

# SYNTHR – Omnichain synthetic assets

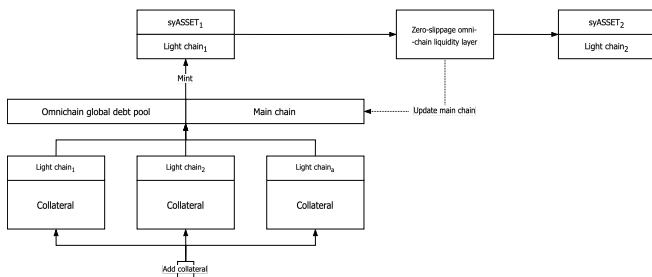
Ritwik Rudra      Vithuran Karvannathan

05/07/2022

## Introduction

SYNTHR utilizes omnichain synthetic assets to power a zero-slippage execution environment. The protocol brings together a combination of pull and push oracles, a zero-slippage omnichain liquidity layer, and multiple independent consensus layers, providing you with a unique cross-chain solution.

### 1. Omnichain global debt pool



The omnichain global debt pool aggregates cross-chain collateral and debt balances, allowing you to add high-quality liquid collateral on multiple chains and mint omnichain syASSETS on any chain. This enables the creation of overcollateralized debt positions with decentralized cross-chain solvency. The omnichain global debt pool acts as the counterparty for all zero-slippage cross-chain swaps.

$$\sum \text{Collateral balance}_a \cup \sum \text{Debt balance}_a$$

a = User 1, User 2, User 3, ...

### C-ratio

$$\frac{\text{Collateral balance}}{\text{Debt balance}} \times 100$$

### Collateral balance

$$\sum \$\text{Collateral}_{ab} + \$\text{Liquidation rewards}$$

> a = axlUSDC/USDC, eETH, rsETH, ...

> b = Arbitrum, Avalanche, BNB Chain, ...

### Debt balance

$$\sum \$\text{syASSET}_{ab}$$

> a = syAAPL, syAVAX, syBNB, ...

> b = Arbitrum, Avalanche, BNB Chain, ...

## A. Architecture

The architecture consists of multiple light chains and a main chain. The main chain exclusively hosts the aggregator contracts, enabling gas-optimized cross-chain synchronicity. This eliminates the need for any off-chain computations to save gas fees and ensures a censorship-resistant framework.

### B. Hedge pool

The hedge pool issues hedge pool tokens and swaps its deposits for the current composition of the omnichain global debt pool, ensuring delta neutrality. It rebalances these hedge pool tokens to capture the evolving composition of the omnichain global debt pool. This enables you to generate delta-neutral yield from protocol liquidations and protects you from any sharp second-party debt balance volatility.

## C. SYNTHR debt shares

Every time you mint omnichain syASSETS, you generate personal and protocol debt. The omnichain global debt pool, which represents overall protocol debt, works on the model of debt load sharing. This means that all users are collectively responsible for the protocol's solvency. Your SYNTHR debt shares correspond to your ownership of the omnichain global debt pool.

### Debt percentage

$$\frac{\text{SYNTHR debt shares}}{\sum \text{SYNTHR debt shares}_a}$$

> a = User 1, User 2, User 3, ...

### SYNTHR debt shares

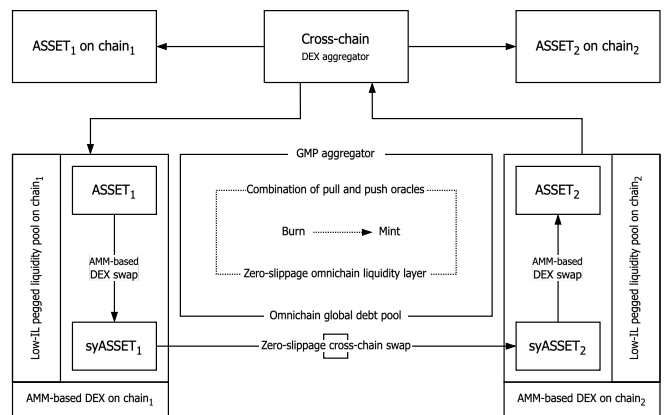
$$\frac{\text{Debt balance}}{\text{Price}_{\text{syUSD}}}$$

> Price<sub>syUSD</sub> == \$1

### Explanation

Let's consider an example to understand how SYNTHR debt shares work. Trader A mints 50,000 syUSD worth \$50,000 and receives 50,000 SYNTHR debt shares, while trader B mints 10 syBTC worth \$50,000 and receives 50,000 SYNTHR debt shares. The omnichain global debt pool, worth \$100,000, consists of 50,000 syUSD, 10 syBTC, and 100,000 SYNTHR debt shares, with traders A and B each owning 50%. Let's say that the price of syBTC doubles, which means that the 10 syBTC are now worth \$100,000 and the omnichain global debt pool is worth \$150,000, split equally between traders A and B. Trader A can exit the protocol by burning the 50,000 syUSD plus an additional 25,000 syUSD, while trader B can exit the protocol by burning 7.5 out of the 10 syBTC. Now, let's consider a different scenario in which trader A utilizes the hedge pool to ensure delta neutrality. The hedge pool swaps the syUSD for the latest composition of the omnichain global debt pool, which is 50% syBTC and 50% syUSD. Trader A, who now effectively holds 5 syBTC and 25,000 syUSD, can exit the protocol without any losses by burning them.

## 2. Omnichain syASSETS



Omnichain syASSETS enable you to seamlessly move between chains without conventional bridges. The zero-slippage omnichain liquidity layer utilizes a combination of pull and push oracles to burn and mint omnichain syASSETS across chains, enabling zero-slippage cross-chain swaps. The combination of oracles ensures price feed accuracy and reliability, while the swaps generate protocol revenue.

### Explanation

Cross-chain DEX aggregators utilize the zero-slippage omnichain liquidity layer to perform low-slippage native asset swaps.

**Step 1:** The cross-chain DEX aggregator utilizes a low-IL pegged liquidity pool on an AMM-based DEX to swap ASSET<sub>1</sub> for syASSET<sub>1</sub> on chain<sub>1</sub>.

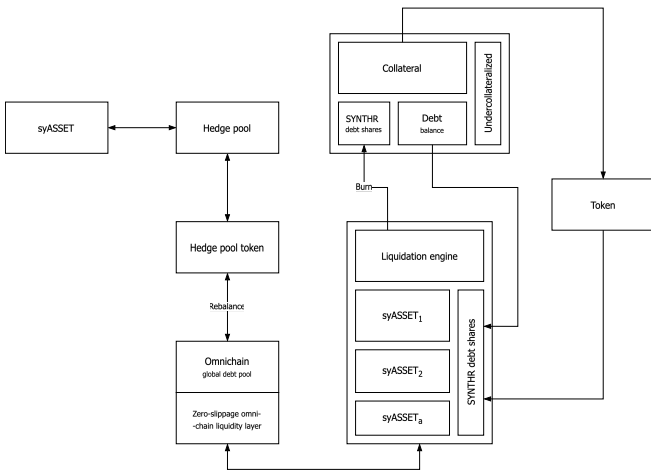
**Step 2:** The cross-chain DEX aggregator utilizes the zero-slippage liquidity layer to swap syASSET<sub>1</sub> on chain<sub>1</sub> for syASSET<sub>2</sub> on chain<sub>2</sub>.

**Step 3:** The cross-chain DEX aggregator utilizes a low-IL pegged liquidity pool on an AMM-based DEX to swap syASSET<sub>2</sub> for ASSET<sub>2</sub> on chain<sub>2</sub>.

**A. GMP aggregator**

The GMP aggregator utilizes multiple independent consensus layers to validate cross-chain messages, ensuring democratic and trustless cross-chain finality. This creates an operational barrier between the core contracts and relayers, preventing collusion between the two.

**B. Liquidation engine**



The liquidation engine liquidates undercollateralized users, preserving protocol solvency and health. It burns the SYNTHR debt shares of liquidated users and redistributes their collateral and debt balances among remaining SYNTHR debt shareholders. This requires the liquidation engine to carry out cross-chain collateral liquidations and swap them for the protocol's native token before distribution.

**Flag for liquidation**

$$C - ratio \leq C - ratio_{Liquidate}$$

**Liquidate**

$$C - ratio \leq C - ratio_{Liquidate} \wedge Time\ remaining_{Flag\ for\ liquidation} = 0$$

**Liquidation balance**

$$\frac{Debt\ balance - Collateral\ balance \times C - ratio_{Minimum}^{-1}}{1 - C - ratio_{Minimum}^{-1} \times (1 + Liquidation\ penalty_a)} \times (1 + Liquidation\ penalty_a)$$

$$Liquidation\ penalty = \begin{cases} b, & a = Self - liquidate \\ c, & a = Liquidate \end{cases}$$

>> b, and c = 10%, 20%, 30%, ...

**Self-liquidate**

$$C - ratio \leq C - ratio_{Minimum}$$

**Withdraw liquidation rewards**

$$C - ratio \geq C - ratio_{Minimum} \wedge Time\ remaining_{Escrow} = 0$$

**C. vesyUSD**

Time-lock syUSD to create vesyUSD. This particular vote-escrow model protects you from any value decay at the end of your time lock and provides you with protocol governance privileges, plus bonus airdrops from ecosystem protocols and early access to ecosystem private sales.

$$syUSD \times \frac{Time - lock}{4\ years}$$

$$> Time - lock = Timestamp_{Unlock\ date} - Timestamp$$

**Real yield**

**1. Farming rewards**

Farm LP tokens to earn farming rewards.

$$Distributable\ farming\ rewards \times \frac{Staked\ balance}{\sum Staked\ balance_a} + \alpha$$

>  $\alpha$  = Unclaimed distributable farming rewards

$$>> Stake\ LP\ token \vee Unstake\ LP\ token \xrightarrow{Assigns} \alpha$$

> a = User 1, User 2, User 3, ...

> Distributable farming rewards = Share of token emissions<sub>Per block</sub>

**2. Liquidation rewards**

The liquidation engine distributes liquidation rewards.

$$SYNTHR\ debt\ shares \times (\alpha_{\infty} - \alpha)$$

$$> \alpha_{\infty} = \alpha_{\infty-1} + \frac{Distributable\ liquidation\ rewards}{\sum SYNTHR\ debt\ shares_a} + \beta$$

>>  $\alpha$  = Infinitely increasing arbitrary variable

$$>>> Escrow \vee Burn \vee Mint \xrightarrow{Assigns} \alpha_{\infty}$$

>  $\beta$  = Unclaimed distributable liquidation rewards

$$>> Burn \vee Mint \xrightarrow{Assigns} \beta$$

> a = User 1, User 2, User 3, ...

$$> Distributable\ liquidation\ rewards = \frac{Liquidation\ balance}{Price_{Token}}$$

**3. Minting rewards**

Mint omnichain syASSETS to earn minting rewards.

$$SYNTHR\ debt\ shares \times (\alpha_{\infty} - \alpha)$$

$$> \alpha_{\infty} = \alpha_{\infty-1} + \frac{Distributable\ minting\ rewards}{\sum SYNTHR\ debt\ shares_a} + \beta$$

>>  $\alpha$  = Infinitely increasing arbitrary variable

$$>>> Mint \vee Burn \vee Withdraw\ minting\ rewards \xrightarrow{Assigns} \alpha_{\infty}$$

>  $\beta$  = Unclaimed distributable minting rewards

$$>> Mint \vee Burn \xrightarrow{Assigns} \beta$$

> a = User 1, User 2, User 3, ...

> Distributable minting rewards = Share of protocol revenue<sub>Per epoch</sub>

**4. vesyUSD rewards**

Create vesyUSD to earn vesyUSD rewards.

$$Distributable\ vesyUSD\ rewards \times \frac{vesyUSD}{\sum vesyUSD_a}$$

> a = User 1, User 2, User 3, ...

$$> Distributable\ vesyUSD\ rewards = (\alpha + \beta)_{Per\ epoch} + \gamma$$

>>  $\alpha$  = Share of token emissions

>>  $\beta$  = Share of protocol revenue

$$\gamma = \sum \text{Unclaimed distributable vesyUSD rewards}_a$$

>>> a = User 1, User 2, User 3, ...

### Revenue distribution

The protocol distributes all of its revenue to its stakeholders based on their contributions, actively fostering a strong sense of community and propagating continued engagement and collaboration among its participants. It also creates a sense of collective long-term alignment.

### Revenue generation

#### 1. Core

All zero-slippage swaps directly contribute to protocol revenue.

#### 2. SDKs

SDKs enable one-click deployments of synthetic asset protocols with custom configurations. 30% of revenues from all SDK deployments flow into SYNTHR, which are either distributed to stakeholders or utilized towards buying back and burning the protocol's native token.

### Security

The GMP and oracle aggregators ensure high levels of transaction security and mitigate front-running risks. Comprehensive insurance, periodic bug bounty programs, regular external audits by third-party firms, and rigorous internal audits intensify overall protocol security.

### SYNTH utility

#### 1. Collateral

Add the protocol's native token as collateral.

#### 2. Farming rewards

Earn farming rewards in the protocol's native token.

#### 3. Flagger fees

Earn flagger fees in the protocol's native token.

$$\frac{a}{\text{Price}_{\text{Token}}}$$

> a = \$10, \$15, \$20, ...

#### 4. Liquidation rewards

Earn liquidation rewards in the protocol's native token.

#### 5. Liquidator fees

Earn liquidator fees in the protocol's native token.

$$\frac{a}{\text{Price}_{\text{Token}}}$$

> a = \$10, \$15, \$20, ...

#### 6. vesyUSD rewards

Earn vesyUSD rewards in the protocol's native token.

### References

---

Breidenbach, C. Cachin, B. Chan, A. Coventry, S. Ellis, A. Juels, F. Koushanfar, A. Miller, B. Magauran, D. Moroz, S. Nazarov, A. Topliceanu, F. Tram'er, and F. Zhang, **Chainlink 2.0**: Next steps in the evolution of decentralized oracle networks

H. Lambur, **Uma**: A decentralized financial contract platform

Pyth Data Association, **Pyth Network**: A first-party financial oracle

S. Liu and I. Lee, Mirror Protocol

Synthetix system documentation