

# UnUniFi

KIMURA Yu, SHIMOJIMA Takeru

March 22, 2023

## Abstract

UnUniFi Protocol is an NFT-Fi Platform with DeFi tools, built via Layer1 app-specific blockchain (ASBC) in the Cosmos Network. UnUniFi provides the tools and features necessary to integrate DeFi functionality into NFT ecosystems, with cross-chain products developed for both retail and institutional users. This includes the first NFTFi Platform to create a proprietary NFT valuation algorithm calculated using real demand data, while allowing users to generate automatic DeFi yield on borrowed assets through an interchain yield aggregator. UnUniFi provides the infrastructure to be the NFT-Fi Hub for Cosmos and beyond. Our mission is to “ Give all NFTs the opportunity to DeFi ” .

## 1 Introduction

### 1.1 Problems faced by NFTs and NFTFi

Many regard NFTs - digital blockchain assets that can sometimes depict or be paired to real-world objects - with healthy skepticism. Liquidity and demand generation issues prevent participants from easily unlocking the intrinsic value of their assets. Institutional users are penalized when looking to diversify into NFT markets at scale. NFT markets suffer from concentrated and easily manipulated liquidity pools. Current pricing models generally fail to consider value changes based on rare NFT assets within a particular collection. Additionally, many current NFTFi solutions are limited in scope, cost-prohibitive, and often create unfavorable lending terms for both platform users and platform lenders.

### 1.2 The rise of NFT + Decentralized Finance (DeFi)

DeFi is short for “ Decentralized Finance ” , a generalized umbrella term used to describe a variety of cryptocurrency and blockchain-based financial applications geared towards disrupting legacy financial institutions. DeFi is not reliant on central financial intermediaries such as brokerages, exchanges, or banks to offer users access to traditional financial instruments, and instead utilizes smart contract-based applications that are built on top of existing blockchain-based platforms. In regard to DeFi, there is a growing movement to make finance more efficient with use-cases not only limited to exchanging tokens, but also realizing existing financial products (such as corporate stocks) on the blockchain. NFT-Fi (NFT  $\times$  DeFi) technology seeks to provide DeFi utility to NFTs, increasing tradability and capital efficiency to establish legitimacy for NFTs, regarding them as a true asset class or store of value. “ NFTs

are undoubtedly becoming a gateway into the DeFi space for mainstream audiences,” said Lauren Stephanian, a principal at Pantera Capital.

### 1.3 Our advantage over other NFTFi solutions

As a technology in its early adoption phase, NFT-Fi needs to be easily accessible, easy to understand, flexible, and scalable to encourage continued adoption. While other NFTFi platforms employ a peer-to-peer or liquidity-pool lending model, UnUniFi combines real demand for the NFT with its intrinsic demand; this is used to facilitate quick liquidity aggregation which gives NFT holders faster and more flexible access to lending. Our technology is scalable for institutional users and can be implemented externally on platforms as an NFT valuation oracle. UnUniFi aims to be a DApps platform with NFT price information at its core.

## 2 NFT collateral lending

### 2.1 Collateral Liquidation Deposit Auction

#### 2.1.1 Definitions

$i \in I$  index of bids

$n = |I|$  number of bids

$\{d_i\}_{i \in I}$  the deposit amounts of  $i$  th bid

$\{r_i\}_{i \in I}$  the interest rates of  $i$  th bid. To unannualize this, an expression `unannualize(r; start, end)` will be used.

$\{x_i\}_{i \in I}$  the expiration dates of  $i$  th bid

$t$  current time

$q$  the average of  $p_i$  and the upper bound of  $s$

$s_d$  the sum of deposits and the amount which lister can borrow with NFT as collateral

$j \in J_i$  index of borrowing from  $i$  th bid deposit

$\{b_{i,j}\}_{i \in I, j \in J_i}$  means the amount of  $j$  th borrowing borrowed from  $i$  th bid deposit

$\{b_i\}_{i \in I}$  means the amount borrowed from  $i$  th bid deposit

$s_b$  the sum of  $b_i$

$\{t_{i,j}\}_{i \in I}$  the start dates of  $j$  th borrowing from  $i$  th bid deposit

$i_p(j)$  the index of the  $j$  th highest price bid

$i_d(j)$  the index of the  $j$  th highest deposit amount bid

$i_r(j)$  the index of the  $j$  th lowest interest rate bid

$i_t(j)$  the index of the  $j$  th farthest expiration date bid

$c$  minimum deposit rate

$$q = \frac{1}{n} \sum_{i \in I} p_i$$

$$s_d = \sum_{i \in I} d_i$$

$$b_i = \sum_{j \in J_i} b_{i,j}$$

$$s_b = \sum_{i \in I} b_i$$

### 2.1.2 Acceptance of the bid

The lister can accept the bid if the favorable bid price is found.

If this process is executed,  $i_p(1)$  th bidder will have the right to settle the trade. To settle the trade,  $i_p(1)$  th bidder have to pay

$$p_{i_p(1)} - d_{i_p(1)}$$

If  $i_p(1)$  th bidder proceeds, he or she will obtain possession of the listed NFT.

The lister will receive

$$p_{i_p(1)} - \sum_{i \in I} \left( d_i + \sum_{j \in J_i} \text{unannualize}(r_i; t_{i,j}, t) b_{i,j} \right)$$

Protocol fees are abbreviated.

Each  $i \in I$  th bidders will receive

$$d_i + \sum_{j \in J_i} \text{unannualize}(r_i; t_{i,j}, t) b_{i,j}$$

If  $i_p(1)$  th bidder decide not to proceed, his or her deposit  $d_{i_p(1)}$  will be forfeited.

### 2.1.3 State transition

When  $(p_{\text{new}}, d_{\text{new}}, r_{\text{new}}, t_{\text{new}})$  will be added to the set of bids, the new bids sequence will be

$$I' = I \cup \{n+1\}$$

$$n' = n + 1$$

$$\{p'_i\}_{i \in I'} = \{p_i\}_{i \in I} \cup \{p_{\text{new}}\}$$

$$\{d'_i\}_{i \in I'} = \{d_i\}_{i \in I} \cup \{d_{\text{new}}\}$$

$$\{r'_i\}_{i \in I'} = \{r_i\}_{i \in I} \cup \{r_{\text{new}}\}$$

$$\{x'_i\}_{i \in I'} = \{x_i\}_{i \in I} \cup \{x_{\text{new}}\}$$

$$q' = \frac{1}{n'} \sum_{i \in I'} p'_i$$

$$s'_d = \sum_{i \in I'} d'_i$$

where the prime means the next state.

The constraint of  $d_{\text{new}}$  is

$$cp_{\text{new}} \leq d_{\text{new}} \leq \min \{p_{\text{new}}, q' - s_d\}$$

In easy expression, it means

$$\begin{aligned} cp_{\text{new}} &\leq d_{\text{new}} \\ d_{\text{new}} &\leq p_{\text{new}} \\ s'_d &= s_d + d_{\text{new}} \leq q' \end{aligned}$$

The first inequation means that the deposit of the new bid must be greater than or equal to minimum deposit amount, that is defined by multiplying  $c$ , the minimum deposit rate, to  $p'_{n+1}$ , the bid price. The second inequation guarantee that  $p_i - d_i \geq 0$ . The third inequation means that  $s'$ , the sum of deposits, must be lower than or equal to  $q'$ , the average of bid prices. To increase the deposit amount of the new bid and increase the probability of getting settlement when the liquidation of collateral occurs, the new bidder must increase  $p_{\text{new}}$  to increase  $q'$ .

When the NFT lister newly borrow assets,  $a_i$  must follow the constraint

$$b_i \leq d_i$$

Additionally the following inequation will be satisfied

$$s_b \leq s_d$$

Deposited amount will therefore be consumed (used for lending resource) in ascending order of interest rates, so the following constraint must be satisfied.

$$b_{i_r(j+1)} = 0 \text{ if } b_{i_r(j)} < d_{i_r(j)}$$

When the  $i_r(k)$  th bid expire,  $x_{i_r(k)} = t$ , there are two distinct paths to select,

First, if the lister configured the automatic refunding when initiating the listing,  $a_{i_r(k)}$  the borrowed amount from  $i_r(k)$  th bid will be automatically repaid. The lister then will automatically borrow the same amount from the deposit whose interest rate is the next lowest.

$$\begin{aligned} b_{i_r(l)} &= d_{i_r(l)} && \forall l \in \{1 : m - 1\} \\ b_{i_r(m)} &= \sum_{l=1}^m (d_{i_r(l)} - b_{i_r(l)}) - \sum_{j \in J_{i_r(k)}} (1 + \text{unannualize}(r_{i_r(k)}; t_{i_r(k),j}, x_{i_r(k)})) b_{i_r(k),j} \end{aligned}$$

where  $m$  is defined as

$$m = \begin{cases} \min & m \in \mathbb{N} \\ \text{subject to} & \sum_{j \in J_{i_r(k)}} (1 + \text{unannualize}(r_{i_r(k)}; t_{i_r(k),j}, x_{i_r(k)})) b_{i_r(k),j} \leq \sum_{l=1}^m d_{i_r(l)} - b_{i_r(l)} \end{cases}$$

If the sum of the deposits of remaining bids are insufficient to cover the refund amount, this process cannot execute and instead liquidation process will be executed.

Second, if the lister didn't configure the automatic refunding, the liquidation process will be executed.

#### 2.1.4 Liquidation process

for  $k \in \{1 : n\}$

if  $p_{i_d(k)} < s$ , continue.

$i_d(k)$  th bidder will have the right of settlement of the deal. To settle the deal,  $i_d(k)$  th bidder have to pay

$$p_{i_d(k)} - d_{i_d(k)}$$

If they do so, then they will have the possession of the listed NFT.

If they don't, then their deposit  $d_{i_d(k)}$  will be forfeited.

Even if all bidders decide not to exercise their settlement right for the deal, the protocol will completely forfeit

$$\sum_{k \in \{1:n\}} d_{i_d(k)} = \sum_{i \in I} d_i = s_d$$

Therefore, there is no possibility of the protocol accruing any losses caused by the liquidation process. If  $i_d(k^*)$  th bidder settles the deal,

$$p_{i_d(k^*)} + \sum_{l=1}^{k^*} d_{i_d(l)}$$

will be temporally sent to the protocol.

If  $\sum_{l=k^*}^n \left( d_{i_d(l)} + \sum_{j \in J_{i_d(l)}} \text{unannualize}(r_{i_d(l)}; t_{i_d(l),j}, t) b_{i_d(l),j} \right) \leq p_{i_d(k^*)} + \sum_{l=1}^{k^*} d_{i_d(l)}$   
each  $i \in \{i_d(l) \mid l \in \{k^* : n\}\}$  th bidder will receive

$$d_i + \sum_{j \in J_i} \text{unannualize}(r_i, t_{i,j}, t) b_{i,j}$$

and protocol will receive

$$p_{i_d(k^*)} + \sum_{l=1}^{k^*} d_{i_d(l)} - \sum_{l=k^*}^n \left( d_{i_d(l)} + \sum_{j \in J_{i_d(l)}} \text{unannualize}(r_{i_d(l)}; t_{i_d(l),j}, t) b_{i_d(l),j} \right)$$

else if  $\sum_{l=k^*}^n \left( d_{i_d(l)} + \sum_{j \in J_{i_d(l)}} \text{unannualize}(r_{i_d(l)}; t_{i_d(l),j}, t) b_{i_d(l),j} \right) > p_{i_d(k^*)} + \sum_{l=1}^{k^*} d_{i_d(l)}$   
each  $i \in \{i_d(l) \mid l \in \{k^* : n\}\}$  th bidder will receive

$$\frac{d_i + \sum_{j \in J_i} \text{unannualize}(r_i; t_{i,j}, t) b_{i,j}}{\sum_{l=k^*}^n \left( d_{i_d(l)} + \sum_{j \in J_{i_d(l)}} \text{unannualize}(r_{i_d(l)}; t_{i_d(l),j}, t) b_{i_d(l),j} \right)} \left( p_{i_d(k^*)} + \sum_{l=1}^{k^*} d_{i_d(l)} \right)$$

## 3 Interchain Yield Aggregator

Describing Auto DeFi Yield with our “ Interchain Yield Aggregator ” Easily utilize assets for yield farming through an Interchain Yield Aggregator directly integrated into UnUniFi Protocol. Yield Farming strategy availability can be extended by deploying an adapter made with CosmWasm via third party communities. Interchain support ensures a variety of high yield strategies are made available to platform users at their selection. Borrowers can leverage their NFT assets for yield farming by using borrowed funds. Liquidity providers can also experience the same benefits by receiving yield on any assets deposited. Auto DeFi Yield ensures capital efficiency, so that tokenized assets deposited into our protocol maintain competitive returns, even when not actively being used in the platform.

### 3.1 The flow of Interchain Yield Aggregator

1. Deposit tokens into the IYA module for UnUniFi Protocol.
2. Specify a strategy based on the information provided by the data provider and select contract.
3. Start investing by sending TX from the contract to the other chain through an interchain account.

### 3.2 The process of receiving rewards

1. Users claim rewards in the Interchain Yiled Aggregator module for UnUniFi Protocol.
2. The Interchain Yield Aggregator receives rewards on the contract the user selected.
3. The contract receives from the chain in which has invested.
4. The contract sends rewards from the other chain to UnUniFi Protocol.
5. Interchain Yield Aggregator sends tokens to the user.

## 4 Derivatives

### 4.1 Counterpart Liquidity Pool

#### 4.1.1 Definitions

$I$  the set of the index of assets in the pool

$\{p_i\}_{i \in I}$  the price of  $i$  th asset in the pool, retrieved from oracle

$\{l_i\}_{i \in I}$  the amount of the liquidity of  $i$  th asset in the pool

$\{w_i^*\}_{i \in I}$  the target weight of the liquidity of  $i$  th asset in the pool

$\{l_i^*\}_{i \in I}$  the target amount of the liquidity of  $i$  th asset in the pool

$p$  the price of the liquidity provider token

$s$  the supply of the liquidity provider token

$\Delta$  the set of derivatives

$\Pi_\delta$  the set of positions of a derivative  $\delta \in \Delta$

$\chi(\pi)$  the counterpart position of the position  $\pi$   
 $r(\pi)$  the imaginary funding rate of the position  $\pi$   
 $s(\pi)$  the total size of the position  $\pi$

$l_i^*$  can be calculated with the given values  $w_i^*, s, p, \{p_i\}_{i=1}^n$  by  $w_i^* = \frac{l_i^* p_i}{s p}$  as

$$l_i^* = \frac{w_i^* s p}{p_i}$$

The price of the liquidity provider token can be calculated as:

$$p = \begin{cases} \frac{1}{s} \sum_{i \in I} l_i p_i & \text{if } s > 0 \\ \sum_{i \in I} w_i^* p_i & \text{if } s = 0 \end{cases}$$

The minting and redemption of the liquidity provider token is executed with this price  $p$ .

#### 4.1.2 Imaginary funding rate

In a Central Limit Order Book (CLOB) model, index prices are retrieved via oracles and mark prices are determined in the CLOB. Funding rates ensure the difference between the mark price and the index price is close to zero.

On the other hand, in the Counterpart Liquidity Pool model, mark prices are equivalent to index prices, and both are retrieved via oracles. The Counterpart Liquidity Pool will take the counterpart position of a trader's order, so there is no time required for matchmaking. However, in this model, if traders acquire positions, the pool and liquidity providers simultaneously incur losses. To tackle this problem, an imaginary funding rate exists. If the net position of traders leans to one side, the imaginary funding rate works to make the net position of traders neutral. The neutral net position of traders means a neutral position for both the pool and the liquidity providers. In the perspective of economics, it can be expressed that this model unifies the conventional funding rate and the time cost of waiting for matchmaking to the imaginary funding rate.

#### 4.1.3 Perpetual futures

PF  $\in \Delta$  serves perpetual futures.

Positions are

$$\Pi_{\text{PF}} = \{\text{long}(i, p), \text{short}(i, p)\}$$

where  $i$  is the index of assets and  $p$  is the position price.

The relations of counterpart position are

$$\begin{aligned} \chi(\text{long}(i)) &= \text{short}(i) \\ \chi(\text{short}(i)) &= \text{long}(i) \end{aligned}$$

$$\forall p \in \mathbb{R}_+, r(\text{long}(i, p)) \propto \int_{\mathbb{R}_+} (s(\text{long}(i, q)) - s(\text{short}(i, q))) dq$$

Notice that  $r(\pi) = -r(\chi(\pi))$ .

#### 4.1.4 Perpetual options

$\text{PO} \in \Delta$  serves perpetual options.

Positions are

$$\Pi_{\text{PO}} = \{\text{call\_long}(i, p), \text{call\_short}(i, p), \text{put\_long}(i, p), \text{put\_short}(i, p)\}$$

where  $i$  is the index of assets and  $p$  is the strike price.

The relations of counterpart position are

$$\begin{aligned}\chi(\text{call\_long}(i, p)) &= \text{call\_short}(i, p) \\ \chi(\text{call\_short}(i, p)) &= \text{call\_long}(i, p) \\ \chi(\text{put\_long}(i, p)) &= \text{put\_short}(i, p) \\ \chi(\text{put\_short}(i, p)) &= \text{put\_long}(i, p)\end{aligned}$$

## 5 Ecosystem Incentive

We recently analyzed the state of various Crypto and DeFi ecosystems and their applications, and asked the question, are dApps truly decentralized? If the viability of a DeFi protocol is dependent on their frontend, how can their dApp claim to satisfy decentralization concerns? Our research indicated that the current state of dApps is that many are not truly decentralized and users must often rely on a single frontend deployment. In this scenario, reliance on a single frontend can create a single point of failure, and overall weakness in the reliability of a protocol.

### 5.1 Single Point of Failure

Generally, users rely on the established Frontend to interact with a specific dApp or Contract. In a real-world example, if a developer understands how a dApp operates, say Opensea for example, they can call the contract directly without needing to use the Frontend. Meanwhile, regular users must rely on the Opensea Frontend in order to interact with its blockchain components. If the Frontend for Opensea stopped working, most users would not know how to interact with the contract. Having one frontend creates a single point of failure. So how do we avoid this? Our solution proposes the ability to create multiple independent frontends, which in turn embodies a true “decentralized” model (eliminating the single point of failure).

### 5.2 Our Solution- Truly Decentralized Frontend(s)

Through our own protocol development, we considered these exact same questions, ultimately allowing us to acknowledge the limitations of current solutions in hopes to create a better decentralized ecosystem. This reflection directed us to create an ecosystem where our API, client library, Bubble App plugin, and frontend incentive module can be utilized by anyone; using a platform like Bubble allows anyone to



create unique applications with their no-code app development tools. Through hyper-accessible frontend deployment, we can be the first ecosystem to successfully decentralize our frontend without a single point of failure.

### 5.3 Ecosystem Development via Community Inclusion

Normally, only experienced developers can contribute to ecosystem expansion when building dApps. However, with our Decentralized Frontend, we can expand the ecosystem to include non-developers. This contribution to community inclusion inherently furthers our ecosystem by allowing anyone to build onto it. A Decentralized Frontend will also simultaneously inspire both localization and mass adoption by allowing the creation of a Frontend that is hyper-focused for each region and category of usage. We believe that the “Decentralized Frontend” is an effective method to expand our ecosystem. In other terms, creating a system that operates like an infrastructure layer for NFT-Fi, allowing our platform to easily integrate into pre-existing NFT ecosystems, marketplaces, and similar use-cases.

### 5.4 How do we Incentivize Frontend Development?

It is one thing to create an environment where our frontend can specifically be distributed and decentralized based on user needs, but how do we encourage these boutique integrations or unique Frontends? Our ecosystem incentive modules aim to reward early adopters who help to expand and realize our decentralized ecosystem. Our latest implementation of this incentive model is a system that distributes rewards to a frontend creator based on the NFT trading fee (excluding gas fee) collected when using that specific creator's frontend. For example, if someone lists an NFT on A's frontend and that NFT is sold for \$10,000 then A will receive \$250 as a reward (assuming trading fee rate = 5% and NFT market frontend reward rate = 5%.) Therefore, reward distribution is perfectly proportional to the actual trading volume. In theory, this module will provide a competitive incentive for the parties which bring value to our ecosystem — motivating Frontend service creators (developers). Some of the various applications for custom Frontends can include NFT markets for real estate NFTs; art collections; financial instruments; PFPs; GameFi, and more... These unique Frontend service creators can then become candidates to receive the Ecosystem Incentive Reward, generated from the various fees accumulated.

## 6 GUU token distribution

Usage	supply[%]	Vesting
Ecosystem Development	30%	Vesting term depending on the situation. Minimum 1-yr linear vesting
Assignment for validators	15%	1-yr full locked, linear 12 months
Assignment for UnUniFi team	15%	1-yr full locked, linear 36 months 1-yr full locked, linear 60 months
Assignment for UnUniFi Development Fund	5%	1-yr full locked, linear 36 months 1-yr full locked, linear 60 months
Marketing	14%	VC(1-yr full locked, linear 36 months)
Treasury	10%	Vesting term depending on the situation. Minimum 1-yr linear vesting
Advisor	1%	linear 6 months from the time of grant
Assignment for business partners	10%	1-yr full locked, linear 24 months

## 7 Tokens specifications

Name	Symbol	Denom in blockchain
GUU	GUU	uguu

## 8 Governance specifications

Minimum deposit for voting	1,000,000uguu=1GUU
----------------------------	--------------------

## 9 Staking specifications

Max validators	100
Bonding denom	uguu

## 10 About Us

The UnUniFi Protocol is owned by the team which is a leading technologist company of Cosmos blockchain technologies.

## 11 Legal Disclaimer

Please be aware of and accept the following risks before using GUU. CAUCHYE ASIA PTE. LTD. shall not be liable for any loss or damage arising out of or in connection with any of the following risks.

### 11.1 Risk of Losing GUU due to Loss of Private Key

The private key itself or a combination of private key shall be necessary for the disposal of the User's GUU, and the management of the private key shall be managed under the User's own authority and responsibility. The loss of the private keys associated with the wallet in which the user's GUU is stored is the same as the loss of the GUU itself. Phishing attacks against you or the GUU on your device may result in loss of GUU due to malware attacks, DoS attacks, consensus-based attacks, or any other form of attack.

### 11.2 Risks Related to the UnUniFi Protocol

Since GUU is based on the UnUniFi Protocol any malfunction, failure or failure of the UnUniFi Protocol may have a material adverse effect on GUU and may render GUU temporarily unusable.

### 11.3 Risk of mining attacks

GUU, like other distributed cryptographic tokens based on public chain protocols, may be subject to mining attacks during the verification of token transactions on the blockchain. These attacks may pose a risk to the recording of transactions related to GUU.

### 11.4 Changes in Laws and Regulations and Taxation Risk

There may be future changes in laws, government ordinances, statutes, regulations, orders, notices, guidelines, or other regulations or taxation systems related to GUU. You are responsible for making your own decisions regarding the taxation of the GUU.

### 11.5 Risks Due to Input Errors and Other Factors by User

There is a risk of unintended transaction results due to input errors or any other actions by the User, failure, malfunction or operational status of the User's or a third party's communication or system equipment, natural disasters, cyber attacks or any other causes.

### 11.6 Relationship between Users

Any transactions, communications, disputes, etc., arising between users and other users or third parties in relation to the Company's website shall be the responsibility of the users.

## 12 Contributions

We have already made some little contributions to the Cosmos ecosystems.  
<https://github.com/cosmos-client/cosmos-client-ts>

## 13 Contact

To contact us on the UnUniFi topic, please create an issue ticket in GitHub.  
<https://github.com/UnUniFi>