2 confirmations before its output can be spent. OTx can be overridden by a conflicting LTx before its 3th confirmation, effectively nullify it.

OTx can have partial lock or none at all. From the protocol view, partial lock is no different than having no lock. But partial lock can provide more confidence and credibility to transaction recipient, that the transaction sender did actually try to lock it first.

Centralization-Resistant Proof of Work

(ProgPOW)

Scalability

In order to be the currency of everyday transaction, YggChain should scale, to any level of adaptation. Scalability is the biggest issue with Bitcoin, Ethereum and all other blockchains. A lot of research and development is ongoing with this issue, most notable projects including:

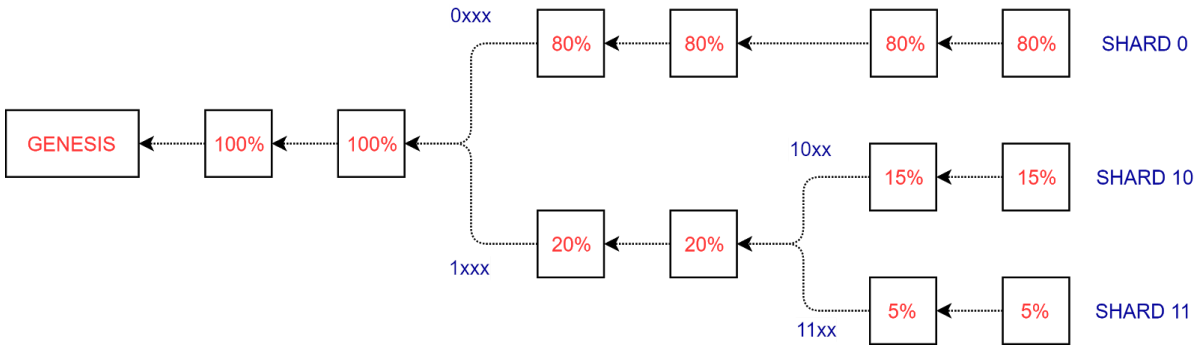
- State Sharding for Ethereum's PoS Consensus by Vitalik.
- Load Balancing of Zilliqa. (They call it 'Transaction Sharding', but it conflicts with the original meaning of database sharding.)

They are all advanced and promising, with their own challenges. YggChain comes with its own Sharding mechanism: Yggdrasil - an economic driven blockchain sharding protocol.

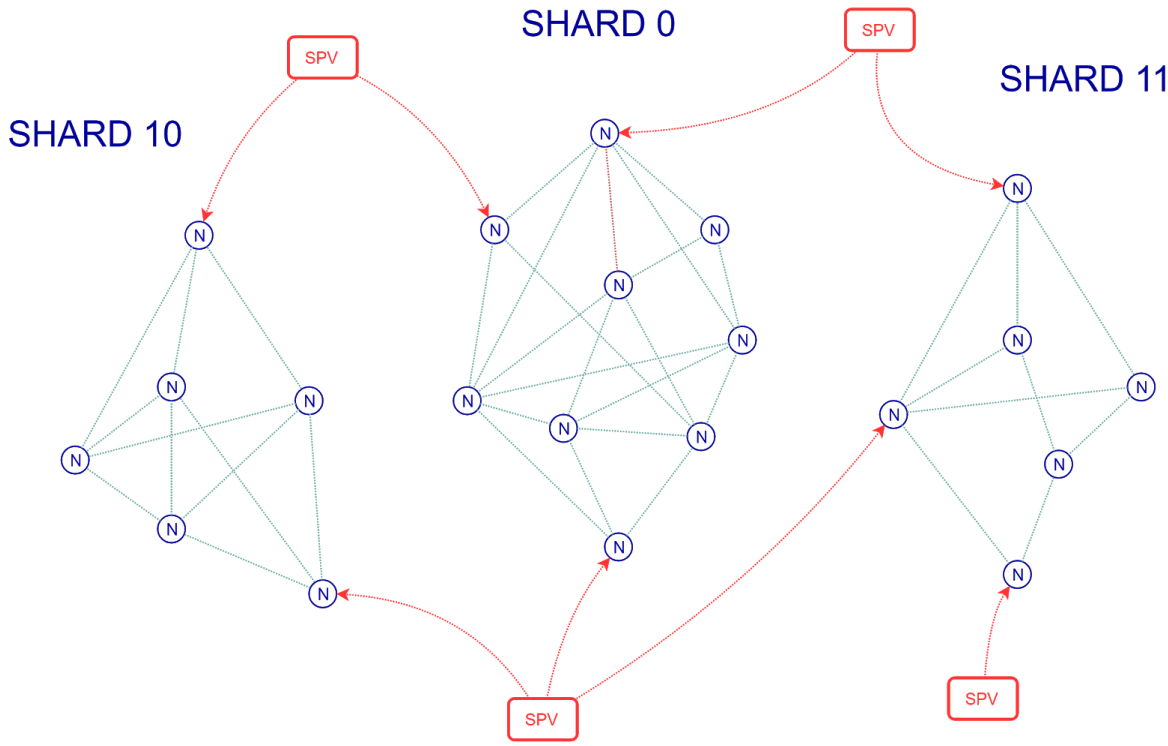
Horizontal Scale: Yggdrasil Sharding Protocol

Yggdrasil shards the YggChain by splitting it into 2 side-chains called high shard and low shard with the following properties:

- Transaction throughputs should be equal between two shards.
- All the transactions with higher value reside in the high shard, the rest stays in the low shard.
- Tokens can be transferred between shards, with normal transaction fee. This will keep the token price the same in all shards.
- Full-client only works on 1 shard at a time only. But SPVs and wallets connect to full-clients of as many shards as necessary.
- The block mining reward, difficulty and Service Node requirements will be split proportional with total value of each shard's transaction set. The total block reward of all shards is always 1.00 YggCoin.



The chain will eventually grow into a tree, where each branch (or shard) independently works on a subset of UTXO from its transactions. Each address can have UTXOs in many shards. This removes the responsibility of managing shards off the network, to the user and client softwares which have the best incentive to secure their own money. See Security for wallet shard management.



Sharding Strategy

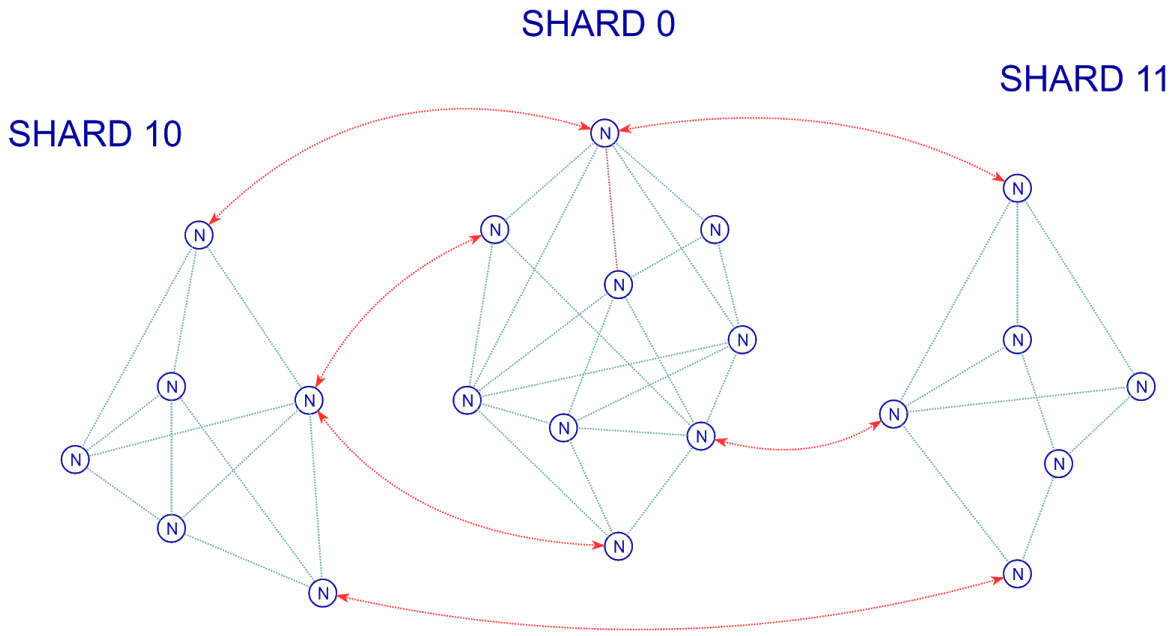
To protect the benefit of both user, Service Nodes and Miners, a shard is split only when a specific threshold of block saturation is reached. Splitting too frequent will create too many unsaturated shards, with transaction fee too low to properly incentive the Service Nodes network. Splitting too infrequent will result in oversaturated shards with transaction fee too high and drive the user always.

A shard is fully saturated, is when all the blocks in an epoch (1 whole calendar week) reach maximum block size. Only transactions with non-zero fee count. A shard will be split when it reaches 90% saturation. (This threshold is subject to change in the development process.)

Cross-shard Communication

Once split, (even though it's technically possible,) shards (or branchs) will never be merged. Most of everyday transactions does not require communication between shards, but occasionally some does. Cross-shard communication is performed by one full-client opening a connection to another full-client of a different shard. This connection will provide the state of transactions from other shard, so full-client can perform the following cross-shard operations:

- Cross-shard transfer.
- Cross-shard 2-ways transactions, or swap.



In a non-finality consensus, all states of confirmation can be reverted. YggChain's Lock & Block consensus is PoS/PoW hybrid, which is non-finality, a locked or blocked transactions can always be reverted when the longest chain has a conflict detected. When a cross-shard operation conflict is detected, transactions from higher order shard takes priority, and the conflicted transactions from lower order shard will be reverted, no matter how long the reverted chain is.

In a pure PoW system, reverting a long chain might sounds really bad, but with the Lock & Block protocol, only the double-spent transactions would be affected on the event of chain reversal. Non-conflicted transactions in a reverted block will return to locked state, and will eventually be included in one of the next blocks.

Security

The security of Yggdrasil sharding protocol rely on its economic-driven property. After a shard split, transaction throughput should be equal between the 2 new shards. Let's say, the high shard has 80% of original shard's YDR value, and the low shard got 20%. The new high shard will have the following properties:

- 80% of YDR value.
- 80% of YggCoin value.
- 80% of mining reward.
- 80% of mining difficulty.
- 80% of Service Node requirements.

The low shard obviously has everything at 20%. This will naturally split the mining power and Service Node network at exactly the same percentage (80/20) because anything else, is economically inefficient for all participants. [TODO: insert math proof here]

This resource split mechanism makes sure that, high value transactions are protected by more mining power and shares. (Transactions worth of 80,000 XDR should be 3 times more secure than 20,000 XDR transactions.) This efficiently protected the network from 51% attack and shard-hopping attack, which is a major challenges of sharding in blockchain.

In pure-technical sharding schemes, transactions are usually split randomly, while mining power and minting stake is split evenly between shards. This allows high value transactions can occur in all shards, while security of each shard is divided. Yggdrasil keeps all the high value transactions in one shard, with higher security, while letting all low value transactions in the other shard, with less security. Any adversary attempts to attack on either shards, should face the same cost versus benefit problem. It's easier to attack the lower shard, but also less worthy.

The protocol itself does not force the transaction value limit on each shard. User can still receive transactions with high value on low shard (for lower fee or for bad intention). But doing so, he or she is risking his own money to get double-spent, or get reverted. It is user responsibility to only accept high value payment in high order shard, and reject ones in low order shard. Every wallet applications should perform this check, and alert its user when there's such a suspicious incoming transaction.

Centralization-Proofness

Sharding (along with ASICS-resistant algorithm) also prevents the centralization of control over the network. By splitting the mining reward and Service Node requirements, more participants can join the network to provide their service. Miner with less powerful rig can mine in lower shard, for smaller, but steadier price. The same with Service Node requirements, owning even a small amount of YggCoin can still allow one to run a Service Node in low order shard.

Vertical Scale: Block Pruning

The Market

YggAuction

YggAuction is an off-chain auction protocol. YggAuction's features:

- Single Phase: one block finality.
- Fair: anyone can participate, and the highest bidders will get the trade.
- Cheat-proof: it's only against the validator benefit if they cheat.

To provide fairness, validators compete for the right (and prize) to the trade by submit their auction result. The one with the highest bidder will be include in the block. That means if a bidder bids for the best price, her bid will almost certainly be included in the next block.

Another reason for validator to always submit the highest bidder, because they own some amount of YggCoin themself, lowering the YggCoin price would only hurt their own capital.

[TODO: continue]

Governance

Compare To Other Projects

Economic

Non-Stable Coins

Along with other stablecoins, YggChain solves the volatility of the cryptocurrency price, to attract usage from regular everyday users. While non-stable coins still serve their own purpose, whether as an investment asset or distributed computing fee, they have been proved that not suitable for everyday exchange transactions.

Cryptocurrencies

Bitcoin and other altcoins are excellent as investment assets, like gold or stock. But with the volatility of their price, they can never be able to replace fiat money, which is what stablecoins like YggChain trying to be.

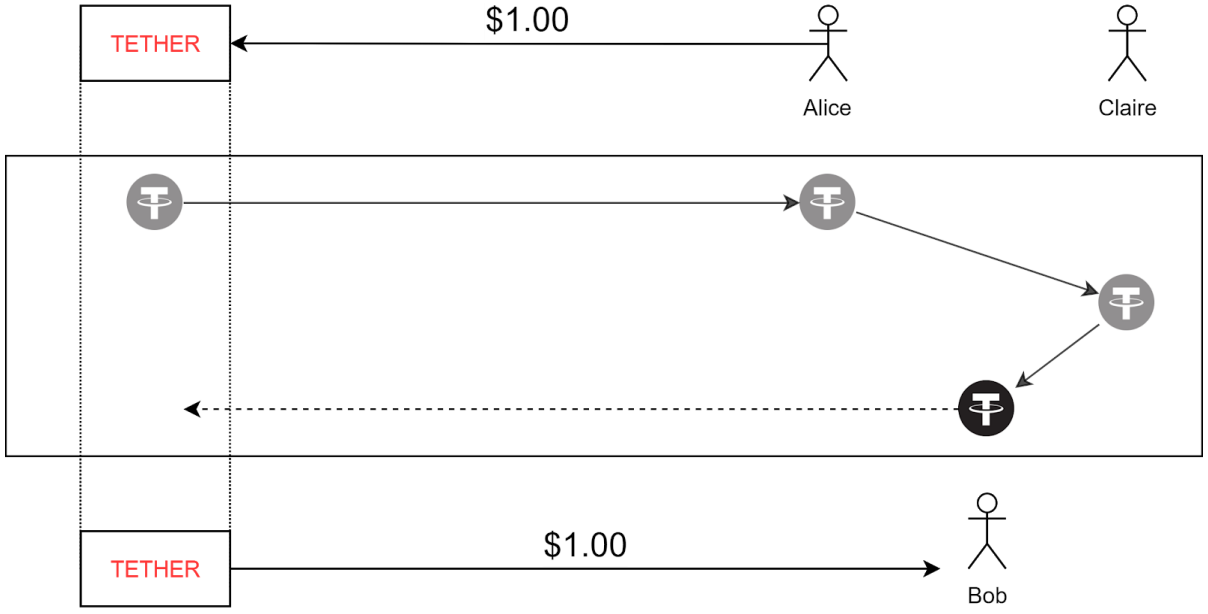
Distributed Computing Platforms

Ethereum and other smart contract platforms run distributed applications, often for a fee. This fee serves as a mean to prevent DOS attack and an incentive for nodes to run the platform. The service these platform provides are not cryptocurrency, but computing power, so the volatility of the fee is not much of a problem.

Stable Coins

Centralized IOU Issuance

Tether and Digix issue IOUs, promise to hold assets in a bank account or vault and issues tokens that represent a claim on the underlying assets. The digital token has value because it represents a claim on another asset with some defined value.

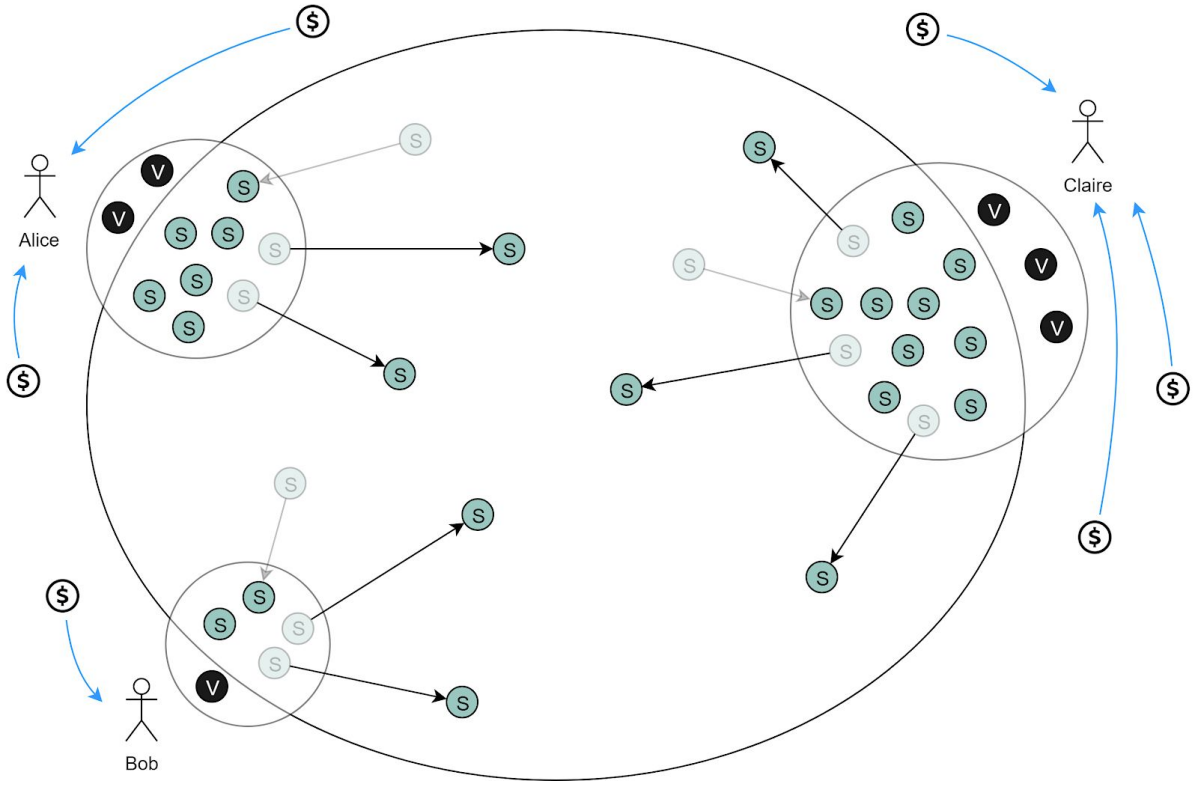


There are 2 main problems of this model:

- It is centralized and it requires trust in the issuing party - that they actually own the assets being represented and that they are willing to honor the IOUs. This model imposes serious counterparty risk on holders of the token. Tether is the canonical example given the serious concerns that the public has about their solvency and legitimacy.
- It is backed by non-crypto currencies, hence depends on those currencies, which make it never be able to replace them. In fact, their purpose is never to replace fiat money, nor provide an everyday-use currency. Their best use is to (short-termly) replace fiat money with crypto asset for convenient crypto trading.

Decentralized Collateral Backed

Bitshares, MakerDAO and the likes try to solve the centralized problem of Tether by decentralize the collateral holder agents. This approach allows users to create stablecoins by locking up collateral in excess of the amount of stablecoins created, either by its own on-chain crypto token (called shares) or by other existing token on sidechains.



The first major problem, of course, is that the collateral backing the stablecoin is often a volatile crypto-asset such as BTS or ETH. If the value of this asset drops too quickly, the stablecoins issued could become undercollateralized, then the best strategy for the issuer is to abandon the locked crypto-asset, and keep the money previously exchanged for issued stablecoins for their own benefit. By doing so, the whole stablecoins system would be flushed down to a spiral of death. For this reason, most of the projects using this model require that the stablecoins be overcollateralized enough to protect against sharp price movements. While this can provide some degree of certainty, there always exists the possibility of a black swan event that causes collateral prices to drop so quick that the stablecoins are undercollateralized.

The second one, is its economy (or rather monetary) efficiency. For each value of stablecoins in circulation, requires at least double of that value in collateral asset need to be locked up. Often, it's much more than double to be overcollateralized enough to protect against sharp price movements, and anything less than double, is playing margin again the stablecoins holders. This provides a banking service, where the bank need to keep gold in reserve with double the value the money in circulation. In reality, most countries worldwide adopts fractional-reserve banking, in which only a fraction of bank deposits are backed by actual cash-on-hand and are available for withdrawal. This allow central bank to expand credit and money supply beyond the amount of the underlying reserves of base money originally created by it. The fraction is about 1% to 30% depend on the country regulation; the more stable the economic is, the smaller fraction it is required. Overcollateralization might sounds attractive for short-term loaner and borrower (just like with Tether), but in the long term, this is a huge limitation, which prevent this model to replace the current fiat money.

Seigniorage Shares

Stablecoins based on Seigniorage Shares paper follow another principle, the core idea is to using another token (called *vol-coin* by Vitalik and *share* by most projects) to absorb the volatility of the stablecoin, but each project comes with different approaches. This section will discuss how each of them is different from YggChain.

Basis (former Basecoin)

How YggChain is different than Basis (more detail analytic below):

- Diverted from Seigniorage Shares paper, Basis adds the 3rd token (Bond) to absorb all the risky volatility for Basis, while keeping the Share token safe and even more rewarded. Basis bond have expiration date making it is the most risky investment, thus defeat the contraction purpose of the token. Basis bond is even set a price limit, which goes against the free market rule, it likes forbid everyone to trade BTC with the price lower than a specific value, like $5k! YggChain otherwise, sticks close to the Seigniorage Shares paper with 2 tokens dynamic, using the YggCoin token alone to absorb all price volatility (both risk and reward) of YDR.
- In expansion process, Basis (along with Fragments and Carbon) distributes new stablecoins to each of token holders, pro-rata. This has 3 main problems:
    - Sleeper Supply: the new coins supply is distributed to all shareholders, whether they want them or not. [TODO: insert data statistic here] Most of the holders keep their shares for long or short term investment, they usually not active when the expansion occur; new coins distributed to them stays inactive in their wallet for sometimes, effectively not in circulation yet. Those new coins then fail their immediate purpose, to provide supply for the market, but instead stays inactive for indefinite time, and then become active later, tipping the balance to the other side of the market force. While the sleeper coins are still sleeping, coin demand are still present, new expansion will happen, causing more sleeper coins. This unwanted effect prevents the market to quickly stabilize the price, and creating more volatility of its own, even black swan event from its own mechanism.
    - Fragmented and dust: beside the top holders, small shareholders will get very small portions of the token, which is unusable due to the network fee.
    - Technical inefficiency: all shareholders will have 1 transaction when the expansion occur, which is half of the time. This will put an ernomost strain to the network, with transactions not directly serving the end users.

  YggCoin, otherwise using off-chain auction protocol (named YggAuction) to sell the newly created YDR (stablecoin) for YggCoin. This has the following advantages over pro-rata distribution:
    - Active Expansion Supply: newly created YDR is given to the active shareholders, who participated and won the auction. New active token is essential to provide supply to the current stablecoins demanding network.
    - No fragment, no dust.

- Technical Efficiency: YggAuction is a fast, fair and cheat-proof off-chain auction protocol, will be the main way for YggChain's YDR contraction and expansion.

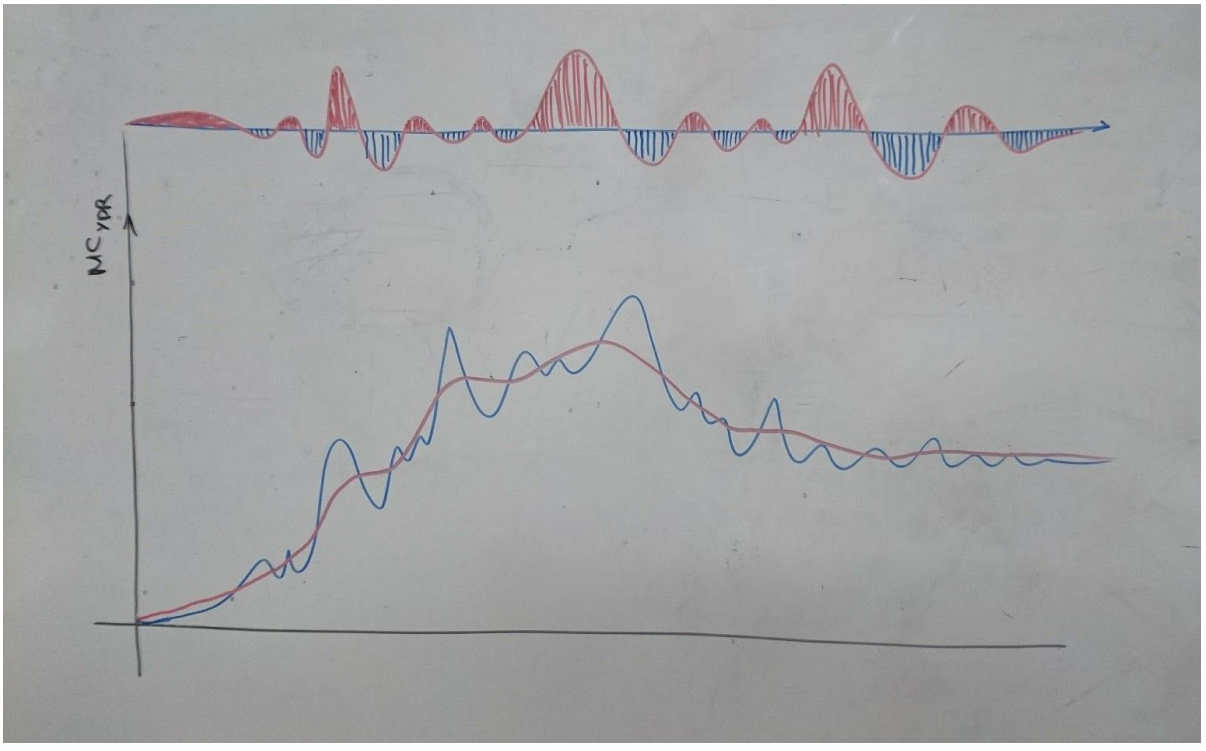
Detail Analytic of Basis

The idea of Basis is rather strange, instead of using the Share token to absorb the price volatility, a third token called Bond is used for that purpose.

There is one important note here, that although sharing the same name, Basis' Bond is very different with financial bond.

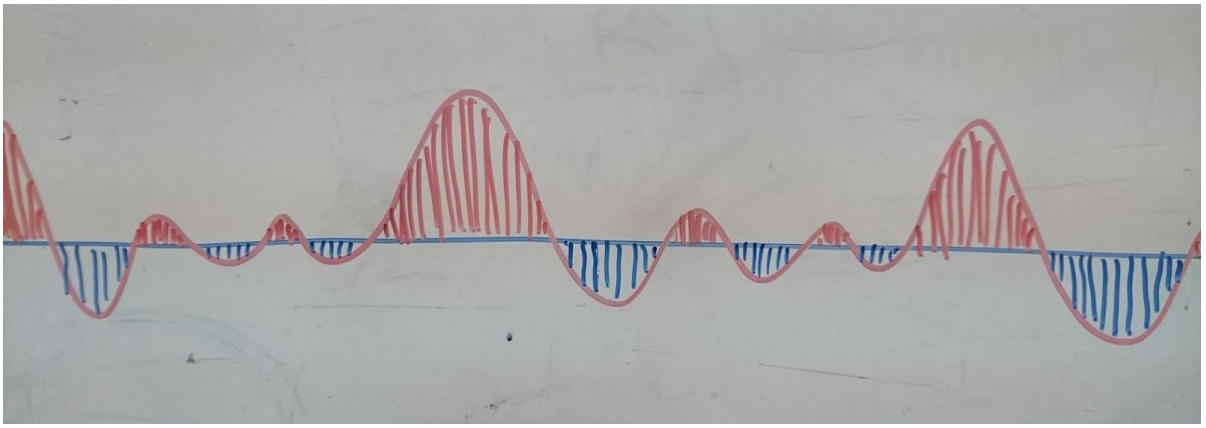
- Financial bond has maturity date, where the bond issuer (government or corporation) is required to repay bond holder. This feature makes financial bond is one of the safest investment, which is important for the money contraction and expansion role.
- On the other hand, Basis' bond does not have maturity date, but expired date instead, where it is disappeared into thin air, making it is one of the riskiest investment.

This feature of Basis shifts all the risky volatility to its Bond token, while keeping all the rewarding volatility to it Share token. When the market interest is increasing, all the new Basis token is distributed to Share token holder, pro-rata; and when the market interest is decreasing for a long time, without recovering quick enough (like after the all-high hype), all the Bond will be expired, take away all the investment in vain.



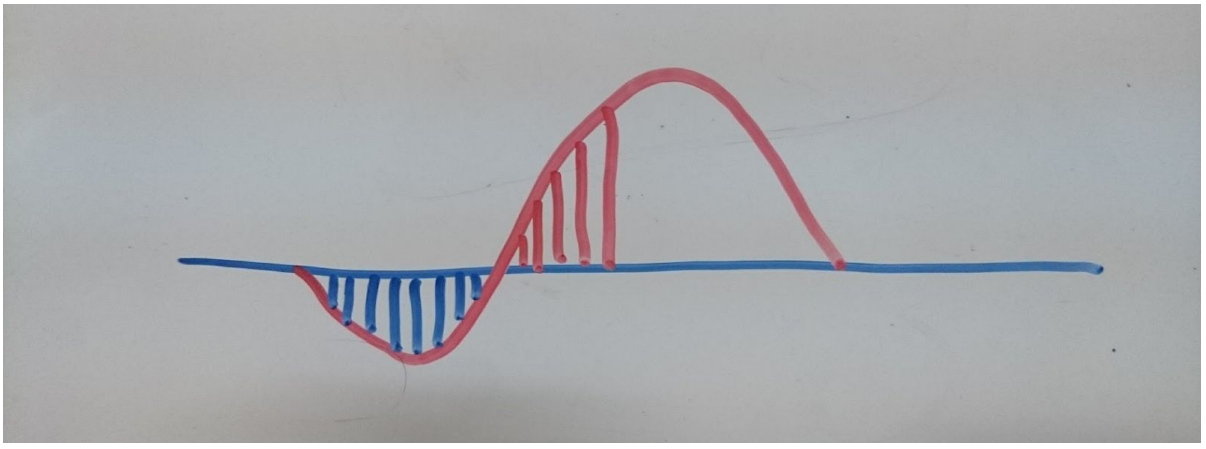
Red graph is the demanding market capacity and blue one is the current market cap of Basis token. The current MC always swings around the demanding MC as the price of the token is

stabilized. When the red graph is above the blue one, the market is expanding (because demand is higher than supply), and vice versa. Straightening the current market cap, we have the demanding market cap graph:

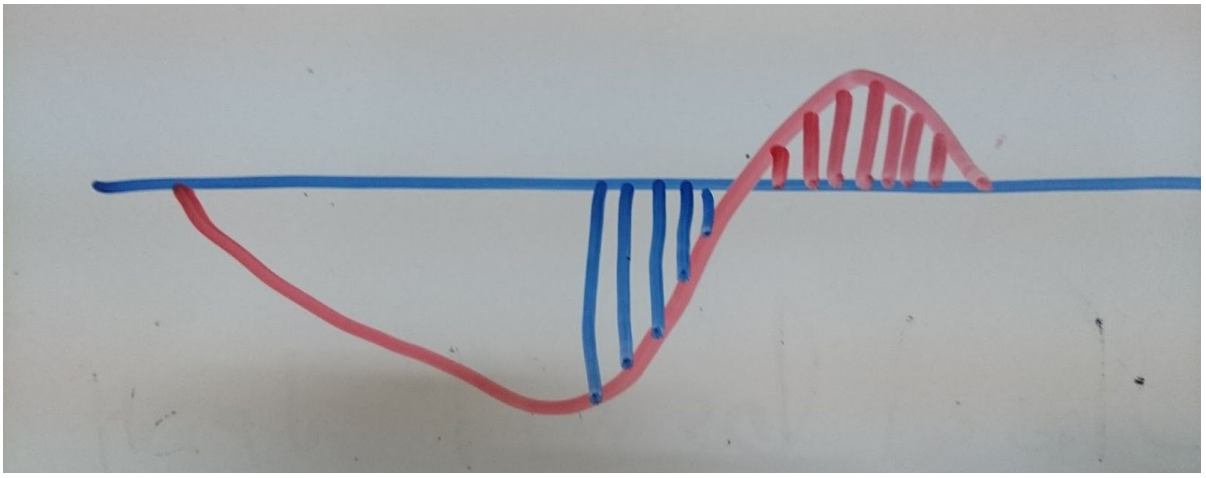


Red area is the total value of token expansion, and blue area is the total value of token contraction.

In a time window (equal bound expiration) where the expansion is larger than contraction value (the market is overall expanding), all the contracted value (blue) is repaid by a part of expansion (stripped red). The rest of the expansion value (empty red) will be distributed to the Share's holder, pro-rata. This makes the Share token a low risk - high reward asset.



And in a time window (bound expiration) where the expansion is smaller than contraction value (the market is overall contracting), only a part of contracted value (blue) is repaid by the expansion (red). The rest of the contraction value (empty blue) will be lost because of bond expiration. This make the Bond token an extremely high risk asset.



Because the risk is high, the reward should be also high, otherwise, buyer could just invest in Share instead, which is already low risk - high reward by definition. Firstly, Basis' bond is always repaid in 1.00 Basis (or 1.00 USD), so the bond price (driven by the market force) should be very low for the risk-worthy reward. Secondly, to prevent spiral of death causing by black swan events, Basis set the lower price limit for Bond in auction. That makes things worse, it likes setting the lower limit for Bitcoin or stock price, when the market force drives the price down lower than the limit, no one can buy bond with a price lower than the limit; but no one would buy anything higher than its expected value, so no bound can be sold at that moment. That is where the purpose of bond (to contract stablecoins supply) is failed.

Fragments

Fragments project is another take on the Seigniorage Shares paper, also using bond token, but with the following distinctions:

- Using ETH as the Reserve Collateral Asset instead of its own (share) token. This makes the currency depends on the ETH, which has its own pros and cons, but not independent nonetheless. The main problem of using side token is, there's no incentive to hold them for reserve. Fragments development team could hold reserving ETH to bootstrap their project, but after a wider adoption, market cap of Fragments would be increased to the point where the volatility can no longer be absorbed by a single party's capital. Then, the stablecoin will solely rely on the secondary mechanism - bond.
- Fragments' bond is designed to be a secondary stabilizing mechanism (after ETH reserve), but (as described above) it will eventually be the main mechanism once the system is well adopted. Currently, there is not much detail of bond in the Fragment document.

How YggChain is different from Fragments:

- Fragment's expansion mechanism distributes newly created Fragments to all Fragment holders, pro-rata. This increase the total value in each user wallet, because there's no token to absorb the positive volatility; Bond only absorb the expansion of previously contraction. [TODO: insert diagram here] This benefits the early users of the Fragments stable-coins, while take away the reward of high-risk bond trader. This

approach is exactly what Seigniorage Shares paper tell us not to do, in section “How not to distribute ∆i” of the paper. YggChain otherwise, clearly separates high-risk-high-reward YggCoin and no-risk-no-reward stablecoin (YDR) for their intended users.

- Fragments share the same problems of distribution new token pro-rata as Basis (see above).

Carbon

Of all the Seigniorage Shares stablecoins, Carbon is different from YggChain the least.
Those different includes:

- Distribution new tokens pro-rata: see above for problems.
- Powered by Hedera Hashgraph, a permissioned DLT, and not a blockchain. YggChain otherwise, is an public permissionless blockchain, where everyone can join to use, trade, develop and even fork their own project if they no longer agree with our development direction. We believe public permissionless blockchain project will lead to the ultimate future of cryptocurrency of free world.