Venus V4 Technical Paper

July 2023

1 Introduction

Launched in 2020, Venus Protocol is pushing the edges of decentralized finance through its novel recombination of existing solutions and deployment on BNB chain, lowering the barrier to entry for millions of new users around the globe. By unifying stablecoin minting introduced by Maker and algorithmic money markets developed by Compound, Venus simplified the user experience and provided core capabilities that enabled decentralized finance to flourish in a single application. As a result, Venus found remarkable success and quickly rose to be one of the most widely used decentralized applications in blockchain finance.

The latest iteration of Venus builds on prior successes and lessons learned to improve across 3 key themes: risk management, decentralization, and user experience.

1.1 Risk Management

Venus V4 improves risk management on several fronts: isolated pools for safely onboarding long tail assets, an industry-first price feed composed of multiple oracles to eliminate a single point of failure, mechanisms for maintaining the VAI peg, more sophisticated risk parameters to give the protocol increased protection against insolvency.

Isolated Pools The first generation of lending protocols, including Venus, combine assets into a single liquidity pool. This introduces significant risk to the entire protocol's liquidity if any token included in the pool experiences extreme volatility. In addition, it is challenging to list new tokens due to the lack of specific risk parameters.

Venus V4 introduces isolated pools, designed to tackle all of the shortcomings of composing a protocol with a single liquidity pool. Isolated pools are composed of segregated collections of assets with custom risk management configurations, offering greater diversification to manage risk and to lend and borrow. Isolated pools quarantine hypothetical failures from spilling over into unrelated markets and affecting the risk state of the entire protocol. Furthermore, isolated pool rewards can be customized per asset in each pool to provide tailored liquidity incentives to users.

Risk Fund and Shortfall Handling Shortfall accounts, accounts that have borrowed beyond the value of their collateral, are a significant risk to decentralized lending protocols. There is little incentive to pay back these loans because the collateral that will be unlocked is worth less than the loan. These accounts are a burden on the protocol's liquidity and prior to Venus V4 there was no mechanism to recover these accounts.

Venus V4 will also maintain a risk fund for each pool, where a percentage of protocol revenue will be allocated and used to cover insolvencies. If after liquidation there is an insolvency shortfall handling mechanism is activated via auctioning the risk fund for the appropriate asset.

VAI - Maintaining the peg VAI is a stablecoin backed by a basket of collateralized digital assets. The original implementation has two major flaws. It lacks a strong mechanism to keep its value pegged to $1 and users were able to mint VAI up to the total supply minus their total borrow, which could put them at risk for immediate liquidation.

Venus V4 introduces a stability fee as designed by Maker to keep VAI's value stable without artificially setting the price to $1. This mitigates risks associated with VAI de-pegging due to low liquidity. The quantity of VAI a user can mint is also updated to account for their weighted supply. A global minting cap on VAI is also introduced. Restricting how much total VAI can be minted will reduce the likelihood of depegging and unwanted liquidations and mitigate risks to the health of the protocol.

Finally, in order to reduce the bad debt associated with VAI, Venus V4 forces liquidators to liquidate VAI first before other assets in accounts with underwater positions.

Liquidations Lending protocols need to handle asset price changes gracefully because shortfall accounts put the protocol's liquidity at risk. To prevent this from happening, lending protocols have a "liquidation" mechanism.

When an account's collateral drops below a predefined threshold, liquidator bots acting for their own profit sell a part of the account's collateral on the market and repay the debt of the borrower. The threshold for liquidation depends on the quality of the collateral – for more volatile assets the liquidation threshold is lower, meaning more collateral is required to secure a position from liquidations.

Liquidations are an important part of risk management. When a market is moving, liquidators need enough time and incentive to react to changes. It is also important to make the protocol fair from the users' perspective because the liquidated user loses a certain percentage of their collateral.

Prior to V4, Venus allowed liquidating up to 50% of the borrow, and had configurable collateral factors for assets and a single global liquidation incentive. Although this setup worked well for the Core Pool, it has several known issues:

- Underwater positions are never liquidated in full. Liquidations happen in "rounds", where 50% of the borrow can be liquidated per liquidation event. For example, if a user has $1,000 worth of collateral, and $900 debt, the loan will be liquidated in chunks – $450 will be repaid in the first liquidation event, $225 in the second event, etc. After 8 liquidations the user will have $3.51 in borrows secured by $3.87 collateral. At this point, liquidating this position is no longer profitable because the gas will be higher than potential profits for the liquidator.
- It isn't possible to determine on chain if liquidators have enough incentive to perform required liquidations. However, to track the total bad debt, it is important to distinguish the actual account insolvency from a position that can potentially be liquidated further.
- Liquidation incentives are not correlated to the quality of the collateral. Liquidators are not incentivized to seize more volatile collaterals. Thus liquidators often choose to seize stablecoins to mitigate the market risks, leaving accounts with volatile collateral underwater.
- It is not possible to decrease the collateral factor for a certain asset without triggering liquidations. This may lead to additional sell pressure for the collateral asset, potentially triggering even more liquidations.

Venus V4 changes the liquidation logic to account for these issues.

1. The liquidation threshold is configured separately from the collateral factor. For example, setting the collateral factor to zero will prevent new borrow positions without affecting the solvency of existing loans. Another benefit is that users will be able to borrow 100% of their borrowing limit without putting them at risk of immediate liquidation.
2. Liquidation incentive can be configured per asset, rather than globally.
3. Two special kinds of liquidations, batch liquidation and account healing, are introduced to allow liquidating the position in full. Batch liquidations incentivize liquidators to liquidate small accounts. Account healing handles bad debt by allowing a liquidator to seize remaining collateral and write off left over bad debt.

Resilient Price Oracle Prior to V4, Venus relied solely on a single price oracle network to support its markets. It didn't have a mechanism to validate prices and to protect against price manipulations or stale data which present an existential threat to the protocol and create a single point of failure.

Venus V4 introduces a resilient oracle that can fetch prices from multiple feeds and validate them using other decentralized sources. A price validation algorithm is used to verify prices between two or more price oracle sources. If a primary source is found to be untrustworthy or fails to return, the resilient oracle can fall back to a secondary source.

Another problem with the original oracle architecture is that it didn't support integrating new oracles. The new resilient oracle design allows for the integration of new price oracles on the fly and supports enabling and disabling price oracles per asset.

1.2 Decentralization

Venus's governance model, originally started as a fork of Compound's token based model, was modified to use a vault based on the native token, XVS. Participants in Venus Governance are required to lock XVS in a vault, and thereby representing their voting power. Protocol updates are made through Venus Improvement Proposals (VIPs), and need to be voted on and queued in a 48 hour timelock. Long voting and timelock periods give users time to vote, provide the community transparency over proposed changes, and allow prevention measures to be made in the event a malicious proposal slips through.

However, this rigid governance model has led to multiple adverse effects on both Venus and Compound. Some examples of potentially damaging delays are:

- Compound's proposal #117 accidentally suspended the price feed for cETH. Although the rollback proposal was submitted 30 minutes after the problem was noticed, the market was offline for 7 days. This could have potentially led to unfair liquidations and/or protocol insolvency should the price of ETH fluctuated.

- Following Venus' pause during the LUNA incident, the protocol stayed offline for several days during the voting and timelock periods. Even though pausing is a serious situation, it is critically important to unpause the protocol after the situation is resolved to prevent further damage brought by changing market conditions.

Additionally, this rigid governance model does not allow for timely adjustment of risk parameters to provide competitive rates. If risk managers are required to recommend collateral factors and interest rates weeks in advance, they need to choose more conservative values to account for unknown market conditions.

Venus V4 introduces several major changes to the governance model. While accounting for the needs of decentralization and transparency, the new model reflects that the changes one can propose can be inherently different. For example:

- It should be possible to pause specific actions on individual markets immediately if the status quo can be abused by an adversary.
- Interest rate model parameters should be adjustable without going through a long governance cycle to offer more competitive rates.
- Upgrading contracts or reassigning roles should require a longer governance cycle so that the community can review the proposed changes and vote against malicious or erroneous proposals.

Taking into account these distinct types of possible updates, Venus deploys a new governance model featuring fast-track VIPs, role-based access control and fine-grained pause mechanism.

1.3 User Experience

Simplified supply and borrow

[pending]

Reward distributor Currently users are rewarded for supply and borrow activities across all markets in the Core Pool with the XVS token. Although this reward system cannot be customized it has worked well so far.

Venus V4 introduces two major upgrades to thereward system. Rewards can be configured per market and lending activity (supplying or borrowing). Multiple reward tokens are supported allowing even greater incentives to users.

Stable rate borrowing Venus earns the difference between interest paid by borrowers for taking out a loan and the interest paid to suppliers for providing liquidity. In previous iterations, these rates were determined by the liquidity and demand for the asset in the market. In this interest rate model, borrowing and lending rates can change dramatically, which means users can have a much higher interest rate at the time of repayment than when they took out the loan. The income a user earns for supplying liquidity can also vary making it difficult to predict income.

Venus V4 introduces stable rate borrowing in addition to variable rate borrowing. Stable rate borrowing offers more predictable interest rates similar to those found in traditional finance offerings. Stable rate borrowing gives users the ability to predict their expenses and to hedge their investments with more certainty.

Venus Prime A new mechanism is introduced designed to incentivize protocol utilization and security via the XVS Vault called Prime. Holders who qualify to receive a Prime Token are given access to supplemental liquidity mining rewards on Prime activated markets.

There are two types of Prime Tokens both having the sole use case of providing access to Venus Prime rewards: irrevocable and revocable, which are earned based on protocol usage and XVS staking history. The rewards apply to both versions of the Prime Token.

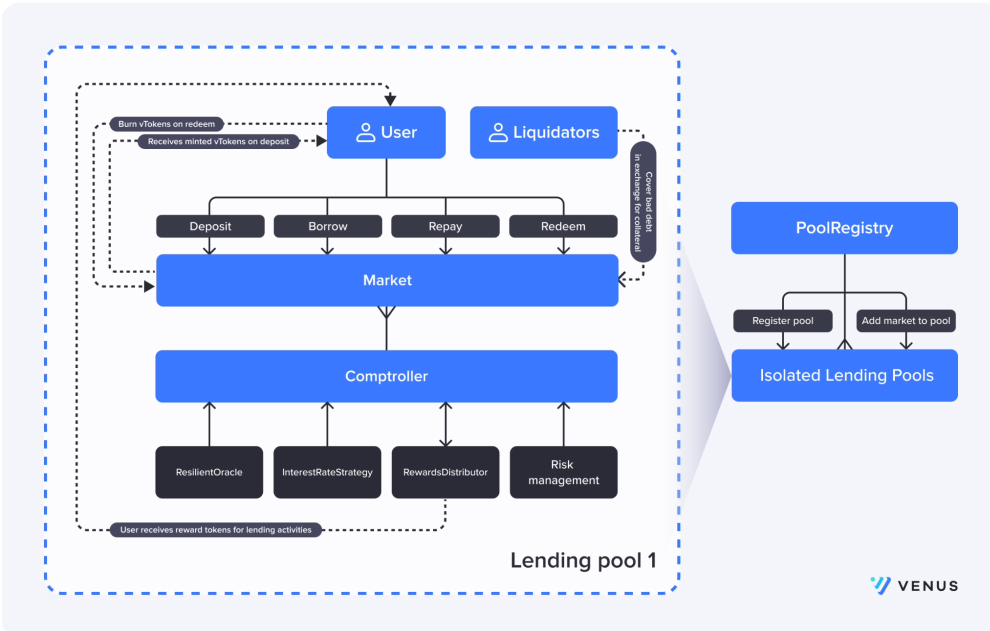
2 V4 Overview

Venus V4 is designed from thoughtful analysis of the evolution of the protocol and its ecosystem to tackle all of the shortcomings that the previous version had. It introduces greater risk management controls, upgrades decentralized governance and gives users new tools and features for allocating their assets in the protocol.

2.1 Risk Management

Isolated pools The Isolated Pools architecture centers around the PoolRegistry contract. The PoolRegistry maintains a directory of isolated lending pools and can perform actions like creating and registering new pools, adding new markets to existing pools, updating pool metadata, and providing getter methods to fetch pool details.

To create a new lending pool, the PoolRegistry deploys the pool's proxy and sets the Comptroller implementation address to the proxy. After setting up the Comptroller, it adds the pool to the pool directory.



Isolated Pools Architecture

To add a new market to any existing lending pool, the PoolRegistry deploys a JumpRateModelV2 or a q contract using the relevant factory, as the interest rate model and then deploys the upgradable VToken for the market before getting the support of the market's Comptroller.

The PoolRegistry also provides setters to update pool details like *setPoolName*, *updatePoolMetadata*, and *bookmarkPool* and getters to retrieve the details of a pool like *getPoolByComptroller*, *getVenusPoolMetadata*, and *getBookmarks*.

Users can perform the following actions on any market in a pool:

1. Deposit: when a user deposits an asset (for a specific market), they receive vTokens that correspond to the liquidity deposited and accrue interest for the deposited underlying assets. vTokens are minted upon deposit and increase with accrued interest until they are burned on redeem or liquidated.
2. Borrow: this action transfers assets to the user in exchange for the collateral that remains locked and cannot be transferred.
3. Redeem: this action allows users to redeem vTokens for the underlying asset as per exchange rate.
4. Repay: this action allows the user to repay the borrowed asset and accrued interest.

Risk Fund and Shortfall Handling Because Isolated Pools can provide lending and borrowing markets for high risk tokens, there is an increased chance accounts activity in these markets can become insolvent due to bad debt caused by extreme fluctuations in prices. Venus provides a sophisticated framework to prevent pool insolvency.

Every pool has an associated risk fund, which receives 40% of the income generated by the pool (interest payment share and liquidation bonus share) in the form of USDT. The risk management framework recovers the bad debt of insolvent accounts by auctioning the risk fund reserve.

When there is a bankruptcy address and no liquidator, the insurance funds can help to repay the bad debt and compensate for the loss.

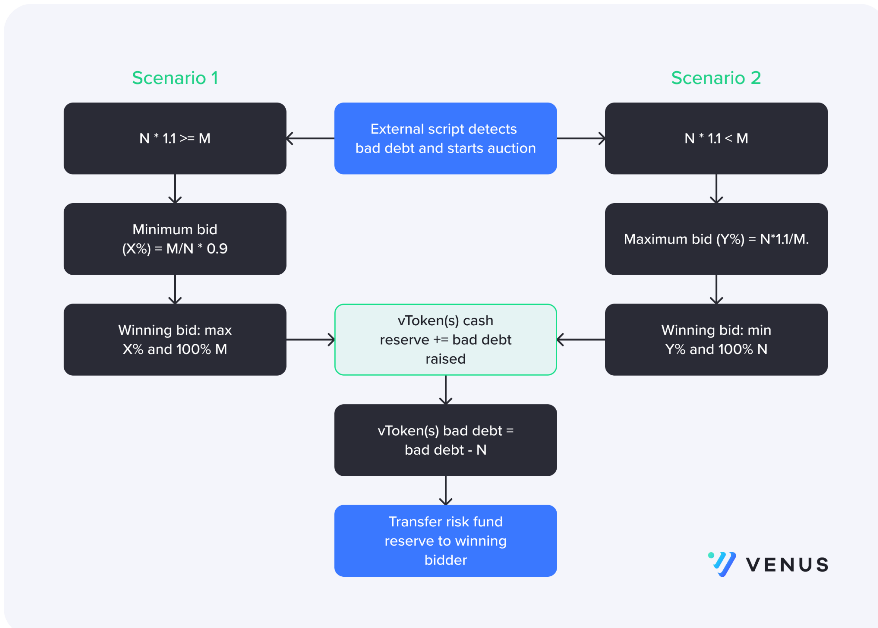
Protocol Share Reserve Protocol share reserve acts as a treasury for all of the isolated pools to which each market can transfer their income in the underlying asset (known as the market reserve). This protocol income is collected from accrued interest and liquidating/seizing accounts.

Bad Debt and Shortfall Handling A set of processes is executed in a market when a shortfall is detected for the borrower to pause the interest accrual on the borrow, write off the borrower's borrow balance and track the market bad debt.

V represents the total bad debt including the accrued interest on the bad debt. To calculate the accrued interest the borrow index when the bad debt is detected is divided by the initial borrow index and then multiplied by the borrowed amount. For example, if the initial borrow index is 1.2 and when bad debt of 100 BUSD is detected the borrow index is 1.5 then the V (total bad debt) becomes 100 * (1.5/1.2) = 125 BUSD.

The calculated total bad debt of the account is written off for the borrower and interest accrual is stopped on the bad debt.
When the pool's bad debt reaches a minimum amount the risk fund reserve is auctioned off to cover the bad debt. Anyone will be allowed to start or restart an auction if the constraints (in terms of the status of the last auction and the bad debt accumulated) are satisfied. Auction participants receive a maximum 10% incentive (configurable by the community via VIP) for covering the bad debt. Depending on the size of the reserve in the risk fund, either 100% of bad debt or a portion of it is raised.

N represents the total pool's bad debt denoted in USD and M represents total risk fund balance in USD. When an auction begins, a starting bid is set to prevent bidders from taking advantage of the auction by opening with an undervalued bid. The highest bidder's funds are locked, and when the auction closes the market(s) total cash reserve is increased,the bad debt of the market(s) is decreased and the risk fund partially/completely transferred to the winning bidder.



Auction scenarios

In scenario 1, X% indicates the percentage of bad debt the bidder is willing to pay and in scenario 2, the Y% indicates the percentage of the risk fund the bidder is willing to seize. During the auction, bidders are only allowed to specify X% or Y% depending on the type of auction.

A bid will be successful only if the bidder has sufficient funds to cover the bad debt they are bidding for and they make the best offer. When a bid is placed it is transferred to the Shortfall contract, and released if they are out bid. Subsequent bids should be placed within 10 blocks (adjustable via VIP) of the previous bid , otherwise anyone (including the winning bidder) can close the auction. If there is no bid for 100 blocks (adjustable via VIP) the auction can be restarted accounting for any changes in the risk fund and bad debt balance.

The auction process attempts to cover as much market bad debt as possible. In scenario 1, all of the bad debt may not be covered by the auction. In this case, the bad debt not covered will be maintained in the system until a new auction is started.

VAI - Ensuring the peg VAI is a stablecoin issued by Venus protocol, which users can mint based on their collateral position in the Venus Core pool.

Stability Fee and Optimised Minting Currently there are few use-cases for VAI so its liquidity is very low, which leads to Venus having to artificially set the VAI price to $1. Because the protocol fixes the value of VAI to $1 so it won't de-peg, it can't reflect the market demand.

Venus V4 introduces a stability fee as an efficient way to keep VAI pegged. Users are required to pay the stability fee when repaying minted VAI. It is continuously added to the user's VAI minted balance every block. The annual fee is calculated as follows:

$$stabilityFee(\%) = baseRate(\%) + max(0, (1 - currentPriceOfVAI)) * floatingRate(\%)$$

- *baseRate* is a flat rate that users always pay
- *max(0, (1 - currentPriceOfVAI)) * floatingRate* is a variable rate users need to pay, based on outstanding VAI. This incentivizes users to burn/mint according to the price of VAI. This variable rate will be always greater or equal 0, so the minimum stability fee will be the *baseRate*. This component of the formula has the following effects:
  - When the VAI market price is below $1, the stability fee is increased automatically. Borrowers will be incentivized to repay the minted VAI, reducing the circulating supply of VAI, and therefore increasing its market price.
  - When the VAI market price is above $1, the stability fee will be fixed to the baseRate, not changed dynamically. Potential borrowers could mint VAI and pay the minimum interest rate.

Revenus from the stability fee will be directed towards handling extreme market conditions like bad debt. Assuming there are ~10512000 blocks per year on the Binance Smart Chain, the *ratePerBlock* is calculated as follows:

$$ratePerBlock(\%) = stabilityFee(\%) / 10512000$$

Interest is continuously accrued at a growth rate per block following the previous formula. To implement calculation of fee a user needs to pay based on *ratePerBlock*, a global index is used and updated during every VAI related transaction. The initial value of global index is 1e18 and updated according to the following formula:

$$globalIndex = globalIndex + ratePerBlock * (currentBlockNumber - lastUpdatedBlockNumber)$$

$$lastUpdatedBlockNumber = currentBlockNumber$$

During minting the *globalIndex i*s set as the user's *minterIndex*. Past interest of the user i.e., interest accrued by the user, will also be tracked. The formula to calculate the total VAI amount a user needs to repay with interest is :

$$repayAmount_t = mintedVAIOfUser_{t-1} + ((mintedVAIUser_{t-1} - pastInterestOfUser_{t-1}) * (globalIndex_t - minterIndex_t))$$

$$pastInterestOfUser_t = pastInterestOfUser_{t-1} + (repayAmount_t - mintedVAIOfUser_{t-1})$$

VAI Minting Rate and Minting Cap Previously, the maximum amount of VAI a user could mint was based on their total supply. This created a risk of immediate account liquidation if a user minted more VAI than their collateral value. Venus V4 uses the user's weighted supply to limit the amount of VAI they can mint. This protects the users from accidentally putting their account at risk of liquidation by minting too much VAI.

A global minting cap is also added which ensures the total VAI minted by all minters cannot pass a specified amount. This works exactly like how the borrow cap works for vToken markets.

Liquidations Liquidations are especially important to support volatile collateral. Venus V4 is built with this use case in mind. It features several improvements over the traditional liquidations flow, including separate liquidation thresholds, per-market liquidation incentives, batch liquidations and account healing.

Liquidation threshold In previous version of Venus, the liquidation threshold is equal to the collateral factor, so the maximum amount you can borrow is strictly equal to the amount that would lead to an immediate liquidation:

$$liquidity = \sum_{i=0}^{N}(collateralFactor_i \cdot suppliedAmount_i \cdot price_i) - \sum_{i=0}^{N}(borrowedAmount_i \cdot price_i)$$

Venus V4 adds some buffer by setting $\forall i$: $liquidationThreshold_i \geq collateralFactor_i$. When checking the possibility of liquidations, Venus will now use liquidation threshold instead of collateral factor. This way, our users can borrow up to 100% of their borrow limit without getting liquidated immediately after that. This change also allows risk managers to decrease collateral factors without triggering liquidations immediately.

Per-market liquidation incentive

Venus V4 allows the risk managers to configure liquidation incentives for each market rather than using a fixed value for the whole pool or protocol. While computing the amount the liquidator can seize, Venus V4 uses the collateral liquidation incentive instead of the global one:

$$seizeAmount = \frac{liquidationIncentive_{collateral} \cdot repayAmount \cdot price_{borrowed}}{price_{collateral}}$$

Batch liquidations It is a common problem for lending protocols to have small undercollateralized positions. Such small positions are usually the result of the liquidations themselves: when liquidators are allowed to repay up to a certain percentage of the borrow, they perform partial liquidations as long as they are profitable and stop afterward.

To avoid this, Venus V4 introduces batch liquidations that are mandatory for liquidating small positions. During batch liquidation, a liquidator must repay all borrows of an account. While this does not guarantee that all liquidations are profitable, this would reduce the number of such undercollateralized positions.

Account healing Although the improved liquidation mechanism will reduce the number of underwater positions, Venus V4 specifically targets high-risk assets, so it must have measures to deal with insolvency. One of these measures is the “account healing” mechanism.

If an account becomes insolvent, it is possible to “heal” it by seizing the entire collateral and repaying as much outstanding debt as possible. When healing happens, the bad debt is increased. This bad debt can later be covered by the risk fund.

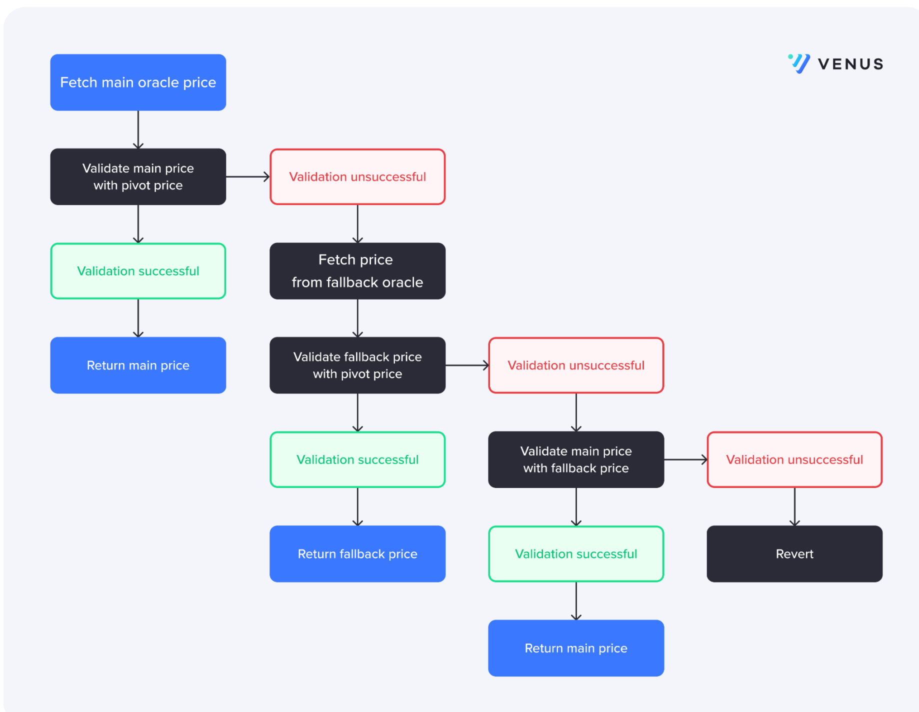
Resilient Oracle DeFi Protocols are usually vulnerable to price oracles reporting incorrect prices. There are various ways in which oracle prices can be manipulated depending on the type of price oracle used which can create a single point of failure and opens several ways for attacking the protocol.

The resilient oracle design avoids a single point of failure by using multiple oracle sources and a price validation algorithm to provide accurate prices and protect from oracle attacks. The V4 launch includes integrations with Chainlink (and any Oracle compatible with the Chainlink Aggregator interface), Pyth Network, Binance Oracle and TWAP (Time-Weighted Average Price) oracles. TWAP uses PancakeSwap as the on-chain price source.

The resilient oracle configures a main, pivot and fallback for every market (vToken). The main oracle is the most trustworthy price source, the pivot oracle is used as a loose sanity checker and the fallback oracle is used as a backup price source.

When fetching an oracle price, for the price to be valid it must be positive and not stagnant. If the price is invalid then we consider the oracle to be stagnant and treat it like it's disabled. To validate the price between two oracles, an upper and lower bound ratio is set for every market. The upper bound ratio represents the deviation between the reported price (price that's being validated) and anchor price (price that is being validated against) above which the reported price will be invalidated. The lower bound ratio represents the deviation between the reported price and the anchor price below which the reported price will be invalidated. So for oracle price to be considered valid the below statement should be true:

$$anchorRatio \ = \ anchorPrice/reporterPrice$$
$$isValid \ = \ anchorRatio \ <= \ upperBoundAnchorRatio \ \&\& \ anchorRatio \ >= \ lowerBoundAnchorRatio$$



Resilient Oracle Price Validation Flow

2.2 Decentralization and Governance

Novel Proposal tracks Venus Governance now has three types of Venus Improvement Proposals (VIPs) – Normal, Fast-track and Critical. Normal VIPs will govern the most relevant updates, such as upgrading the contracts or changing the access controls. Fast-track VIPs will be used to adjust risk parameters such as interest rates or collateral factors. Critical VIPs will be used during emergencies or changes demanding the quickest reaction. Each of these VIP types will have its own proposal threshold, timelock and voting periods. This reflects the level of risk and impact of the changes introduced to Venus.

The exact parameters for these types of VIPs will be configured by Governance. Initially, these values will be:

- Normal VIP: 24 hours voting + 48 hours delay
- Fast-track VIP: 24 hours voting + 6 hours delay
- Critical VIP: 6 hours voting + 1 hour delay

Role-based access control

Instead of checking whether the caller is the "admin", many contract methods now rely on a separate Access Control Manager contract. The Governance can allow certain actions to go via the "fast-track" or the "critical" route, or even allow guardians to bypass voting and allow certain actions such as pausing to be called directly with a multisig. This is useful to set borrow and supply caps, pause certain actions on the markets or take other non-destructive measures in response to rapidly changing market conditions.

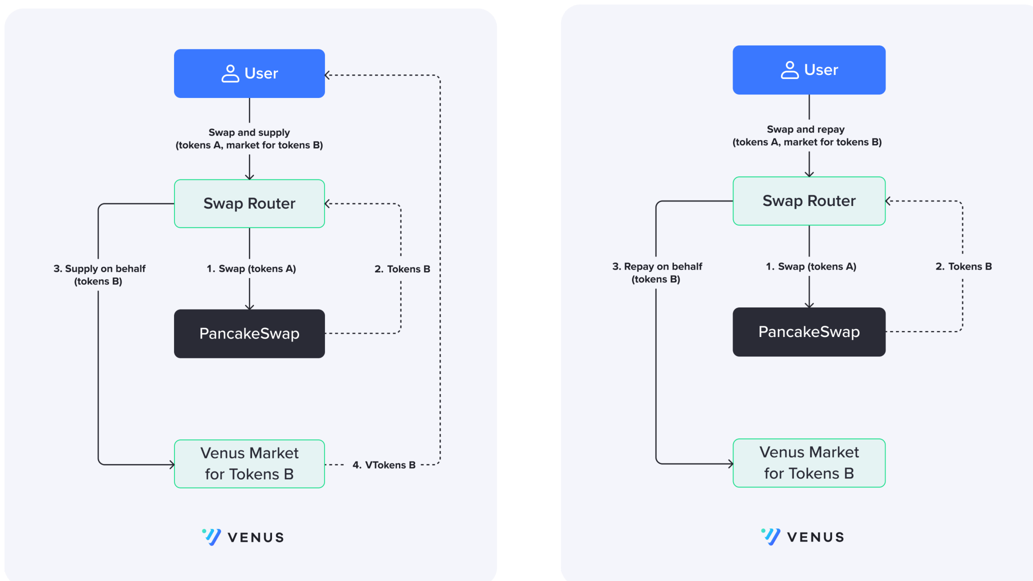
Fine-grained pause

It is now possible for the pause guardian to pause any action on any market individually. Instead of pausing the entire protocol to protect against an attack or to do damage control, the guardians can pause individual market actions such as supply, borrow, and enabling collateral.

2.3 User Experience

Venus V4 creates a better user experience by integrating swaps into supply and repayment behind the scenes via a third-party DEX. This is beneficial to users because they can participate in supplying to a pool of their choice without needing to own the supplied coin at the moment of the transaction.

In the first case, the user will be able to supply tokens B (that means, mint VTokensB), sending to the new SwapRouter contract tokens A. Internally, the swap is performed in PancakeSwap. In the second case, the user will be able to repay their debt in tokens B, providing tokens A. In a similar way, internally the swap is performed using PancakeSwap.



Swap tokens A for tokens B and supply tokens B, in one transaction. Swap tokens A for tokens B and repay the debt of tokens B, in one transaction

SwapRouter is the smart contract where this new logic is encapsulated.

In addition to creating a seamless user experience for lending activities, V4 includes features to help users with pool discovery. Users can browse pools by category to participate in supplying and borrowing tokens in a particular vertical such as gaming. Users can also view tokens ranked by APY and filter pools by token.

Behind the scene swaps and easy discovery gives users the powerful user experience of being able to find the pools they are most interested in and empowering them to interact with that pool without requiring they hold any particular token.

Reward Distributor Reward distributors are used to reward lenders and borrowers with the associated reward token based on the user's activities for the associated markets. One or more reward distributors can be added to a pool through *addRewardsDistributor*. Reward distributors can customize distribution rates for lenders and borrowers per market. Whenever a user interacts with a market rewards are allocated according to the vTokens accrued by the user according to the current borrow/lend reward speed for the respective assets in the pool.

The user experience for claiming rewards has been improved to allow users to claim rewards for individual markets. This new design reduces gas fees compared to the previous implementation because we only attempt to claim rewards from markets where rewards are claimable. Vault rewards can also be selected and claimed from the same model further streamlining the claiming process.

Stable Rate Borrowing

Stable rate borrowing gives users an alternative to variable interest rate loans. Stable-rate loans have their interest rates set at issuance until rebalancing. Variable rate borrowing has a per transaction resolution; it can change drastically with the involvement of each user (like deposit, borrow). On the other hand, users who borrow at a stable rate will be unaffected by individual actions of other users. Stable rates may change if there are large fluctuations in the market but will remain relatively unchanged.

Stable Rate Calculation At any point t, the utilization rate and stable loan adoption rate will be calculated as:

$$u_t = \frac{total\_borrow_t^{StableRate} + total\_borrw_t^{VariableRate}}{total\_supply_t}$$

$$stable\_loan\_wt_t = \frac{total\_borrw_t^{StableRate}}{total\_borrw_t^{StableRate} + total\_borrw_t^{VariableRate}}$$

Where, $total\_supply = cash + borrows + bad\_debt - reserves$

At any point t, the **variable interest rate** will be determined in the same way as the current protocol, for example using the Jump Rate Model:

$variable\_borrow\_rate_t\,(u_t) = a_1 u_t + b$, when $u_t < kink$

$variable\_borrow\_rate_t\,(u_t) = a_1 kink + a_2 * (u_t - kink) + b$, otherwise

At any point t, the **stable interest rate** will be calculated as:

$$stable\_borrow\_rate_t(u_t, stable\_loan\_wt_t) = variable\_borrow\_rate_t + base\_premium + premium_t(stable\_loan\_wt_t)$$

where,

$$premium_t(stable\_loan\_wt_t) = stable\_rate\_slope \times max(0,\, stable\_loan\_wt_t - optimal\_stable\_loan\_wt)$$

Supply rate will be calculated as:

$$supply\_rate_t = market\_average\_borrow\_rate_t * u_t * (1 - reserve\_factor)$$

$$market\_average\_borrow\_rate_t = \frac{total\_borrow_t^{VariableRate} \times borrow\_rate(u_t) + total\_borrow_t^{StableRate} \times wt\_average\_stable\_rate_t}{total\_borroww_t}$$

where,

$$wt\_average\_stable\_rate_t = \frac{\sum_{i=1}^{n} borrow\ at\ fixed\ rate\ i \times fixed\ rate\ i}{total\_borrow_t^{StableRate}}$$

Model Parameters

- a_1: variable interest rate slope1
- a_2: variable interest rate slope2
- b: base rate per block for variable interest rate
- $kink$: optimal utilization rate, at which the variable interest rate slope shifts from slope1 to slope2
- $reserve\_factor$: portion of interest income taken out from protocol, i.e. not distributing to supplier
- $base\_premium$: base premium added to the prevailing variable rate if stable interest rate is chosen
- $stable\_rate\_slope$: a premium added when the stable rate loan adoption rate is above the optimal stable loan adoption rate. A factor will be applied to this premium before adding to the rate
- $optimal\_stable\_loan\_wt$: the optimal stable loan adoption rate, i.e. weightage of stable loan borrow in the total borrow
- $reserve\_factor$: portion of interest income taken out from protocol, i.e. not distributing to supplier

The stable rate provides predictability for the borrower; however, it comes at a cost, as the interest rates are higher than the variable rate. The rate of a stable loan is fixed until the rebalancing conditions are met:

1. When $utilization\_rate > rebalance\_up\_ur\_threshold$, and
2. $market\_average\_borrow\_rate_t <$
   $variable\_borrow\_rate(u = rebalance\_up\_ur\_threshold) * rebalance\_up\_rate\_fraction\_threshold$

For example, if we set:

- $rebalance\_up\_ur\_threshold = 0.9$
- $rebalance\_up\_rate\_fraction\_threshold = 0.5$

The conditions become the following:

1. When $utilization\_rate > 90\%$, and
2. $market\_average\_borrow\_rate_t < variable\_borrow\_rate(u = 90\%) * 0.5$

Stable Rate Rebalancing The vToken contract contains the public $rebalanceStableBorrowRate$ function which can be called with an account address to rebalance the stable interest rate of a specific user. Suppliers are incentivized to rebalance rate when the stable supply rate is low because they can earn more interest. Initially, Venus will provide an agent that will periodically monitor all of the stable rates positions and rebalance the ones deemed necessary. This does not increase centralization to the protocol because even if the agent stops working, anybody can call the rebalance function of the contract and rebalance the stable borrow rate.

The rebalance strategy will be decided off chain by the agent. This means that users who satisfy the rebalance conditions may not be rebalanced immediately. Since those conditions depend on the liquidity available and the state of the market, there might be some transitory situations in which an immediate rebalance is not needed.

Venus Prime A dynamic rewards program, hinged upon liquidity mining rewards, has been devised to offer superior borrowing and lending rates to committed users of the Venus Protocol. Contrary to the typical approach of many protocols that rely on their finite token supply to bolster rewards, Venus Prime will utilize protocol revenue to reward selected users.

Users who meet certain conditions are eligible to receive Soul Bound Tokens. These are unique, non-transferable NFTs that allow them to participate in the Venus Prime program.

Two variants of the Prime Token exist, each with specific conditions that must be met for a user to be selected:

1. Irrevocable Prime Token:

- The user must have staked XVS for the preceding 12 months.
- The user must have engaged with the Venus Protocol monthly over the last 12 months. This signifies that the user must have performed at least one borrow and supply operation per month, consistently for 12 consecutive months.

2. Revocable Prime Token:

- The user must stake a minimum of 1,000 XVS in the vault for a duration of 90 days.
- Upon reaching the staking threshold of 90 days with at least 1,000 XVS, users are eligible to claim the Prime Token.
- If users withdraw XVS, causing their balance to drop below 1,000 at any given time, their Prime Token will consequently be burned.

Venus streams (per block) 10% income of boosted markets to the prime token holders . The interest earned by prime token holders depends on their total “Qualified Value Locked” (QVL).

Calculate Income Per Block

Formulas to calculate both stable and variable income per block for a market are:

$$income\_from\_stable\_borrows = stable\_borrows * average\_stable\_borrow\_rate * reserve\_factor$$

$$income\_from\_variable\_borrows = (total\_borrows - stable\_borrow) * borrow\_rate\_per\_block * reserve\_factor$$

$$total\_income\_per\_block = income\_from\_stable\_borrows + income\_from\_variable\_borrows$$

Boosted Rewards Once the user has claimed a prime token they begin to receive boosted rewards for the governance selected markets in the form of rewards per block.

The Cobb-Douglas function is used to calculate rewards for a user. Protocols like 0x, Goldfinch or The Graph also use this formula to distribute rewards.

$$rewards(i) = (t * \mu) * \frac{T(i)^{\alpha} * V(i)^{1-\alpha}}{\sum_{j=0}^{N-1} T(j)^{\alpha} * V(j)^{1-\alpha}}$$

t = total income generated by the market
μ = % of income to distribute as boosted rewards
α = amplification weight of borrow and supply versus locked XVS
T(i) = total XVS staked by user i
V(i) = total borrow/supply by a user i in the market
N = total users

or the formula can be written as:

$$sumOfMembersScore = \sum_{j=0}^{N-1} T(j)^{\alpha} * V(j)^{1-\alpha}$$

$$memberScore(i) = \frac{T(i)^{\alpha} * V(i)^{1-\alpha}}{sumOfMembersScore}$$

$$rewards(i) = totalRewardsForDistribution * memberScore(i)$$

Here is how Venus implements the above formula for a market to calculate rewards per blocks:

There is a global $lastUpdatedBlock$ variable that is calculated as:

*incomeInPastBlocks = (block.timestamp - lastUpdatedBlock) * incomePerBlock*
lastUpdatedBlock = block.timestamp

There are also a global $rewardIndex$ and $sumOfMembersScore$ variables. $sumOfMembersScore$ represents the current sum of members' scores. And $rewardIndex$ is updated whenever a user's staked XVS or supply/borrow changes.

delta = incomeInPastBlocks / sumOfMembersScore
rewardIndex = rewardIndex + delta
numberOfTimesScoreUpdated++
*sumOfMembersScore = sumOfMembersScore - (T(old)^α * V(old)^1-α) + (T(new)^α * V(new)^1-α)*

Whenever a user's supply/borrow or XVS vault balance changes rewards are accrued and added to their account.

*rewards = (rewardIndex - userRewardIndex) * ((numberOfTimesScoreUpdated - userNumberOfTimesScoreUpdated) * scoreOfUser)*

Finally $userNumberOfTimesScoreUpdated$ and $userRewardIndex$ are updated to their current global values.

Significance of α A higher value of α reduces the weight on stake contributions in the determination of rewards and increases the weight on supply/borrow contributions. The value of α is between 0-1. The default value will be 0.5 which distributes rewards evenly between borrowers and suppliers. A higher value for example, can be used to attract more supply from the prime token holders.

XVS and Market Caps There will be a minimum requirement of 1000 XVS required to mint a prime token. The maximum XVS cap taken into account when calculating the score of a user is 100,000 XVS.

Per market borrow and supply caps will be set for each user. The cap will be calculated by multiplying the XVS balance of the user with the prime rewards multiplier.

*borrowCapOfUser = xvsBalanceOfUser * marketBorrowMultipler*
*supplyCapOfUser = xvsBalanceOfUser * marketSupplyMultipler*

Update Cap Multipliers and Alpha Market multipliers and alpha are configurable by VIP. Changes to these values will be applied gradually so users can update their borrow/supply positions and their new scores can be calculated.

There is also a public permissionless function *updateScore* which can be invoked by anyone to update a user score anytime. If the parameters haven't changed it won't update the score. Venus will have an automation script to call this function for all users whenever these parameters change. Anyone in the community can also invoke this function independently.

Venus will also provide a tracking system for more visibility into the smart contract which tracks the total number of accounts whose scores have pending updates to provide transparency to the process.