Final Report
Dec 2020

Author: Memento Blockchain Pte Ltd

Table of Contents

Table of Contents 2

Executive Summary 3

Overview of the experiment 4

Our Proposed Solution 4
- Initial assessment 4
- Critical revision of the three solutions 5
  - Solution 1 5
- Summary of the custody solution 6
- The Dutch auction 8

Solution Architecture 9

System UI 11

The front-end is a web-based app that can run on any browser, allowing the system to be platform independent. 11
- Fund Manager UI 12
- Investor UI 12

User Interface Rebalancing Workflow 13
- Initial fund configuration 13
  - Stakeholder: fund manager 13
- Investor mints the fixed-proportion fund tokens 15
  - Stakeholder: investor 15
- Managed fund creation 17
  - Stakeholder: fund manager 17
- Investor mints the managed fund tokens 18
  - Stakeholder: investor 18
- Second fund configuration 20
  - Stakeholder: fund manager 20
- Propose the Dutch Auction 21
  - Stakeholder: fund manager 21
- Investor checks the Dutch auction 22
  - Stakeholder: investor 22
- Start the Dutch Auction 23
  - Stakeholder: fund manager 23
- Participate in the Dutch Auction 24
  - Stakeholder: market makers 24

End the Dutch Auction 25
Stakeholder: fund manager 25
Investor checks the rebalanced fund 26
Stakeholder: investor 26
Limitations and future works 26

Executive Summary

Digital assets such as security tokens continue to disrupt the market and there is growing interest in the traditional financial industry to experiment in this space. Today, digital assets are everywhere we look. Every single currency, asset, real estate, and even arts or wines has the potential for tokenization. As such, the term digital asset will continue to

encompass a growing number of items. For now, tokenization appears to be the path towards the future.

The digitalization of value — digital assets — is a growing trend and in this process, existing financial assets, or real assets are digitized on the blockchain with the advantage of removing intermediaries in their distribution, lowering transactional cost and also addressing the issue of counterparty risk.

The world of digital assets — value which is represented by tokens issued on a public blockchain that can then be distributed in the same way that other cryptocurrencies are — creates more opportunity in this increasingly digital world. More and more, traditional intermediaries and new entrants into the financial services industry are realizing that the potential of blockchain, its widespread distribution of financial products, is the benefit of liquidity.

Liquidity is very crucial in financial markets; the lack of it is very costly and very problematic for both the issuers and the holders of the financial assets, the blockchain technology presents another way to distribute and get your product out there to those that want it.

Effectively enabling and making distribution of financial products fairer, cheaper, more cost effective and therefore, enhancing the liquidity profile of what typically has been very illiquid assets.
Our mission is to transform the traditional asset management and lower the barrier to entry for capital and talents. Our technology stack can be used to create, launch and invest in funds with any tokenized digital assets and can be customized to the needs of large financial institutions.
With this Proof of concept we want to demonstrate that traditional asset management can be converted into digital using the blockchain, making the industry more efficient, more compliant and accessible to everyone. While there are some limitations in the existing technology the path and the speed of innovation will overcome any issues that arise in the financial industry.

Overview of the experiment

Our Proposed Solution

Initial assessment

In this POC we want to investigate how it is possible to create a new type of fund tokens that give the possibility to the fund manager to periodically change the allocation of the

component tokens. We define the *rebalancing problem* as following: *given that a number of component tokens are stored in a smart-custodian contract, how can the fund manager change the current fund allocation without having the possibility to take full, or partial, control of the component tokens themselves?*
In the rebalancing problem there is a sub-problem that stands out right away. Whichever way we solve it, since the cost of creating any token is marginal, the fund manager could create a fake token, of which he completely controls the supply, and rebalance 100% of the fund into this token. While this is a real problem we have devised a very simple solution: we limit the tokens that can take part in a rebalancing event. This can be accomplished, for example, by restricting the universe of tokens to a white list of vetted tokens.
In this POC we explored three different possibilities to solve the rebalancing problem:

1. Using centralized exchanges. Possibly give control to the fund manager to trade the component tokens without having himself the possibility to take the tokens out of the exchange account
2. Using decentralized exchanges. The fund manager can sign trade transactions that are sent to decentralized exchanges. However the tokens are taken and sent back to directly to the custodian smart contract
3. By introducing a third-party type of users: the arbitrageurs, or market makers. These market makers would help the rebalancing by purchasing and, respectively, selling directly to the smart custodian the component tokens that are in surplus and, respectively, in demand. These users would make a small profit for each transaction, however ensuring that the component token custody stays with the smart custodian.

Note that in all cases the portfolio has some extra components that should be sold in exchange of the new components. Therefore any solution should allow us to exchange fund components

Critical revision of the three solutions

We have determined that the best way going forward was to use solution 3, as we are about to explain.

Solution 1

In the proposed solution 1 the fund manager would follow the following procedure:

- take the extra components out from the smart custodian into his own wallet;
- send these components to a centralized exchange;
- on the centralized exchange, swap the extra components for the new components;
- take out the new components from the centralized exchange;
- send the new components to the smart custodian so that the fund has the new allocation

This solution has the great advantage that when the swap of the extra components for the new ones is done on a centralized exchange, it can be done fastly, with minimal slippage and transaction fees. This is because centralized exchanges have high liquidity. On the other hand this solution has a number of issues. The first one is that the whole

procedure is very time consuming, as taking tokens in and out from centralized exchanges can take a long time (hours, or even days). The second problem is that the fund manager needs permission to take out components from the fund itself. This poses a security threat and totally renders the digital custody solution useless. For these reasons we discarded this solution.

Solution 2

In this proposed solution, the fund manager sends the extra components directly from the smart custodian to the decentralized exchange. At first this solution seems to be the most natural as there is no need to take out the extra component from the smart custodian. However there are at least two technical issues. The first problem is that, at the time of writing (namely December 2020), decentralized exchanges have very little liquidity. This means that if the fund has a reasonable size, the exchange will not have enough tokens to perform the swap. The other reason is that, currently, decentralized exchanges have very high slippages and it is not unheard of that people pay up to 3% for a single transaction. For these technical reasons we have decided to discard solution 2.

Solution 3

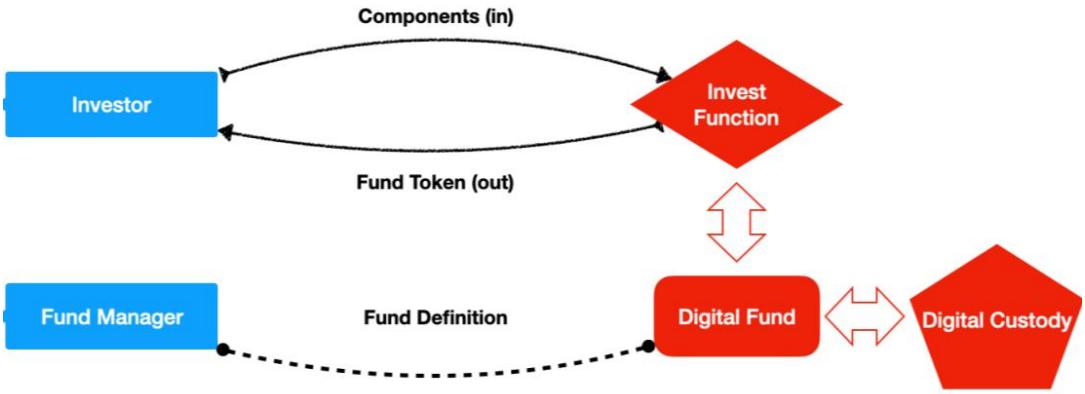
We think that solution 3, while technically harder to implement, allows the fund manager to exchange the extra components for the new ones with reasonable costs and without giving the fund manager a total control of the underlying components. In this solution there is a rebalance function in the smart custodian that allows the exchange of the extra components for the new ones. A new category of users, the market makers, can take advantage of this function to make a small-arbitrage profit in exchange for the burden to help with the portfolio rebalance. More details on this solution are given in later paragraphs.

Summary of the custody solution

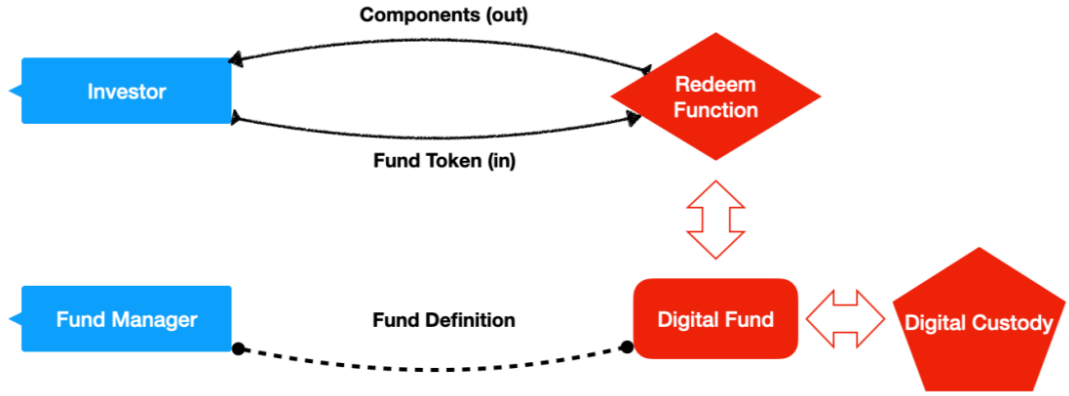
In order to understand how *Solution 3* may work, we first need to look at the results obtained by our previous POC. Indeed we have shown how the creation of smart-custody contracts allowed us to generate digital funds with a fixed proportion of constituent components. We summarize here that solution because it already contains the seeds of how to proceed to create a managed fund. There are three main operations that can be performed on the fixed-allocation digital funds:

1. Creation. The fund manager can create a fund by specifying what are the exact proportion of the component tokens
2. Investment. An investor can send to the fund components in the exact proportion specified by the fund manager to obtain a corresponding number of fund tokens
3. Redemption. An investor that owns a certain number of tokens of a specific fund can send them back to the smart contract to obtain the component tokens in the proportion specified by the fund manager

The following chart outlines the fund creation and the fund investment process

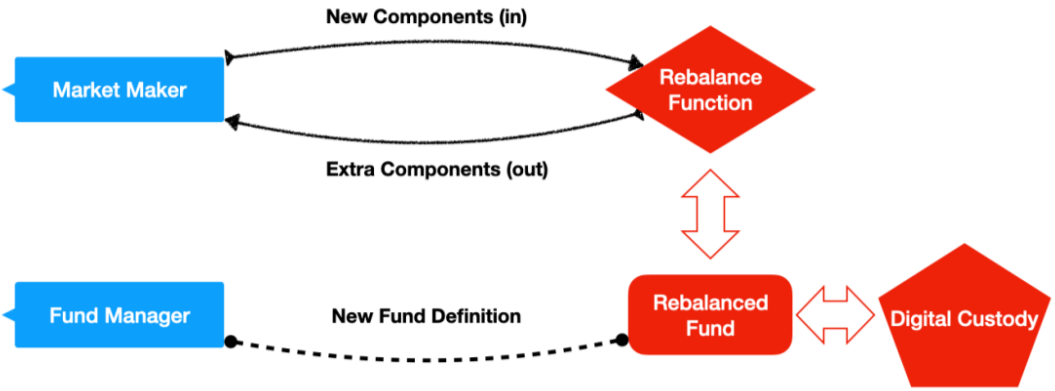


In the chart above we show with blue squares the fund manager and the investor, similarly we denote in red the blockchain components. The dashed line indicates that the digital fund was created by the fund manager. Then we show that the investor can receive a fund token by providing the components in the proportions specified by the fund manager.
The converse operation, i.e. obtaining the fund components by providing the fund token is described in the chart below.



For completeness we show again that the fund was created according to the exact proportion of the components, as specified by the fund manager. As already described in the final report of our previous POC, we have already successfully implemented the functionalities described above.

As seen in the two examples above it is possible to create a smart-contract function (technically a *method*) both to upload and download components in predetermined proportions to the Digital Custody. We can combine these two behaviours into a single smart-contract function, namely the *Rebalance Function*, that takes in the new components and *spits out* the extra components. In the following chart show the schematic of this function:



In order to make this trick work we need a new type of user: the market maker. When the fund manager provides a new fund definition the system computes what are the new components from the difference between what the new-fund components and what is already in the digital custody. Similarly the system computes what the extra components are as the tokens that the fund manager is ready to get rid of.
Typically the value of the extra components is a few basis points higher than the value of the new components so that the market makers are incentivized to perform the exchange. The operation of rebalancing is usually limited to a short time, typically of the order of a few hours.

The Dutch auction

In creating our solution we have found another problem: from the time when the fund manager proposes a new allocation to the time it is executed the market value of the exchange components might change.
As a consequence either of two cases might happen:

1. The component exchange is very convenient for the market makers. In this case it will happen almost instantly, however at a high cost for the fund investors
2. The component exchange is not convenient at all for the market makers. In this case it will not happen.

In order to eliminate this problem we propose that the rebalancing, i.e. the exchange of components, is allowed using a Dutch auction. In this Dutch auction, there is a limited time in which the new components can be exchanged for the old ones. For example this time could be 10 minutes or 1 hour. The price at which the old components are sold in favor of the new ones starts high (above market price) and ends low (below market price) at the end of the auction.
In this way there is an optimal time between the start and the end of the auction in which the price of the old component is both low enough, so that the market makers can purchase them, but not too low so that the fund investors do not lose a lot of money.

Solution Architecture

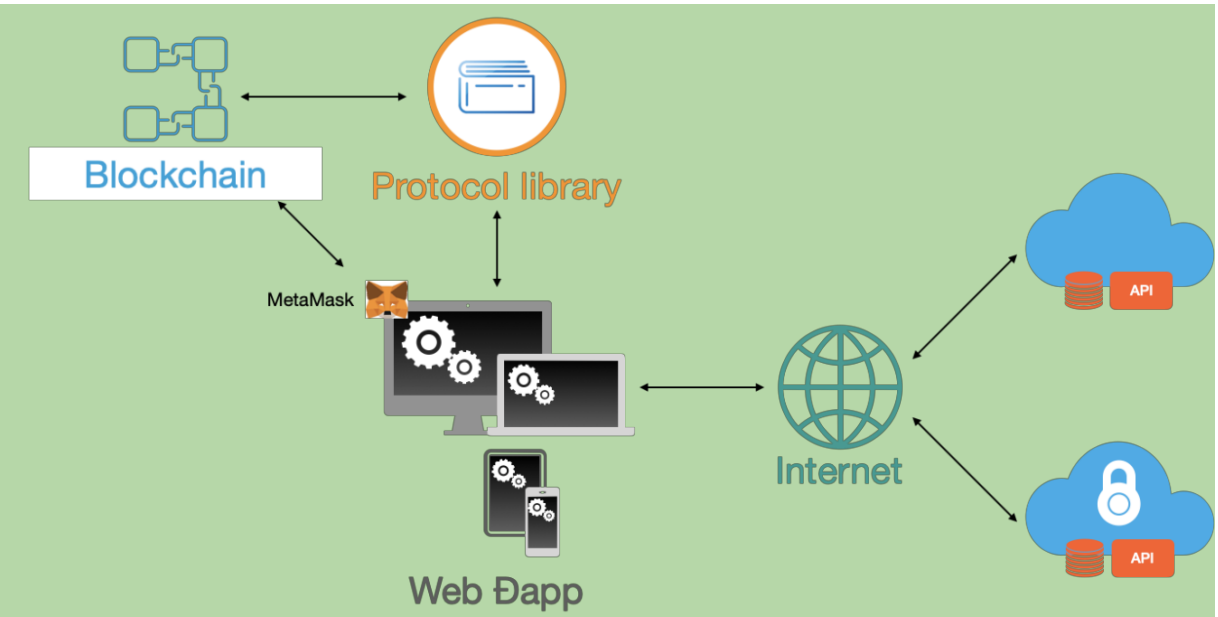
For the front-end/back-end we used cloud service to achieve:

- Agility: the IT team does not bother about server provisioning and maintenance and can concentrate on the core system development
- Elasticity: enjoy instant scale up/down in accordance with the needs of the business, reducing costs and improving the ability to meet the users' demands
- Velocity: easily deploy the application in multiple physical locations within minutes, allowing to provide a lower latency and better experience to the users

We are committed to achieve the following key features:

- Modular
- Robust
- Secure
- Measurable
- Debuggable
- Maintainable
- Reusable
- Extensible
- Lightweight

The layered (n-tier) high level architecture is depicted in the following picture:

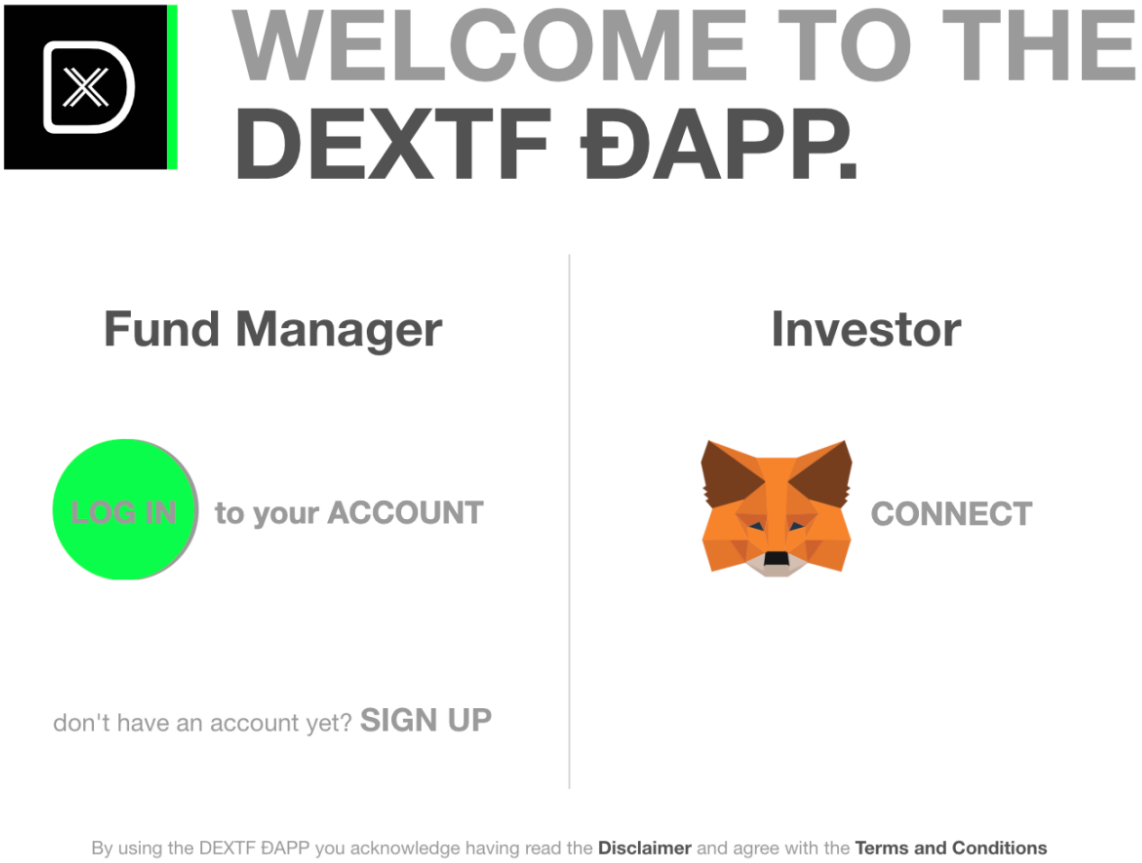


High level architecture

The key parts are:

Protocol library (based on the Javascript SDKs)
A set of JS modules to manage the interactions with the blockchain (creation of a fund, issuance of a fund, etc.)

Web Đapp
The front-end is a web-based interface, developed using Vue.JS (progressive framework for building user interfaces), Semantic UI (development framework that helps create responsive layouts) and Axios (promise based HTTP client for the browser and Node, allowing to send asynchronous HTTP requests to REST endpoints and perform CRUD operations)



Cloud services

Amazon Web Services (AWS) services like Cognito (secure user sign-up, sign-in, and access control), DynamoDB (no-sql database), Lambda Functions (server-side business logic), API Gateway (RESTful APIs) and Step Functions (serverless workflows)

System UI

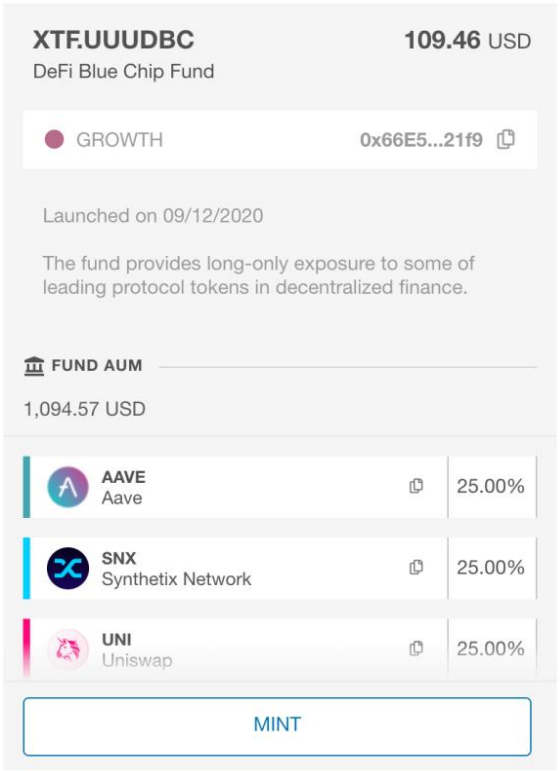
The front-end is a web-based app that can run on any browser, allowing the system to be platform independent.

We invested a good amount of time in designing the User Experience (UX) so that the actions of the various users (fund managers and investors) are as simple as possible.

We implemented the following pages:

- Welcome page for the user to login or register to the service
- Main landing page, called the “Command Centre”, with 2 main areas:
    - “Manage Funds”, for to Fund Managers to manage their funds
    - “Invest/Redeem Funds”, for the Investors to invest into the publicly available funds

Each fund is represented by a card with relevant information like name, issuing company, description, composition, etc…



Fund Manager UI

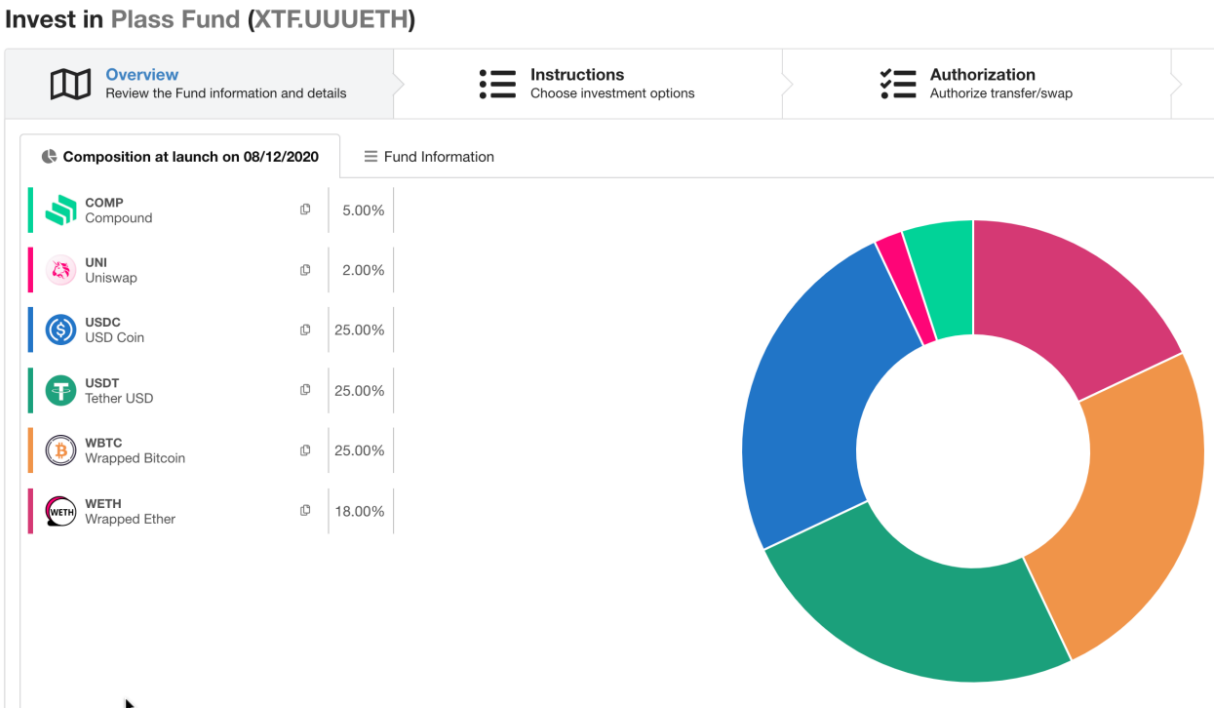
The “Manage Funds” section shows all the funds that the Fund Manager created and manages. A step-by-step wizard will guide the Fund Manager through the creation of a new fund.

Once the Fund Manager is ready to launch the newly created fund he can press a Launch button, that will trigger a call to a specific function of the Protocol library and the creation of the Smart Contract representing the new fund.

Investor UI

The “Invest/Redeem Funds” section shows all the publicly available funds. The Investor can perform two operations: Invest in a fund and Redeem fund.

An Invest button starts a step-by-step wizard to drive the user through the process to invest into a fund. When the Investor confirms the operation a call to a specific function of the Protocol library is triggered; this will start the blockchain transaction to complete the investment.



The reverse operation (redemption of the fund) is initiated with a Redeem button, that starts a step-by-step wizard to guide the user through the process to redeem a fund. When the Investor confirms the operation a call to a specific function of the Protocol library is triggered; this will start the blockchain transaction to complete the redemption.

User Interface Rebalancing Workflow

We consider here a complete example of rebalancing starting from a given initial and ending with another final allocation. While in the reported example given here, in order to simplify the discussion, we only use two assets, in a production environment any number of assets can be used. Also each of the screenshots shown are part of an actual working software solution that does connect to a real blockchain. Hence all the examples are actually working and could be, in principle, used for real-life trading.

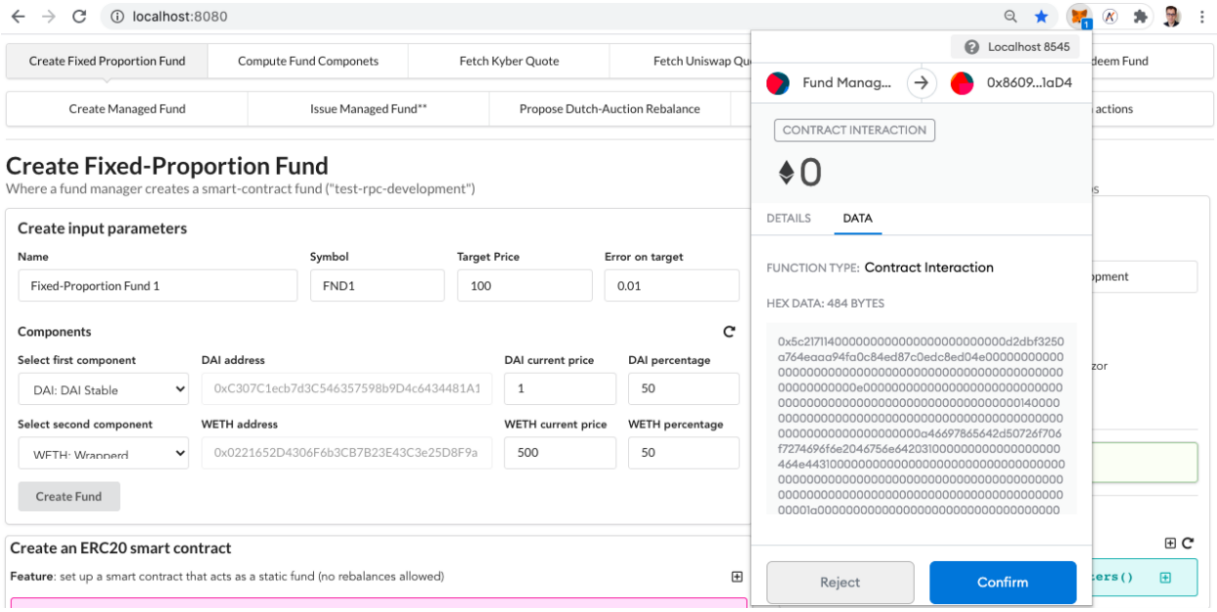
Initial fund configuration

Stakeholder: fund manager

We consider a fund manager that wants to create a fund, which we will call FND1, with with only two tokens, namely the stable coin DAI and the ERC20 representation of ether, WETH

- 50% DAI, assuming a current price of 1.00
- 50% WETH, assuming a current price of 500.00

The fund manager then will start by simply creating a fixed-proportion fund FND1 using the appropriate tab and sign the transaction.



In the above screenshot we show the act of signing the transaction to create the FND1 fund, using the popular plugin MetaMask. After the transaction has been approved by the blockchain we receive the fund address: 0x751f05AeA6D13Baee179777ADB1CA5af11aeAAce

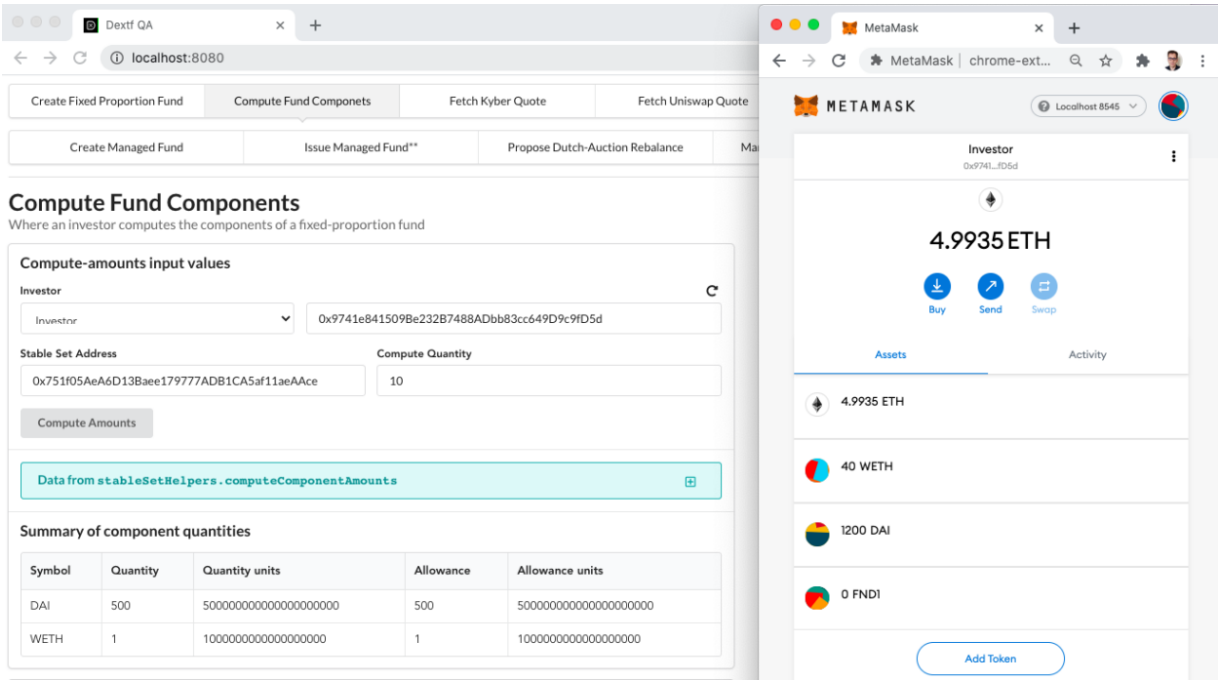
Investor mints the fixed-proportion fund tokens

Stakeholder: investor

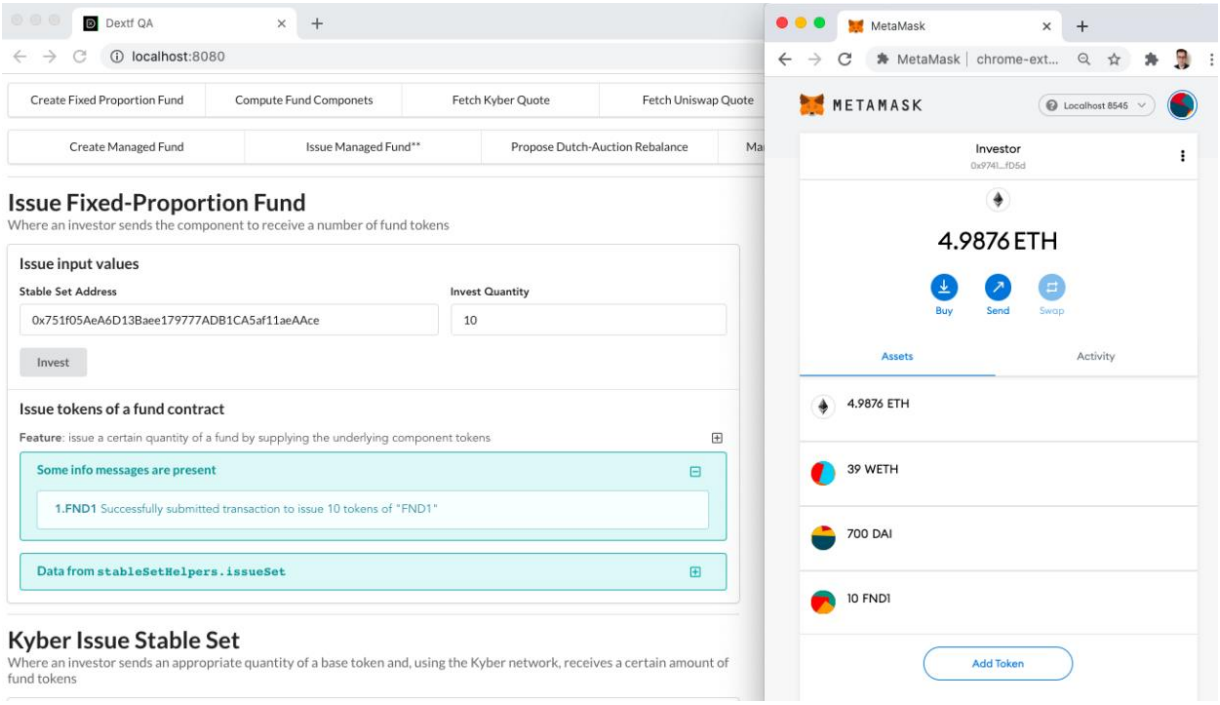
An investor wants to purchase 10 tokens of FND1. In order to mint the fund he needs to have

- 500 DAI (current price 1)
- 1 WETH (current price 500)

The investor can check that these are indeed the components to issue 10 fund tokens by querying the blockchain in the appropriate tab. Notice how he initially has 0 tokens of FND1 in his portfolio, however he has 40 WETH and 1200 DAI.



He can then proceed to approve and mint 10 tokens of SWA.FND1, by trading in 500 DAI and 1 WETH



In the above screenshot the investor mints 10 FND1 tokens. His token holdings are then automatically updated in the MetaMask window.

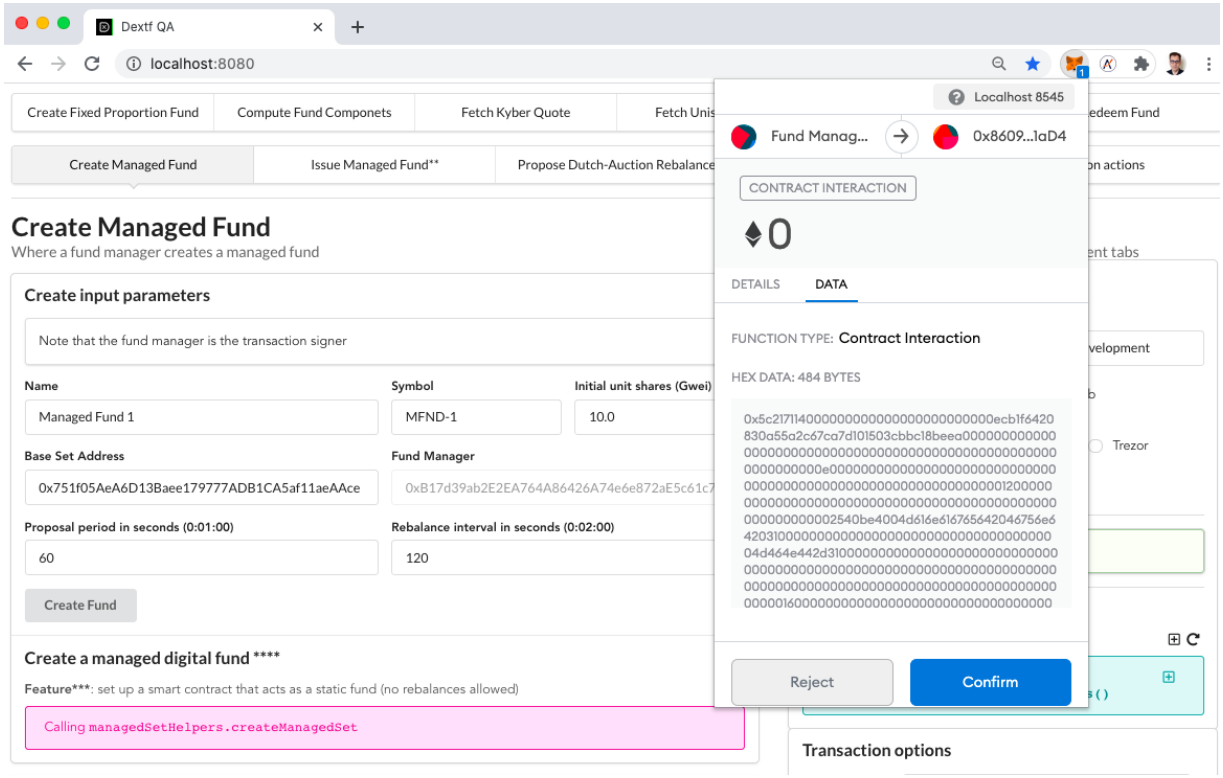
Managed fund creation

Stakeholder: fund manager

As the fund manager wants to be able to change the allocation of his fund, he creates a wrapper, i.e. another tradable fund, around the initial fixed-proportion fund FND1.
In other words, the fund manager creates a wrapper token, called MFND-1, with the following properties:

- There is only 1 component: FND1
- The component is held in proportion 1 to 1
- In the fund creation already decides some of the rebalancing parameters

The managed fund MFND-1 is now in the Default state

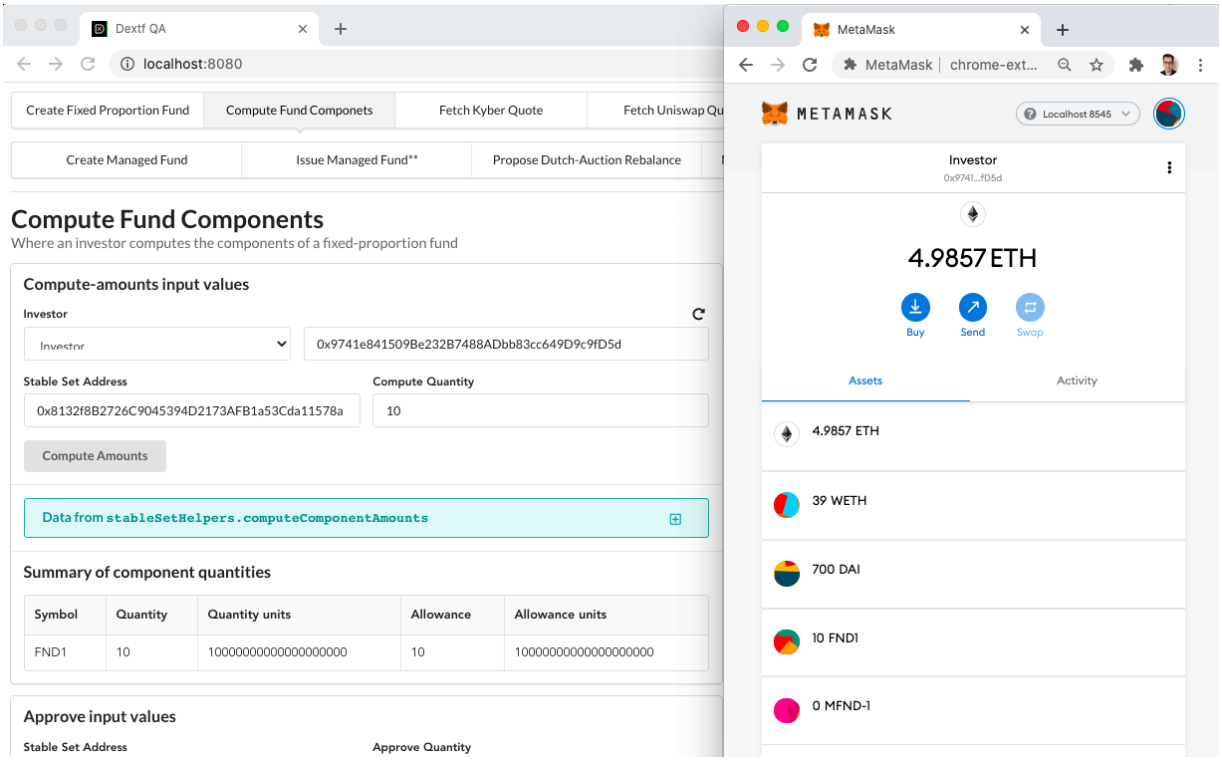


The above screenshot shows the transaction used to create the managed fund called MFND-1. After the transaction has been approved by the blockchain we receive the fund address: 0x8132f8B2726C9045394D2173AFB1a53Cda11578a

Investor mints the managed fund tokens

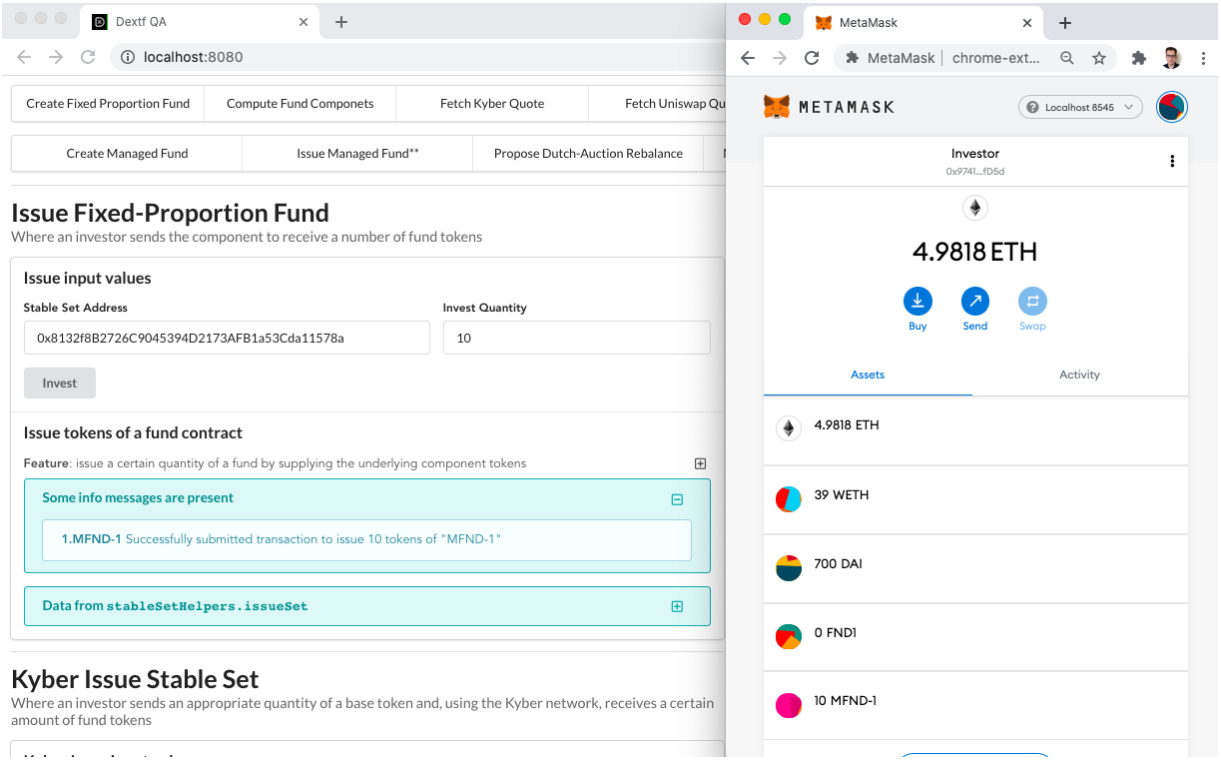
Stakeholder: investor

Since the managed fund was created with MFND-1 in proportion 1-to-1 with respect to FND1, the investor can now trade his 10 tokens of FND1 for 10 MFND-1 tokens.



In the above screenshots we can see the components of MFND-1. Also we notice that the investor, at the moment, has 10 FND1 and 0 MFND-1 .

The investor can now simply mint 10 MFND-1 tokens by trading in 10 FND1 tokens:

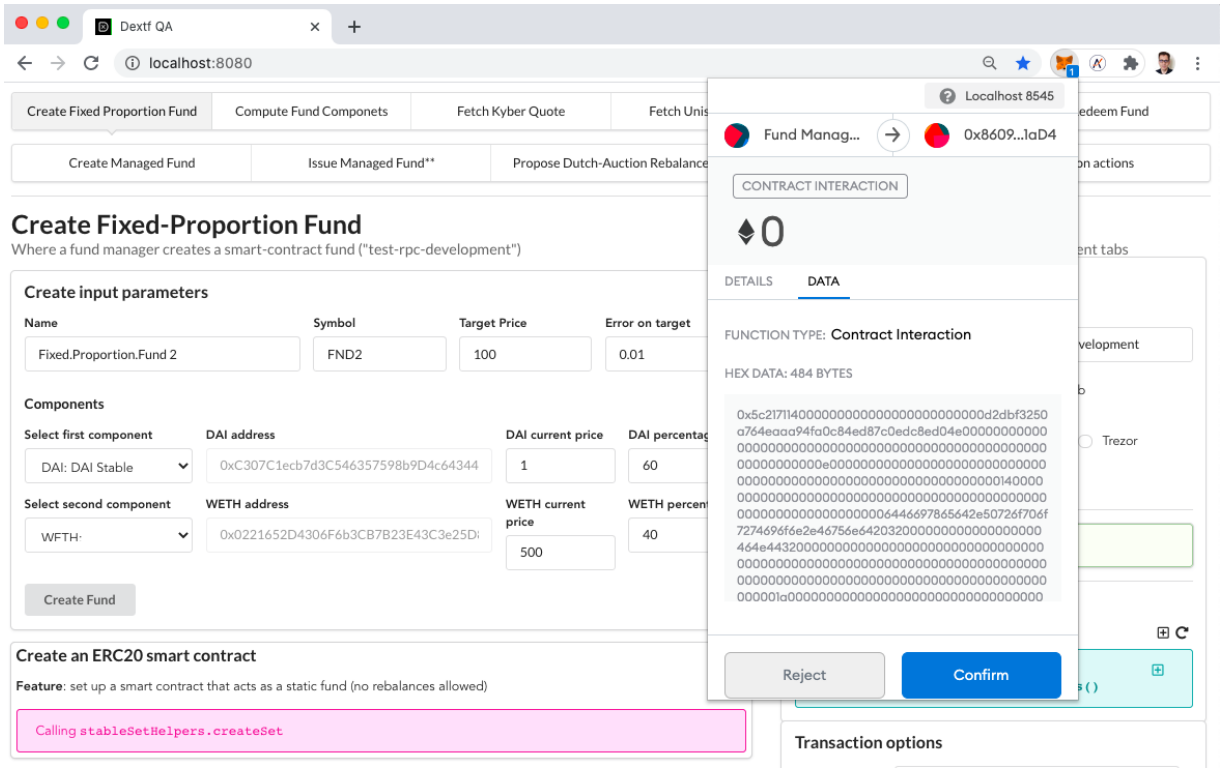


Second fund configuration

Stakeholder: fund manager

At this point the fund manager wants to rebalance the fund into a new allocation. In order to perform this operation he creates a new fixed-proportion fund, which we will call FND2, with the following allocation:

- 60% DAI, assuming a current price of 1.00
- 40% WETH, assuming a current price of 500.00

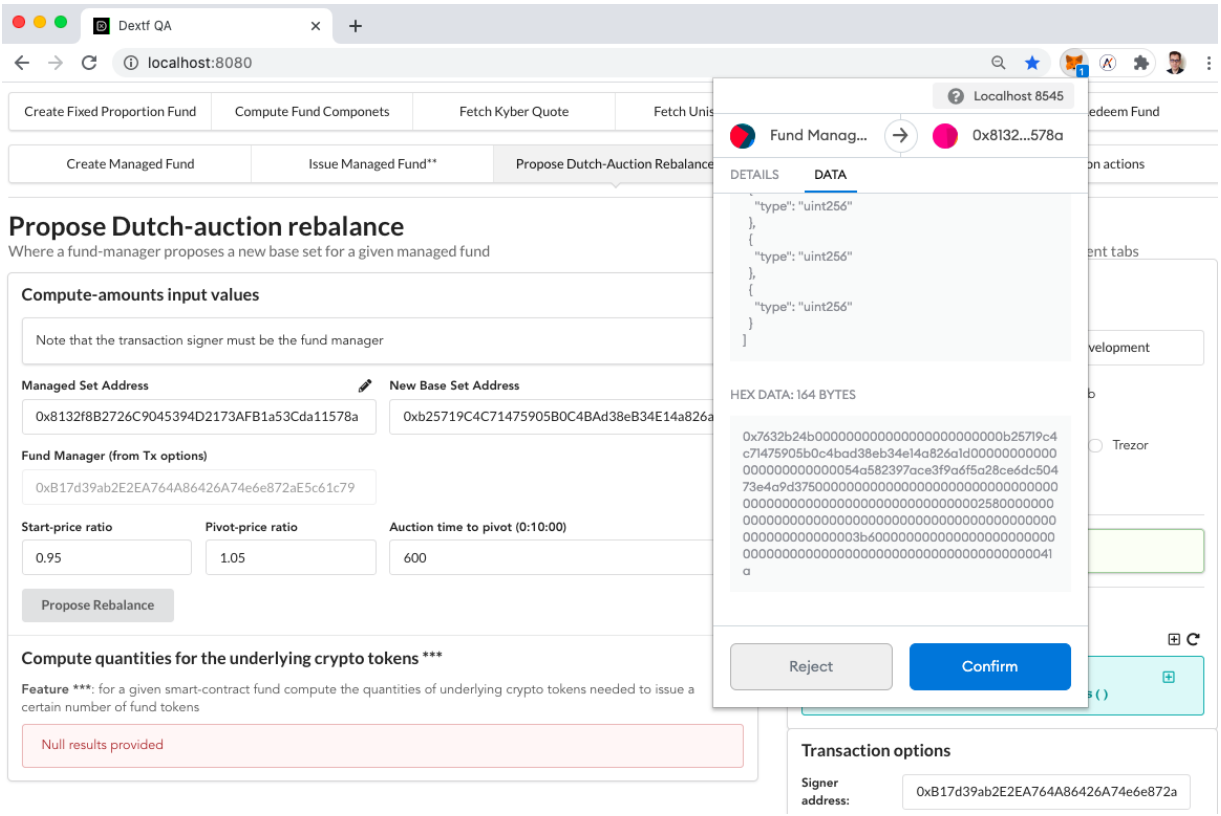


After the transaction has been approved by the blockchain we receive the fund address: 0xb25719C4C71475905B0C4BAd38eB34E14a826a1d

Propose the Dutch Auction

Stakeholder: fund manager

In order to have the market makers help in changing the fund allocation from FND1 to FND2, the fund manager proposes a Dutch auction.

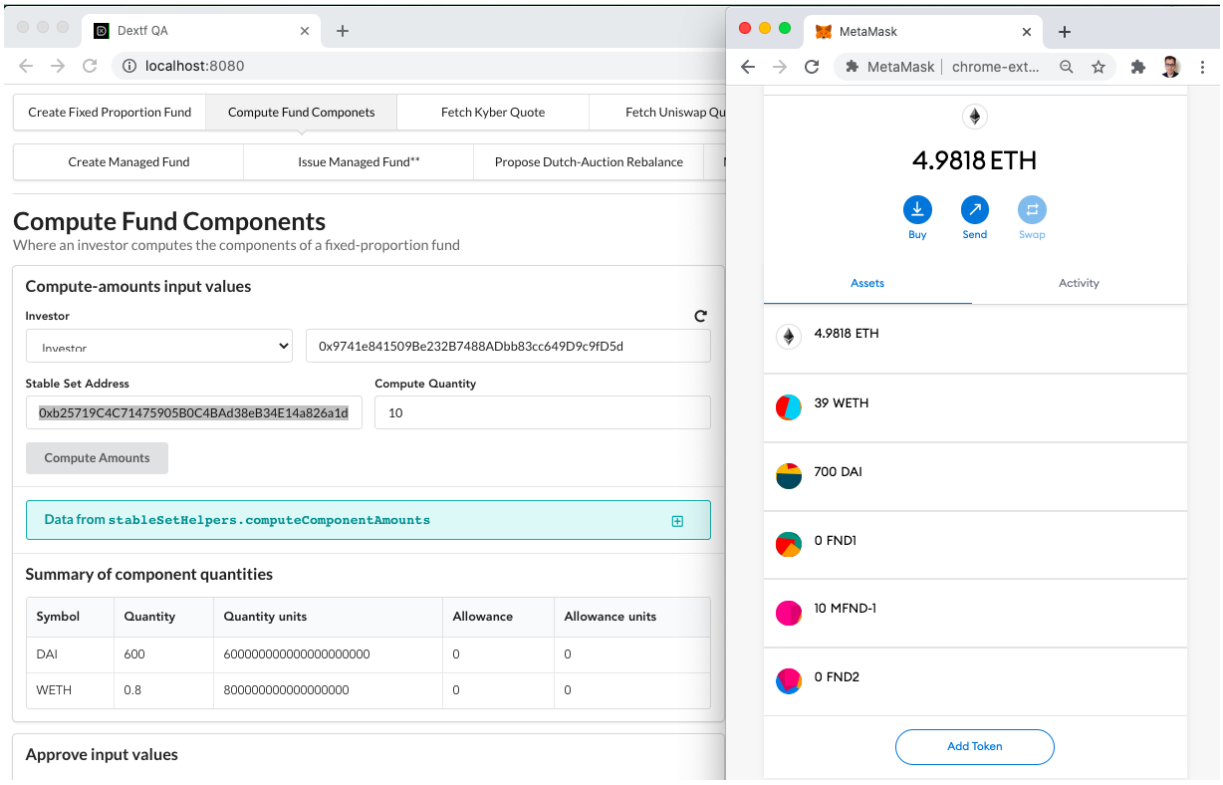


The above screenshot shows the confirmation of the transaction that proposes the Dutch auction. Note that the auction will run from 5% above market price to 5% below market price in 10 minutes.
Note that when the Dutch auction is proposed, the smart contract checkers that all new components are white listed (for brevity we do not show the user interface for handling of the white list).
After the smart auction is confirmed by the blockchain the managed fund MFND-1 turns to the Proposal state

Investor checks the Dutch auction

Stakeholder: investor

Since the fund is now in the Proposal state, the investor can check what the proposed new fund allocation is.

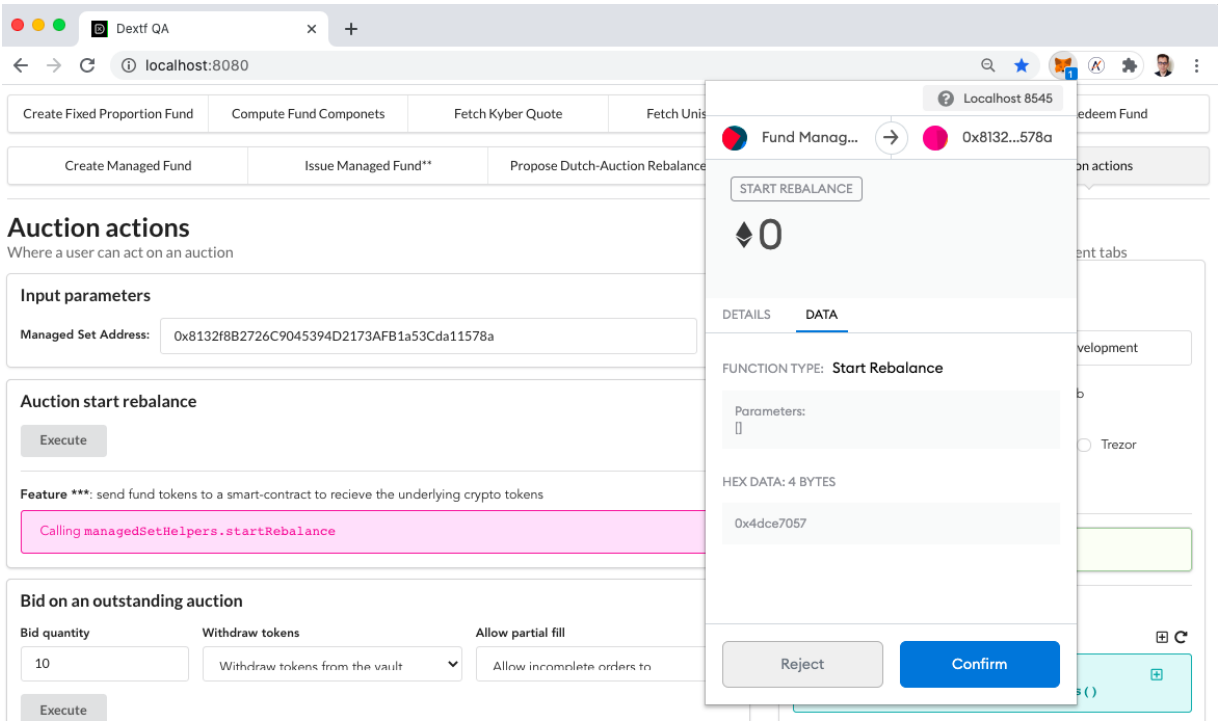


At this point, if the investor is confident with the new allocation he simply waits for the market makers to perform the exchange. If the investor is not happy with the new allocation he can simply redeem the FND1 from MFND-1 and keep the current allocation.
Note that the investor *does not* have to mint FND2 tokens.

Start the Dutch Auction

Stakeholder: fund manager

After the proposal period, as specified in the managed fund smart contract, has passed the fund manager, or anybody else with access to the blockchain, can officially start the Dutch auction by sending the appropriate transaction to the blockchain.



After the start-auction transaction has been mined by the blockchain, the managed fund MFND-1 enters the Rebalance state. During the Rebalance state there cannot be any issue or redemption of the MFND-1 fund.

Participate in the Dutch Auction

Stakeholder: market makers

The market maker observes the ratio between SWA.FND1 and SWA.FND2 and decides when to step in.

With ratio 0.95 we have, for 10 tokens

- Out SWA.FND1 i.e. 500 DAI & 1 WETH
- In SWA.FND2/0.95 i.e. 600/0.95 DAI & 0.8/0.95 WETH (632 DAI & 0.842 WETH)

Since only the in/out balance is actually exchanged

- In 132 DAI and out 0.158 WETH; MM receives 1 WETH for 835 DAI sent

With ratio 1.05 we have, for 10 tokens

- Out SWA.FND1 i.e. 500 DAI & 1 WETH
- In SWA.FND2/1.05 i.e. 600/1.05 DAI & 0.8/1.05 WETH (571 DAI & 0.762 WETH)

Since only the in/out balance is actually exchanged

- In 71 DAI and out 0.238 WETH; MM receives 1 WETH for 298 DAI sent

Starting with a ratio of 0.95 and ending with a ratio of 1.05, the market makers are able to exchange

- At the beginning of the auction receives 1 WETH for 835 DAI sent
- At the end of the auction receives 1 WETH for 298 DAI sent

There will be an intermediate time in which a market maker can make money, but not so much to be front run by another market maker

In our little demo we assume that the Dutch auctions ends fairly and that the exchange of tokens happens when the ratio is exactly 1.0.

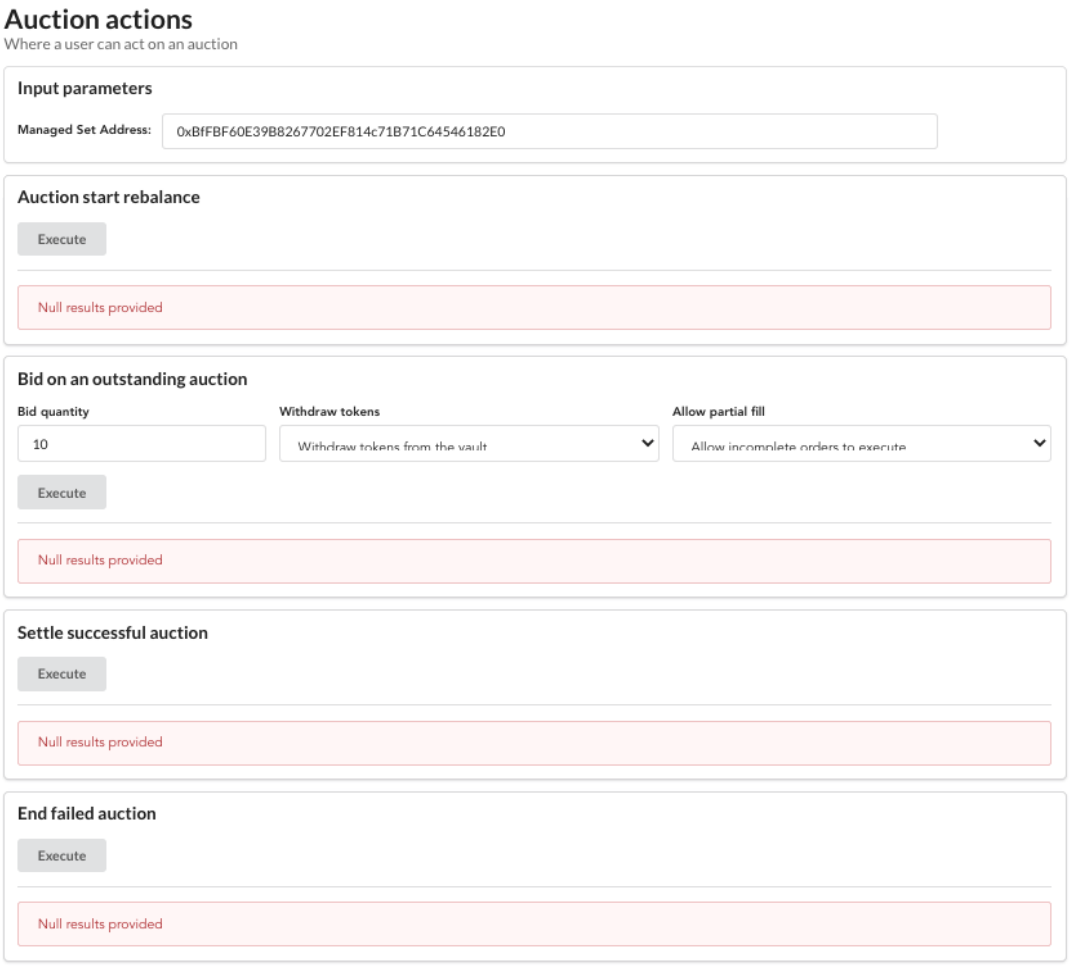
End the Dutch Auction

Stakeholder: fund manager

After the auction time has elapsed, or when all tokens have been exchanged, the fund manager has the following possibilities:

- Declare the auction a success and set the managed fund into the Default state. The component is now SWA.FND2
- Declare the auction a complete failure and set the managed fund into the Default state. The component is still SWA.FND1
- Declare the partial failure and set the managed fund into the Drawdown state. There is not component and the investors can redeem the portion of their funds

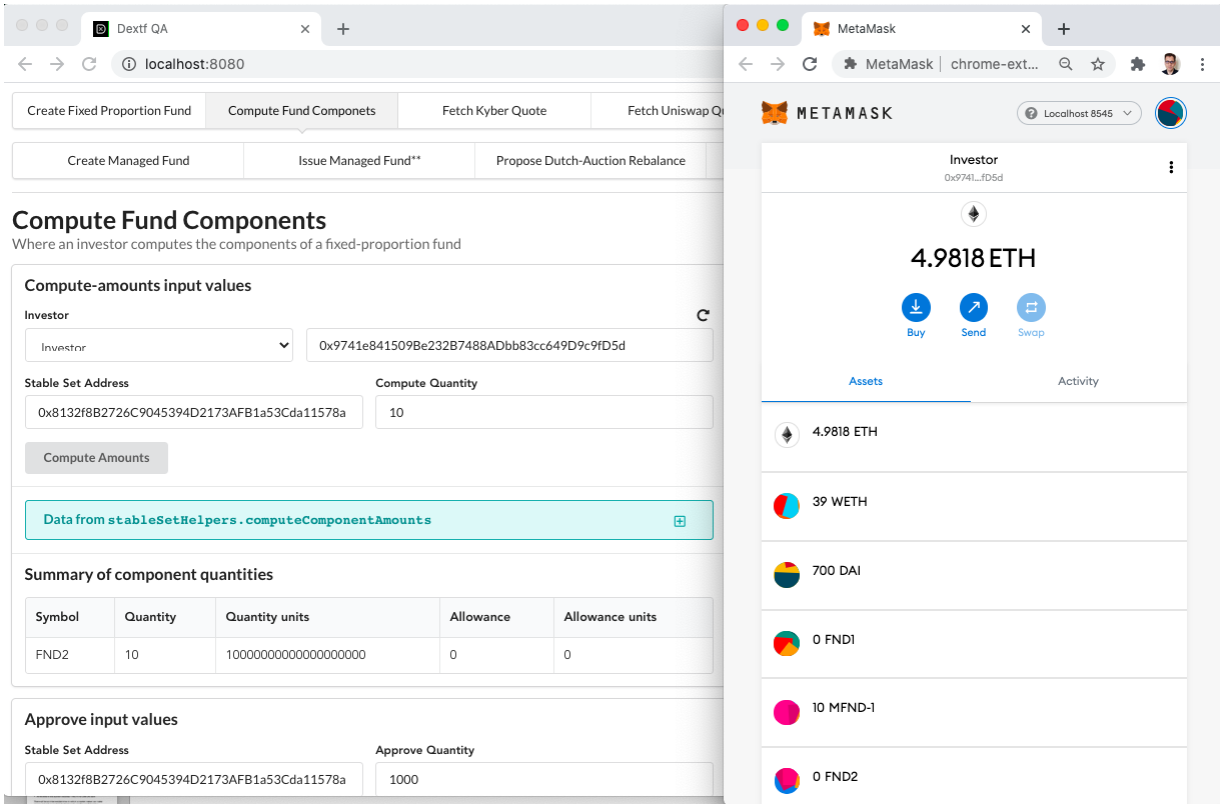
Here we show a screenshot of the possible auction actions related to a given managed fund.



Investor checks the rebalanced fund

Stakeholder: investor

Assuming that the fund manager settled the auction successfully, the managed fund has now a new single component: FND2



The screenshot above shows the new composition of MFND-1 which now has FND2 instead of FND1. Notice that, in principle, since the value of FND1 and FND2 might not be the same, the proportion of MFND-1 tokens to FND2 tokens might be different from 1.

Limitations and future works

The structure of smart contracts described in this work allows the creation of funds where the fund manager defines the asset allocation and under no circumstances he will have control of the funds' tokens, whether they are representing the newly minted digital funds or the tokens representing the balance of each investor custodized by the Smart Custody contract. As a result, the completion of this POC enabled us to introduce and build a working solution, if compared to the complex financial products issued in the market today. With the current state of the protocol, we are able not only to issue passive investments based on short-term strategies but also active managed funds which require periodic rebalancing of the underlying components. Furthermore, we recognize that in order to realize the ideal but achievable goal of distinguishing Singapore as a trendsetter in the digital asset management industry, we need to extensively develop on top of the inefficiencies that corrode investors' value and stifle asset managers' performances within the traditional AM framework. Our plan into 2021 and beyond is to work with market participants to understand their needs to accelerate the innovation and contribute to the creation of a regulatory framework that can help financial institutions to enter the digital asset management space in a controlled and compliant way.

While this proof of concept has been developed on the Ethereum blockchain there are some limitations that might prevent this platform to be adopted as an end-all solution especially for enterprise adoption. The primary problem with Ethereum is scalability. Transactions are still very slow, as Ethereum's public blockchain can only process roughly 20 transactions per second (TPS) compared to other newer blockchains. Enterprises require a very high throughput of transactions, and Ethereum can not yet offer that on its public mainnet as such we are exploring other blockchains like Algorand that are more efficient and built for the financial industry. The Algorand blockchain is quite unique in terms of security and ease of use and thanks to its atomic transfers allows for secure and immediate transaction settlement for multiparty transactions directly built on Layer 1.

Going forward we will also work to involve market makers and exchanges (decentralized and non) for our rebalancing function to get access to best execution (best rates and minimal slippage).

We are extremely thankful for being awarded the MAS FSTI POC grant, and it has enabled us to complete this first backbone infrastructure that prepares us for our next challenge. We are also grateful for the feedback from the evaluation panel, who has taken time out to review the proposal for this proof of concept.