# Omni: A personalized, composable, and dynamic lending protocol

**Anonymous Authors**[1]

## Abstract

The Omni protocol is a novel composable, dynamic, and capital efficient money market primitive. In comparison to existing money market designs that make trade-offs between asset capital efficiency and liquidity to support more assets, the Omni protocol is able to support a wide array of collateral and borrow assets with zero fragmentation and maximal capital efficiency. The protocol introduces a novel concept of "risk tranches" for asset pools that allows lenders to opt-in and opt-out of lending to certain collateral assets, so lenders earn the maximum yield for their risk profile and borrowers have access to maximum liquidity. In addition, the protocol introduces collision-free sub-accounts for asset management, high efficiency borrowing modes, a joint risk and utilization interest model, timed collateral, proportional loss socialization, and dynamic liquidations using dutch auctions. Through these advancements, the Omni Protocol elevates the capital efficiency and fluidity of the money market landscape, offering a robust foundation for fostering a more inclusive and efficient financial ecosystem.

## 1. Introduction

Lending has always been a foundational part of financial systems. From helping people buy their first home by taking out a mortgage loan to supercharging the growth of corporations worldwide, lending is a catalyst for growth in economies everywhere.

In the world of decentralized applications, the need for robust lending systems is even more crucial. In the fiat world (off-chain), banks are the major provider of loans, considering parameters like credit scores when underwriting loans. In the world of crypto applications, achieving true decentralization requires replicating aspects of a bank in a permissionless and noncustodial manner. This is where decentralized lending comes into play.

Decentralized lending brings three main benefits: an increase in capital efficiency on-chain, a decrease in risk for lenders, and an overall boost in on-chain credit that stimulates further activities and decentralized applications.

- **Capital Efficiency:** Lending and borrowing activities breathe life into crypto markets, fostering liquidity and promoting the efficient allocation of resources. Borrowers seek funds for various purposes such as liquidity provisioning, leveraging positions, and maintaining exposure. Lenders supply these funds to earn interest on their capital. This continuous flow of capital ensures liquidity and supports the seamless operation of crypto markets and decentralized protocols.
- **Risk Management:** Lending provides a mechanism for transferring risk from those less inclined or equipped to bear it (borrowers) to those better suited (lenders). Traditionally, lenders evaluate borrowers' creditworthiness and price loans accordingly, facilitating risk management and distribution within the financial system. In decentralized finance, credit scores are replaced by over-collateralized loans, enabling truly permissionless and open lending for all.
- **Credit Creation:** Lending acts as a catalyst for credit creation within the decentralized economy. When a money market extends a loan, it effectively generates new money, appearing as a deposit in the borrower's account. This credit multiplication mechanism has a multiplier effect on the money supply, stimulating economic activity both on and off-chain.

However, there are still significant unmet needs in lending within existing models. For Peer-to-Peer (P2P) models, like EthLend and Dharma, there is often difficulties scaling due to inadequate liquidity from fragmentation and yield dilution. Analyzing and establishing fair loan configurations, especially in the volatile crypto landscape, remain challenging. This inefficiency hampers liquidity attraction, making it difficult for borrowers to secure loans and occasionally results in unfavorable configurations leading to lender losses. Furthermore, legacy P2P models necessitate locking liquidity until specific custom terms are violated, prohibiting instant withdrawals.

Meanwhile pool-based models, like Aave and Compound, only enable a few select assets to be managed on their protocols due to low risk tolerances. The few number of supported assets and low turnover leads to fragmentation for

users across multiple protocols. Not only is this a poor user experience, but it creates significant overhead for users to manage their assets across multiple protocols and keep track of which assets are where.

Some protocols, such as Beta and Euler, have tried to improve the model by enabling permissionless listing of assets via a Uniswap V3 pool. However, in practice we have seen low usage of permissionless listings, due to the complexities of risk management and attracting liquidity. Additionally, while this enables more assets to be listed for depositing, it does not solve the problem of enabling these assets as collateral. This is because these protocols do not distinguish between different collateral assets, meaning every collateral asset listed has fair access to the underlying pools for borrowing. Protocols are still required to create a separate isolated pool for each collateral type configuration they wish to support, creating fragmentation and overhead for users.

Here we introduce the Omni protocol, a composable, dynamic, and safer money market capable of handling all collateral types and borrow types with zero fragmentation and maximal capital efficiency—the first of its kind. For lenders, personalize your loan risk appetite with zero liquidity fragmentation and compartmentalize positions into subaccounts. For borrowers, use long-tail assets, like SHIB, as collateral in addition to stablecoins, LSDs, and more, to borrow one or multiple assets all at the same time and all from a single wallet. The core of the protocol revolves around a new concept we introduce: risk tranches. Risk tranches can be thought of as pyramid-like layers within an asset pool, where liquidity of higher risk tranches is always available to lower risk tranches. As an analogy, water that starts at the top flows downwards to every other level, while water that starts in the middle doesn't flow upwards to the top. In addition, the protocol introduces a number of new features for lending including collision-free sub-accounts for asset management, high efficiency borrowing modes, a joint risk and utilization interest model, timed collateral, proportional loss socialization, and dynamic liquidations using dutch auctions.

## 2. Architecture

The architecture of the protocol is structured as shown in Figure 1. There are three main groups that comprise interactivity of the protocol: External, Risk Management, and Asset Pools. The External group consists of users and administrators that are responsible for the configurations of the protocol via the risk management layer and also deploying new asset pools. Next, the Risk Management layer is responsible for handling risk assessment of all loans. It tracks the risk parameter configurations of each of the asset pools, determines how much interest should be paid in a given state, as well as interfaces with oracles to determine

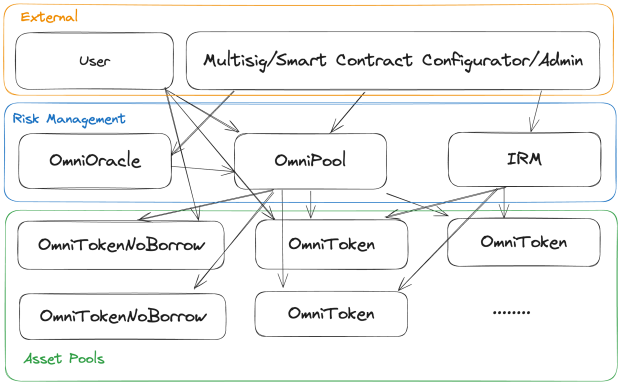


*Figure 1.* Architecture diagram of the Omni protocol, with arrows showing relationships.

the healthiness of a loan. Within the Risk Management layer, the OmniPool contract serves as the main point of entry for borrowers of the protocol. Finally, is the asset pool layer which consists of all the assets that are tracked by the risk management layer. These assets are separated into two categories: borrowable and non-borrowable.

## 3. Asset Pools

The Asset Pools layer of the protocol consists of the assets tracked by the protocol for lending and borrowing. The protocol is able to support ERC20, ERC721, and additional non-standard assets as underlying assets. The protocol does not make specific decisions about how to wrap each of the underlying assets, however the protocol does define a wrapper interface that must be followed in order for these assets to be supported. There are two main types of assets that are wrapped into to be supported by the Omni protocol: OmniToken and OmniTokenNoBorrow, termed oTokens. These are names assigned to borrowable assets, i.e. assets that can be used as collateral and/or used for loans, and non-borrowable assets, i.e. assets that can only be used as collateral. One of the major advantages of the Omni protocol is that there is only one borrowable pool per asset, therefore there is zero liquidity fragmentation to lenders and max liquidity for borrowers. Secondly, within each borrowable pool lenders are able to explicitly opt-in to lending to a certain set of assets. This means that it is possible for user Alice to lend to only USDC and ETH collateral, and user Bob to lend to USDC, ETH, and BETA collateral. Both Alice and Bob will earn the interest paid by borrowers who use USDC or ETH as collateral, but only Bob will earn the interest from users using BETA as collateral. Similarly, borrowers using USDC or ETH as collateral can borrow both Alice's and Bob's deposits, and borrowers using BETA as collateral are only able to borrow Bob's deposits.

### 3.1. Risk Tranches

For borrowable tokens, the Omni protocol introduces a novel concept in lending protocols we term "risk tranches". A risk tranche is defined as a set of unique assets that may be used as collateral to borrow the assets deposited into a given asset pool. Tranches have an ordering from lower risk to higher risk, e.g. if USDC is low risk and BETA is high risk. There are two important invariants that tranches follow:

1. The deposits ($d$) available to higher risk tranches ($t$) are always available to lower risk tranches.

$$d_{t1} \geq d_{t2} \geq d_{t3} \geq d_{t4}$$

2. Interest ($i$) earned from lower risk tranches ($t$) is split with all higher risk tranches proportionately.

$$i_{t1} \leq i_{t2} \leq i_{t3} \leq i_{t4}$$

Invariant (1) draws on the practical insight that if you are willing to lend to more risky assets, you are also willing to lend to less risky assets. For example if you wish to lend to BETA, you will also be willing to accept lending to USDC as well. This invariant significantly simplifies the user experience for the lender, while maintaining separation to tailor risk to a lender's risk profile.

Next, Invariant (2) follows logically from Invariant (1), where as interest should be earned where deposits are available. Since deposits are available from higher risk tranches to lower risk tranches, this means that interest earned by lower risk tranches is also paid to higher risk tranches. This guarantees that there is zero fragmentation of capital available to borrowers of assets, and ensures that there is maximal yield earned.

The risk tranche that a collateral asset is assigned to determines the risk tranche that a borrower is able to loan liquidity from. For example, if USDC is assigned to the lowest risk tranche, then it is able to borrow from liquidity deposited in all tranches. If SHIB is assigned to the highest risk tranche, then it is only able to borrow from liquidity deposited in the highest risk tranche, while all deposited liquidity from tranches lower than it are not exposed to the risks of borrowers who use SHIB as a collateral asset.

### 3.2. Actions

There are four main actions users can take when interacting with a borrowable asset: deposit, withdraw, borrow, and repay. When interacting with a non-borrowable asset (i.e. collateral only) there will be two actions: deposit and withdraw.

#### 3.2.1. Deposit

When depositing an asset, i.e. an oToken, users need to select a risk tranche that they wish to deposit their tokens in. As mentioned before, this risk tranche determines the set of collateral allowed to borrow the deposited tokens. In return for the deposits, the user will receive a share of the underlying pool, that can be redeemed for the proportional amount of the pool. For oTokens that are not borrowable, a risk tranche does not need to be selected.

#### 3.2.2. Withdraw

To withdraw, users need to select a risk tranche as well, as the shares a user has are specific to a risk tranche. Users are only allowed to withdraw if their account positions are healthy, and if the risk tranche is not paused by the protocol. Protocol pausing is explained in the risk management section. Additionally, in order for a withdrawal to be allowed there must be available capital. Due to the incorporation of risk tranches, it is possible for a lower risk tranche to borrow assets from a higher risk tranche. This will influence the raw deposit amount, however there is an invariant that we follow to ensure all withdrawals and borrows are valid. It is that the cumulative total borrow of the asset must be less than or equal to the cumulative total deposit when summed from the highest to lowest tranches. The invariant is as follows, for the given max tranche count $T$:

$$\sum_{i=T}^{t} b_i \leq \sum_{i=T}^{t} d_i \quad \forall t \leq T$$

By using this invariant, it greatly simplifies the algorithmic and implementation complexity of the problems introduced by the risk tranche concept.

#### 3.2.3. Borrow

To borrow, users must interact with the OmniPool contract as the entry point. Prior to borrowing, the user must mark the collateral assets they wish to use as collateral, and also mark the borrowing asset so that the OmniPool knows which assets to track. When marking an asset as collateral or for borrowing, the same state is updated. Therefore, if a user marks an asset for borrowing it is also marked as collateral, and can be used as collateral as long as the collateral factor of the market is non-zero. A borrow is considered valid as long as there is sufficient collateral and the same invariant followed by withdraw is also valid. As a reminder, the invariant is as follows, for the given max tranche count $T$:

$$\sum_{i=T}^{t} b_i \leq \sum_{i=T}^{t} d_i \quad \forall t \leq T$$

#### 3.2.4. REPAY

To repay, users must interact with the OmniPool contract as the entry point. When a user repays, the user sends the assets that were loaned back to the asset pool.

#### 3.2.5. INTEREST

Interest is paid out proportionately based on the deposit amounts in each of the risk tranches. When determining which risk tranches are eligible to receive interest for a given interest rate payment of a loan, we look at the second invariant we follow: interest $(i)$ earned from lower risk tranches $(t)$ is split with all higher risk tranches proportionately.

$$i_{t1} \leq i_{t2} \leq i_{t3} \leq i_{t4}$$

### 3.3. Supply and Borrow Caps

The protocol implements supply caps specifically for assets that are non-borrowable, and borrow caps specifically for assets that are borrowable. The protocol maintains that high risk assets should not be simultaneously shared for collateral usage and borrowing at the same time. It is possible to create two separate token wrappers of an underlying high risk asset, where one is non-borrowable and can be used as collateral, while the other is borrowable but cannot be used as collateral. Assets that are able to be used as collateral and borrowable should exclusively be high quality assets as determined by the protocol. Assets that are considered high risk by an external risk evaluation framework should always be wrapped using the non-borrowable OmniTokenNoBorrow contract, whereas low risk assets or high risk assets that are borrowable but not able to be used as collateral should be wrapped using the OmniToken contract. Supply caps are not dependent on tranches, while borrow caps are dependent on tranches, as when borrowing using a given asset it is specific to a single tranche.

## 4. Risk Management

Risk management for the protocol is composed of two parts: on-chain and off-chain. The on-chain risk management components involve parameter configuration, oracles, and interest rate models. The off-chain risk management component involves risk analysis prior to listing markets, monitoring loan positions for liquidations, and resolving bad debt if it occurs.

### 4.1. Asset Configurations

The protocol is able to configure the collateral factor, borrow factor, expiration time, risk tranche, and isolated collateral flag for every listed asset on the Omni protocol. The collateral factor is a multiplier applied to the assets value if it is being used as collateral. The borrow factor is a multiplier applied to the assets value if it is being taken out as a loan. Additionally, if the same asset is being used for lending and borrowing then there is a special self collateralization factor due to the price correlation. The purpose of these factors is to apply a buffer to the asset's value to ensure that there is sufficient time and incentive to liquidate an unhealthy loan without accruing bad debt to the protocol.

The risk tranche assigned indicates what deposits will be available to the account using the asset as collateral. Next, the purpose of the expiration timestamp is based on the notion that crypto markets are cyclical and require active market configuration re-evaluation from time to time for most assets. By providing an expiration timestamp up front, this enables the protocol to be transparent with lenders and borrowers how long a loan is able to be outstanding using a given asset as collateral. If the current block timestamp is greater than the configured expiration timestamp, then that means the collateral asset can be liquidated.

Finally, the isolated collateral flag is a unique feature for more risky collateral assets. This flag indicates that no other isolated collateral assets can be used in conjunction with the current isolated collateral asset, however non-isolated collateral assets can be used as cross-collateral at the same time. The purpose of enabling assets in the 0 risk tranche to be used as collateral in conjunction with a single isolated collateral asset, is to ensure that if the value of the isolated collateral asset depresses the loan position can be saved from liquidation with other high quality assets. Otherwise, if a borrower is only able to top up their loan position with the same asset it could potentially lead to a doom loop that makes it even more difficult for a safe liquidation to occur. Additionally, if the isolated collateral flag is false, than the asset must be set to the 0 risk tranche.

### 4.2. Mode Configurations

The protocol is able to make better risk assumptions when it knows specific assets are being used for lending and borrowing. This feature is most similar to the high efficiency modes in the Aave protocol, where if a borrower is only using stablecoins then they get more favorable loan terms. When determining a mode, it enforces that there are only certain markets that a user can count as collateral and only certain markets that they are able to borrow. By agreeing to only interact with a restricted set of markets, the protocol is able to make more aggressive risk assumptions and give a borrower higher collateral and borrow factors resulting in greater capital efficiency.

Modes are also configured with an expiration timestamp, so that once the timestamp has passed users are no longer able to enter into the mode anymore. However, for users that previously entered the mode account prior to the expiration timestamp, their accounts will not be affected. The expira-

tion of a mode does not affect the liquidatability or health of the account that has already entered the mode.

### 4.3. Enter/Exit Markets/Modes

To successfully be able to take out loans from the Omni protocol a user must have entered markets or entered a mode to determine which assets will be counted as collateral and which assets are available to be borrowed. Users are not allowed to enter a market more than once, and are not allowed to enter the market if it is expired. Additionally, only one isolated collateral market is able to be used at any given time. For the isolated collateral market as well, a user is only allowed to enter an isolated collateral market when they have no active borrows. This is because the isolated collateral market can influence the borrow tier to be non-zero, whereas non-isolated collateral markets should not influence the borrow tier for the account.

In order to enter a mode, a user must not have entered any markets, isolated or not. In order to exit a market or exit all markets, a user must not have any borrows active on their subaccount. Similarly, for a user to exit a mode they must not have any borrows.

### 4.4. Oracles

For oracles, the protocol allows the support of Chainlink, Band, and a Custom oracle. The protocol should most commonly use Chainlink or Band oracles when available, but if unavailable may defer to a Custom oracle that may be a Uniswap V3 oracle implementation or another similar custom oracle. The protocol normalizes the prices reported by these oracle providers to return the price of a token in one base unit, e.g. wei, satoshi, etc. depending on the underlying assets decimals. Additionally, the protocol configurators should be very careful when setting the oracle address for an asset market to a provider that is not Chainlink or Band, as it is a privileged method.

### 4.5. Interest Rate Model

The interest rate model uses a linear kink model, where prior to a kink there is a flatter slope and after the kink there is a steeper slope. However, the interest rate is not only dependent on the utilization, but also the risk tranche of the collateral asset being used for the loan. This is because higher risk collateral assets should also pay a higher interest rate to compensate the lenders for taking greater risks, while a lower risk collateral asset should be able to pay a lower interest rate as it is a safer asset. Therefore, when configuring the interest rate model for each market, there is a different set of parameters for the curve for each risk tranche of the asset.

### 4.6. Reserve

The protocol can be configured to have a reserve that is paid a fee for all interest paid by borrowers. This is an optional configuration to the protocol.

## 5. Subaccounts

The Omni protocol introduces a novel implementation for handling subaccounts for users that requires zero gas frees to create and is guaranteed to be collision free, that is the first of its kind for lending protocols. One of the biggest grievances of existing lending protocol UX is the difficulty of managing loan positions and maintaining different loan configurations simultaneously. With Omni, users are able to create as many as $2^{96} - 1$ subaccounts for a single wallet address. This is accomplished by taking advantage that each key slot in a mapping is a 32 byte object, and addresses are only 20 bytes. Thus, we can take advantage of the extra 12 bytes of data available to create unique subaccount addresses based on the user's wallet address.

This also means that whenever users perform an action with the protocol: deposit, withdraw, borrow, and repay, users will also be selecting which subaccount they want to interact with in addition to the other required parameters. Each subaccount can be considered as a clean address, where the deposits and borrows of one subaccount are completely isolated from all other subaccounts.

Through subaccounts, managing positions for Omni is now 10x easier compared to any other lending protocol that exists.

## 6. Liquidations

The Omni protocol uses dutch auctions based on the health factor of the loan position when conducting liquidations. Additionally, liquidations are allowed to occur in two situations:

- If the factor-adjusted value of deposits is less than the factor-adjusted value of borrows.
- If the collateral used for the loan position is expired, i.e. the current block timestamp is greater than the expiration timestamp configured for the collateral asset parameters.

For liquidations that result in bad debt, the Omni protocol will automatically pause the protocol such that users are no longer able to withdraw or borrow from the the risk tranche with bad debt and any risk tranches greater.

It is strongly recommended that users exercise caution when depositing into the protocol when the risk tranche is paused,

as it could lead to potential loss as the protocol has the ability to socialize the bad debt loss proportionately amongst all depositors.

### 6.1. Dutch Auction Liquidations

Liquidations follow a dutch auction based on the health factor of the position. Furthermore, liquidation bonuses are asset specific, so some assets that are less risky to liquidate will have lower liquidation bonuses, while those assets more difficult to liquidate will have higher liquidation bonuses.

There is also a separate liquidation bonus given to assets where the current timestamp has exceeded the expiration timestamp and the loan position is still healthy.

### 6.2. Soft Liquidations

Instead of liquidating a fixed amount or percentage of the violator's unhealthy loan position, the Omni protocol instead opts for a soft liquidation. This means that liquidators are able to liquidate a user's account up to a specific health factor. The threshold is also asset dependent, so safer assets will be liquidatable up to a lower health factor threshold compared to more risky assets.

The protocol additionally provides a way for specific accounts to have special liquidation thresholds. The purpose of this is to add flexibility to how credit is managed on the protocol in the future.

### 6.3. Bad Debt Pausing and Socialization

The protocol will automatically trigger a pause to the risk tranches of the protocol whenever bad debt occurs from a liquidation. When the risk tranche is paused, users are unable to withdraw and borrow, but are allowed to deposit and repay to keep any active loan positions healthy to reduce risk of further liquidations and further bad debt to the protocol.

The protocol has the ability to give an administrator the right to socialize the bad debt losses from specific loan positions. The loss socialization method can only be called when there is no more collateral left to liquidate for the given position and bad debt has accrued to the protocol. In favor of fairness, the administrator is able to call a custom loss socialization method that will proportionately distribute the bad debt loss to all lenders of a token that were in the risk tranche of the affected loan or higher risk tranches. It is recommended that a liquidation call is always chained with a loss socialization call to ensure that the bad debt account does not have any collateral leftover prior to loss socialization. Otherwise, the bad debt account will be able to have their debt removed while keeping ownership of any collateral not taken already.

### 6.4. Additional Notes for Liquidators

Additionally, when a liquidator is receiving their reward for liquidation, the protocol opts to greedily try to obtain as much reward as possible. However, it is potentially possible that an account does not have sufficient rewards to reward the liquidator, e.g. when a liquidator liquidates an account that will trigger bad debt. The liquidation will still process correctly regardless of the reward received by the liquidator. It is strongly encouraged that the liquidator check the returned values from the liquidation call and perform any necessary checks and reversions based on the amount of rewards received for liquidating.

## 7. Conclusion

The Omni protocol introduces a groundbreaking approach to decentralized lending, addressing many of the challenges faced by existing models. By leveraging a novel concept of "risk tranches," Omni ensures maximal capital efficiency and zero liquidity fragmentation. This allows lenders to tailor their risk profiles while ensuring borrowers have access to maximum liquidity.

Omni's architecture is designed to be dynamic, composable, and safer, accommodating all collateral types and borrow types. It offers unique features such as collision-free sub-accounts for asset management, high-efficiency borrowing modes, and dynamic liquidations using dutch auctions. Furthermore, the protocol's risk management is both on-chain and off-chain, ensuring robustness and adaptability in the ever-evolving crypto landscape.

The Omni Protocol aims to push the decentralized lending space forward by offering a more inclusive, efficient, and fluid money market system. Its innovations aim to foster a more inclusive and efficient financial ecosystem, bridging the gap between traditional and decentralized finance.