Kresko v1.0: A novel synthetic asset protocol with a single CDP-per-account model

Kresko Founding Team

Abstract

Accessing wealth using stocks and commodities is still a big challenge globally. Our motivation is to create a protocol that democratizes access to wealth without borders and censorship.

In this Whitepaper, we introduce Kresko, a permissionless synthetic asset protocol, that enables creation of overcollateralized synthetic assets such as stocks, commodities, and crypto. Our goal is to explain how Kresko works and highlight its key innovations in DeFi.

Kresko uses a single-CDP-per-account model with distinct risk parameters to maintain the protocol's health. The Kresko protocol introduces novel improvements over existing solutions. This includes improved capital efficiency, user experience, frictionless listing of assets, and risk management.

Table of Contents

1. Introduction

People in emerging economies lack access to wealth-building tools such as international stocks and commodities, making it difficult for individuals in places like Nigeria to invest in companies like Tesla or Apple. Traditional finance (TradFi) is exclusive due to access based on geography, high account balance requirements, high fees, and cumbersome paperwork. Decentralized finance (DeFi) is open and borderless, removing middlemen, lowering costs, and censorship-resistant.

DeFi encompasses decentralized exchanges, lending markets, stablecoins, and synthetic asset protocols, but innovation in synthetic assets has been limited.

Synthetic assets provide access to stocks and commodities, leveling the playing field by making wealth-building tools available to all. They offer several benefits such as:

- **Access:** Providing access to previously inaccessible financial tools due to geography, high cost, or cumbersome paperwork.
- **Fractional Ownership:** Synthetic assets allow purchasing of fractional assets. You have the freedom to buy $1 or $1000 worth of assets.
- **Reduced Fees:** Since the counterparty is a smart contract with no middlemen, the fees are substantially lower.
- **Censorship Resistant:** Removing reliance on third parties, you avoid censorship.
- **Flexibility:** Synthetic assets allow the conversion of one asset to another within a few seconds. Transfer of assets from one person to another is possible at the click of a button. None of which is possible with traditional stocks and commodities.

In this paper, we describe Kresko's unique approach to synthetic assets, which improves capital efficiency, debt management and antifragility compared to existing models.

2. State of the Art and Related Work

DeFi revolutionized blockchain applications by bringing practical uses that users wanted, rather than speculation on future possibilities.

This section provides a brief overview of DeFi and its relation to synthetic asset protocols.

2.1 Decentralized Exchanges

Automated market makers (AMMs) are one type of decentralized exchange (DEX), which are essential for many DeFi services.

Previously, users had to use centralized exchanges with a central limit order book (CLOB) model, which required a third party to match buyers and sellers. While decentralized exchanges such as LocalBitcoins existed, they had high friction due to their peer-to-peer model, which required buyers and sellers to be present at the same time.

Uniswap (Adams et al., 2020) introduced a new method of trading that did not require buyers and sellers to be present at the same time. Uniswap popularized AMMs on which market makers could pool assets in a smart contract. And traders could later trade 24/7/365 using the pooled assets.

2.2 Lending Protocols

Lending markets are decentralized versions of traditional money markets.

Lending markets such as Aave (Aave Protocol, 2020) and Compound (Leshner & Hayes, 2019) allow users to earn interest on their unused assets without giving up custody to a third party. Borrowers provide collateral in the form of crypto assets and pay an interest to the lenders.

Smart contracts act as market makers by pooling assets and matching lenders and borrowers asynchronously, with interest rates set algorithmically based on demand.

2.3 Stablecoins

Crypto tokens such as BTC and ETH are volatile, making them less suitable for use as a medium of exchange. Stablecoins, on the other hand, maintain stable value and are more practical for paying for goods and services.

A stablecoin is a token pegged to the value of 1 USD or another reserve asset. A good example of an overcollateralized stablecoin is DAI (The Maker Team, 2017), which users can generate in a permissionless manner by depositing collateral and borrowing against it, incurring interest charges.

In this case, users borrow from a smart contract that acts as the counterparty.

2.4 Synthetic Asset Protocols

A synthetic asset is a derivative instrument whose value tracks the price of another asset or a group of assets. While synthetic assets can track the price of any arbitrary asset, stablecoins are a subset of synthetic assets as they track the value of 1 USD.

Synthetic asset protocols such as Synthetix (*Synthetix,* 2022) and Mirror protocol (Liu & Lee, n.d.) enable the creation of synthetic stocks, commodities, and crypto assets.

In a way, synthetic asset protocols are similar to lending protocols, in that users borrow synthetic assets by depositing collateral. Borrowers can then use these assets to make markets on DEXs for fees, or take a short position by selling them on a DEX.

3. Motivation and Limitations of Existing Solutions

Synthetic asset protocols are a form of derivatives protocols, yet it is still challenging to access simple forms of derivatives such as stocks and commodities in DeFi. Existing synthetic protocols require substantial capital and technical knowledge.

Some centralized exchanges allow users to trade synthetic stocks and commodities. Centralized exchanges share the same drawbacks as traditional finance counterparts, such as censorship, cumbersome onboarding requirements, high fees, and lack of adequate insurance.

3.1 Synthetix

The Synthetix protocol (Synthetix) uses its native governance token (SNX) exclusively as collateral. However, this poses a systemic risk to the protocol as the collapse of the SNX token

would affect all synthetic tokens on the platform. Additionally, the amount of synthetic assets that can be created on the protocol is limited by the circulating SNX supply.

Minters on the protocol share a global debt pool and the protocol allows any-to-any arbitrary conversion of synthetic assets using oracle prices, enabling zero-slippage trading. However, minters take on arbitrary risk as they cannot estimate their risk and their debt can increase or decrease independent of their original minted value, based on the exchange rates and supply of Synths within the network (Synthetix, 2022).

3.2 Mirror Protocol

The Mirror protocol shares similar risks as Synthetix by using a single collateral token, UST, a stablecoin on the Terra blockchain. Later versions allowed other tokens such as LUNA and ANC as collateral, however, they were all interdependent tokens within the same Terra ecosystem, which has already proven disastrous to the protocol with the implosion of UST and the Terra ecosystem (Kelly, 2022).

The protocol valued stable assets more than volatile ones when using them as collateral to adjust risk, but did not perform similar risk adjustments for the borrowed assets. This meant that borrowing stable assets and volatile assets had the same collateral ratio requirements. For example, gold is relatively stable compared to TSLA stock, but minters could borrow those assets as though they had the same risk profile.

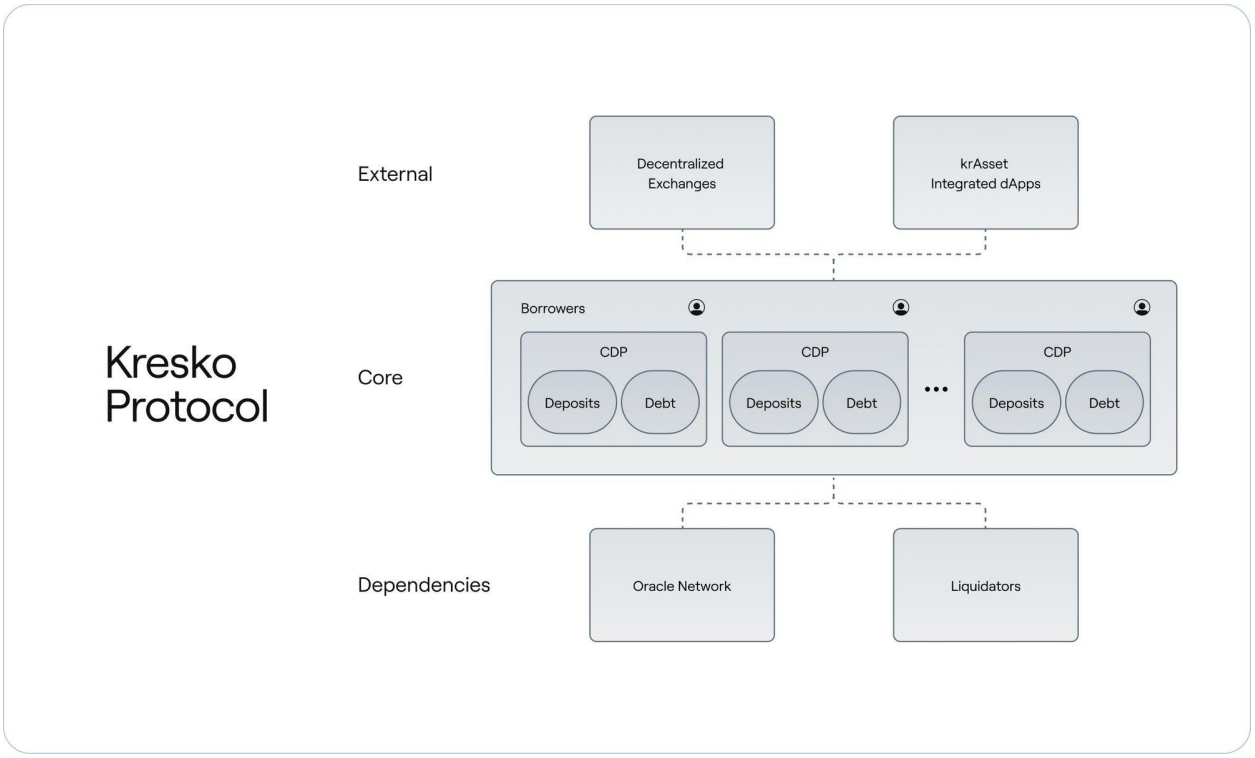
Additionally, the debt management process was laborious. If a user had many debt positions, they had to monitor each position and ensure they were sufficiently collateralized. For instance, if a user had 10 debt positions, 5 of which were at risk of being liquidated, they had to provide collateral to 5 different vaults. The higher the number of debt positions, the worse the user experience.

The protocol was also not capital efficient. Excess collateral from a non-risky position could not be automatically applied to an at-risk position. Users had to either supply new collateral to the at-risk position or manually withdraw collateral from another position and use it towards the at-risk position.

4. The Core Protocol

The Kresko synthetic asset protocol is an open derivatives protocol. Users can provide collateral and borrow synthetic assets by taking debt positions.

A synthetic asset is a derivative instrument whose value tracks the price of another asset or group of assets. Synthetic assets allow users to get exposure to an external asset without holding the underlying asset.



4.1 Protocol Actors

The following provides a list of the actors on the protocol:

- **Protocol:** A set of rules that can be implemented to create the Kresko synthetic asset protocol. The protocol is implemented using a set of smart contracts on an EVM (Ethereum Virtual Machine, n.d.) compliant blockchain. Unless otherwise noted, the protocol is the main actor taking the actions listed in the whitepaper.
- **Borrower:** An account that creates a collateralized debt position (CDP) by depositing collateral and borrowing synthetic assets from the protocol.
- **Liquidator:** An account that liquidates CDPs whose positions are below a certain collateral ratio (liquidation threshold) required by the protocol.
- **Oracle:** An entity that posts external asset prices from trusted data sources to the oracle network for consumption by Kresko protocol.
- **Trader:** An account that trades synthetic assets listed on an external DEX.

4.2 Collateralized Debt Position

A collateralized debt position (CDP) is a system introduced by the MakerDAO team with their decentralized stablecoin DAI (The Maker Team, 2017).

A CDP is an overcollateralized lending contract where borrowers deposit collateral and borrow assets against the deposited collateral. The contract itself acts as a counterparty when borrowers open a debt position.

4.2.1 Synthetic Asset Creation

The protocol mints synthetic assets when borrowers open a CDP. The protocol burns the minted assets when borrowers repay them.

The synthetic assets created on Kresko are called *krAssets*. For instance, a synthetic asset that tracks the price of Tesla with a ticker 'TSLA' is referred to as krTSLA.

4.2.2 Single CDP With Multiple Collaterals and Multiple Debt Positions

On the protocol, borrowers can create only a single CDP per account. One or more collaterals back each CDP and borrowers can take multiple debt positions on a single CDP.

4.2.3 No Global Pools

The protocol does not pool the collateral deposited by all the borrowers on the protocol, i.e., the collateral deposited in a CDP is localized only to that account and does not affect other accounts.

Similarly, the protocol does not pool the debt taken by all the borrowers on the protocol, i.e., the debt taken in a CDP is localized only to that account and does not affect other accounts.

4.3 Tokens and Assets

The term 'tokens' is used to refer to on-chain assets. The term 'assets' is used broadly to refer to both on-chain and off-chain assets. The following asset types are relevant here:

- Crypto assets such as ETH or USDC
- Synthetic or derived assets such as krETH or krTSLA
- Off-chain assets such as stocks and commodities like TSLA or IAU

The protocol allows different assets as collateral and debt.

4.3.1 Pricing Assets

The protocol prices tokens in USD. The protocol obtains the prices of the tokens using a network of oracles.

4.3.2 Collateral Types

The following provides a list of collateral asset types under consideration:

- **Native cryptocurrencies:** assets native to the blockchain ecosystem the protocol is implemented on. Example: ETH, BTC, etc.
- **Stablecoins:** assets pegged to a stable fiat currency. Example: USDC, DAI, etc.
- **Wrapped cryptocurrencies:** wrapped assets representing a non-native cryptocurrency that has been bridged. Example: wBTC, wETH, etc.
- **Liquid staked assets:** liquid staked assets. Example: stETH, etc.
- **Synthetic assets**: synthetic assets minted by the protocol could be supplied back as collateral. Example: krTSLA, krIAU, etc.

4.3.3 krAssets and Types

The following provides a list of asset types under consideration as krAssets within the protocol:

- **Stocks:** Assets traded on traditional exchanges. Example: AAPL, TSLA, etc.
- **Commodities:** Commodities such as gold, silver, and oil. Example: IAU, USO, etc.
- **Exchange-traded Funds (ETFs):** A basket of assets based on an index or another asset(s). Example: QQQ, VTI, etc.
- **Synthetic Crypto assets:** Synthetic representation of other crypto assets. Example: ETH, BTC, etc.

4.4 Collateral Deposits

The protocol requires borrowers to deposit collateral before they can borrow krAssets.

4.4.1 Collateral Factor

The protocol values each collateral differently based on its risk profile. Generally, the higher the volatility of an asset, the higher the risk.

Collateral factor (CF) is a fraction between 0 and 1 that is used to calculate the risk-adjusted valuation of deposited collateral.

Given an asset *a*, CF can be represented as follows:

$$CF_a = [0, 1]$$

4.4.2 Collateral Deposit Value

The quantity, price, and the collateral factor are used to determine the *deposit value (v)* of a collateral. Deposit value enables the protocol to properly weight different collaterals and calculate the total deposit value of the assets in real time.

Given a user's collateral *a*, oracle price P_a, quantity Q_a, and collateral factor CF_a the deposit value v_a can be represented as follows:

$$v_a = Q_a * P_a * CF_a$$

For example, if Alice has deposited 1,000 USDC, oracle price of 1 USDC = $1.01, CF for USDC = 0.99, then the deposit value can be calculated as follows:

$$v_{USDC} = 1,000 * \$1.01 * 0.99 = \$999.90$$

For multiple collaterals, the *total deposit value* (V) is calculated by adding the deposit values of all the individual assets deposited.

Given *n* collaterals, a user's total deposit value *V* is calculated as follows:

$$V = \sum_{i=i}^{n} v_i = \sum_{i=1}^{n} Q_i * P_i * CF_i$$

For example, if Alice has deposited 1,000 USDC (v = $999.90), 1 ETH (v = $2,734.01), and 30 krIAU (v = $1,072.55), then the total deposit value is:

$$V = \$999.90 + \$2,734.01 + \$1,072.55 = \$4,806.46$$

4.5 krAsset Debt

A borrower takes on a debt position(s) by borrowing krAsset(s) from the CDP.

4.5.1 krFactor

Similar to collateral, the protocol values each krAsset differently based on its risk profile. Generally, the higher the volatility of an asset, the higher the risk.

krFactor is a number between 1 and infinity that is used to calculate the risk-adjusted valuation of a krAsset. It shows the debt taken to borrow $1 worth of krAsset.

Note: This is a premium borrowers pay on top of the minimum collateral ratio to borrow an asset and it is higher for riskier assets.

Given a krAsset *a*, krFactor can be represented as:

$$krFactor_a = [1, \infty]$$

Additionally, krFactor can be used to encourage (i.e., by setting low krFactor) or discourage (i.e., by setting high krFactor) borrowing of certain krAssets based on the needs of the protocol.

4.5.2 Debt

The quantity, price, and krFactor are used to determine the *debt* (d) incurred by borrowing a krAsset.

Given a borrower's krAsset *b*, oracle price P_b, quantity Q_b, and krFactor $krFactor_b$, the debt d_b is calculated as follows:

$$d_b = Q_b * P_b * krFactor_b$$

For example, Alice wants to borrow 1 krTSLA, Oracle price of 1 TSLA = $1,000, krFactor for TSLA = 1.05, then the debt is:

$$d_{krTSLA} = 1 * \$1,000 * 1.05 = \$1,050$$

For multiple krAssets borrowed, the *total debt* (D) is calculated by adding the values of all the individual debts.

Given *n* krAssets, total debt *D* is:

$$D = \sum_{i=i}^{n} d_i = Q_i * P_i * krFactor_i$$

For example, if Alice has borrowed 1 krTSLA (d = $1,050), 1 krAAPL (d = $180), and 1.2 krIAU (d = $48), then the total debt is:

$$D = \$1,050 + \$180 + \$48 = \$1,278$$

4.6 Collateral Ratio

The *collateral ratio* (CR) is the ratio of a CDP's total deposit value to total debt. Given a CDP's total deposit value *V* and total debt *D*, the collateral ratio *CR* is:

$$CR = \frac{V}{D}$$

For example, if Alice's total deposit value is $4,806.46 and total debt is $1,278, then her collateral ratio is:

$$CR = \frac{\$4,806.46}{\$1,278} = 3.7609 = 376.09\%$$

The following sections describe the different conditions that dictate how CDPs operate.

4.6.1 Minimum Collateral Ratio

The *minimum collateral ratio* (MCR) is the minimum CR that must be maintained by an account to borrow krAssets. Borrowers cannot borrow krAssets if their *CR* is below *MCR*.

For example, if MCR = 150% and Alice has a *total deposit value* of $150, then

- She can only borrow up to $100 worth of krAssets.
- After she has borrowed $100, she can no longer borrow additional krAssets until the CR increases to more than 150%.

4.6.2 Liquidation Threshold

The *liquidation threshold* (LT) is the CR below which liquidations can occur. That is, if a CDP's *CR* falls below *LT*, a liquidator can liquidate the CDP until the CR is greater than or equal to LT.

For example, if MCR = 150%, LT = 140%, and Alice has a total deposit value of $150

- She can borrow up to $100 worth of krAssets and after doing so,
- If CR ≤ 140% due to the collateral dropping in value or krAssets increasing in value, then
- Her positions are subject to liquidation to bring her CR equal or above 140% (LT)

Refer to the Liquidation section for more details of the liquidation process.

4.6.3 Withdrawal

Borrowers can only withdraw collateral from their CDP's CR is above MCR. Withdrawing collateral will decrease the CR of the CDP.

4.6.4 Repayment

A borrower can reduce their debt by repaying some or all of their borrowed krAssets. The protocol burns the repaid krAssets and removes the corresponding debt from the CDP.

During repayment, the protocol collects a portion from the CDP's collateral as a *repayment fee* The repayment fee is assessed from the most recently deposited collateral asset. If the user's balance of the selected collateral isn't enough to cover the entire repayment fee then additional collateral assets are selected in descending order until the entire repayment fee value is assessed.

The fee is a percentage of the repaid debt value.

For example, if a borrower repays $100 worth of krAsset and the repayment fee is 1.5%, then the protocol collects $1.50 worth of repayment fee from the collateral.

Note: During liquidation, the repayment fee is collected from the liquidated CDP's collateral.

5. Markets

Borrowers can list borrowed krAssets on an external DEX to earn trading fees and other incentives such as farming rewards (when provided). This creates a market for traders and investors to get exposure to krAssets.

5.1 krAsset Price Deviation

When the price of a krAsset on a DEX deviates from the oracle price, it creates an arbitrage opportunity for the borrowers of the protocol.

If the price of a krAsset is below the oracle price, borrowers with existing krAsset debt position are incentivized to buy the krAsset from the open market and return it to the protocol for a discount.

Similarly, if the price of the krAsset is above the oracle price, then borrowers are incentivized to borrow the krAsset at a discount and sell it on the open market.

5.2 Stock Splits and Stock Merges

Kresko handles stock splits and merges gracefully. krAssets can be rebased accordingly with a positive or a negative index to adjust for these events.

Kresko governance can affect an increase or decrease in total supply of a krAsset to account for stock splits and merges.

Note: krAsset balances will reflect the rebases instantly. No user action is required to obtain the rebasing tokens.

5.3 Dividends

Since synthetic assets are created on the protocol based on an external oracle price of an instrument, the protocol does not pay any dividends.

6. Risk Management and Protocol Fees

Kresko uses whitelisting and risk parameters to manage different risk scenarios. These are governance-controlled parameters. The *Governance* section describes the governance process.

In the following sections, we will describe the different ways of managing risk.

6.1 Asset Whitelisting

- **Collateral Whitelisting:** To use a token as collateral in the protocol, it must be explicitly whitelisted globally.
- **krAsset Whitelisting:** To enable minting of a krAsset on the protocol, it must be explicitly whitelisted globally.

All assets whitelisted on the protocol require reliable sources of price feeds. Refer to the *Oracle* section for more details.

6.2 Asset Delisting

Assets could be delisted based on changes to their risk profiles or other external conditions that could lead to protocol insolvency.

Once an asset is delisted as a collateral, it cannot be deposited but previously deposited collateral will still be active. To encourage users to withdraw a delisted collateral, the governance process can progressively decrease its CF.

Similarly, once an asset is delisted as a krAsset, it can no longer be borrowed but previously borrowed assets can be returned. Additionally, delisted krAssets cannot be used as collateral. To encourage users to repay a delisted krAssets, the governance process can progressively increase its krFactor.

6.3 Global Parameters

Global parameters are applied to all the CDPs created on the protocol. As they affect the entire protocol, these numbers are expected to change infrequently.

- **Minimum Collateral Ratio (MCR):** It is currently set at 1.5 or 150%.
- **Liquidation Threshold (LT):** It is currently set at 1.4 or 140%.

6.3.1 Repayment fee

The **repayment fee** is a governance configurable parameter that is currently set at 1.5% of the amount being repaid.

Repayment fees are used as protocol revenue and prevent users from spamming the network. One use case of the revenue is to help backstop the protocol in case of insolvency.

6.3.2 Liquidation Incentive

The **liquidation incentive** is a governance configurable parameter that is a percentage of the krAsset being liquidated.

Liquidation incentives are used to incentivize the liquidators to help keep the protocol healthy and solvent.

6.4 Local Risk Parameters (Per Asset Parameters)

The risk profile of individual assets—both collateral and krAssets—varies from asset to asset. Before whitelisting a token, a risk analysis of how such a token would affect the rest of the protocol will be performed and the token's risk factors must be calculated.

These numbers are expected to change with moderate frequency.

- **Collateral factor (CF)**
- **krFactor**

Exponentially Weighted Moving Average (EWMA) on the returns will be used to model and calculate *CF and krFactor.*

The protocol enforces safety limits on many parameters of the protocol for the security of user funds. The governance can increase the safety limits progressively after ensuring safe functioning of the protocol.

The following describes three safety limit parameters.

6.4.1 User Parameter: Minimum Debt Value

Dust positions have several negative externalities including liquidation difficulty and healthy position management. Borrowers and liquidators should be influenced by economic considerations and not gas costs.

To help with this, the protocol enforces a floor on the debt a user can take per asset.

The *minimum debt value* is a limit that is applied to each debt position that a user takes. The gas fee on the blockchain Kresko is running on will be used to determine this limit.

6.4.2 Maximum Collateral Cap

The *maximum collateral cap per asset* is the limit on how much a particular collateral borrowers can deposit in total on the protocol. It is different for each collateral.

Note: This cap is especially useful for krAssets used as collateral. The limit prevents protocol solvency issues in case of price crashes.

6.4.3 Maximum krAsset Supply

The *maximum krAsset supply* is the limit on how much krAsset borrowers can borrow in total on the protocol. It is different for each krAsset.

The limit allows introduction of new krAssets to the protocol in a safe manner. By slowly increasing the cap, the governance can study the effect of a particular krAsset on the overall protocol health. This helps prevent risky scenarios that could affect a large amount of capital.

7. Governance

Kresko is a decentralized protocol with censorship resistance and openness as its core tenets. However, in the early days a set of *Governors* will help in maintaining the safety of the protocol. As the protocol matures, the governance will gradually be transitioned over to the community.

The Governors will consist of an initial set of stakeholders of the protocol that will control the multi-sig admin key.

Any changes to the protocol parameters will be notified to the community in advance unless there are security concerns.

The following are some actions that are controlled by the admin key.

- Update protocol parameters
- Use of protocol revenue
- Oracle related parameters
- Upgrade functions
- Transfer control over to the community
- Emergency pausing of the protocol

Note: Emergency pausing of protocol can be activated with a smaller Governor set for a faster turnaround time. This is to ensure safety of user funds in case of critical issues.

The Kresko documentation site will provide additional details regarding the conditions during which the Governors can use the admin key.

8. Liquidations

Liquidation is the act of a liquidator repaying debt on behalf of a CDP whose collateral ratio (CR) is below liquidation threshold (LT) in order to bring it above the liquidation threshold (LT). The repaid krAsset is burned and removed from the Minter's debt.

The protocol maintains solvency by ensuring that all CDPs are over collateralized. CDPs whose CR is below LT are considered *underwater* and can be subject to liquidation. When a CDP is underwater, the protocol incentivizes liquidators to liquidate a portion of the CDP to bring it back to safe levels.

Liquidation incentives are provided from the CDP being liquidated and in one of the collateral assets from the CDP.

8.1 Liquidation Process

While a CDP's collateral and debt can be composed of several assets, liquidations operate exclusively on one debt asset and one collateral asset at a time.

Liquidators can make liquidations calls to an underwater CDP by paying one of the krAssets in the CDP and receive one of the collaterals from the CDP. If the Minter's position is still below the LT following a single liquidation call, the Liquidator can continue to make additional liquidation calls until the call that brings the CDP equal to or greater than LT.

For instance, if the liquidation incentive is 5% and the repayment fee is 1.5%, during a liquidation, the Liquidator will receive 105% of the USD value of the krAsset repaid and the protocol will receive 1.5% of the USD value of the krAsset repaid.

8.2 Maximum Liquidatable Value

Maximum liquidatable value (MLV) is the maximum amount of a particular collateral of an underwater CDP a liquidator can liquidate.

Since the risk adjusted valuation, i.e., CF, is different for different collaterals, liquidators have an incentive to target low CF assets during liquidation. That is, at current market prices, they could receive a higher value of collateral (i.e., assets with low CF) as compared to high CF assets.

In order to calculate the MLV, we first calculate how much a particular pair—collateral and krAsset being returned—is underwater (*ValueUnderLT*) and a repayment divisor (*RD*).

Given, liquidation threshold *LT*, liquidation incentive *LI*, repayment fee *RF*, and

- Collateral *a* with
    - collateral factor CF_a
    - value of the deposit v_a
- krAsset *b* with
    - krFactor $krFactor_b$
    - debt d_b

$$ValUnderLT_{ab} = d_b * LT - v_a$$

$$RD_{ab} = (krFactor_b * LT) * (1 - RF) * \frac{CF_a}{(LI - 1)}$$

In order to safeguard low CF assets, the protocol differentiates between CDPs with only one collateral and CDPs with multiple collaterals.

8.2.1 Calculating MLV: Single Collateral CDP

For an underwater CDP containing only one collateral, we can calculate the MLV_{ab} for the krAsset being returned as follows:

$$MLV_{ab} = \frac{ValueUnderLT_{ab}}{RD_{ab}}$$

8.2.2 Calculating MLV: Multiple Collateral CDP

The calculation for the MLV_{ab} for a CDP collateralized by multiple collaterals is different. To safeguard low CF assets being targeted by liquidators, we use a normalizing function to calculate the MLV as follows:

$$MLV_{ab} = \frac{ValueUnderLT_{ab}}{RD_{ab} * \frac{v_a}{CF_a^4}}$$

9 Oracle Price Feeds

The protocol needs live prices of all whitelisted assets—krAssets and collaterals. The protocol requires external entities to report the prices of the assets in terms of USD.

Oracles act as external sources of price feed data to the protocol.

9.1 Market Hours

The price feed availability differs based on the type of asset.

- **Traditional finance assets:** Assets such as stocks, commodities, and ETFs typically trade on exchanges whose markets trade only during normal market hours. For instance, TSLA listed on NASDAQ trades between 9:30 am and 4 pm (Eastern Time) on weekdays only. It doesn't trade on weekends or on market holidays.
- **Crypto assets:** for crypto assets, price feeds are generally available 24/7 throughout the year.

When a price feed of an asset is paused, borrowing and debt closures are paused as well until the market for those assets reopens. Liquidations remain active when the price feed is paused using the last reported price.

Note: Under exigent conditions such as flash crashes or hacks, the protocol or governance mechanisms can pause borrowing, debt closures, and liquidations.

9.2 Kresko Price Feed Operations

Kresko expects oracles to post prices under the following scenarios:

- **Every *n* seconds:** report asset price at least every *n* seconds.
- **Deviation:** report prior to *n* seconds if the latest retrieved price differs from the current on-chain price by *x%*, where *x* is configurable (per asset and per blockchain). For instance, if the retrieved price deviates from the current price by greater than 0.5%, then a new price is expected to be posted.
- **Unexpected scenarios:** If there is an unexpected interruption in trading of the asset in conditions such as a stock exchange closing outside of schedule or during a flash crash.

Kresko will use a primary Oracle system and have another failover system for redundancy.

10. Improvement Over Existing Solutions

The Kresko protocol introduces novel improvements over existing solutions. This includes capital efficiency, better user experience, frictionless listing of assets, and better risk management.

10.1 Capital Efficiency

Each account maintains a single CDP. All the user's deposited collaterals are shared between all the debt positions.

For instance, if a user borrows 10 krAssets, and one of them increases in price and one or more decrease in price, then there is no need to add additional collateral if the shared collateral is greater than minimum collateral ratio.

10.2 Simplified CDP Management

Since there is only one CDP to manage, in case of underwater scenarios, a user needs to add more collateral to just one CDP. In addition, if the user wants to withdraw collateral, then can do so from just one CDP rather than having to deal with multiple CDPs.

10.3 Frictionless Listing of Assets

Kresko uses a risk adjusted valuation for collaterals and krAssets. This makes the protocol flexible to handle the risk of each asset uniquely.

Asset listing of arbitrary assets can occur quickly without having to change global ratios or affecting other assets. This makes the protocol agile.

10.4 Localized Risk

The protocol does not pool the collateral deposited or the synthetic assets borrowed from different users. That is, the assets in a CDP are localized and do not affect other CDPs.

10.5 Improved Risk Management

The protocol uses several risk controls using global collateral ratios, per asset risk adjustments, and safety limits. This helps keep the protocol secure and solvent.

10.6 Composability

krAssets are designed to follow the ERC-20 standard (*ERC-20 Token Standard*, n.d.). That is, users and applications can use krAssets like any other token—transfer them between accounts, create liquidity pools on DEXs, trade, etc in a permissionless manner.

In addition, the protocol facilitates composability through the use of trusted contracts. Trusted contracts can take actions such as borrow krAssets and withdraw collateral on behalf of users.

Developers can create useful end user applications with features such as automatic debt management, rebalancing, etc using the trusted contracts feature.

Governance needs to approve trusted contracts after a thorough vetting process.

11. Limitations and Future Work

The protocol consists of the core functionality of a capital-efficient synthetic protocol. We have designed it with minimal features to ensure safety. Once the model is battle tested, we will make improvements and add additional features.

The following sections highlight limitations of the current model and propose improvements to overcome the limitations.

11.1 Mixed Asset Type Debt Positions

While the single-CDP-per-account model allows simplified debt management, it is risky to pool different asset type debt positions. For instance, if a borrowed asset rises in price suddenly, other borrowed assets could get liquidated. Having the ability to borrow volatile assets and non-volatile assets from the same pool adds risk to CDPs.

In the next version, we will consider the creation of multiple CDPs per account. Different classes of debt assets can be added to different CDPs. Applications built on top of the protocol can allow creation of CDP templates based on different risk profiles. Users can share them, enabling social market making.

11.2 Incentive Mechanism

Currently, borrowers are not directly incentivized with fees or rewards. Borrowers can make markets on DEXs to earn trading fees and farming rewards (if any).

We are currently exploring other incentivization mechanisms such as a governance token, external token rewards, and incentive mechanisms on an application layer above the protocol.

11.3 Inverse Assets

The protocol currently allows users to only create one way synthetic assets, i.e., the price of the synthetic asset follows the prices of the underlying. That is, it does not not allow users to create inverse synthetic assets whose prices are inversely related to the underlying.

Currently borrowers and market makers might have to use TradFi markets to de-risk their positions. In addition, they can indirectly de-risk their positions using krAssets as collateral to borrow inversely correlated assets.

Inverse assets are helpful in creating delta neutral strategies allowing borrowers to de-risk their positions. This is a feature under consideration for the next version.

11.4 Better Price Parity

The protocol sets the price of a krAsset to its underlying during the creation of the asset. It relies on external liquidators to help maintain overcollateralization and price parity in an indirect way. Additionally, borrowers who make markets maintain the price parity using an external DEX. See section 5.1 for more information.

Based on the amount of liquidity on a DEX, the price of a krAsset could drift from the underlying asset's price. The above mentioned processes are slow to bring the krAsset prices in sync with underlying assets' prices.

For the next version, we plan to include an oracle based DEX within the protocol that can react to price changes in real time.

11.5 Capital Efficiency Improvements

Integrating a DEX within the protocol can enable the protocol to use a portion of the collateral to provide liquidity for krAssets. This can improve capital efficiency over current models by 50% or more.

Pooling of assets to lend collateral and borrow synthetic assets as with lending markets is another avenue of exploration to create capital efficiency.

12. Miscellaneous

12.1 Protocol Specs and Implementation

The whitepaper has been written as a specs document for the implementation of the protocol. The protocol has been implemented in Solidity to run on an Ethereum Virtual Machine (EVM) compliant blockchain.

12.2 Composability and Frontend

The frontend is just a client that can be used to access the protocol implementation. Users can use a terminal client to interact with the protocol without a frontend.

It is expected for third-parties to create frontend interfaces to the protocol. Other dApps and protocols can also use Kresko based on their needs.

12.3 Smart Contract Security Audits

The implementation of the protocol has undergone a third-party audit by Quantstamp, a well known blockchain security firm. Additional audits will be conducted to ensure the security and correctness of the protocol.

In addition, the Kresko team and partners have conducted several security audits internally.

12.4 Economic Risk Assessment

The Kresko team is also conducting an economic risk assessment of the protocol by studying the various scenarios that could lead to risky situations. In addition, this study will be used to propose safety parameters to use on the protocol.

A third-party audit might also be conducted to achieve an unbiased risk assessment.

13. Conclusion

Our goal is to democratize access to wealth for anyone, anywhere. We have introduced a novel synthetic asset protocol, Kresko, that uses a single-CDP-per-account model for capital efficiency and improved user experience.

The protocol is agile by using distinct risk parameters: global parameters that change infrequently and local parameters that change with moderate frequency. This enables frictionless listing of arbitrary assets.

We have taken a security-first approach and added minimal features. Once the model is battle-tested, we will add additional features to further enhance capital efficiency and user experience.

14. References

Adams, H., Zinsmeister, N., Robinson, D. (2020, March). *Uniswap v2 Core*. https://uniswap.org/whitepaper.pdf

Aave Protocol Whitepaper V2.0, (2020, December). https://github.com/aave/protocol-v2/blob/master/aave-v2-whitepaper.pdf

Leshner, R. & Hayes, G. (2019, February). *Compound: The Money Market Protocol Version 1.0*. https://compound.finance/documents/Compound.Whitepaper.pdf

The Maker Team. (2017, December). *The Dai Stablecoin System Whitepaper.* https://makerdao.com/whitepaper/Dai-Whitepaper-Dec17-en.pdf

Synthetix Litepaper Version: 1.5. (2022, March). https://docs.synthetix.io/litepaper/

Liu, S. & Lee, I. (n.d.). *Mirror Protocol V2*. https://730577551-files.gitbook.io/~/files/v0/b/gitbook-x-prod.appspot.com/o/spaces%2F-MLRzugf7mxc4ryNhTuq%2Fuploads%2F0t3znySsjF6CiLrcT0mI%2FMirror_Protocol_V2.pdf?alt=media&token=b5728c7d-7f12-4f41-8ce6-8347da02b9ff

Kelly, L.J. (2022, May 14). *How Terra's UST and LUNA Imploded*. https://decrypt.co/100402/how-terra-ust-luna-imploded-crypto-crash

Ethereum Virtual Machine (EVM), (n.d.), https://ethereum.org/en/developers/docs/evm/

ERC-20 Token Standard, (n.d.), https://ethereum.org/en/developers/docs/standards/tokens/erc-20/