Genshiro Introduction

Equilibrium brings its technology to Kusama. The project name for Kusama is Genshiro. Kusama positions itself as Polkadot’s sister network where all of the field testing of exciting new technology happens. Genshiro will follow this approach and will launch its product on Kusama with the aim to battle test its technology by introducing a much wider range of assets than initially supported in polkadot, lowering critical collateralization requirements to 100%.

Further, Genshiro will roll out the DEX and introduce a wide range of derivative assets based on traditional financial assets (e.g users will be able to trade derivative products such as perpetual swaps and/or future contracts on commodities, stocks, ETF’s, e.t.c.).

Last but not least, Genshiro will aim to fully decentralize the Kusama based network right from the go by cleverly allocating it’s utility tokens to the Kusama community.

Below is a quick overview of the core components of the Genshiro protocol.

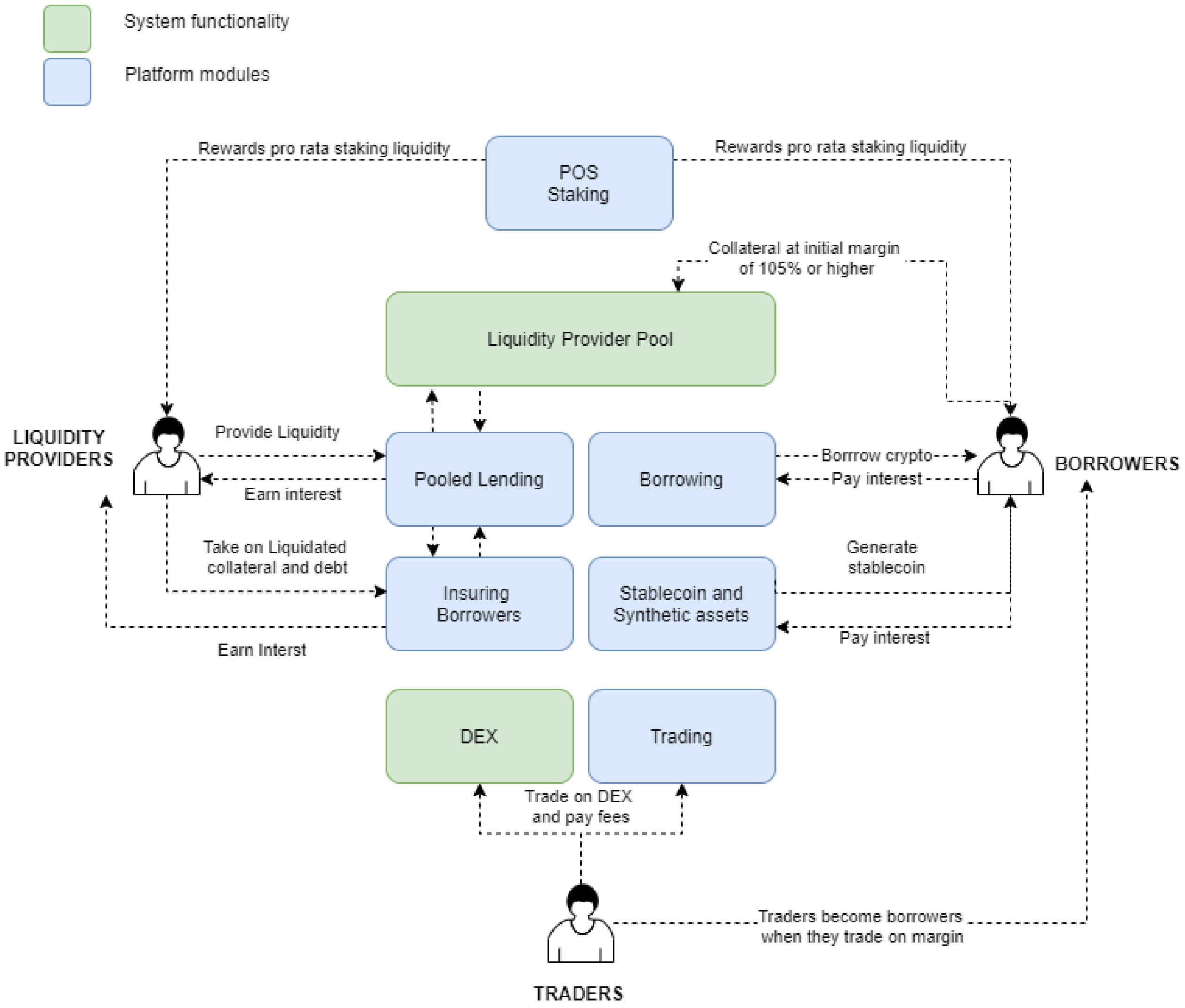
Overview

We propose a novel approach to the collateralized stablecoins and lending facility backed by a portfolio of crypto assets.

The stablecoin will be a multiple-collateral stable unit of account pegged to USD. Stability comes from overcollateralization of borrower positions. We utilize novel approaches to on-chain pricing (interest rate calculation) and risk calculations (determination of the health of the entire system), which distinguishes our approach in several ways from stablecoins like DAI:

1. There are no arbitrary governance-set interest rates, they are determined by a borrower’s portfolio, borrower debt amount, overall system liquidity, and price risk.
2. There are no arbitrary set LTV requirements. The system makes sure every position remains solvent at a 100% collateralization ratio.
3. There are no arbitrary set liquidation penalties and no hidden fees when borrowers default. There is no need for keepers (those who participate in liquidation auctions).
4. By design there are always two sides to the system: liquidity providers on one side and borrowers on the other side. Borrowers pay fees to liquidity providers when borrowing crypto assets/generating stablecoins. Liquidity providers in turn bear liquidation risk of borrowers.
5. There is a generic assets module which supports double entry book keeping (separation of assets and liabilities) and tracks portfolios of user assets. There is no such thing as position with 1 collateral, we work with portfolios.
6. System design allows for building a spot trading DEX with desired levels of leverage.

Further, the business processes behind the system may be explained via following high level overview chart:



Risk model

Decentralized stablecoins will maintain stable value if borrowers have either excess collateral or if the liquidity provider pool is sufficiently capitalized. We assess system risk using the credit risk model. We use a structural model approach, which assumes that a borrower default can be explained by a specific trigger point. For example, it can be caused by a decrease in asset value below some threshold (like the value of the debt). The value of assets itself is modelled as a stochastic process.

The initial assumption is that asset prices are distributed log-normally, and risks are calculated from there. Yet, crypto prices are anything but log-normal, fortunately our pricing model is easily extendable to more sophisticated models like the jump diffusion or hyper exponential jump diffusion, as will be shown further.

Here's a detailed algorithm of how we assess the overall system risk:

1. Calculate value of borrower collateral, borrower debt and liquidity provider pools in normal market conditions.
2. Calculate risks of these pools (measured as portfolio volatility).
3. Calculate discounts (Value at Risk) for each pool given confidence value and pool's volatility.
4. Calculate pool values in stressed market conditions using discounts.
5. Calculate insufficient collateral:
   insufficient_collateral = max[0, debt_value * (1 + debt_discount) - collateral_value *(1 - collateral_discount)]
6. Compare insufficient collateral with the stressed value of the liquidity provider pool to calculate the risk scale:

   scale = (insufficient_collateral / stressed liquidity provider pool value)^alpha
   scale is initially bounded between 0.5 and 2. Alpha is a parameter that drives the steepness of the scale curve.

Pricing model

Output of the risk model (scale) serves as a parameter scaling input to the pricing model. This is a feedback loop we're looking for: If there is more insufficient collateral than there is stressed liquidity provider liquidity, the scale is > 1 and the borrowers' interest rates scale higher. If there is excess liquidity provider liquidity, the scale is < 1 and borrower interest rates scale lower. This market mechanic is designed to achieve equilibrium in system solvency. This is market mechanics that assures enough capital in the liquidity provider pool to cover for borrower losses.

To price borrower loans we turn to research from traditional finance.The problem of pricing a collateralized loan comes from the realms of traditional finance and stock loans. Associated research on this was pioneered by Xia and Zhou (2007). Under the Black–Scholes model, they derived a closed-form pricing formula for the infinite-maturity stock loan by solving the related optimal stopping problem. We adapt the approach proposed by Xia and Zhou, and come up with an elegant pricing solution that depends on borrower portfolio risk and the level of portfolio collateralization. We scale the formula to account for excessive crypto volatility, making interest rate numbers plausible.

The table shows approximate interest rate breakdown for a borrower portfolio, given its collateralization ratio and the ratio of dollar liquidity in liquidity providers and collateral pools.

Collateralization	**Liquidity pool / Collateral pool ratio**					
	0	0.4	0.8	1.2	1.6	2
300.00%	14.69%	1.86%	1.86%	1.86%	1.86%	1.86%
250.00%	16.21%	1.95%	1.95%	1.95%	1.95%	1.95%
200.00%	19.25%	3.36%	2.14%	2.14%	2.14%	2.14%
150.00%	28.38%	14.99%	6.30%	4.01%	3.01%	2.71%
120.00%	55.75%	55.75%	21.92%	12.86%	8.93%	6.80%

Table.1. Interest fee based on a 5% daily volatility assumption.

Governance

There are several system parameters that are subject to change via decentralized governance. The list might grow / change as the solution develops and matures. We propose to govern the system by staking special purpose utility token and voting for system parameters.

parameter	**description**
Critical margin	The absolute minimum below which liquidations of borrower positions are triggered. By default = 100%
Initial margin	The absolute minimum below which no collateralized borrowing may happen. It governs the borrower’s maximum leverage. By default = 105%
Maintenance margin	The minimum margin where maintenance margincall happens - borrower has some time to top up his balance to or above initial margin, or otherwise get liquidated. By default = 102.5%
Liquidity target	Shows what fraction of stressed system losses should be held in aggregate in the liquidity provider pool. It governs the bankruptcy risk. By default = 1, meaning that in a stressed

	market scenario there should be enough liquidity in the liquidity provider pool to cover all of the collateral pool (borrower) losses.
Scale bounds	Risk model volatility scaling bounds. By default lower bound = 0.5 and upper bound = 2
Fee reserve weight	Shows what fraction of borrower-generated fees go into the stability fund. This is needed for covering tail risk and possible system default.
Price data parameters	Number of points and frequency to consider in statistical calculations.
Asset discounts and/or VAR confidence	Parameters used to stress-test collateral and debt portfolios for upside and downside price risks.
Borrowing limits	Limits how much of the bailout pool will be available for borrowing assets.
Supply caps	Defines the maximum allowable collateral asset concentration across the entire system.

Governance mechanics

1. Users propose referenda for voting.
2. Users must lock/stake their tokens to vote on a referendum.
3. The longer users lock their tokens the more weight they are assigned. Voting without a lock slashes users' voting power to some fraction of their token amount. Each doubling of the lock time increases the power of a user's vote, all the way up to six times the account's balance (which would be a lock of 32 enactment periods).
4. There will be a notion of Council - a list of 12 members which gets periodically elected and represents passive token holders.
5. Adaptive quorum biasing tunes the passing threshold according to the turnout in a referendum. For public proposals, the lower the stake turnout, the higher the passing threshold, favoring the status quo and preventing one large token holder from swinging a low-turnout vote. Referenda that have unanimous approval from the Council have the opposite threshold. As turnout approaches 100%, all thresholds converge to a simple majority.

Further considerations.

1. The proposed solution requires per-user calculations of interest fees and distribution quantities among liquidity providers. To organize these calculations and minimize on-chain usage of resources we will use off-chain workers
2. The proposed solution works with portfolios of crypto assets, so the success and diversity of assets largely depends on bridge availability.
3. Peg can be any currency - USD, EUR, e.t.c., or even IMF SDR, as long as we know prices of all collateral assets in reference to the peg.
4. System design allows to build DEX and allow leveraged trading both in crypto assets and other derivative products.