



# Mainframe Lending Protocol

Doug Leonard  
doug@mainframe.com  
July 2020

## Abstract

The Mainframe Lending Protocol enables fixed-rate lending and borrowing — something much needed in decentralized finance today. Tokenizing on-chain debt allows anyone access to instant liquidity against collateral assets approved by Mainframe Governance. Similar to a zero-coupon bond, the debt obligation settles on a specific future date. Mainframe Governance is the community organized process of managing the various aspects of the Mainframe Lending Protocol.

Liquidating at-risk collateral-accounts is a time-sensitive problem addressed by many other DeFi protocols. Existing solutions rely on external actors motivated by some form of arbitrage opportunity to perform this essential task. Recently, this approach has proven unreliable during times of extreme volatility. Furthermore, a dependence on external actors drives ecosystem resources out of the system and away from a protocol's core value proposition. The Mainframe Lending Protocol proposes a novel liquidation architecture that drives resources back to ecosystem participants and is designed to function reliably even during times of extreme volatility.

The system works by balancing incentives between borrowers, lenders, and guarantors, each representing a distinct and complementary position of economic exposure. Borrowers deposit collateral and mint tokens, representing a debt obligation. Lenders purchase the tokenized debt obligation, typically at a discount, and redeem them for face value at maturity. Guarantors purchase collateral at a discount, when collateral-accounts fail to satisfy the collateral requirement.

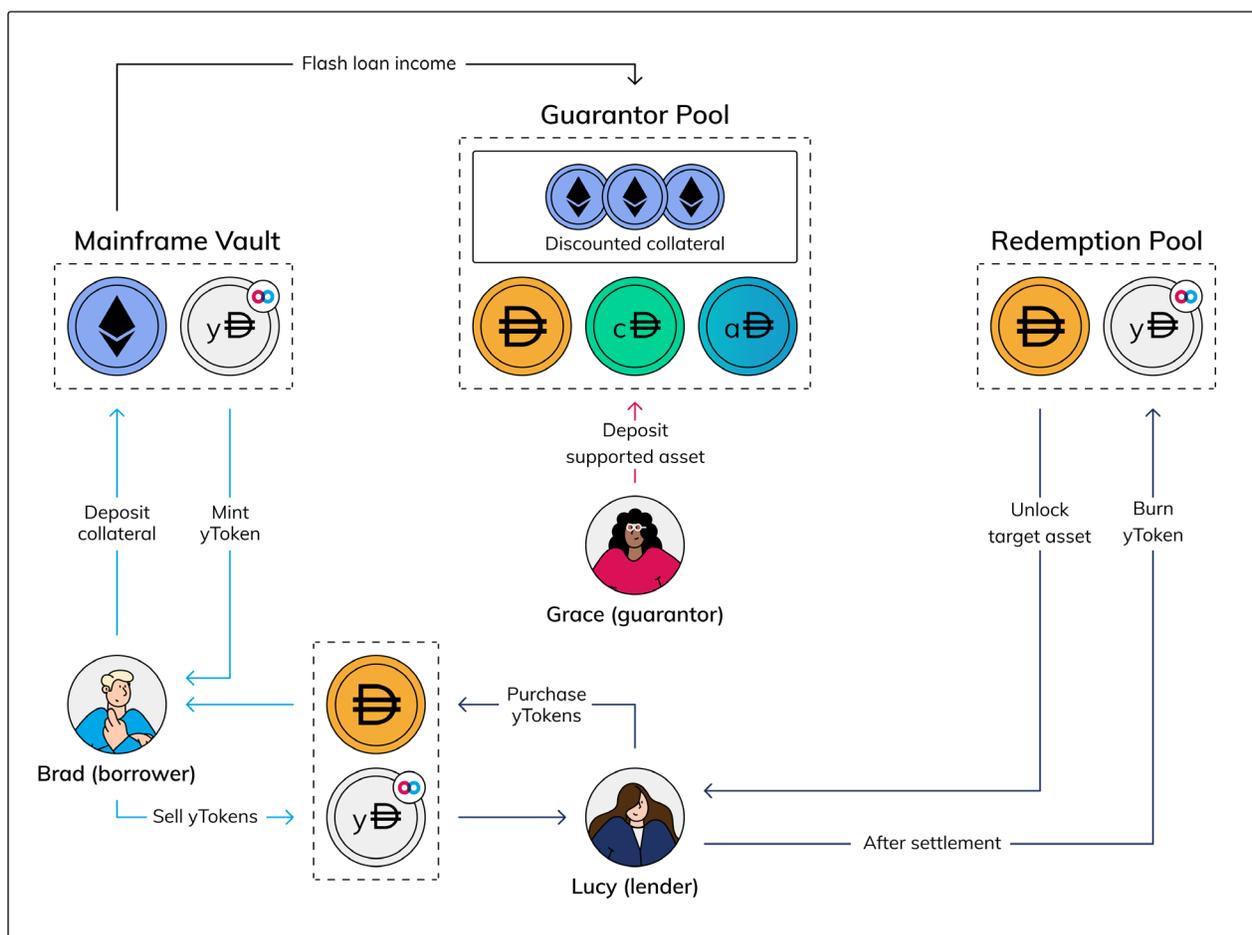
Throughout the Lending Protocol, Mainframe Token (MFT) may be utilized to vote on governance proposals, unlock discounts, achieve greater capital efficiency, increase protocol rewards, and access additional leverage. Together, the strategies for MFT utilization, collateral rehypothecation, liquidation, and settlement enable lower collateral requirements and allow for a more efficient increase in leveraged exposure to base assets.



# 1. Introduction

Decentralized Finance (DeFi) is the home to financial building blocks that can be reused and extended in new ways. Leveraging existing infrastructure components and reimagining how each one connects to another accelerates innovation. The system outlined in this paper would not be possible without these open and reusable building blocks.

The Mainframe Lending protocol is an implementation of Dan Robison's Yield Protocol, yTokens are the vehicle to synthetically lend or borrow target assets. The initial infrastructure borrows heavily from the Maker and Compound Protocols. Notably, the system takes inspiration from Maker's token utilization, Governance, and Vaults[2]. Price oracles use Compound's Open Oracle System[3]. Reprehypothecation leverages parts of Hayden Adams' Uniswap Protocol v2[4]. To provide a comprehensive resource in describing the Mainframe Lending Protocol, many excerpts in this paper directly quote the respective protocols. If you have read the documentation of these foundational protocols, you will be comfortable in your understanding of this system. This document is not an effort to reinvent proven technologies, but rather a novel positioning of many existing DeFi building blocks in hopes of driving additional adoption and liquidity for trustless fixed-rate lending and borrowing instruments.



Protocol overview (fig. 1)



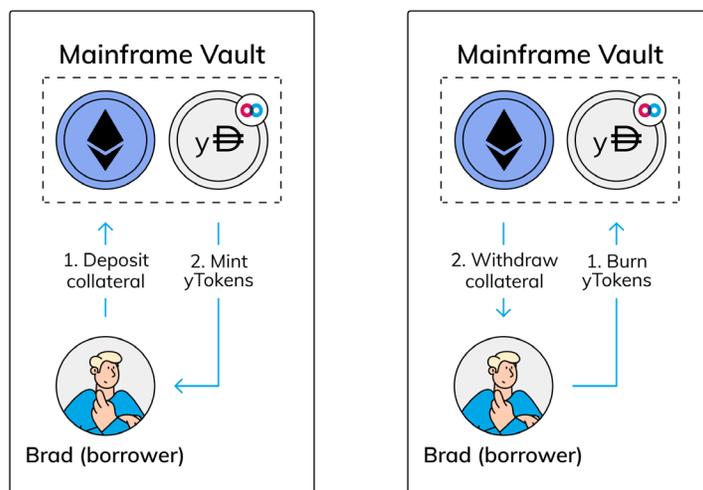
## 2. Basic Example

Lucy lends, Brad borrows, and Grace guarantees the transaction.

The term “yToken” generically refers to tokenized debt backed by collateral. Suppose that on October 1, 2019, Brad owns 1 ETH, currently worth 100 Dai, and he wants to enter a 1.5x leveraged long position on ETH. Brad looks up the Dai/ETH yToken contract that expires at the end of the quarter (on December 31, 2019) with a collateralization requirement of 150%.

Brad creates a vault in that yToken contract, deposits 1 ETH as collateral, and takes out 50 yDai as debt. Brad then sells those yDai. yDai of that maturity is currently trading at a discount—say, 0.97 Dai—so he only receives 0.485 ETH. He deposits that ETH into his vault as additional collateral. He now has exposure to 1.485 ETH.

If Brad wanted, he could continue to take out yDai, trade it for ETH, and redeposit the ETH as collateral, repeating until he reaches his desired amount of leverage or he bumps up against the collateralization requirement.



Mint yTokens (fig. 2)

Repay (fig. 3)

Since Brad’s vault has 50 Dai of debt outstanding and a collateralization requirement of 150%, it gets liquidated if the value of his collateral falls below 75 Dai, which corresponds to the ETH price falling to about 50.5 Dai. If his vault liquidates at that exact price, resources from the Guarantor Pool are used to pay off the debt of 50 yDai, and 75 Dai worth of ETH collateral is distributed back to the Guarantor pool, giving them a profit of 25 Dai, plus a little more based on the discount at which the yDai are currently trading. If the price of ETH gets too close to his liquidation price, Brad should try to close his vault or deposit more collateral to avoid paying this penalty.

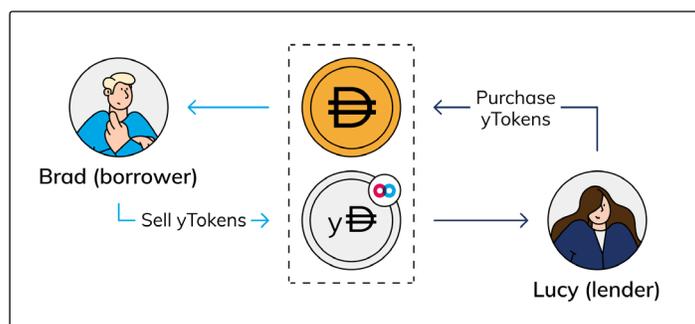


Suppose Brad closes his position just before expiration and the price of ETH is 200 Dai. He sells approximately 0.25 ETH of his collateral for 50 yDai. Using his vault, Brad burns 50 yDai unlocking the rest of his 1.235 ETH collateral, worth 247 Dai.

But, if Brad fails to repay his debt upon expiration, a requisite amount of his ETH collateral is sold to the Guarantor Pool at a discount so yToken holders can redeem their tokens for Dai. Any unsold collateral remains in his vault.

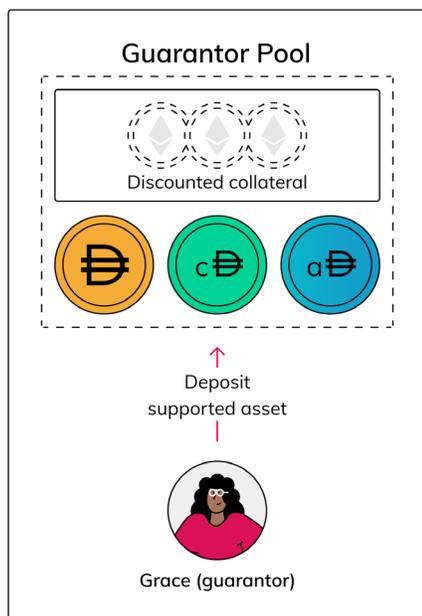
Alternatively, Brad can exit the position early at any time by purchasing yTokens (likely, though not necessarily, at less of a discount than he initially sold them for), and burning them to repay his debt.

Now suppose Lucy wants to lend 100 Dai and earn interest on it until December 31, 2019. Lucy can simply purchase yDai. At the current discounted price of 0.97 Dai, Lucy's 100 Dai can buy about 103.09 of these yDai. At maturity, Lucy's yTokens will be redeemable for 103.09 Dai, so she will have earned 3.09 Dai in interest. If Lucy wants to exit her position before that, she could do so by selling her yTokens (likely, though not necessarily, at a higher price than she initially paid for them).

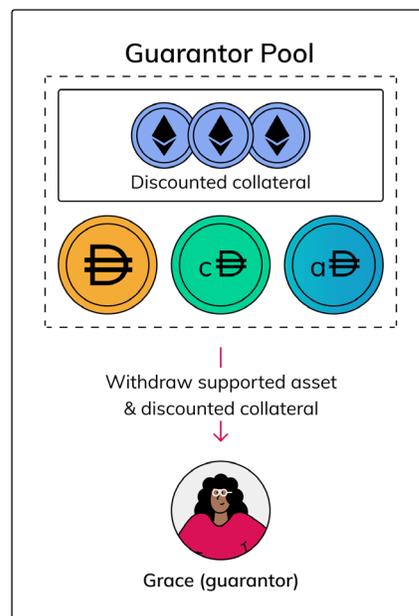


Lend & Borrow (fig. 4)

Finally, suppose Grace would like to earn multiple streams of income and have the opportunity to purchase ETH at a discount. Grace deposits 50 Dai into Compound and then puts her cDai into Mainframe's global Guarantor Pool. Both Guarantor deposits and vault collateral like Brad's are used for flash loan liquidity; notably, the revenue it generates is paid to Grace and other Guarantor Pool members. In the scenario above, when Brad was liquidated, Grace bought ETH at a 33% discount. Simultaneously, her deposit earned interest income from Compound and collected flash-loan fees generated from both her cDai deposit and borrower's vault-collateral.



Deposit (fig. 5)



Withdraw (fig. 6)

### 3. Yield Protocol

The following excerpts come directly from Dan Robison's white paper defining the Yield Protocol. More advanced users should refer to the full Yield Protocol specification for a comprehensive understanding of the Mainframe Lending protocol.

The Yield Protocol is a standard for a token that settles based on the value of a target asset on a specified future date, and which is backed by some quantity of a collateral asset. We call these tokens "yTokens."

You can create yTokens by depositing collateral, then sell them to effectively borrow (and short) the target asset. Buying yTokens is economically similar to lending the target asset. The effective "interest rate" received by yToken holders is determined by the discount at which yTokens currently trade, as well as the time to maturity.

The design of the Yield Protocol is heavily influenced by other projects on the Ethereum blockchain that offer synthetics, borrowing, lending, and/or leverage. Yield primarily differs in that its interest rates are implicit and set by market prices, rather than being set by Governance or a formula. Additionally, whereas most lending protocols use floating interest rates, Yield enables term loans with fixed interest rates, while still maintaining some degree of fungibility[1].



### 3.1 yToken Mechanism

This section describes the yToken mechanism at a high level. For simplicity, this description assumes the existence of a synchronous free on-chain price oracle for the underlying asset.

yTokens differ from each other in four dimensions: target asset (or oracle), collateral asset, expiration time, and collateralization requirement. Anyone can define a particular yToken by specifying those four parameters. For example, there would be one yToken for a given Dai oracle, backed by ETH, settling at 11:59 PM on December 31, 2019, with a 150% collateralization requirement.

One example of a yToken might be “yDai/ETH (2021-01-01)”. This refers to a yToken with Dai as the target asset, ETH as the collateral asset, expiring at 12:00 AM on January 1, 2021. If referring generally to any yToken that targets Dai, we just use the term “yDai”.[1]

#### 3.1.1 Minting yTokens

Once a token contract exists for a particular yToken, anyone can deposit collateral to create a vault. These vaults are analogous to (and named after) the vaults in the Maker system. The owner of a vault can mint yTokens, which adds to the vault’s debt. They can also burn yTokens to reduce their debt. The debt of a particular vault must not exceed the value of its collateral plus some required margin, or it will be liquidated. A yToken resembles a secured zero-coupon bond. Upon expiration, it can be redeemed from the yToken contract for its face value[1].

Depositing the target asset into the yToken contract’s redemption pool will mint an equal amount of the respective yToken. This additional minting mechanism is useful for market makers who need instant access to yToken liquidity without exposure to the base collateral asset.

### 3.2 DeFi Primitive Compatibility

Governance may elect to support other DeFi primitives, like Compound’s cTokens, AAVE’s aTokens, or mStable’s mAssets, enabling certain groups within the protocol to earn multiple streams of income on idle resources. New risks are introduced and will need to be considered thoroughly for each primitive Governance votes to support.

### 3.3 Flash Loans

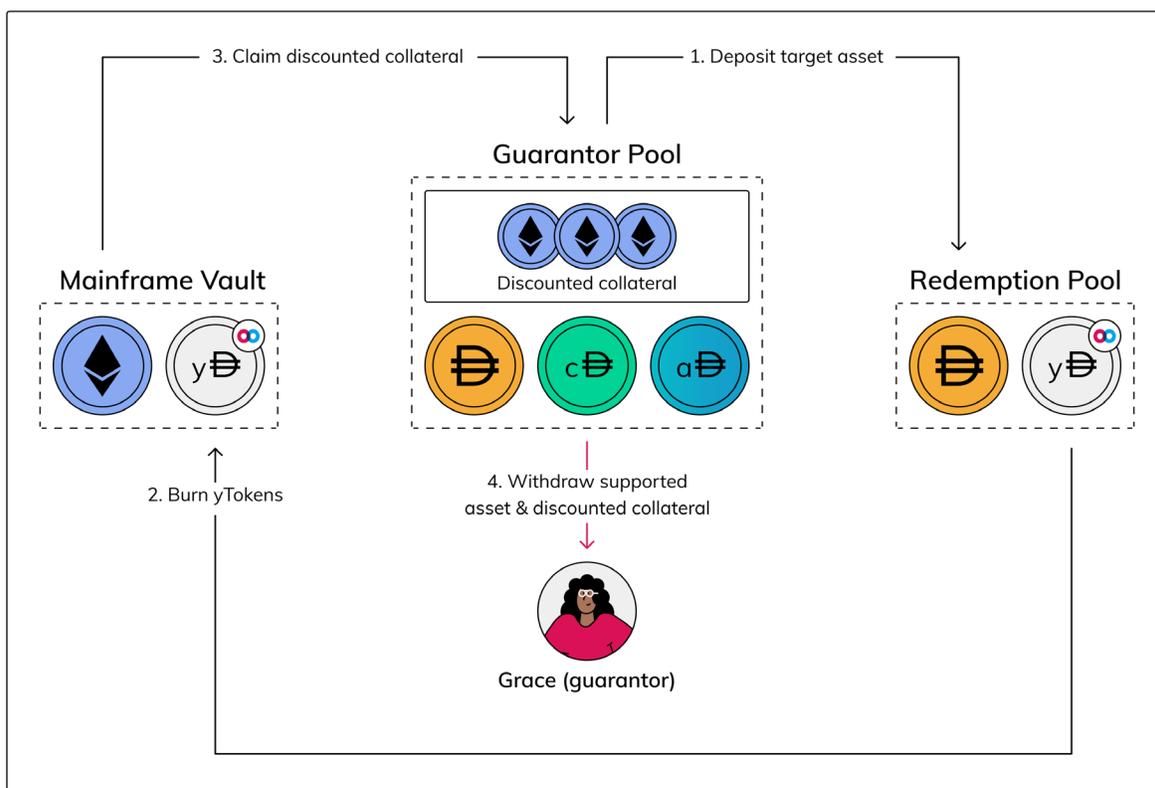
The Mainframe Lending Protocol will allow users to leverage the various pools of liquidity for flash loans. Vaults and redemption pools may generate income from flash loans. Income derived from the use of vault collateral benefits Guarantors. Whereas, income generated from redemption pools contributes to the system buffer where excess resources are used to burn MFT.



### 3.4 Liquidation

If the price of the collateral asset falls relative to the target asset before a yToken expires, some of the vaults and thus the yTokens they back may become undercollateralized[1]. To prevent this from happening, Guarantors pool together the target asset type or a supported derivative, like Compound's cTokens. Guarantors have the benefit and obligation to purchase the base asset at a discounted rate. This role is economically similar to writing a put option on vault collateral with the exercise price equal to the vault's liquidation point. Although it is anticipated that with proper incentives, Guarantors will enter and exit the pool freely, liquidity is not guaranteed.

The combined value of the Guarantor and Redemption Pools acts as a natural debt issuance ceiling. The face value of all yTokens must not exceed the sum of value within the Guarantor Pool and balances of the yToken redemption pools. This ceiling is enforced at the smart contract layer.

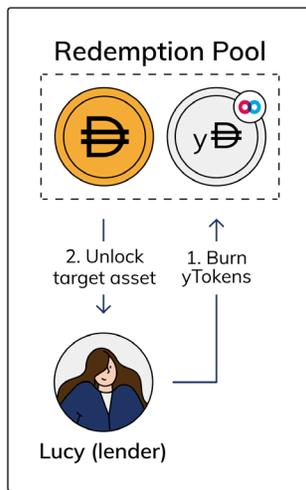


Liquidate (fig. 7)



### 3.5 Settlement

The purpose of the settlement mechanism is to ensure that yTokens trade at the same price as the target assets at the moment of maturity[1]. Once the yToken contract matures, liquidation may be triggered for any vault with outstanding debt. When triggered, Guarantors purchase the requisite amount of collateral and deposit the respective target asset balance into the yToken contract's Redemption Pool. At maturity, yToken holders burn yTokens to redeem the target asset from the Redemption Pool.



Settlement (fig. 8)

## 4. Protocol Incentives

Mainframe Token is an ERC-20 asset that empowers community governance of the Mainframe Lending Protocol. MFT aligns and balances incentives within the system and is critical for the sustainability of protocol upgrades. Throughout the Lending Protocol, MFT may be utilized to vote on governance proposals, unlock discounts, achieve greater capital efficiency, increase protocol rewards, and access additional leverage.

MFT may be staked, streamed, or burned. It is anticipated that staking MFT will be a necessary part of our Delegate Governance System and will be the subject of a future publication. Managing protocol incentives by Governance is expected to be iterative in nature. Appropriately, questions about how much streaming or burning will be required and where each will be used will be managed by Governance.

### 4.1 Streaming

Streaming tokens is the process of automating access to an asset as a function of time. An initial sum of tokens are locked in a smart contract, and the recipient gains access to claim the



tokens as they become available. Interestingly, when the sender and recipient are the same entity, this streaming mechanism incurs a cost measured in terms of the streamed asset's time-value. Streaming tokens to oneself acts as an effective locking mechanism while avoiding sudden shocks to the asset's circulating supply.

## 4.2 Burning

Burning tokens is the process of sending tokens to a destination that effectively makes them unrecoverable. Many implementations of token burning send tokens with the following destination address: `0x00`. Burning tokens is often desirable to reduce the total supply of tokens.

## 5. Protocol Roles

The Mainframe Lending Protocol breaks down into four primary roles, Borrowers, Lenders, Guarantors, and Delegates. Users can choose to play any combination of these roles and can fine-tune their desired type of exposure and amount of risk by changing their level of participation within each role. Over time it is expected that Governance will iterate on how MFT is utilized for each role. For example, it may be advantageous first to require Borrowers and Guarantors to Lock and Stream MFT to themselves during the initial bootstrapping phase since this mechanism more aggressively drives down the circulating supply of MFT in the short term. Later this mechanism might be replaced with a simple burn requirement once the Mainframe Lending Protocol has secured a position and reputation within the DeFi ecosystem. For this reason, each role's incentives are defined abstractly in anticipation that MFT utilization will evolve and improve in time.

### 5.1 Lenders

Lenders earn income from yToken appreciation.

### 5.2 Borrowers

Borrower's liquidation penalties are implicit in their vault's collateral requirement. For example, a vault with a collateral requirement of 150% stands to forfeit 33% of its collateral in the event of liquidation. While borrowing does not strictly require the use of MFT, the collateral requirement and liquidation penalty are functions of MFT utilization. Borrowers who desire additional leverage or want to hedge against potential liquidation with a reduced penalty can do so with MFT. Because the novel liquidation mechanism enables much lower collateral requirements, everyone has access to collateral requirements competitive with those seen in alternative lending protocols. Borrowers who utilize MFT optionally unlock greater capital efficiency, lower liquidation penalties, and increased leverage.

### 5.3 Guarantors

Guarantors may have multiple income streams. The income earned from Borrower collateral being used for flash-loan liquidity goes to Guarantors. This rehypothecation might be considered the premium borrowers pay guarantors for effectively writing put options on vault



collateral with the exercise price equal to each vault's liquidation point. As support for other DeFi primitives, like Compound's cTokens or AAVE's aTokens, are added by governance vote, Guarantors will be able to stack their income streams from multiple sources. Guarantors who optionally utilize MFT will have a greater proportional claim on pool income. Finally, Guarantors purchase vault collateral at a discount set by the vault's collateral requirement.

## 5.4 Delegates

It is expected that Delegates within the Governance system will be required to stake MFT. MFT holders will also have the option to stake MFT and be able to delegate their participation in the governance process of debating, proposing, and voting on all changes to the protocol if desired. Staking and delegation will also be the mechanism by which resources are allocated to specific efforts or protocol priorities as control of the protocol is decentralized and handed over to MainframeDAO.

## 5.5 MFT Holders

All MFT holders benefit from the decrease in total supply when the protocol burns MFT. A portion of every liquidation and income from redemption-pool flash-loan liquidity goes to the system buffer where excess resources are used to burn MFT.

# 6. Governance

## 6.1 Use of the MFT Token in Mainframe Governance

The MFT token—the governance token of the Mainframe Lending Protocol—allows those who hold it to vote on changes to the lending protocol. Note that anyone, not only MFT holders, can submit proposals for an MFT vote.[2]

MFT holders will be able to vote on the following:

- Adding new collateral asset types.
- Add or change the Risk Parameters for collateral asset types.
- Modify Incentive Parameters.
- Modify the Price Feeds.
- Modify Emergency Shutdown Procedure.
- Trigger Emergency Shutdown.
- Upgrade the system.

Governance can also allocate funds from the Mainframe Buffer to pay for various infrastructure needs and services, including Oracle infrastructure and collateral risk management research. The funds in the Mainframe Buffer are revenues from a Protocol Endowment Fund, Liquidation Fees, Flash Loan Income, staking rewards, and other income streams.

## 6.2 MainframeDAO

Mainframe will start with centralized control of the protocol at first and will transition control



over time to MFT stakeholders. Mainframe Group, Inc. will dissolve after the transition of control is complete. Mainframe currently plays the primary role in maintaining the Mainframe Lending Protocol, expanding its usage worldwide, and facilitating Governance. However, Mainframe Group, Inc. plans to dissolve once MainframeDAO can manage Governance completely on its own. Should MainframeDAO fail to sufficiently take the reins upon the dissolution of Mainframe Group, Inc., the future health of the Mainframe Lending Protocol could be at risk.

MFT holders are incentivized to prepare for the dissolution after it completes “gradual decentralization” of the project. Moreover, successful management of the system should result in sufficient funds for Governance to allocate to the continued maintenance and improvement of the Mainframe Lending Protocol.

## 7. Conclusion

The Mainframe Lending Protocol allows anyone to create fungible on-chain debt obligations economically similar to zero-coupon bonds. The tokenized debt obligations are backed by a surplus of collateral that are escrowed into audited and publicly viewable Ethereum smart contracts. A novel system of incentives, including penalties, discounts, and arbitrage opportunity, protect the protocol from under-collateralization. Compatibility with other DeFi primitives enables participants to earn income from multiple DeFi protocols at once. Mainframe tokens align the incentives of each stakeholder, balances the participation of ecosystem members, and provides certain desirable benefits within the system.

## 8. Glossary

Rehypothecation - reuse of collateral, occurs mainly in the financial markets, where financial firms re-use the collateral to secure their own borrowing[5].

Secured zero-coupon bond - A zero-coupon bond is a debt security that does not pay interest but instead trades at a deep discount, rendering a profit at maturity, when the bond is redeemed for its full face value[6].

## 9. Citations

[1] Dan Robinson The Yield Protocol: On-Chain Lending With Interest Rate Discovery. Apr. 2020, <https://research.paradigm.xyz/Yield.pdf>.

[2] MakerDAO. MakerDAO Whitepaper Maker, 2020, <https://makerdao.com/en/whitepaper>.

[3] Leshner, Robert, and Geoffrey Hayes. Compound, Feb. 2019, <https://compound.finance/documents/Compound.Whitepaper.pdf>.

[4] Adams, Hayden, et al. Uniswap v2 Core. Uniswap - Automated Liquidity Protocol, Mar. 2020, <https://uniswap.org/whitepaper.pdf>.

[5] “Hypothecation.” Wikipedia, Wikimedia Foundation, 11 Dec. 2019, [en.wikipedia.org/wiki/Hypothecation](https://en.wikipedia.org/wiki/Hypothecation).

[6] Chen, James. “Zero-Coupon Bond.” Investopedia, Investopedia, 6 Mar. 2020, [www.investopedia.com/terms/z/zero-couponbond.asp](https://www.investopedia.com/terms/z/zero-couponbond.asp).



## **10. Acknowledgments**

This work is based on various iterations of ideas and conversations with present and former Mainframe team members, namely Ricky Stamton, Mick Hagen, Rafaella Baraldo, Reed Hagen, Paul Le Cam, Willian Mitsuda, Robson Silva Jr., Luca Tavazzani, Guilherme Viana, and Sarah Wiley. A special thanks to all the authors and open source contributors of Maker DAO, Yield Protocol, Uniswap, and Compound.

## **11. Disclaimer**

This paper is for general information purposes only. It does not constitute investment advice or a recommendation or solicitation to buy or sell any investment and should not be used in the evaluation of the merits of making any investment decision. It should not be relied upon for accounting, legal, or tax advice or investment recommendations. This paper reflects current opinions of the authors and is not made on behalf of Mainframe or its affiliates and does not necessarily reflect the opinions of Mainframe, its affiliates, or individuals associated with Mainframe. The opinions reflected herein are subject to change without being updated.