MYSO Finance
Balancer Grant Application

January 2023

MYSO Finance Association
Aetienne Sardon (as@myso.finance)

1 Value-Add for Balancer

- **How does the grant aim to provide value to the Balancer Ecosystem**: the proposal aims to directly contribute TVL growth on the Balancer Protocol by making liquidity provisioning in early-stage project tokens less risky and allowing early-stage projects to attract more risk-averse LPs to build more market depth.
- **What is the requested grant and individual milestone/tranche amounts**: $90K in Bal tokens, to be paid out in 3 tranches of $30K each. First upfront, second after completion of basic MVP, and third after successfully completing an audit.
- **What are the milestones and the timeline**: milestones are (1) develop smart contract MVP (1st month), (2) develop UI MVP (2nd month), (3) go through security audit (3rd month).
- **Can you quantify the expected value add**: a somewhat similar project was Ondo's v1 vaults, which helped facilitate $210M in liquidity (see section 2 and [1]).
- **Is this proposal value for money for the Balancer Ecosystem**: we expect audit costs to be around $50-75K, hence the remainder of $40-15K is to cover MYSO Finance's development costs for 3 months. Any additional development costs would effectively be subsidized by MYSO Finance.
- **Who is the team and what is their experience to successfully deliver the project**: MYSO Finance comprises a team of 5 people, 3 of which are developers. The team has previously developed a v1 protocol for non-liquidatable loans, so called "Zero-Liquidation Loans", and has hands-on experience in going through a security audit as well as shipping dapp UIs.

2 Problem: DEX Liquidity for early-stage DAOs

Early-stage DAOs typically struggle to grow secondary market liquidity for their native token. This is because stablecoin or ETH holders often times are hesitant to provide two-sided liquidity on a CFMM DEX for any early-stage DAO token as they don't want to be exposed to a risky native token and suffer from impermanent loss. Currently, the only way to go about this is to either (a) provide DEX liquidity mining incentives to attract more risk tolerant liquidity providers, who speculate on incentives to outweigh the potential impermanent loss, or, (b) to focus on CEX liquidity with the help of established market makers.[1]

- **Liquidity Mining Incentives**: the problem with DEX liquidity mining incentives is that they often times involve complex bribing systems, which essentially function as 'markets for liquidity'. Rolling out a liquidity mining program through these bribing economies often comes with significant coordination overhead and can be challenging to budget for due to their dynamic nature (to attract x liquidity over a period of t, how much bribes y are needed? How does this change if another project bribes?).
- **Market Maker Deals**: when partnering with market makers, these typically ask the DAO to lend them their native token and in exchange the market maker uses his own stables (or ETH) inventory to provide a two-sided market. The only caveat with this is that the market maker usually demands from the DAO a call option on the native tokens it borrowed, allowing them to buy the token at a discount in case it pumps. This gives the market maker upside potential in the native token, while the DAO is essentially giving this upside up, and the extent to which it does is mainly determined by the strike price and expiry of the call option. However, there's usually a significant information asymmetry between a DAO and market makers, and little public information available to help DAOs figure out what actually constitutes a fair deal, making negotiations with market makers somewhat challenging. This is all the more the case, given that –once such a deal is signed– market makers often times have significant influence over the secondary market dynamics of the DAO's token.

3 Solution: DEX-LP Loans for Delta-Neutral LPing

An alternative solution to build DEX liquidity is that DAOs enable their users to do 'delta-neutral' liquidity provisioning, meaning that the user's LP position

[1]While [1] also pioneered a protocol to connect token holders to collectively provide two-sided DEX liquidity, this wasn't specifically targeted at DAOs and also has been decommissioned meanwhile.

value is immune against changes in the DAO's token price. This way, users who don't want to be 'too heavily' exposed towards the DAO's native token could still provide liquidity and help build DEX market depth without taking 'too much' directional risk in the native token. One can achieve this through 'DEX-LP Loans', meaning the DAO lends its native token to users, who pair it with their stables or ETH holdings to then provide two-sided DEX liquidity. The received LP token can then be atomically pledged as collateral with the DAO. This way the DAO lends its native tokens in a purpose-bound way (i.e., for DEX liquidity provisioning), ensuring that they cannot be used in a way that would be misaligned with the DAO's interests (e.g. for short-selling or the like).

As it is in the DAO's interest to build and maintain liquidity for longer periods (e.g. 6 months), any native token loan would have to have a certain minimum time period before liquidity providers can repay and unwind their LP position. Using a conventional crypto-collateralized loan setting for this would be problematic because of (a) liquidity providers would be vulnerable to liquidation risk as their position would be locked-in for a certain time period without being able to unwind it to avoid being liquidated, (b) it's challenging to find a reliable price oracle for the LP token value[2], and (c) loans typically have a repayment amount that is higher than the loaned amount, however, in order to attract liquidity providers it might be necessary for a DAO to offer a debt discount (i.e., negative interest rate). In contrast to conventional liquidation-centered DeFi loans, 'Zero-Liquidation Loans', as described in [2] [3], can be used to facilitate such a purpose-bound DEX-LP loan.

4 Lifecycle

4.1 DAO grants DEX-LP Loan

Assume a DAO has some native token $\mathbb{X}$, where each token $\mathbb{X}$ is currently worth $P_{\mathbb{X}} = 1$ ETH. Further assume, the DAO treasury has $10,000$ $\mathbb{X}$ in reserves, and would like to utilize 10% to build two-sided DEX liquidity for $1,000$ $\mathbb{X}$ and $1,000$ ETH. For this it offers Pepe the following loan:

- **Loan Amount**: $1,000$ $\mathbb{X}$
- **Collateral**: $2,000$ $\mathbb{X}$-ETH-LP-Token
- **LTV**: 50%
- **Tenor**: 6 months
- **Earliest Repay**: after 5 months

[2] Especially because the LP token price would be susceptible to maliciously skewing the LP token composition to manipulate the LP token NAV to forcefully trigger a liquidation.

- **Repayment Amount**: 950 $\mathbb{X}$, i.e., 5% debt discount[3]

Assume Pepe holds $1,000$ ETH and would like to earn yield on it but wants to maintain primarily ETH exposure. Pepe could deposit their ETH with Aave but would only be earning 0.01-2% p.a. So instead, Pepe decides to borrow $1,000$ $\mathbb{X}$ from the DAO, pair it with their $1,000$ ETH, add it as liquidity to a Balancer pool, receive $2,000$ $\mathbb{X}$-ETH-LP-Token[4], and pledge the LP tokens immediately as collateral with the DAO.

4.2 Resulting Position Value at Expiry of DEX-LP Loan

Now that the loan is initiated, what does Pepe earn or lose if the price of token $\mathbb{X}$ changes, e.g. to $P_{\mathbb{X}} = 0.8$ ETH or $P_{\mathbb{X}} = 1.2$ ETH? And what's the resulting payoff for the DAO? In section 4.2.1, we'll first go over the resulting payoff from the LP's perspective, and then in section 4.2.2 over the DAO's.

4.2.1 LP Payoff

Let x denote the amount of $\mathbb{X}$ tokens held by the Balancer Pool, and y the amount of ETH. Since Pepe pledged $1,000$ $\mathbb{X}$ and $1,000$ ETH into the pool, we have $x = 1000$ and $y = 1000$. Given x, y and $P_{\mathbb{X}}$ (ETH price of the DAO token), we can determine the ETH value of the LP token by:

$$\text{50-50-LP-Value} = 2 \cdot \sqrt{x \cdot y \cdot P_{\mathbb{X}}} \tag{1}$$

At the same time, Pepe owes the DAO the following ETH denominated repayment amount:

$$\text{Repay} = 950\ \mathbb{X} \cdot P_{\mathbb{X}} \tag{2}$$

But because the loan is structured as a Zero-Liquidation Loan, there's no forced liquidation and instead Pepe has the right –but not the obligation– to repay and reclaim their collateral. Being a rational Pepe, they will only repay if the LP token is worth more than the debt owed. Hence, Pepe's overall position value is given by:

$$\text{DEX-LP-Loan} = (\text{50-50-LP-Value} - \text{Repay})^{+} \tag{3}$$

The resulting position value is illustrated below (see fig. 1) and compared with the situation where (a) Pepe would've just held 100% in ETH and where (b) they would've just done a regular 50-50 LP position. Obviously, if Pepe

[3] For illustration purposes, we assume the DAO wants to incentivise LPing through a debt discount. However, if DEX swap fees are 'high enough', a DAO might not have to do this to attract LPs.

[4] $= 2\sqrt{1000 \cdot 1000}$.

would've just held 100% in ETH then their RoI would've been independent of any price moves in token 𝕏 (flat dotted orange line). If Pepe would've done a 50-50 LP, their RoI would've been the yellow curve (b1 or b2), essentially with more upside potential if 𝕏 appreciates but also more downside risk when it depreciates. In contrast, if Pepe provides liquidity in conjunction with the DEX-LP Loan as described before (see section 4.1), their RoI will be largely immune against any token 𝕏 changes as indicated by the blue line, which outperforms a simple 100% ETH holding for token 𝕏 price changes of +/-50%, all while being able to earn any DEX swap fees (e.g., assuming a 5% DEX yield we can see that the resulting blue line c2 is shifted upwards compared to c1 in fig. 1; note that such pool yields could also be splitted between the DAO and LP). Depending on the DAOs preference for incentivising DEX LPing, it could offer a higher or lower debt discount (or none at all), which essentially translates into a higher base line yield for Pepe (i.e., the higher the debt discount, the higher the blue line is shifted upwards, meaning that e.g., if the token 𝕏 price doesn't change at all Pepe earns the full debt discount, in this example, 5%).

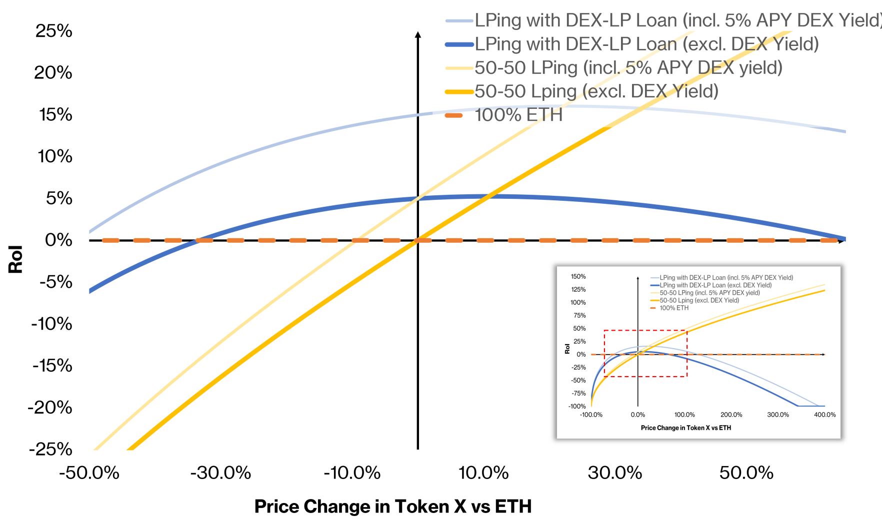

Figure 1: **LP RoI** - Comparing RoI for (a) 100% ETH, (b1) 50-50 LPing (excl. DEX yield), (b2) 50-50 LPing (assuming 5% APY DEX yield), (c1) LPing with DEX-LP Loan (assuming 5% debt discount and excl. DEX yield), (c2) LPing with DEX-LP Loan (assuming 5% APY DEX yield)

4.2.2 DAO Payoff

Figure 2 illustrates the DAO's cost from granting a DEX-LP Loan vs. not giving such a loan to LPs. One can see in fig. 2 that for price moves of token 𝕏 vs ETH between $[-50\%, +60\%]$ the cost is between $[-2.5\%, -8.5\%]$ for the numerical example described earlier in section 4.1. Only in the very extreme case where token $\mathbb{X}$ significantly outperforms ETH would the DAO have been better off to hold on to its token $\mathbb{X}$ allocation instead of granting a DEX-LP Loan. Note that in this extreme case the borrower would not have repaid as the token $\mathbb{X}$ debt would've exceeded the value of the LP token collateral. In this scenario, the DAO would've been left with the LP token collateral, which essentially means that the LP token collateral would've turned into protocol owned liquidity.

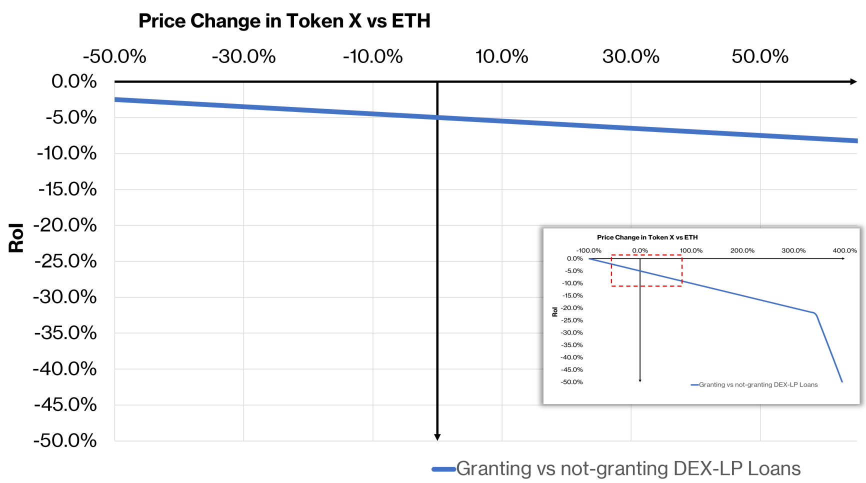

Figure 2: **DAO RoI** - Comparing RoI for (a) Not Giving DEX-LP Loans, and (b) Granting DEX-LP Loans (assuming 5% debt discount)

Note that one can calculate the exact price $P^*_{\mathbb{X}}$ where the owed token $\mathbb{X}$ debt (denominated in ETH) is higher than the reclaimable LP token collateral:

$$P^*_{\mathbb{X}} = 4 \cdot \frac{y}{x} \cdot \frac{1}{(1-d)^2} \tag{4}$$

where $d \in [0, 1)$ denotes the debt discount. For $y = 1000, x = 1000, d = 0.05$ we find that $P^*_{\mathbb{X}} = 4.43$, meaning that token $\mathbb{X}$ vs ETH would've to appreciate by +343% before the DAO significantly loses on potential token $\mathbb{X}$ upside (as indicated by the kink in fig. 2). Assuming a loan tenor of e.g. 6 months this potential upside loss may be significantly lower than the value-add of having otherwise idle token $\mathbb{X}$ inventory put to good use to build DEX liquidity. Moreover, the DAO could increase or adjust d to its individual risk-reward preferences, e.g. to push the point $P^*_{\mathbb{X}}$ farther out or closer down. Obviously, depending on how the DAO chooses d it will incur a cost (or profit if $d < 0$), which, however allows for very straight forward budgeting to drive a certain amount of DEX liquidity. For example, if the DAO granted a DEX-LP Loan as outlined in section 4.1, it would know in advance that it pays 50 $\mathbb{X}$ token for 2,000 ETH TVL in DEX liquidity for at least 5 months. This makes is very easy for the DAO to understand what the exact cost is to attract a certain target DEX liquidity. For example, assuming that the $\mathbb{X}$ vs ETH price stayed between $[-50\%, +60\%]$ the DAO would know in advance the DEX liquidity cost not only in $\mathbb{X}$ token but could also easily translate this into ETH terms, given the corresponding price scenarios (e.g., for price moves between $[-50\%, +60\%]$ the cost for 2,000 ETH TVL would be between $[25, 85]$ ETH). Note that the numerical example is for illustrative purposes only, i.e., the DAO is free to choose whichever loan structure is most suitable for its DEX liquidity goals and $\mathbb{X}$ token budget. For example, the DAO could issue loans over time in tranches to better average the cost in ETH terms, or it could choose rather shorter tenors to reduce percentage costs while retaining more long-term upside, or it could decide to have DEX yields be split in a certain proportion between the LP and the DAO.

All in all, one can see that such a setup would allow the DAO to devise alternative or complementary liquidity bootstrapping strategies, which allow for (a) very easy cost planning in terms of how much $\mathbb{X}$ token are spent per attracted ETH liquidity, allow to (b) retain upside in its token and, potentially, (c) even allow for upside participation in DEX yields by splitting corresponding rewards between the LP and itself, and lastly, (d) allows to distribute token rewards to LPs directly instead of having to go through intermediary bribing systems.

5 Implementation Sketch

One can construct such system as described in section 4 as follows:

1. Allow the DAO to deposit part of its $\mathbb{X}$ token inventory into a vault contract.

2. Allow the DAO to quote loan terms (which are valid for a certain time period) as off-chain signed messages that the LP can then accept and submit (and otherwise, e.g. if prices have changed significantly the DAO can sign a new quote).

3. Allow the LP to flashborrow the 'loanToken' (here $\mathbb{X}$ token) such that they can then use a callback function to directly pair the flashborrowed token with their stables or ETH inventory, provide as two-sided liquidity to balancer and then atomically pledge back as collateral into the DAO's vaul.

4. Allow the LP to repay any time prior to expiry of the loan, but not earlier than the earliest repayment date, as described in section 4. When repaying, allow the LP to flash-reclaim (part of) the LP token collateral such that the LP can decompose it into its LP token constituents, swap any missing $\mathbb{X}$ token if needed, and repay their token $\mathbb{X}$ debt.

5. Allow the DAO to collect the resulting proceeds after loans have been settled.

Most steps are straightforward to implement (in particular 1. depositing and 2. off-chain message signing). For example, a rough implementation sketch for 3. would look as follows:

```
function borrow(
    DataTypes.LoanQuote calldata loanQuote,
    address callbacker,
    bytes calldata data
) external nonReentrant {
    ...

    IERC20Metadata(loanQuote.loanToken).safeTransfer(
        msg.sender,
        loanQuote.loanAmount
    );
    if (callbacker != address(0)) {
        IVaultFlashCallback(callbacker).vaultFlashCallback(loan,
            data);
    }
    IERC20Metadata(loanQuote.collToken).safeTransferFrom(
        msg.sender,
        address(this),
        loanQuote.pledgeAmount
    );

    ...
}
```

Similarly, one a function skeleton for 4. would look as follows:

```
function repay(
    DataTypes.LoanRepayInfo calldata loanRepayInfo,
    address callbacker,
    bytes calldata data
) external nonReentrant {
    ...
    IERC20Metadata(loanRepayInfo.collToken).safeTransfer(
        msg.sender,
        reclaimCollAmount
    );
    if (callbacker != address(0)) {
        IVaultFlashCallback(callbacker).vaultFlashCallback(loan,
            data);
    }
    IERC20Metadata(loanRepayInfo.loanToken).safeTransferFrom(
        msg.sender,
        address(this),
```

```
            loanRepayInfo.repayAmount +
                loanRepayInfo.loanTokenTransferFees
        );
        ...
    }
```

Lastly, for step 5., the DAO could unlock any left unlcaimed LP token collateral through the following function:

```
    function unlockCollateral(
        address collToken,
        uint256[] calldata _loanIds
    ) external {
        uint256 totalUnlockableColl;
        for (uint256 i = 0; i < _loanIds.length; ) {
            uint256 tmp = 0;
            DataTypes.Loan storage loan = loans[collToken][_loanIds[i]];
            if (!loan.collUnlocked && block.timestamp >= loan.expiry) {
                tmp =
                    loan.initCollAmount -
                    (loan.initCollAmount * loan.amountRepaidSoFar) /
                    loan.initRepayAmount;
            }
            loan.collUnlocked = true;
            totalUnlockableColl += tmp;
            unchecked {
                i++;
            }
        }
        lockedAmounts[collToken] -= totalUnlockableColl;
    }
```

References

[1] Ondo. Ondo v1 — vaults & laas. https://docs.ondo.finance/ondo-v1-vaults-laas, Jan 2023.

[2] A. Sardon. Zero-liquidation loans: A structured product approach to defi lending. https://arxiv.org/abs/2110.13533, Oct 2021.

[3] A. Sardon. Myso v1 core: A trust-minimized protocol for zero-liquidation loans. https://figshare.com/articles/preprint/MYSO$_v1_Core_ATrust-Minimized_Protocol_for_Zero-Liquidation_Loans/21581328, Nov 2022.