EigenYield

Buying the swap-fee cash flow of locked pools,
and pricing it as a single factor

Robinhood Chain, chain id 4663 · September 2, 2026

eigenyield.me

Abstract

An index of a launchpad market holds its tokens and is therefore long their price. This paper describes an instrument that is long their *trading* instead. On Robinhood Chain, a graduated Pons pool charges no liquidity-provider fee; a hook charges 1% of every swap and pays a fixed share of it to an address the current holder can transfer. That transferability makes the fee stream a separable asset. EigenYield acquires such streams, and issues one ERC-4626 share against them.

Because pool fees are driven by a common factor, the streams are not priced individually. We build the epoch-by-pool matrix of accrued fees, take its first principal component, and use the non-negative part of that eigenvector as the allocation. The result is an instrument whose value moves with market activity and is structurally insensitive to the price of any constituent. We give the estimator, the eligibility conditions, the harvest accounting, and an attribution rule that remains correct when the fee escrow can only be swept in full.

1 Two different bets

Let a pool trade volume V_t during epoch t at price P_t. A holder of the underlying token has a position whose value is proportional to P_t. A holder of the pool's fee stream has a position whose value over the epoch is proportional to ϕV_t, where ϕ is the fee rate.

These are not the same exposure, and they are not reliably correlated. The clearest illustration is a sell-off: P_t falls sharply while V_t reaches its maximum, because a crash is the most heavily traded event in a token's life. A position in ϕV_t is paid most on the day a position in P_t is worth least.

Nearly every index product sells exposure to P_t. The instrument described here sells exposure to ϕV_t, and never holds the underlying at all.

Two properties follow immediately, and we state both because the second is a genuine limitation rather than a feature:

1. The share price cannot fall because a constituent fell. There is no inventory to mark down.
2. The share price can go flat. If nobody trades, nobody pays. The instrument is a factor bet on aggregate market activity, and it earns nothing while that activity is absent.

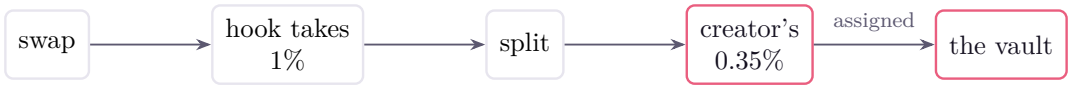
2 The instrument on Robinhood Chain

A Pons launch that completes its bonding curve *graduates*: its liquidity is minted as a single full-range Uniswap v4 position into a locker that never releases it. The pool's own fee is zero. Instead the protocol hook charges $\phi_{\text{hook}} = 1\%$ of each swap and splits it,

$$\underbrace{30\%}_{\text{protocol}} + \underbrace{35\%}_{\text{buyback}} + \underbrace{35\%}_{\text{creator}} = \phi_{\text{hook}},$$

so the creator's residual is $\phi = 0.35\%$ of traded volume. That residual is paid to a *creator fee recipient*, an address recorded on the launch factory which the current holder may transfer.

This is the entire mechanism. A creator assigns the recipient to the registry for a term and is paid up front; the registry claims the accrued fees each epoch and returns the recipient at term end. The token never moves, the locked position never moves, and no party takes price exposure to the constituent.



3 Eligibility

A stream enters the basket only if all of the following hold. Three of the four are answered by the chain rather than by a data provider, because a provider can be wrong and, on this chain, demonstrably is: several pools it files under Pons are not registered on the factory at all.

1. **Registered and graduated.** The factory reports the token exists and is in the graduated phase.
2. **No pending redirect.** The protocol may propose a new fee recipient behind a three-day timelock. A pool with a proposal outstanding leaves the basket immediately.
3. **Depth.** In-range liquidity, read from the v4 state view, is at least $D_{\min}$.
4. **Age.** At least $\tau_{\min}$ has elapsed since graduation.

The age condition is calibrated to observed behaviour rather than chosen for neatness. Graduates on this chain turn over in days: the deepest pools on the board are routinely one to three days old, so a fortnight-long threshold gates out the entire asset class. We use $\tau_{\min} = 1$ day at the contract level and a three-day floor for inclusion in the estimator, which excludes launch-day volume without excluding the market.

4 The factor

4.1 Estimator

Let $F \in \mathbb{R}^{T \times N}$ collect the fees accruing to N eligible streams over T epochs, so F_{ti} is the fee earned by stream i during epoch t. Standardise each column,

$$Z_{ti} \;=\; \frac{F_{ti} - \mu_i}{\sigma_i}, \qquad \mu_i = \frac{1}{T}\sum_t F_{ti}, \qquad \sigma_i^2 = \frac{1}{T}\sum_t (F_{ti} - \mu_i)^2,$$

so that a large pool does not enter the estimate merely for being large. Form the sample covariance and take its leading eigenpair,

$$\Sigma \;=\; \frac{Z^\top Z}{T-1}, \qquad \Sigma v_1 \;=\; \lambda_1 v_1, \qquad \lambda_1 = \max_j \lambda_j.$$

We compute (λ_1, v_1) by power iteration with Gram–Schmidt deflation for the subsequent components. The explained-variance ratio

$$\rho_1 \;=\; \frac{\lambda_1}{\operatorname{tr}\Sigma}$$

is the share of fee variation attributable to the common factor, and is published live. On the current basket ρ_1 sits near 0.46.

4.2 Allocation

Allocation weights are the non-negative part of the leading eigenvector, renormalised:

$$w_i \;=\; \frac{\max(v_{1,i},\,0)}{\sum_j \max(v_{1,j},\,0)}.$$

Negative loadings are set to zero rather than shorted. A stream that earns well but only when its own community is active carries idiosyncratic rather than market risk; the vault declines to pay for it. Weights govern where newly harvested capital is deployed, not what is already held.

4.3 What the factor is not

ρ_1 measures co-movement, not profitability. A basket can have a high ρ_1 and earn very little, if the shared factor is itself near zero. The factor answers "how much of this is one thing?", and the level of that one thing is reported separately as realised fee flow.

5 Vault accounting

The vault is an ERC-4626 over wrapped ether. Let A_t be total assets and S_t total supply; the share price is A_t/S_t, and a deposit of a mints aS_t/A_t.

5.1 Vesting

A harvest of size H at time t_0 is not credited at once. Writing $L(t)$ for the still-locked portion,

$$L(t) \;=\; H\cdot\max\!\left(0,\;1-\frac{t-t_0}{\Delta}\right), \qquad A(t) \;=\; B(t)-L(t),$$

where $B(t)$ is the vault's balance and Δ is one epoch. Without this, an observer could deposit one block before a harvest and redeem one block after, capturing a full epoch of fees earned by others' capital. The linear release makes that arbitrage worth approximately nothing.

5.2 Fees

A performance fee of 10% is taken from each harvest. There is no management fee, no deposit fee and no withdrawal fee. Harvesting is permissionless: the first caller in an epoch receives a fixed tip.

6 Attribution under a single-sweep escrow

The fee escrow exposes `claim()`, which takes no arguments. It therefore sweeps everything owed to the caller in a given currency and cannot be invoked per pool. When the registry holds several streams in the same currency, a single claim returns one aggregate amount R that must be divided among them.

Let w_i denote the pending fees of pool i read from the hook *immediately before* the claim, and let $\mathcal{G}$ be the set of pools in the currency group. We attribute

$$R_i \;=\; R\cdot\frac{w_i}{\sum_{j\in\mathcal{G}} w_j}, \qquad \sum_{i\in\mathcal{G}} R_i = R.$$

Proposition 1. *The attribution is invariant to whether the hook reports a pool's total fee or only the creator's share of it.*

Proof. Suppose the hook reports $w_i = k\,c_i$, where c_i is the creator's accrued share and $k > 0$ is a constant of proportionality fixed by the split policy. Then

$$\frac{w_i}{\sum_j w_j} = \frac{k c_i}{k \sum_j c_j} = \frac{c_i}{\sum_j c_j},$$

so k cancels and R_i is unchanged. □ □

The measurement must be taken before the claim, since afterwards every w_i is zero. Reading order is therefore part of the correctness argument, not an implementation detail.

One consequence is not removable at the contract level and is stated plainly: a pool excluded by an eligibility condition still has its fees swept by the same call, and those fees are distributed among the pools that were attributable. The escrow offers no per-pool selector, so this is managed operationally, by grouping, rather than solved in code.

7 Risk

Stream redirection.
The protocol may propose a new fee recipient behind a three-day timelock. Any position built on these fees can therefore be terminated with three days' notice by a party other than the holder. The registry drops such a pool on sight, which stops further capital being committed but does not recover what was already paid for the term. This is the dominant risk and it is structural.

Silence.
The instrument earns only while the market trades. A quiet chain produces a flat share price, not a falling one, but flat is not free.

Contract risk.
The vault, the registry and the underlying launch protocol are unaudited. Permanently locked liquidity removes the withdrawal risk of the underlying pool; it does not remove implementation risk here.

Administrative powers.
The owner sets the fee recipient, the performance fee, the eligibility parameters and the swap adapter. Adapter changes are timelocked and reinvestment is bounded per call, but the role remains privileged and belongs behind a multisig.

Price risk.
Structurally low. The vault holds claims, never tokens. A constituent falling 80% is a high-volume day for the harvest rather than a drawdown.

8 Deployment

Contract	Address
EigenYield vault (ERC-4626 share)	`0xb6e54f359A4F03EB02De3D75e267c7A0ea403006`
Fee rights registry	`0x058e2Ed9BEc0ab122a75f29e1f70D132FF872448`
Fee claim encoder	`0xfeDA448f38aa1979341D77818d15F951E922a36b`
Swap adapter	`0xDC35aaB51FB38d9a778cE5168133Ee43557D8265`
Vault asset (WETH)	`0x0Bd7D308f8E1639FAb988df18A8011f41EAcAD73`
Pons factory	`0x7eD598BcEf8bd9Edd8C97A195C6d13f40801EC7e`
Pons fee hook	`0xE5e702641Ea86F4ae6cC3cDaeD2B886f976Be044`
Pons fee escrow	`0xd3AFEB2a57f70eF218Aa82451c51B2fb0416Ac9e`
Uniswap v4 state view	`0xf3334192d15450cdd385c8b70e03f9a6bd9e673b`

The basket, the decomposition, the share-price series and a health check are served without authentication at eigenyield.me, under `/api/pools`, `/api/eigen`, `/api/nav` and `/api/health`. Every figure in this paper can be recomputed from them.

This document describes a smart contract, not a registered fund, and is not investment advice. The contracts are unaudited.