Sapling

Protocol Whitepaper

v0.1

Tim O'Shea, Gareth Campbell, Farrukh Yakubov

June 2022

Abstract

Sapling is an open-source and decentralised unsecured lending platform enabling anyone (individual, institution, community, syndicate, family or group of friends) to provide credit to small and medium enterprise (SME) borrowers in a cost-efficient and agile manner. Sapling is building the most simple and secure Web3 lending platform in the Ethereum ecosystem, owned by the Sapling community and focussed on returning SME lending to a relationship-based business. Sapling will help to close the SME financing gap by bringing SME lending on-chain and using the unique characteristics of blockchain technology to build a lending ecosystem founded on verifiable trust.

This whitepaper introduces the Sapling protocol and provides details of its architecture, protocol mechanics, actors and their actions.

Contents

1 Introduction

1.1 Our Mission

Sapling is an uncollateralized lending protocol created to help solve one of the most significant challenges for the global economy - getting capital into the hands of small and medium sized enterprises (SMEs) to help them grow. SMEs represent over 90% of businesses worldwide and are estimated to be responsible for 50-70% of employment globally, but find it increasingly difficult to obtain credit due to structural and regulatory changes in the global banking industry in recent decades.

We are strong advocates of SMEs, recognise the critical role they play in contributing to global GDP and believe that supporting SMEs around the world will have an incremental, positive impact on global growth. We are equally passionate about Web3 and blockchain technologies, are convinced that they can solve real-world problems and believe that using these new technologies will create (i) greater access to capital for SMEs and (ii) markedly more efficient deployment of capital to businesses that are under-served by the current banking system.

1.2 Sapling Protocol

To address this problem, we are building Sapling, an open-source and de-centralised unsecured lending infrastructure enabling anyone (individual, institution, community, syndicate, family or group of friends) to provide credit to SME borrowers without having to run a full-stack banking operation.

Sapling will build the most simple and secure lending Web3 platform, owned by the Sapling community and focussed on returning lending to a relationship-based business founded on trust. By building a lending infrastructure that puts money into the hands of SMEs who so badly need it, we help improve SME productivity, improve the livelihoods of the people who run them, those who work for them, and everyone who relies on the products or services they provide.

1.3 Glossary

- **Smart Contract** - a program that runs on Ethereum based virtual machines, and implements a certain intended logic.
- **Lending Pool** - a smart contract that facilitates lending and borrowing. A Lending Pool has a Pool Manager and issues tokens that represent the Lender's share of the pool capital.
- **Protocol Pool** - a smart contract similar to the Lending Pool contract, which issues tokens that represent Lender's positions.
- **Lender** - a user, an Ethereum address, who provides capital into any Lending Pool or the Protocol Pool with the intention of earning interest on the funds provided.
- **Borrower** - a user, an Ethereum address, who borrows funds from a Lending Pool under agreed conditions with the intention of paying back the borrowed funds with interest.
- **Pool Manager** - a user, an Ethereum address, that originates loans, manages loans and manages a set of Lending Pool parameters.
- **Protocol Governance** - a decentralised entity that can take governing actions and controls a set of parameters across the protocol components.
- **Capital Token** - ERC20 token (USDC) representing the capital provided to Lending Pools by the Lenders and Pool Managers.
- **Pool Token** - ERC20 tokens that represent Lender positions, and are issued and controlled by the Lending Pools.
- **Solidity** - a programming language used for writing smart contracts.

2 Protocol Mechanics

The core Sapling protocol consists of a Protocol Pool and any number of Lending Pools, with three main actors: Borrowers, Lenders and Pool Managers. Figure 1 provides a high-level overview of interactions between Sapling core components and those protocol actors:

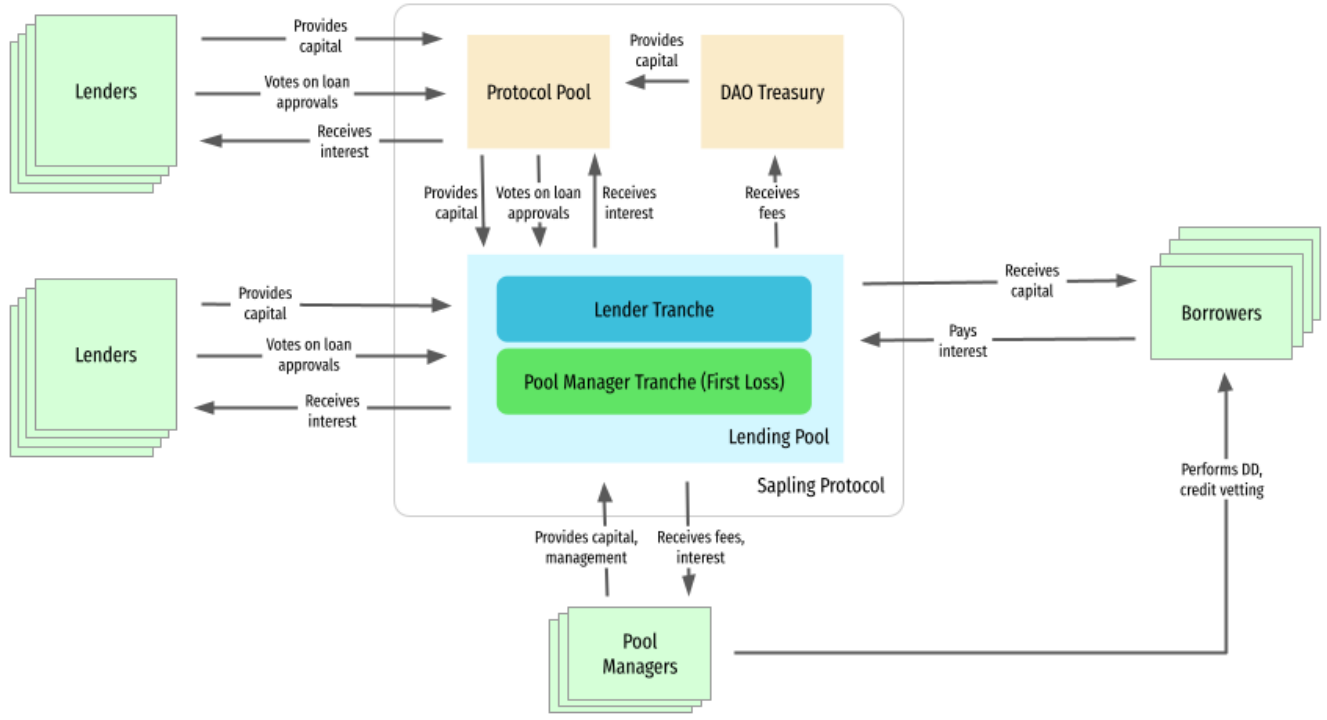

Figure 1. Sapling Protocol Overview

2.1 Protocol Components

2.1.1 Lending Pool

The Lending Pool is a smart contract that allows a single Pool Manager to accept capital (USDC) from Lenders and issue loans using that capital to Borrowers. In order to issue loans to Borrowers, the Pool Manager must first stake an amount of USDC into the Lending Pool smart contract to establish the Pool Manager Tranche; this tranche will act as first loss capital in the event of any Borrower default.

2.1.2 Protocol Pool

The Protocol Pool is a Lending Pool operated by the Protocol Governance, whose purpose is (i) to provide a diversified investment opportunity for Lenders and (ii) to provide loss cover for the lending activities within the protocol. Protocol Governance will nominate and vote for a Governance Board to act as Pool Manager for the Protocol Pool, then the Governance Board will direct investment from the Protocol Pool into any active Lending Pool on the protocol. The Sapling DAO Treasury will provide the first loss capital required to create the Protocol Pool, and the Pool will accept additional capital from Lenders, giving Lenders diversified exposure across many Lending Pools.

When depositing capital into the Protocol Pool, Lenders will receive an equivalent amount of Protocol Pool tokens. Lenders can withdraw their capital at any time (whole or partial withdrawal) by redeeming their Protocol Pool tokens. Like Lenders into any other Lending Pool, Lenders into the Protocol Pool will receive an interest rate return on their capital.

2.2 Protocol Actors

Protocol actors perform specific roles within the protocol, and have different access rules enforced across the protocol contracts. Within the same Lending Pool roles are exclusive and cannot be used interchangeably. For instance, a Borrower cannot be a Lender at the same time within the same pool and vice versa. The role of Protocol Governance is exclusive across the protocol.

2.2.1 Pool Manager

The Pool Manager is a user (an Ethereum address) that controls which Borrower requests to the Lending Pool are approved or denied, and which loans to default. Any individual or entity that wants to create a Lending Pool on the Sapling protocol first has to undergo successful KYC / KYB verification, undertaken by Protocol Governance. Once successfully verified, the Pool Manager may create a Lending Pool, add first loss capital and start lending.

2.2.2 Lender

A Lender is a user (an Ethereum address) that provides capital into one or many Lending Pools with the intention of earning interest on the funds provided. Lenders can deposit capital into any open Lending Pool and decide which Pool or Pools to invest in based on key Lending Pool

information which will be accessible publicly. Lenders can also withdraw the capital they have provided.

2.3.3 Borrower

A Borrower is a user (an Ethereum address) who borrows capital from one or many Lending Pools and pays back the borrowed funds with interest. Borrowing is subject to on-chain loan approval by the Pool Manager and to any on-chain or off-chain KYC / KYC or identity verification that the Pool Manager specifies.

2.3.4 Protocol Governance

Protocol Governance is a decentralised entity with an Ethereum address that controls a set of parameters across the protocol components and takes governing actions regarding the protocol. These governing actions include, but are not limited to, enacting an emergency pause on a Lending Pool or Pools and approving updates to the upgradable components of the protocol.

2.4 Identity Verification

In order to leverage the on-chain and off-chain KYC / KYB solutions that are available currently, Sapling will create a 'verification hub' which will allow the protocol to integrate with an unlimited amount of current and future KYC / KYB solutions. Over time, the Protocol Verification Hub will offer participants a choice of using a number of potential identity verification solutions, and will also act as the verification gateway into the protocol allowing any user of the protocol to interrogate which participants have or have not been verified to lend or borrow.

Initially, it is anticipated that protocol participants will have to rely on off-chain identity verification solutions as follows:

- Protocol Governance will carry out off-chain verification for any prospective Pool Managers and will not allow Pool Managers to create pools until they have successfully completed the protocol verification process.
- Pool Managers will have the choice of whether to mandate identity verification for Lenders who supply capital into their Pool or Borrowers who they loan capital to from their Pool. If a Pool Manager chooses to mandate identity verification then they will be able to specify any off-chain process that they wish to use.

It is the intention of the protocol to ensure that an appropriate level of verification information is made transparent to support the decision-making of other protocol participants. For example:

- A Pool Manager will be able to see whether a Borrower has been verified by another Pool Manager.
- A Lender will be able to see what percentage of a Pool Manager's portfolio of Borrowers have successfully completed a verification process.
- A Lender will be able to evaluate the correlation between default rates and verification rates on a Pool-by-Pool basis.
- A Lender who has provided capital into a Pool will be able to view the uploaded verification evidence for a Borrower prior to voting to approve the relevant loan.

2.5 Lender Voting

Lender voting will be implemented via a protocol voting contract to provide Lenders with a control mechanism to guard against a Pool Manager becoming a bad actor.

Once a loan has been originated by the Pool Manager and a Borrower request has been received, the Pool Manager can only issue the loan to the Borrower when Lenders into the relevant Pool have voted to approve the loan. The voting mechanism will be weighted based on each Lender's share of the pool.

3 Protocol Pool Mechanics

The Protocol Pool will deploy capital by investing as a Lender into any number of underlying Lending Pools across the protocol. The aim of the Protocol Pool will be to generate an interest spread between the interest rate earnings it achieves from its investments and the interest rate it pays Lenders into the Protocol Pool. The capital generated by this spread will either be retained in the Protocol Pool to act as protocol loss cover or will be returned to the DAO Treasury where Protocol Governance will determine its use.

The Protocol Pool Governance Board will be responsible for determining which Lending Pools to lend into and will propose loans to its community of Lenders for votes of approval. The Governance Board will also be responsible for determining what proportion of the returns it generates are retained in the Protocol Pool or moved into the DAO Treasury.

As a Lender into other Lending Pools, the Protocol Pool will by default acquire the ability to vote to approve the relevant Lending Pool's loan originations. The Protocol Governance Board will be responsible for exercising the Pool's voting rights. It is anticipated that the Protocol Pool vote will act as a signal for other Lenders in the ecosystem.

The loss cover within the Protocol Pool operates on a tiered basis. Should the Protocol Pool experience any loan defaults then those losses will be absorbed initially by the tranche of first loss capital seeded into the Protocol Pool by the DAO Treasury at the time of Pool creation. Thereafter the losses will be absorbed by the net interest return generated by the Pool and finally by the tranche of Lender capital that has not yet been deployed.

Subject to the amount of capital available within the Pool at any given time, the Protocol Pool may choose to divert excess capital to cover any losses incurred by other Lending Pools. The Protocol Pool Governance Board will be responsible for choosing whether to cover these losses.

4 Lending Pool Mechanics

4.1 Pool Funds Limit

The Pool Funds Limit (PFL) is the total amount of capital allowed in a Lending Pool, limited by the amount of first loss capital staked into the Lending Pool by the Pool Manager and the Target Stake Percentage, a parameter controlled by Protocol Governance. The default value of Target Stake Percentage is 10%.

PFL is calculated as:

$$PFL = stakedFunds \times \frac{100\%}{targetStakePercent}$$

4.2 Stake-to-Pool Ratio

The relationship between the Pool Manager's stake and the total capital in the pool define the stake-to-pool (STP) ratio :

$$stakeToPoolRatio = \frac{stakedCapital}{totalPoolCapital}$$

It is essential for the Pool Manager to maintain a healthy stake-to-pool ratio given that this ratio controls lending and unstaking. No new loans can be approved if the STP ratio falls below the Target Stake Percentage parameter. Furthermore, only the portion of the Pool Manager's stake that results in an excess STP ratio can be unstaked.

4.3 Pool Tokens

Lender positions in the Lending Pool are represented by Pool Tokens. Pool Tokens allow efficient and fair distribution of (i) the earnings from loan interest payments and (ii) any losses that cannot be absorbed by the Pool Manager's stake. Pool Tokens are minted at the point of deposit into the Pool (or at the point of staking for Pool Managers) and are burned at the point of a Lender redeeming them in the relevant Lending Pool. Pool tokens belonging to the Pool Managers are also burned on loan defaults.

Given a Capital Token (USDC) amount, current total Pool Tokens and total Pool Capital, the corresponding Pool Token amount is calculated as:

$$poolTokenAmount = capitalTokenAmount \times \frac{totalPoolTokens}{totalPoolCapital}$$

Given a Pool Token amount, current total Pool Tokens and total Pool Capital, tokens are calculated as:

$$capitalTokenAmount = poolTokenAmount \times \frac{totalPoolCapital}{totalPoolTokens}$$

Pool Tokens may be exchanged outside of Lending Pools and the Sapling protocol, providing alternative liquidation options for the Lenders as opposed to redeeming them in the Lending Pools.

4.4 Exit Fees

Exit fees will be applied for any Lender redeeming Pool Tokens and withdrawing funds. The default value of the Exit Fee, controlled by Protocol Governance, is 0.5%. Exit fees are added back into the Lending Pool, and distributed proportionally amongst the positions in the Pool.

Staking and depositing are considered Pool entry actions, while unstaking and withdrawing are considered Pool exit actions.

4.5 Loans

Loans represent the on-chain agreement between Lending Pools and Borrowers. A loan, in terms of the smart contract, consists of an application with loan parameters and a detail with repayment statistics.

Each loan has the following application parameters:

- Loan ID
- Borrower's Ethereum Address
- Amount
- Duration
- APR (annual interest rate applied to the loan amount)
- Late APR Delta (increase in Loan APR for overdue periods)
- Grace Period for Payments

- Loan Request Timestamp
- Repayment Schedule Type (None, Monthly/30day, or Quarterly/90day)
- Loan Status

Loan application parameters cannot be modified after loan funds have been withdrawn by the Borrower. A borrowed loan will maintain Loan Details data with the following parameters:

- Loan ID
- Total Amount Repaid
- Principal Amount Repaid
- Interest Paid
- Approved Timestamp
- Borrowed Timestamp
- Last Payment Timestamp

A loan can go through a number of states based on actions taken on the loan by the actors in the Lending Pool. Refer to Figure 2 below for a complete loan states flow:

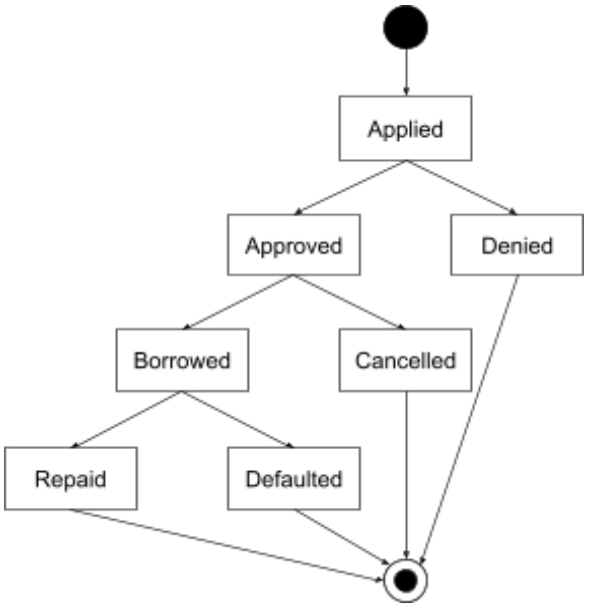

Figure 2. Loan States

4.5.1 Repayment Schedule

A repayment schedule can be selected by the Borrower when requesting a loan from a Lending Pool, and can be adjusted by the Pool Manager before the loan terms are accepted by the Borrower. This schedule cannot be changed once the loan has been borrowed.

Repayment schedules are represented as payment interval types and will be relative to the time elapsed since the loan was first borrowed. For instance, a loan with a Monthly / 30 day repayment schedule borrowed on June 21 will have the next payment due on July 21 - exactly 30 days from the borrowed time.

The following repayment schedules are available:

- Monthly /30 Day
- Quarterly / 90 Day
- None (for short-term loans).

Although the repayment schedule for a loan cannot be changed once the loan has been borrowed, it should be noted that schedules do not limit early payments or repaying a loan in full and closing the loan ahead of schedule. However, loan repayments that miss a scheduled repayment due date will incur an interest rate increase for the duration of the late period. If a loan repayment misses a scheduled repayment due date and remains outstanding past the loan repayment grace period, the loan may be subject to default by the Pool Manager.

4.5.2 Calculating a Loan Balance

The balance of a loan due at any given time is calculated in the following manner:

$$loanBalanceDue = principalAmountDue + principalAmountDue \times interestPercent$$

Where *interestPercent* depends on loan APR, the number of days passed since the loan Borrowed Timestamp, and whether or not the payment is regular or late. Interest percentage at the time of balance checking or payment is calculated in the following manner:

$$interestPercent = effectiveApr \times \frac{daysPassed}{365}$$

Where *effectiveApr* is,

For regular payments:

$$effectiveApr = loanApr$$

For late payments:

$$effectiveApr = \frac{loanAPR \times daysPassed + lateAPRDelta \times lateDays}{daysPassed}$$

For loans with a repayment schedule the $principalAmountDue$ corresponds to the current instalment amount and any past principal amounts that are overdue. Moreover, $lateDays$ will be counted from the last missed payment date.

4.5.3 Calculating Interest and Principal Repayment Components

The amount of interest paid as part of a loan repayment is calculated as follows:

$$interestPaid = amountPaid \times \frac{interestPercent}{100\% + interestPercent}$$

The principal amount paid as part of a loan repayment is calculated as the difference between the amount paid and the interest paid:

$$principalAmountPaid = amountPaid - interestPaid$$

4.6 Loan Interest Distribution

When a Borrower pays interest on a loan, the interest is distributed amongst the parties that have a claim on that interest payment: the Lenders into the relevant Pool, the Pool Manager and the Protocol.

Firstly, the protocol earnings are allocated by taking a percentage of the paid interest. This percentage value is determined by the Protocol Earning Percent (*protocolEarningPercent*) parameter. This parameter can be modified by the Protocol Governance and is bound between 0% and 10%.

Secondly, the Pool Manager's earnings are calculated. The Pool Manager's earnings are dictated by the Manager Earning Factor (*managerLeveragedEarningPercent*) parameter, controlled by the Pool Manager. This is a percentage value representing the Pool Manager's relative earn factor compared to a Lender, per unit of funds in the pool. A Manager Earning Factor value of 100% corresponds to 1x leveraged earnings when compared to the Lenders in the Pool, meaning that

the Pool Manager and the Lenders earn the same per unit of fund. Similarly, a Manager Earning Factor value of 150% corresponds to 1.5x earning leverage.

As an example, if a Pool Manager is looking to offer an average APY of 10% to Lenders into the Pool and wants to achieve a 15% APY on their staked funds, the Manager would set the Manager Earning Factor to 150%. Besides the Manager's Earning Factor, Lender APY depends on a number of parameters such as funds borrowed, protocol fees, and stake-to-pool ratio (see Section 4.7 for more details). Any Pool Manager earnings above a leverage factor of 1x will be made available for the Pool Manager to withdraw.

Lastly, the remaining interest is added to the Pool funds, resulting in a proportional increase in balances. This remaining interest includes the Pool Managers' earnings that correspond to 1x earnings leverage factor, recognising the interest that is being accrued by their first loss capital and necessary to maintain the Pool Manager's stake-to-pool ratio.

The following schematic summarises the process for splitting earnings from loan interest:

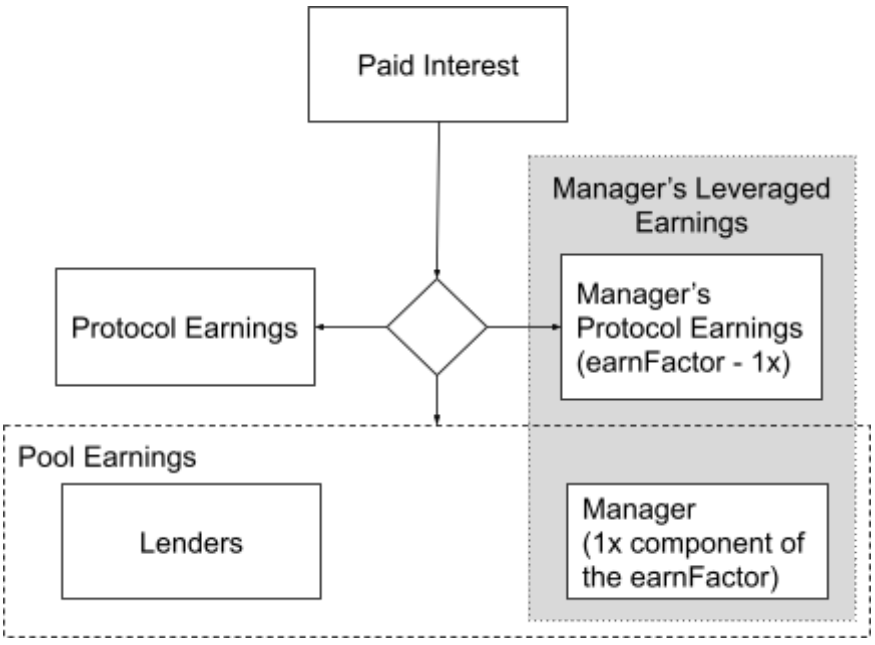

Figure 3. Splitting Loan Interest Earnings

The formulas for splitting the interest earnings are as follows:

Given:

$$currentStakePercent = \frac{stakedPoolTokens}{totalPoolTokens} \times 100$$

Then, protocol earnings:

$$protocolEarnings = interestPaid \times protocolEarningPercent$$

Pool Manager's withdrawable earnings, or MWE:

$$MWE = (interestPaid - protocolEarnings) \times currentStakePercent \times (earnFactor - 1)$$

Pool earnings:

$$poolEarnings = interestPaid - (protocolEarnings + MWE)$$

4.7 Lender APY

Lender APY depends on a set of variables comprising the Pool state and a number of parameters. These variables are:

- Weighted Average Loan APR
- Borrow Rate (as a ratio of Borrowed Capital to Total Pool Capital)
- Protocol Fee Percent
- Stake-to-Pool Ratio
- Manager Earning Factor

Lending pools publicly provide information on Projected Lender APYs for current and arbitrary borrow rates. The Projected Lender APY is calculated in the following manner:

First, the Pool APY is calculated:

$$poolAPY = weightedAvgLoanAPR \times \frac{borrowedCapital}{totalPoolCapital}$$

Next, the Protocol Fee APY is calculated, representing the portion of the Pool APY allocated to protocol fees:

$$protocolFeeAPY = poolAPY \times protocolEarningPercent$$

Then, the portion of the APY corresponding to the Pool Manager's leveraged earnings is calculated:

$$stakeToPoolRatio = \frac{stakedCapital}{totalPoolCapital}$$

$$leveragedEarnPercent = stakeToPoolRatio * (managerEarnFactor - 100\%)$$

$$managerWithdrawableAPY = leveragedEarnPercent * (1 - protocolEarningPercent)$$

Finally, the Lender APY is calculated as the difference between the Pool APY and the sum of the Protocol Fee APY and Pool Manager's Withdrawable Earnings APY:

$$lenderAPY = poolAPY - (protocolFeeAPY + managerWithdrawableAPY)$$

4.8 Math

Lack of native support for fixed point math in Solidity created the need for workarounds using integers.

Working with Percentage Values

Percentage values are represented using integers with extra digits for decimals. There are constants in the contract to help convert percentage values when interacting with it:

PERCENT_DECIMALS - how many decimal digits do the percent values in the contract have.
ONE_HUNDRED_PERCENT - a value representing 100% in the context of the contract.

Example:

Using PERCENT_DECIMALS value of 2, 100% is represented as 10000, and 30.05% is represented as 3005.

5 Lending Pool Actions

5.1 Stake

Staking is essential for Lending Pool operation and is exclusive to the Pool Manager. A Lending Pool cannot be created without initial capital staked by the Pool Manager, and this stake will act as first loss capital in the event of loan defaults. Upon staking, an equivalent amount of Pool Tokens are minted and locked in the Lending Pool. A Pool Manager's staked position will also appreciate in value from interest payments in the same way as the Lender positions do.

There is no limit for how much capital can be staked by the Pool Manager. The Pool Manager should be aware that staking funds into the pool will increase the Pool Funds Limit, i.e. the total amount of capital a Pool can hold. See Section 4.1 for additional details.

The Pool Manager's staked balance is publicly visible as part of available Lending Pool data.

5.2 Unstake

Unstaking is the way in which a Pool Manager is able to exit their position in the pool, and is exclusive to the Pool Manager. Upon unstaking, an equivalent amount of the locked Pool Tokens are burned.

Unstaking is subject to limitations. When the Pool is not closed, a stake-to-pool ratio greater than or equal to the Target Stake Percentage must be maintained by the Pool Manager. When a Pool is closed, the Pool Manager is able to unstake the full amount of capital available in the staked position.

Unstaking is also subject to exit fees. See Section 4.4 for more details.

5.3 Deposit

Depositing provides additional capital to the Lending Pool, and is exclusive to Lenders. An equivalent amount of Pool Tokens are minted and allocated to the Lender on each deposit. See Section 4.3 for more details on Pool Tokens.

A Lender's Pool Token balance is public on the Pool Token contract (ERC20). Both the Lender's Pool Token and equivalent capital balance are publicly visible as part of available Lending Pool data.

5.4 Withdraw

5.4.1 Withdrawing Lender Funds

Withdrawals remove a Lender's capital from the Lending Pool, and is available to the holders of the relevant Pool Tokens. Lender capital can be withdrawn by redeeming an equivalent amount of Pool Tokens, which are burned in the process. See Section 4.3 for more details on Pool Tokens.

Lender capital withdrawals are subject to exit fees. See Section 4.4 for more details.

The amount of capital that can be withdrawn by a Lender is limited by the available liquidity. A Lender's withdrawable capital balance is publicly visible as part of available Lending Pool data.

5.4.2 Withdrawing Fees

The Pool Manager and Protocol Governance wallets acquire earnings (fees) on loan interest payments. See Section 4.6 for a full description of the interest earnings distribution process.

These accrued earnings can be withdrawn from the Lending Pools at any time by the respective wallet owners. Protocol earnings withdrawals are not subject to exit fees and can only be withdrawn in full only (i.e. there are no partial withdrawals of Protocol earnings).

The Protocol earnings balance is publicly visible as part of available Lending Pool data.

5.5 Loan Actions

The following section describes the loan-specific transactions available in the Lending Pool. See Section 4.5 for a full description of loan parameters and mechanics.

5.5.1 Request

Borrowers can request a loan for a specific amount, duration and repayment schedule. The rest of the loan parameters assigned to a loan request will be taken from the current loan template parameters maintained by the Lending Pool (see Section 4.5).

Requesting a loan is exclusive to Borrowers, and only one request per Borrower address is allowed at any given time. Subsequent applications from the same Borrower can be submitted once the current request is either approved or denied.

The amount of loan requests (current and historic) that a Borrower has made, together with whether those requests have been approved or rejected, is a Borrower statistic and is publicly visible.

5.5.2 Approve

Pool Managers can approve pending loan requests; loan approvals are exclusive to Pool Managers. In the loan approval transaction, the Pool Manager may choose to amend the loan parameters in the request, or leave them unchanged. Approving a loan request will remove the loan funds from Pool liquidity and make them available for the Borrower to withdraw.

All approval counts (current and historic) are Borrower statistics and are publicly visible.

5.5.3 Deny

Pool managers can deny pending loan requests; loan denials are exclusive to Pool Managers. Denying a loan request closes the request in question and no further action on the request can be taken by the Borrower or Pool Manager.

Loan denial counts (current and historic) is a Borrower statistic and is publicly visible.

5.5.4 Cancel

Pool managers can cancel previously approved loans as long as the loan in question has not been withdrawn at the time of cancellation. Cancelling a loan will release the allocated loan funds back to the Lending Pool and no further action can be taken by Borrower or Pool Manager on the loan in question.

Loan cancellation is not exclusive to the Pool Manager. An approved loan can also be cancelled by Protocol Governance if the loan has neither been borrowed nor cancelled within a certain amount of time, also known as the Pool Manager's Inactivity Period. The Pool Manager's Inactivity Period is a Lending Pool parameter and has a default value of 90 days.

Loan cancellation counts (current and historic) is a Borrower statistic and is publicly visible.

5.5.4 Borrow

Borrowers can borrow the loans that have been approved by the relevant Pool Manager. Borrowers must inspect the loan parameters before borrowing as the parameters may have been changed by the Pool Manager since the Borrower submitted a request. Borrowing is exclusive to the Borrower and the transacting address must match with the one recorded as the Borrower in the original loan request.

Borrowing a loan will transfer the allocated loan funds, in full, to the Borrowers wallet (i.e. partial withdrawals are not supported). The loan in question will obtain an Outstanding status.

Current outstanding loans count and borrowed amounts are Borrower statistics and are publicly visible.

5.5.5 Repay

Once borrowed, a loan can be repaid by the Borrower or by anyone paying on behalf of the Borrower. Loan payments can be made in part or in full. See Section 4.5.2 for more details on loan balances and interest payment calculations.

Once the current loan balance due is fully paid, the loan in question will obtain a Repaid status.

Loan repayment counts and repaid amounts (current and historic) are Borrower statistics and are publicly visible.

5.5.6 Default

Pool Managers can default previously borrowed loans if the loan in question qualifies for a default.

Defaulting a loan will burn an amount of the Pool Manager's Pool Tokens in the Lending Pool equivalent to the loss amount from the default. If the Pool Manager's stake does not cover the loss the Pool funds will be reduced, with the excess loss being distributed proportionally across the Lenders' positions. See Section 4.3 for more details on Pool Tokens.

Loan defaulting is not exclusive to the Pool Manager. A qualifying loan can also be defaulted by Protocol Governance if the outstanding amount due has not been paid and the loan has not been defaulted within the Pool Manager's Inactivity Period. The Pool Manager's Inactivity Period is a Lending Pool parameter with a default value of 90 days.

Loan default count is a Borrower statistic and is publicly visible.

5.7 Manage

The Pool Manager will be responsible for the following Pool management actions:

- Loan origination.
- Vetting and identity verification of Borrowers.
- Coordinating Lender voting on loan approvals.
- Pausing and resuming lending from the Lending Pool.
- Defaulting loans.

In addition to these management actions, Pool Managers can also manage their Lending Pools using the following parameters:

Parameter Name	**Default Value**	**Purpose**
Pool-Level Parameters		
Manager Earning Factor	150%	Pool Manager's tool to control their earning leverage within the Pool
Target Liquidity Percent	0%	Defines the proportion of the Pool to keep liquid (i.e. not loaned to Borrowers)
Minimum Loan Amount	100	Enforces a lower bound for the loan amount on loan requests
Minimum Loan Duration	1 day	Enforces a lower bound for the loan duration on loan requests
Maximum Loan Duration	51 * 365 days	Enforces an upper bound for the loan duration on loan requests

Parameter Name	Default Value	Purpose
Template Loan APR	30%	Provides a default loan APR for new loan applications
Template Late Loan APR Delta	5%	Provides a default APR delta (increment) applied to late payments
Template Loan Grace Period	60 days	Provides a default grace period for late loan payments
Loan-Level Parameters (Pool Manager can change on a loan-by-loan basis)		
APR	Template Loan APR	Allows Pool Manager to change the APR for a specific loan from the Template Loan APR
Late Payment APR Increment	Template Late Loan APR Delta	Allows Pool Manager to change the Late Payment APR for a specific loan from the Template Late Loan APR Delta
Loan Duration	Borrower-set	Allows Pool Manager to change the specific loan duration from the value submitted on the Borrower loan request
Loan Grace Period	Template Loan Grace Period	Allows Pool Manager to change the specific loan grace period for late payments
Loan Amount	Borrower-set	Allows Pool Manager to change the specific loan amount from the value submitted on the Borrower request

5.8 Govern

With respect to Lending Pool management, Protocol Governance will be able to:

- Transfer the Lending Pool management to a new wallet address.
- Withdraw Protocol Earnings from the Lending Pool.
- Transfer the Protocol wallet address (along with the accrued earnings from the Lending Pools) to a new wallet address.
- Enact emergency pauses on Lending Pools.
- Enact emergency resumption of Lending Pools.

In addition to these actions, Protocol Governance can also set the following parameters that affect Lending Pool Management:

Parameter Name	Default Value	Purpose
Target Stake Percentage	10%	Enforces the stake-to-pool ratio for the Pool Manager and dictates maximum Pool size
Protocol Earning Percentage	5%	Determines the share of the paid interest allocated to the Protocol
Maximum Manager's Earning Factor	500%	Enforces a limit the Pool Manager's leveraged earnings factor
Exit Fee	0.5%	Fee levied on Lenders who withdraw their funds from a Lending Pool

6 Failure Mode Analysis and Safeguards

We have implemented a number of solutions (i) to safeguard the protocol against potential malicious actors and (ii) to facilitate exit routes from potential deadlock scenarios.

6.1 Deposit Entry Attack

A malicious party could anticipate when loan payments are due, deposit a significant amount of capital just before the loan payments are made and then withdraw their capital shortly after the loan payments. Having the most significant share of the pool would give the malicious party the largest earnings from the paid interest in a very short amount of time.

Moreover, these parties could be the Borrowers themselves, flash loaning deposits, making loan repayments and withdrawing the flash loan deposit in the same transaction. This would result in abuse of the Lending Pool and render loans effectively near-zero interest.

Two measures have been implemented to prevent such abuse of the Lending Pool and to prevent the potential stealing of the loan interests:

1. Firstly, enforcing the Target Stake Percentage parameter sets a limit on maximum pool size and limits the deposit amount at any given time, reducing the potential damage caused by a deposit entry attack. See Section 4.1 for an explanation of Pool Funds Limit.

2. Secondly, enforcing an exit fee for withdrawals will apply a percentage exit fee to the withdrawn capital that scales with the withdrawal amount. The exit fee is distributed to the Lenders that remain in the Pool. See Section 4.4 for an explanation of exit fees.

The combination of these two measures above makes the deposit entry attack impractical.

6.2 Inactive Pool Manager

Pool Manager inaction could cause a number of issues:

- Loan requests being ignored
- Funds not being made available to Borrowers following loan approval votes
- Funds being stuck in an allocated state where the Borrower has simply not taken action to withdraw loans which have been approved (and an absent Pool Manager not at hand to cancel those loans to return the funds to the Lending Pool).

A Pool Manager could also neglect to default loans, delaying the inevitable loss to their stake and keeping the leveraged earning income stream in the interim. This would result in reduced earnings for the Lenders and an artificially-maintained Lending Pool score.

To prevent the scenarios above, qualifying loans can be cancelled and/or defaulted by Protocol Governance after the Pool Manager's Inactivity Period has passed on the loans in question. See Section 5.5.4 on Loan Cancelling and Section 5.5.6 on Defaults for more details.

6.3 Loss of Access to a Borrower Wallet

If a Borrower has lost the keys to their original wallet, there would be no way to repay their loan and the loan would have to be defaulted.

To prevent this, the Lending Pool has an additional method to allow 3rd parties to repay a loan on behalf of the original Borrower. In practice, willing Borrowers could open a new wallet and close their original loans. The 'Repaying on Behalf' transaction is designed to be intentional, as it requires both the Loan ID and matching original borrower address to be specified as valid. See Section 5.5.5 on Repaying for more details.

6.4 Pool Manager Position Lockout

A situation could arise where the Pool Manager is never able to unstake funds from the Lending Pool due to Pool mechanics that require the Pool Manager's stake to act as first loss cover for loan defaults, and due to the fact that the Lenders may choose not to withdraw their capital.

Because of the uncollateralized nature of the lending undertaken using the protocol, removing the first loss staking requirement is not an option.

Instead, the Lending Pool provides the Pool Manager with a Pool closure mechanism. Once there are no outstanding or approved loans, the Pool Manager can close the Pool. A Pool in a closed state would allow anyone, including the Pool Manager, to withdraw their positions, whilst the Pool Manager can unstake their funds when the Pool is in a closed state. See Section 5.7 for further details on Lending Pool Management.

7 Conclusion

Closing the credit gap for SMEs is not a new problem, and has indeed attracted a lot of attention from a large number of firms who have both recognised the failings of the current banking system and tried to fix the issue. Alternative sources of lending for SMEs have sprung up in the form of microfinance firms, peer-to-peer lending platforms, crowdfunding platforms and other shadow banking institutions, who have all taken a slightly different approach to tackling the problem.

The improvement that these firms have made to the lending process for SMEs is extremely welcomed, and a significant step forward when compared to traditional bank lending. However, we believe that by leveraging blockchain and crypto technology SME lending needs can be addressed even more effectively, for the following reasons:

- **Efficiency** - the design of Sapling protocol, together with the underlying distributed ledger technology that it builds upon, makes the lending process for SMEs quicker (instant settlement) and more efficient (less administration).
- **Increased Participation and Access to Capital** - by using smart contract technology Sapling will allow for a far wider participation of lenders in the SME finance market; once cleared by Protocol Governance, anyone will be able to launch a Lending Pool in an instant to issue loans. Sapling will provide a “permissionless lending infrastructure in a box” for anyone wanting to deploy capital to support SME growth - an individual, an investment club, an existing fintech lending platform, a micro-financing firm, a DAO or a charitable society. By allowing any of these potential participants to become a lender on the blockchain Sapling will immediately provide SMEs with a wider pool of potential finance options to draw on.
- **Transparency** - a key advantage to using blockchain technology lies in its ability to inject transparency into the lending process. Any activity undertaken using the Sapling protocol will be on-chain, immutable and open for all to see, whether that is the performance of a Pool Manager or how reliable a Borrower is when it comes to repaying a loan. Over time this transparency becomes more powerful, whether to the benefit of Lenders who can easily interrogate a Pool Manager’s long-run performance before investing in that Pool, or for Borrowers who can point to (and market) their on-chain repayment history (i.e. they will own their credit score and history) to negotiate lower

interest rates on subsequent loans. Blockchain transparency therefore leads to more efficient investment decisions and fairer, more affordable lending fees for Borrowers.

- **Open and Borderless Capital** - blockchain technology supports the flow of capital across borders, allowing borderless allocation of capital to where there is demand and where it can generate returns. Sapling will provide an efficient way for (e.g) a Lender in India to allocate capital to an SME in Bolivia in a simple and quick manner, generating a return for the Lender and fulfilling the demand for capital from the SME. In extreme cases this can circumvent inefficient local banking systems or even bad actors (governmental or within the banking industry) which fail to supply the capital that local SMEs demand.

- **Open Source** - unlike current fintech solutions, Sapling will be built as an open source protocol solution with the express intention of stimulating others to improve the current credit offerings to SME around the world. If this means that the community identifies a way to improve the protocol that the core team has missed, we see this as success. If it means that the community builds a suite of tools to help participants, we see this as success. If someone forks the code and builds a better version of the protocol that gives SMEs a better source of affordable credit, we see this as success. An open source approach focuses the team on creation of value for the community rather than extraction of value for the DAO or token holders.

- **Incentivisation** - as with many crypto protocols, Sapling will be able to use digital primitives (in the form of utility tokens or NFTs) to incentivise and reward good actors, whether that is Borrowers who keep to repayment schedules or community members who develop tools to help protocol participants.

- **Community** - It is hard to escape the impact of community in crypto, with many protocols achieving success in no small part because of the passionate communities that support them and are willing to go the extra mile in order to help protocol growth. We intend to build and leverage a strong community to grow Sapling globally, crowdsource ideas, use the community to source Pool Managers, Lenders and Borrowers throughout the world and transition ownership of the protocol to that community over time. We have seen the difference a strong community makes in terms of protocol traction and see this as a key pillar of our strategy.