PayStream V1 Whitepaper

Maushish Yadav

December, 2024

Abstract

PayStream V1 provides APY-optimized rates through matching lenders and borrowers with its P2P engine and allows them to repay the lending amount through various customizable streams of funds while also giving the same rewards that both parties get in traditional peer-to-pool protocols like Kamino and Margin-Fi.

Contents

1 Motivation

The decentralized finance (DeFi) landscape has grown significantly since the FTX crash on Solana. Still, existing protocols often lack flexibility in repayment options and fail to optimize lending rates for all participants.

Let's take the Kamino Protocol as an example:

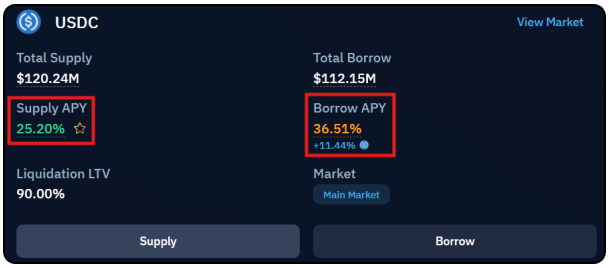

Figure 1: APY spread in app.kamino.finance

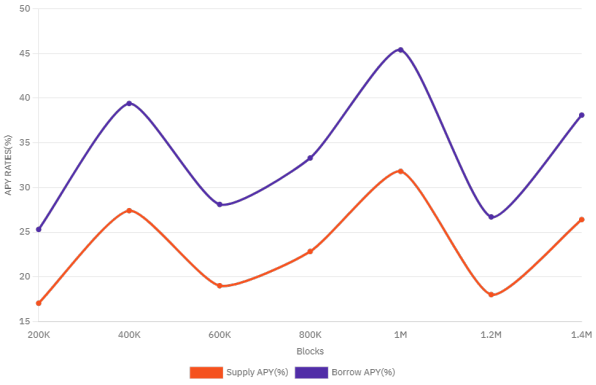

Figure 2: APY Spread of USDC on Kamino over 1.4M Solana Blocks.

The spread between APYs is intentional, as keeping utilization rates below 100% enables users to both withdraw current funds or borrow new funds at any time. This interesting property is sometimes referred to as the "liquidity" of the money market. The design choice of this "pool model" keeps suppliers and borrowers motivated whilst preserving the liquidity of positions.

However, this model has proven very inefficient as suppliers are not competing with each other. Moreover, one may also remark that rates are not decided by the offer and demand of the market but are biased by the borrowing rate curve and Protocol reserve factor, chosen by the platform. Like here Kamino supplying and borrowing rates is well documented:

$$B_t = B_F + \frac{B_C - B_F}{U_C - U_F}(U_t - U_F)$$

$$S_t = U_t \cdot B_t \cdot (1 - R_t)$$

Where:

- B_t: Current borrow interest rate

- B_C: Borrow interest rate at the ceiling utilization knot point
- B_F: Borrow interest rate at the floor utilization knot point
- U_t: Current utilization rate
- U_C: Utilization rate at the ceiling knot point
- U_F: Utilization rate at the floor knot point
- S_t: Current supply interest rate
- B_t: Current borrow interest rate
- R_t: Current reserve factor (protocol take rate)

Thus the spread is given by:

$$B_t - S_t = B_F + \frac{B_C - B_F}{U_C - U_F}(U_t - U_F) - U_t \cdot B_t \cdot (1 - R_t)$$

A natural idea would be to build some sort of order book to register every position in a P2P fashion.

The PayStream Protocol leverages the currently existing PLFs (Protocol-Level Frameworks), such as Margin-Fi and Kamino, to create an efficient - yet liquid - P2P market of supply and borrow positions with near-zero spread.

Users eventually get permanent positions with self-adjusting rates, being, at best, the exact rate that the matched borrower is paying, and at worst, the rate of the PLF that PayStream falls back on. PayStream can therefore be described as a liquidity pool optimizer, where both borrowers and suppliers benefit from improved rates while preserving the same guarantees and the same liquidity.

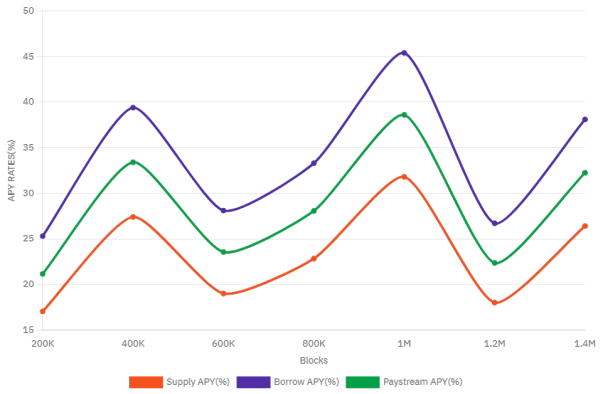

Figure 3: Kamino rates compared to the Paystream P2P APY on USDC over 1.4M Solana blocks

While most DeFi platforms focus solely on lending and borrowing, repayment processes remain rigid and unsuitable for modern use cases. Freelancers, startups, and businesses often require flexible payment structures to match irregular cash flows or milestone-based funding needs.

PayStream addresses this gap with its customizable streaming services:

- Borrowers can repay in increments over time, aligning with their income cycles.
- Payment streams support various models, including linear, cliff-based, and step-wise schedules.

The PayStream Streaming Module allows Solana-based businesses to repay loans in flexible streams, aligning with cash flow, while freelancers receive steady, real-time payments. This ensures predictable income for recipients, improves cash flow management for borrowers, and fosters efficient fund utilization within the Solana ecosystem.

2 The PayStream Protocol

2.1 High-level description

PayStream combines APY-optimized lending with programmable payment streaming solutions, tailored for different types of users. Leveraging a peer-to-peer (P2P) matching engine, PayStream aligns borrower and lender requirements, ensuring competitive rates while maintaining liquidity through integrations with PLFs like Kamino and MarginFi. Borrowers benefit from customizable repayment streams, enabling precise, real-time fund disbursements based on linear, cliff, or step-wise payment curves.

Now in this section it is assumed that the Paystream Protocol is only integrated with kamino and is then called Paylend-Kamino

From a user point of view, Paylend-kamino operates very similarly to Kamino: users can supply, withdraw, borrow and repay assets witht the same liquidity as Kamino. Liquidators can liquidate undercollateralized credit lines according to the same collateral factors and the same price oracles as Kamino. One should not experience any different from what they are used to, except that the rates are more interesting. Paylend-Kamino acts as a proxy between the user and Kamino. Let's take at how assets flow in this setting.

- **Supply:** The user supplies tokens to PayStream (Step 1). In the background, the protocol deposits these tokens into Kamino (Step 2) and generates KMNO tokens (Step 3). PayStream holds the KMNO tokens and uses them to manage incentives from the Kamino pool.

Figure 4: A supplier deposits liquidity to PayStream

- **Borrow:** The user first provides collateral, such as SOL tokens, with collateral factors determined by the underlying Protocol-Level Framework (PLF) like Kamino (Step 4), and triggers the borrow function (Step 5). In the background, the protocol performs the following steps:

1. **Matching Engine Activation (Step 6):** The PayStream P2P matching engine attempts to match the borrower's request with available lender liquidity directly within the protocol.
2. **Fallback to PLFs (Step 7):** If the required liquidity is not matched through P2P, the protocol allocates funds from Kamino's pool or similar PLFs.
3. **Liquidity Transfer (Step 8):** PayStream moves liquidity directly from the matched lenders or PLFs into the borrower's wallet as the borrowed asset.

Once the borrower receives the funds, repayment begins via the PayStream streaming module (Step 9). This module allows borrowers to repay in real-time or using customized schedules, such as linear, step-wise, or cliff-based streams. This ensures flexibility for borrowers while maintaining predictable, consistent returns for lenders.

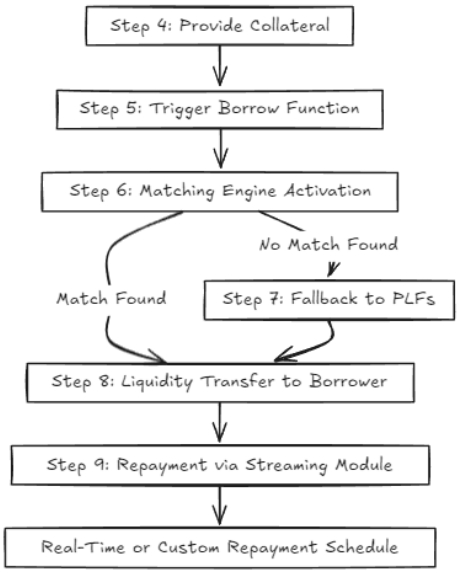

Figure 5: A borrower takes liquidity from PayStream

During the P2P position, users are moved out of the Kamino pool and seamlessly hold a P2P position with a utilization rate of 100%. When the borrower requires a match, PayStream moves liquidity from Kamino's pool, ensuring that matched lenders do not share rewards with the rest of the pool. This results in improved APYs for both parties: lenders benefit from higher returns compared to remaining in the pool, and borrowers receive lower borrowing costs. For example, instead of an underlying supply APY of 25.2% and borrow APY of 36.5% in Kamino's pool(USDC token), PayStream creates a win-win scenario where both supply and borrow APYs converge to near 30.7%.

2.2 P2P Engine

PayStream is a hybrid interest rate mechanism that combines Peer-to-Peer (P2P) matching with fallback integration into Protocol-Level Frameworks (PLFs) like Kamino and MarginFi. It is designed such that, if a borrower or lender is unable to find a counterpart via the P2P mechanism, PayStream seamlessly falls back onto PLFs by allocating unmatched funds into their liquidity pools. Under this configuration, PLFs act as the supplier or borrower of last resort, ensuring users are at least as economically rewarded as they would be by directly using the PLFs.

2.2.1 Liquidity

One may ask how PayStream ensures full liquidity in specific scenarios, such as when a lender wants to exit a Peer-to-Peer (P2P) position where their capital is fully borrowed. The core principle is that in situations where a PayStream user cannot exit, the protocol seamlessly falls back to the underlying Protocol-Level Frameworks (PLFs), such as Kamino or MarginFi, ensuring liquidity and flexibility.

For example, instead of an underlying supply APY of 25.2% and borrow APY of 36.5% in Kamino's pool (USDC token), PayStream creates a win-win scenario where both supply and borrow APYs converge to near 30.7%. In this scenario, if a lender (Alice) wishes to withdraw her supplied USDC but the borrower (Bob) has not yet repaid the loan, PayStream borrows USDC from Kamino using Bob's collateral as security.

Here, PayStream also offers Alice the option to withdraw her funds through the streaming module, allowing her to receive payments over a defined period (e.g., weekly or monthly) instead of a lump-sum withdrawal. This gives Alice a flexible repayment structure while maintaining liquidity, and it helps Bob continue repaying the loan through flexible streams as well. If Alice re-engages in the future, PayStream reconnects them at the original P2P rate, offering both parties improved rates compared to traditional pool-only models.

2.2.2 Liquidation Mechanism

PayStream has its own liquidation mechanisms, but directly copies the parameters of the underlying PLFs, such as Kamino or MarginFi. The protocol mechanically mirrors the same collateral factor, liquidation conditions, and price oracles fetched on-chain from the underlying PLF. In this way, the liquidation guarantees for PayStream users are the same as those on the underlying PLF.

It is important to note that PayStream's contract may sometimes have a borrow position on the underlying PLF. However, PayStream itself can only be liquidated if its aggregate position, which consists of all PayStream users' positions, becomes eligible for liquidation. To prevent this, PayStream ensures that users' positions are liquidated when necessary, maintaining the safety of PayStream's overall position in the PLF.

2.2.3 Supply and Borrow Balances: onPool vs inP2P

In PayStream, the supply balance and borrowing are each split into two variables: onPool and inP2P. This reflects whether the user's liquidity or borrow request has been matched with a counterpart in the Peer-to-Peer (P2P) market, where they benefit from the P2P APY, or if the supply/borrow position is in the underlying Protocol-Level Framework (PLF) like Kamino.

For the **onPool** case, when the user's liquidity has been supplied or borrowed on Kamino, PayStream directly tracks the user's deposit or borrow position within the Kamino pool. In this case, the user's balance and generated yield are based on the rewards provided by Kamino's liquidity pool.

In the **inP2P** case, PayStream uses a mechanism inspired by current liquidity protocols. PayStream introduces a unit called 'p2pIndex', whose underlying value grows over time. This unit is used to describe the "on PayStream" debt (both for suppliers and borrowers). The value of the debt is linked to the token's unit by the variable `p2pIndex`, according to the following formula:

$$\text{valueInUnderlying} = \text{valueInP2PUnit} \times \text{p2pIndex}$$

The `p2pIndex` is updated based on the mid-rate yield per block via an internal function, which is called each time a user calls a function that requires the conversion to this unit. Note that the complexity remains constant.

Example: Assume 1 SOL = 200 SOL (representing the deposit amount in Kamino). Alice supplies 1 SOL to PayStream, so her supply balance becomes:

- onPool: 200 SOL (deposited in Kamino)
- inP2P: 0 p2pSOL

Now Bob borrows 1 SOL. At this moment, 1 SOL = 100 p2pSOL. Alice's supply balance becomes:

- onPool: 0 SOL (Kamino pool)
- inP2P: 100 p2pSOL

Note that Bob shares the same numbers for his borrowing balance. A year later, if the mid-rate remains 1.4%, the price of p2pSOL should be approximately 1 SOL = 98.6 p2pSOL.

Market Imbalance: In current PLFs, there is a great imbalance between the volume of loanable funds compared to the volume of demand, which is intentional since liquidity pools need more suppliers than borrowers to function. This is not the case with PayStream, which can handle more borrowers than suppliers and still remain fully liquid.

Moreover, in PayStream, the imbalance is not necessarily in favor of suppliers since the rates may be different. The imbalance is highly dependent on market conditions and thus on the P2P APY itself. The P2P APY in the example above is an arbitrary choice and should be flexible to reflect supply and demand. For example, the P2P APY could be adjusted closer to the supply APY of the PLF instead of being positioned in the middle to attract more borrowers.

Without needing to build a complete competitive interest rate market, PayStream introduces flexibility by updating the P2P APY according to market conditions, ensuring optimal balance between supply and demand.

2.2.4 P2P Positions and Matching Engine in PayStream

P2P positions on PayStream benefit from a 100% utilization rate, meaning there is as much supply as borrow demand. This is a key difference from other pool-based Protocol-Level Frameworks (PLFs), where there is often more loanable liquidity than borrow

demand, leading to low utilization rates in their pools and consequently, creating the APY spread.

In PayStream, there is often an imbalance between supply and demand. For example, one might expect n units of suppliers attempting to match k units of borrowers, with $n > k$ (or vice versa). The protocol needs to select k suppliers to benefit from the P2P APY, while the remaining $n - k$ suppliers will be placed into the underlying PLF, such as Kamino. The module responsible for choosing and matching the k suppliers is referred to as the **matching engine**.

There are several ways to design a matching engine:

- Maximize matched volumes.
- Minimize gas usage and avoid small leftover amounts (dust).
- A passive user or contract should be able to benefit from PayStream simply by supplying or borrowing.
- The use of PayStream should benefit as many users as possible.

It might seem that efficiency is gained at the advantage of a small group of matched suppliers, but this is not the case. The issue lies in the current DeFi interest rate market, which disincentivizes borrowing demand. With PayStream, more borrowers are attracted, and the entire cash-flow market grows. Additionally, as described in Section 2.3, the P2P APY will self-adjust according to supply and demand, attracting even more users.

One might worry that this matching engine requires looping over all users, making it inefficient and non-scalable with blockchain constraints. The PayStream algorithm does iterate through users; however, the gas cost for matching is chosen by the DAO, which sets how many matches are done for each user. When there is insufficient gas for matching, the algorithm falls back to the PLF for the remaining liquidity. This ensures the full scalability of PayStream while maintaining efficiency.

2.2.5 Rewards and inverted spreads

THere are rare incidents where a token APY spread gets inverted i.e. $APYborrow < APYsupply$,

In this scenario Paystream guarantees at least the liquidity inflated APY but it will be less likely to have a strictly better APY.

PayStream operates under the following regime:

- APY Spread Not Inverted: PayStream moves supply and borrowing positions in and out of Kamino to maximize APYs(Most likely scenario)
- APY Spread Inverted:PayStream and user can swap Earned rewards from $PAYSTREAM$ token, incentivizing users by providing abonus.

Thus, in every scenario, PayStream users are better off by using the platform, as it efficiently optimizes APYs through P2P matching without relying on additional rewards or external incentive mechanisms.

2.3 Streaming Module

The PayStream streaming module is built on-chain using Solana program constructs, which enable flexible, time-controlled withdrawal functionality. This feature allows users to receive their funds over a predefined period. In PayStream-Streaming module , token distribution is supported through three distinct mathematical curves: Linear, Cliff, and Step-wise

There will be more and custom curves that will be added in the future.

2.3.1 Linear Stream

A Linear Stream ensures that tokens are distributed at a constant rate over a specified time period. The mathematical equation for the linear distribution of tokens is:

$$y(t) = m \cdot t + b$$

Where:

- $y(t)$ represents the amount of tokens distributed at time t.
- m is the constant rate of distribution (the slope).
- b is the initial amount of tokens (the y-intercept).

In a Linear Stream, tokens are distributed uniformly over time, ensuring consistent and predictable payouts for users who have borrowed or lent tokens.

2.3.2 Cliff Stream

A Cliff Stream utilizes a piecewise function to describe its distribution mechanism. Initially, no funds are distributed during a specific cliff period. Once this period ends, the stream begins to distribute tokens according to the agreed-upon function. The function is defined as:

$$y(t) = \begin{cases} 0, & \text{for } t \leq t_0 \\ f(t), & \text{for } t > t_0 \end{cases}$$

Where:

- $y(t)$ is the amount of tokens distributed at time t.
- t_0 is the cliff duration, marking the initial period during which no funds are distributed.
- $f(t)$ is the function describing the continuous distribution of tokens after the cliff period ends.

The Cliff Stream structure is useful for situations where payments are contingent upon meeting certain conditions. The initial cliff period provides time for the parties involved to meet specific goals or performance metrics, after which the distribution of funds begins.

2.3.3 Step-wise Stream

The Step-wise Stream represents a distribution model based on fixed time periods, such as weekly or monthly payments. It disburses tokens at regular intervals, allowing for consistent and predictable payouts over the engagement period. The mathematical representation of this stream is:

$$y(t) = k \cdot n(t)$$

Where:

- $y(t)$ represents the total amount distributed at time t.
- k is the fixed amount disbursed per time period (e.g., weekly or monthly).
- $n(t)$ is the function that counts the number of time periods that have passed by time t.

The Step-wise Stream model is ideal for long-term lending or borrowing engagements, ensuring that repayments or lending returns are predictable and consistent over the term of the agreement.

3 Use Cases

3.0.1 Aggregators

Aggregators like Earnpark constantly try to find the best rates between different supply or borrow markets. Remark that if Paylend-kamino is aggregated with Kamino, an aggregator will never switch back to Kamino. Moreover, PayStream's interfaces are the same for Paylend-kamino, Paylend-MarginFi and others. An integrator will be much more friendly with a single interface rather than many. This way, one integrator can consider only having the Paystream.

3.0.2 Strategists

Protocols like Lulo build strategies to maximize the earnings. Such protocols use Kamino or MarginFi and thus can use Paylend-Kamino or Paylend-MarginFi to generate even better returns without taking additional market risks.

3.0.3 Individuals

Individuals like freelancers can interact with PayStream to either borrow funds with customizable repayment streams or lend assets for competitive interest rates. Lenders earn returns through PayStream's P2P matching engine, while borrowers can access liquidity with flexible payment options, making it suitable for both simple and advanced financial strategies.

3.0.4 Stablecoins

Decentralized stablecoin protocols create strategies for their collateral to generate returns. However, these strategies must remain fully liquid to allow the protocol to redeem tokens and maintain the asset peg. PayStream optimizes these strategies by integrating with Kamino, maintaining liquidity while improving yield rates. With PayStream's P2P engine, stablecoin protocols can further optimize yields, offering higher returns for users supplying collateral.

4 Conclusion

PayStream V1 bridges the gap between traditional peer-to-pool lending models and innovative programmable payment solutions. By offering optimized APY rates, flexible repayment streams, and robust integration with established PLFs, PayStream is poised to become a cornerstone of the DeFi ecosystem.

5 Acknowledgement

Paystream aims to become a decentralized common good. This White Paper itself is the product of intense research. In particular the author would like to express their gratitude to Shek, Sabir, Nitt for their invaluable feedbacks

References

Email address: maushishbusiness@gmail.com