NVO

DECENTRALIZED
EXCHANGE

Table of Contents

1.1 Abstract

The key function of a decentralized cryptocurrencies exchange is cryptocurrencies and assets are not held by any third party. They are directly transferred from one owner to another in a P2P model. NVO decentralized exchange is operated through two components. The first one is a multi-cryptocurrencies open source wallet. All keys are created locally on user's device and never sent to any server.

Users control their private keys, can send and receive funds, export wallet, and encrypt the wallet with a password or seed phrase. Developers can develop and add new assets through an open plugins system keeping the wallet lightweight.

The second component is a validator hosted on the Safenetwork. Users will connect to the Safenetwork through the wallet to exchange. An application will match the orders issued by the users and check all transactions and order informations are valid. Unlike centralized and semi-centralized exchanges, there isn't a wallet acting as a relay. Transactions are processed simultaneously through an order matching engine from one wallet to another. The validator on the Safenetwork will check if the transactions are valid prior to being exchanged. The orders are then matched and transactions broadcasted in their respective networks. The validator process raw transactions and doesn't hold any private keys or assets nor can it sign transactions resulting in a trustless decentralized exchange using the Safenetwork to validate orders.

1.2 Goals

Main objectives are :

- Building a secure product that end users will love, and use easily.
- Viable and steady development workflow with transparent release model.
- Storage and exchange of cryptocurrencies and tokenized assets
- Utilizing functionalities from other projects in a dynamic and modular decentralized environment
- Automating all modules and payments to establish an autonomous exchange vehicle.

1.3 Requirements

A set of requirements will be provided to enable the most suitable environment for a steady development curve and release.

The modules required by NVO rely on the functions and features of each other. They have their own set of requirements and are of equal importance.

To bring a reliable environment, the API Clusters will be required first as they are a key element of synchronisation for the whole system either for the wallet or the exchange. At some points, the plugins may also require the API Clusters.

1.4 Language

NodeJS will be used to develop the project and it’s modules. The common point of all the cryptocurrencies is NodeJS, a dynamic language because of the RPC interfaces provided by all the cryptocurrencies. It is a versatile language and can be used for different platforms and functionalities from a wallet to a load balanced server cluster.

The API Clusters required for NVO will be made using NodeJS to enable quick deployment, development, and provides a high level customisation regarding the resources consumption and management of each node.

The validator will take advantage of the versatility of NodeJS. It is one of the main component that will let SafeNetwork applications work with different cryptocurrencies.

1.5 Safenetwork

A working proof of concept can be created from the tools already provided by Maidsafe and then implemented on the latest testnets without having to wait for the the Safenetwork’s full development. The validator hosted on the Safenetwork doesn’t need to read the blockchain or use hosted nodes at this stage of development. This will be done through API clusters. When the exchange is production ready, an independent double-checking system will also be implemented by adding nodes to the validator. The proof of concept can run on the latest testnets, but the exchange does require the Safenetwork to be past Alpha phases for it to be production ready as Safenetwork testnets are reset frequently.

SafeNetwork aims to provide distributed computing in a secure environment by splitting the informations and contents into encrypted chunks and storing them in the network. The SafeNet applications will use the resources provided by the Farmers, either CPU, memory, disk space, bandwidth. As a reward, the farmers will receive Safecoins. To limit the resources spoil, the Safenetwork apps will have to use Safecoins to pay for network’s resources.

Due to the nature of the validator, it will use Safenetwork because of the advantages provided and decentralized nature of the network. The main issue with the implementation of the validator in the SafeNetwork is that it will be bound to the success or failure of the Safenetwork project.

In the failure of SafeNetwork, there are other alternatives. The most interesting one is deploying a smart contract on the Ethereum blockchain or using the promising Waves smart contracts. The validator can also be hosted in a server owned by NVO, but this solution would make the exchange semi-decentralized, so it won’t be used unless it’s necessary as a temporary measure.

Why use the SafeNetwork instead of a Smart Contract :

Smart contracts and SafeNetwork are two different products. They have their own set of advantages and disadvantages. When it comes to creating an autonomous organizations, Smart Contracts are ideal as the code is saved in the blockchain and can be run remotely. The problem is the amount of data that can be stored in the blockchain even if Ethereum provides more storage than most of the other blockchains.

Regarding the SafeNetwork, it is a distributed file storage system and a distributed computing system that provides resources to the applications and websites built for it. This environment is more suitable for the validator, as it’s size and resources consumption will grow over time. The other advantage is that it respects the decentralized requirements as the data will be securely stored within the network.

2 Modular projects

2.1 Wallets

Initially the wallet will be provided with limited features then these enhancements will be implemented :

- Multisignature addresses
- Addresses management
- Account management
- Offline storage
- Offline management
- Assets creation and management
- Support for Ethereum tokens
- Light Nodes
- Staking wallets
- Multiple languages

These features will be implemented at different stages of development.

The wallet will use random password and pass phrases dynamically generated using the CPU entropy or another provider of random data.

Each wallet will generate a Unique User ID. It will be used by the validator for order origin and transactions history. This ID will be used to manage the user accounts in the wallet and enable the support for multiple accounts in the same wallet. This ID will never be used to track users. Personal data will never be asked, collected or sent to any third party.

As an account management system will be implemented in the wallet, it is normal that multisignature addresses will be enabled as well as part of the advance features created for experienced users. Users can generate as many addresses as they need. The address management let users select which address will be used to receive or send transactions comparable to Coin Control. It will display the available receiving addresses and sending addresses.

All these features will be available online and offline. Users will have the ability to “prepare transactions”. Prepared transactions can be either broadcasted when the wallet goes online, or be deleted.

The wallet will support popular cryptocurrencies on default. Some of these cryptocurrencies have tokens creation functionalities. This can be supported due to dynamic nature of the wallet through a flavour plugin system. It’s also possible to get the transactions history for these tokenized assets. The main technical difficulty here is the support of Ethereum tokens and smart contracts as they require the usage of a node to retrieve the required informations.

Efforts have been provided by the original creators and maintainers of cryptocurrencies to provide “Light nodes” possibilities to developers. These light nodes will be progressively added to the wallet. The main problem is the possibility to run them all at once. Supposing that a wallet supporting 10 different cryptocurrencies. Each one with a Light Node implemented means that the wallet will have to manage 10 different networks at once. This will lead to an increased bandwidth usage and CPU usage.

As the wallet should notify the users about incoming transactions, either on mobile or desktop, the networking problem is an issue for development. Depending on the feedbacks received from the users, NVO may either enable networking via Light Nodes for features like transactions broadcasting or to trigger notifications for incoming transactions. Or disable it and rely on the API clusters via subscription method which means that each address owned by the user will have to be passed to the API cluster to keep track of the balance.

An updater will be provided with the wallet. It will be used as a base module for the development of the plugins system allowing cryptocurrencies developers to create integrations in the wallet and allow users to choose which cryptocurrency they want to hold in their wallet. The plugins will be developed by either by any third party or by the NVO team. These will provide the users with features they would like to add to the wallet, reduce the size of the initial downloaded package and extend the possibilities of the future integrations and innovations from other projects.

The wallet could be used to stake POS coins, as part of the objective to provide an alternative to the usual wallets and strategy to grow the userbase. This feature will be enabled in the later development stages as it requires a lot of setup, a mature code, and an extended test flow.

2.2 API Clusters

The Demo release uses different APIs to gather the required informations for the wallet either transaction history, broadcasting, balance checking and even cryptocurrencies prices.

To provide a continuous service without having to supply any information to third parties, an API cluster network will be deployed. It is a simple architecture meant to provide different informations to the wallet and ease the integration of tokenized assets and functions from the different cryptocurrencies that may not be available using the common API.

The advantages are :

- Lower running fees than subscribing for a paid API provider.
- Better response time
- No country limitation
- Customized data formating
- Usage of advanced functions
- Support for all cryptocurrencies
- Doesn’t require to send high level informations
- Availability guaranteed with API failover

Inconvenience :

- Data stream interception (MITM)
- DDOS vulnerability
- Brute forcing

Regarding the inconveniences,, many solutions can be applied. NVO could use some of the available crypto projects to encrypt the data streams (Namecoin, Nexus) and communications even if the informations exchanged won’t be of critical level, users may want to keep their transaction history or orders history private. NVO will never send the private keys, seeds or the password to unlock the wallet, these will left at user's’ local storage.

The DDOS vulnerability isn’t an issue because the architecture of Clusters means that NVO will deploy several API clusters, so if a cluster is under DDOS, the failover mechanism will automatically switch to another cluster. The unique scenario where a DDOS can really harm the clusters network is if the entire network is attacked at the same time. In this scenario, NVO will have to deploy more clusters. Users will have to wait for these clusters to be usable, meanwhile a quick fix will be released to enable the usage of the common APIs. At the time of writing, it is the best solution available in case of a general DDOS targeting all the clusters at the same time.

What is a Cluster ?

An API Cluster is the regroupement of several servers, each one dedicated to a single usage.
As an example, if NVO were to use this system at the release of the Demo, a typical cluster would consist of :

- 1 server for bitcoin blockchain
- 1 server for Ethereum blockchain
- 1 server for Ripple ledger
- 1 Processing server

The blockchain servers will be running pruned blockchains in order to save disk space, and it will reduce the cost of a full blockchain as long as it is not required. The issue when working with pruned blockchains is the lack of historical informations about some addresses which may bring incomplete informations. In this scenario, several solutions are available. The processing server could ask the Bitcoin pruned blockchain for the oldest transaction, and if it is out of reach for the actual state of the pruned blockchain, it could grab this information from either a full node run by NVO or an external source.

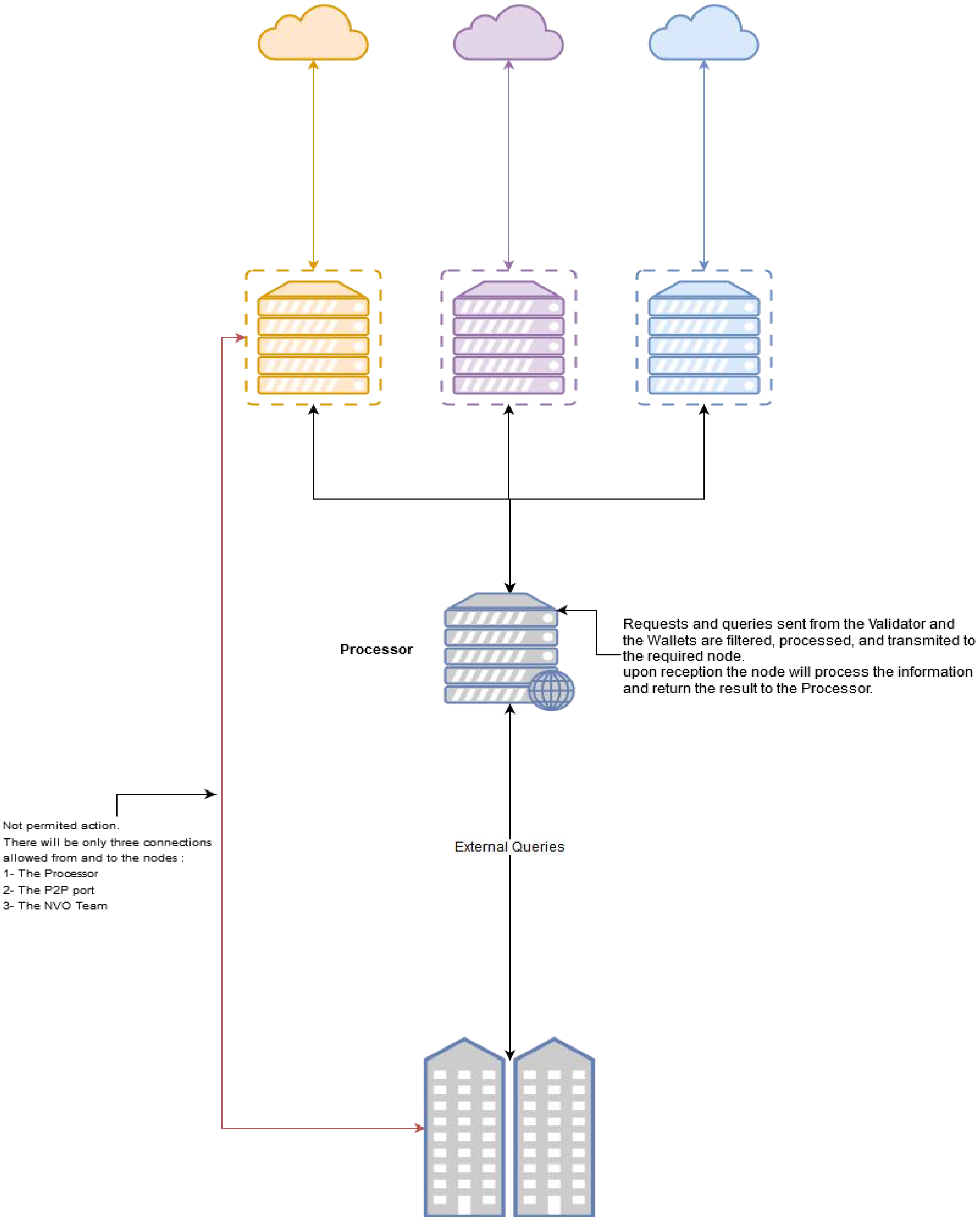
This scenario isn't expected to happen soon as all the addresses generated by NVO will be new addresses, except if a user imports an old wallet. In this case, the API processing server will solve the problem by asking an external source for the historical transactions of the imported wallet.

Why use pruned blockchains ?

The running cost of a full Bitcoin blockchain is expensive compared to a pruned one. The downside is the limited depth of transactions history.

Full nodes may be considered if some conditions are met like a consequent Cluster network, a popular demand from the users. This will happen if it's reliable enough to be able to sustain for full nodes.

The figure is a representation of an API cluster:



Requests and queries sent from the Validator and the Wallets are filtered, processed, and transmited to the required node.
upon reception the node will process the information and return the result to the Processor.

Not permited action.
There will be only three connections allowed from and to the nodes :
1- The Processor
2- The P2P port
3- The NVO Team

These clusters don’t require a lot of maintenance and are easily deployable. The most important part is to create the required calls and functions in the processor to return the required datas to the wallets and the validator.

NVO will consider different strategies regarding the calls, and the technologies to use. For example, a websocket can notify users when a payment arrives to the wallet, but it is not handy when it comes to broadcasting a transaction, as it is usually made to receive live data feeds while a REST API is great at processing data on demand, but it is poor with live feeds as it relies on AJAX calls. Webhooks may be interesting but they are known to be a source of security issue if not implemented wisely.

This architecture enables the usage of failover. If any API cluster is down for any reason, is unreachable, or with a high response time, the wallets will move to another cluster. We could limit the amount of users connected to a single cluster to reduce the payload and direct the wallets to another cluster to provide a short as possible response time.
If at any moment, the communication between the API clusters and the users needs to be encrypted to provide a better security level, the usage of encrypted data streams will be enabled using SafeNet or TrustNet or Namecoin encryption methods.

Ultimately the API clusters may be removed or kept in a minimal state depending on the evolution of NVO wallet and exchange. The ultimate goal of each module is be entirely automated and to rely on it’s own resources to get the required informations.

2.3 Validator

The validator is a SafeNet application that will parse the orders sent from the wallets. Each order will contain informations about the user’s request. It will verify these informations, queue the order and check for a matching order.

Each request sent from the wallet will contain these informations. These can change during development :

- Order information
- Trading Pair
- Order Type
- Amount
- Price
- Address A
- Address B

Validation Process :

The first verification made by the validator upon the reception of a request is to validate the balance of Address A. If the amount has changed during the issuing process, the request is rejected and a message is returned. If the balance is valid the request turns into an order then it is queued for matching.

Once an order can be matched with another, the validator sends a payment request with order information, sending addresses, receive addresses, and the amount of each currency to both request issuers. On the wallet side, when the message is received, the balances are checked. If the amounts are sufficient to cover the fees of the transaction and the trade, a raw transaction is created and signed then returns to the validator in Hex format.

Once the raw transaction is received, the validator will parse it and verify the addresses and the balances validity. If everything is valid, the transactions are broadcasted. Transaction hashes, order summary and useful informations are sent back to users.

The downside of this system is that each wallet must be online to conclude the trade resulting in users no longer to be able to shut down their devices after placing an order. Offline trading won’t be supported. A solution may come up, but it would be hard to apply it without compromising the decentralization of the trade at this particular time of the Safenetwork’s development. To compliment this downside, UX developers will make sure orders are seen and executed faster than ordinary exchanges by several methods. One example is by putting the latest trade order in a section of the wallet seen by all wallet users, as well as a maker-taker fees system encouraging users to provide liquidity rather than fulfilling existing orders. This downside won’t affect the user experience as long as the exchange process is convenient and can be fulfilled quickly through a large userbase from the wallet. This also means that it's important to implement several strategies to acquire more users to use the wallet.

For offline trading, a probable, yet conditional solution may be applied. It consists on a 1of2 multisignature address shared between the user and the validator to enable offline trading in a decentralized environment. The user will provide a currency pair to trade, and an amount, the order will be listed for matching, and when the validator finds a matching pair it will sign the transaction using its key. It can be considered as a shared account between the validator and the user. This solution poses some questions regarding decentralization. The functions will be secured and automated by safenetwork and a smart contract, but it isn’t fully decentralized as the validator is now holding assets.

Another probably theory for offline trading that was given from an outside source is to implement both the wallet and validator on the Safenetwork. This means keys will be securely stored on the Safenetwork accessible at any time. If the Safenetwork is unhackable this is an approach that could be taken and can provide other benefits as well such as hosted recovery features. This won’t be reasonable to implement during testnet or Alpha stages.

2.4 Exchange limitations

Supposing the wallet supports 10 cryptocurrencies, and the NVO Exchange enables the trading of each currency with another. Users will have 100 pairs to trade and most of them would be just irrelevant due to liquidity being so spread out.The temptation to enable the possibility of exchanging to each currencies with another is great, however, the users will be lost in the exchange process.

This is why two different markets will be enabled, a main market and a sub-market.

The main market: Enabled the trading of the major cryptocurrencies, like Bitcoin, Ethereum, Litecoin, Monero, Dash ...
The Sub Market: some cryptocurrencies has the ability to create tokenized assets. These will be traded on the sub Market. The pairs matching will be Cryptocurrency/Tokenized asset ie: **ETH/GNO, BTC/MAID**, **WAVE/INCNT**.
It means that for each supported currency that can create tokenized assets, a sub-market will be enabled to allow the trading of these assets as long as they are supported by the wallet. Assets created from the submarket can also be exchanged with Bitcoin.

This option of dual market brings ease of use for users and is easier to scale as running on Safenetwork won’t be free, thus the available resources are used for the most popular pairs. Creating low volume pairs will result in more storage space and a waste of costly resources.

The exchanging operations will be held on the SafeNetwork. Each step will happen in a different decentralized settings without the intervention of a 3rd Party. This means that the exchange won’t suffer from any country limitation as the mechanism will be hosted on a distributed network using the capacities of different decentralized providers.

2.5 Plugins system

This system will be built on top of the updater and will use the same routes and servers. On the wallet side, the users will have access to a tab displaying the plugins.

The wallet may have to be reloaded depending on the nature of the plug-in. Some can be loaded directly after downloaded while others will trigger a wallet reload.

The most important part of implementing a plug-in system is to enable a reverting possibility. This will be a mandatory prerequisite to any plug-in.

Plug-in types:

- Cryptocurrency support.
- Prototype feature.
- Flavour plugin.

These plugin categories are the main ones that will be added. Other types may be added later depending on the needs for them.

Cryptocurrency support plug-ins enable integration of new cryptocurrencies into the wallet.

Prototype features will be released by the NVO Team. During the development, the NVO Team will release alpha features. Users that wants to participate in these test can download them separately. This will provide the NVO Team with better feedbacks regarding the stability and bugs of each feature.

Flavours are wallets with different features than the basic wallet, as an example, a trading flavoured wallet will provide a trading focused user interface with enhanced features like enabled arbitrage with different exchanges.

The plugins system open room for third party services and companies to build services within the wallet. An ecosystem can be created that will help acquire users for the wallet and ultimately to the exchange. A payment system can incentivize developers to add productive features to the wallet or help companies create paid products as a plugin.

2.6 Acknowledgement of Risks

Problems and solutions evolve with development. There is almost always a solution to a problem, and sometimes there are too many solutions. The focus will always be finding and choosing the ideal solution.

Assets will be lost and unrecoverable if:

- The Safenetwork is compromised resulting in the validator being breached. Only assets being exchanged can be stolen through the force validation of orders. Assets in storage and not being placed for exchange can’t be accessed by the validator.
- Data Stream manipulation and interception. In case of any MITM, Double Spend, or SideJacking, the users could suffer from a coin loss. There are many methods that can be set up to counter these key risks.

- If MaidSafe is unable to deliver Safenetwork, or if the Safenetwork encryption is compromised, the validator will be impacted as it’s purpose is to take advantage of the encryption and resources of the

Safenetwork. If this scenario happens, the validator will be moved to an alternative option until hosting on the Safenetwork is viable again such as using smart contracts or using servers as a temporary solution resulting in the exchange being semi-decentralized.

- If the data sent/received to/from the wallets is improperly encrypted, a third party could intercept these datas, they won't have access to much informations except the raw transaction. If the raw transactions are modified, it won't be accepted by the validator.

2.7 Releases

NVO Wallet will be released for :

- Windows 64x
- Mac 64x
- Linux 64x
- IOS
- Android

The demo on the website is an exception as there is no point to engage efforts in developing it for mobile support at this time.
The mobile version won't have all the functionalities of the desktop version. The main objective of this version is to enable trading from the user's tablet or smartphone. To do so, the Plug-in system will be enabled on the mobile wallet to keep the ability of selecting coins. The only problem with developing for IOS is the uncertainty of the App Store terms and conditions.

Both desktop and mobile wallets are complementary of each other and will let users move their funds directly from mobile to Desktop and vice versa without having to pay transaction fee, simply by importing private keys.

2.8 Source code policy

- The wallet will be open source, users will be granted the ability to use, modify, distribute and redistribute the wallet. NVO used an ASAR archive instead of a plain folder to avoid wrong manipulation of the wallet's source. Users who wishes to amend the code, or read it can find guides on how to open an ASAR archive, in case of complications, users can contact the support of NVO.
- The validator will be closed source until a production ready version is available. Then a voting process will be taken by token holders for the validator to be open sourced.
- The plugins will be created by both the NVO Team and the developers community. As they are part of the wallet, the same source code policy is applied.

2.9 Development flow

The development of the different modules will happen simultaneously, as each module has to be adapted to each other.

NVO wallet updates will be released on a monthly basis with detailed release notes. As there will be a plug-in system, it is better to fix the release dates for the wallet to have enough time to proceed to the required test prior to any release.

3 Business model

3.1 Fees per trade

There will be a competitive 0.2% fees per trade. A maker and taker fees system will be implemented to provide liquidity for new markets of up to 0.35%. As of May 8, the 24h volume of the top exchange is $963,790,086 with a growing cryptocurrencies marketcap of $40 billion. Fees on the exchange will be automatically distributed in a decentralized manner after the exchange is production ready using counterparty.

3.2 The risks of centralized exchanges

Over $1B is traded through exchanges on a daily basis as of May 2017, and over 95% of this volume are processed through centralized exchanges. This is a risk most cryptocurrencies users can understand. An ecosystem that fully relies on the ability of centralized software to process and store decentralized cryptocurrencies is unhealthy and can easily be manipulated.

A more balanced ecosystem is needed for a decentralized economy. Centralized solutions are crucial, as they offer great financial solutions to connect with the mainstream economy. However, decentralized solutions are necessary in a decentralized economy because of their nature to be secure, transparent and trustless.

Traditional networks are insecure

Traditional networks are inherently flawed. There are endless points of failure that can lead to a chain reaction. So a vulnerability affecting a single service can leave all users vulnerable as demonstrated by the latest Cloudflare memory leak incident and heartbleed.The development of the Safenetwork has been carefully reviewed. Documentations and demonstrations have been provided showing that the project does work backed by an active development team. Although the Safenetwork biggest hurdles right now won’t be a technological one but a marketing one. All disruptive technologies will face doubts from the public and mainstream adoption.

3.3 Competitive advantages

- Trustless and decentralized exchange.
- A wallet and an integrated exchange all controlled by users.
- Wallet’s users are encouraged to try out the exchange.
- Users can enable as many different assets as they want through an open plugins system.
- Low in operating cost and cheaper to scale.
- Security bounty program.
- Multiple languages support means the exchange won’t be restricted by location.
- Streamlined user flow with simple learning curve
- Asset-to-asset exchange. No proxy tokens.
- Due to being a crypto integrated platform, Ripple Gateway could be added so users could trade fiat, support Ethereum tokens or support the creation of assets and using them through Waves or Omnilayer.
- Being a dynamic platforms means anyone can contribute resulting in the possibility of creation of services revolving around the wallet.

3.4 Allocation of funds

This is a rough estimate of how funds will be allocated :

- 50% will be reserved for the development and design of the entire project
- 15% for marketing and community management
- 10% for legal consulting
- 5% for distribution fees for NVST holders through NVSX.
- 10% for jumpstarting liquidity when the exchange is opened.
- 10% as reserve for security bounty program.

3.5 Fiat management

Fiat won't be supported by NVO. Using the plugin system on NVO, exchanges and other service providers can add fiat management to the wallet for users.

4 Token details

4.1 Token details and fees distribution

The NVO team will use counterparty to generate 15.000.000 NVO tokens.

Token supply:	15 million
Emission rate:	No new coins created
Token distribution date:	July 1st to July 7th 2017

50% of all fees per trade on the exchange will be automatically distributed to addresses with NVO tokens on a weekly basis in the form of NVSX when the exchange is production ready. NVSX will have a fixed price of $0.99 and can be traded for Bitcoin automatically using Counterparty exchange or through the validator without fees. NVST price will not be fixed and will be decided by the market.

NVST and NVSX are not the same. NVSX will be issued proportionally to the amount of fees collected by the validator's usage. They will be distributed among the holders of NVST and will have to be stored/sent to an NVO wallet for redeem. Upon the detection of NVST inside a user's wallet, the validator will issue a 2OF2 Bitcoin address containing the amount of NVSX based on the user's amount of NVST. It will then create a redeem script and send it to the holder wallet. The wallet will endorse the redeem script and allow the transaction.

This will enable an automated process of fees distribution. NVST holders will still have to send funds into a NVO wallet to allow the process to take place. Later the whole process will be automated, using the new features from counterparty like smart contracts allowing the exchange and payout process to be fully autonomous.

This system enables an automated distribution process amongst the addresses holding NVST, external exchanges filtering, and keeps NVO legal as only tiny amounts are processed because the price of NVSX will be fixed at 0.99 USD. The total available amount of NVSX will change proportionally to the amount of fees collected. It is a way to control the inflation and the price to enable a stable value asset for the holder's usage.

It can be considered as a POS, as the only way to receive fees distribution is to hold the funds in an

NVO wallet. If a holder wants to store the NVST in a paper wallet, he will still receive the NVSX when he redeem it using NVO wallet as they will stay in circuit and stored for them.

NVST will be exchanged like any other asset or token. For NVSX, we will use the decentralized exchange of counterparty or using the validator. The most important part is to keep the payout process decentralized and automated and as far from the control of NVO team as possible while being secure to realize the vision of being a decentralized exchange.

NVO will pay for all transactions fees during the exchange of NVSX to Bitcoin. Counterparty calculates the fees dynamically. NVO will try to issue as less fees as possible for transactions. As an example, let's suppose that 100 holders issued a redeem order, the validator will create 100 multisig addresses and send to each one of these addresses the proportional amount of NVSX. It will create a single transaction as long as it is lower than the size of a block.

The fee estimation for such a transaction is (This part is provided for information only and isn't necessary to understand.) :

Inputs size =~ 147 bits

Outputs size =~ 34 * (amount of NVST holders = 100)

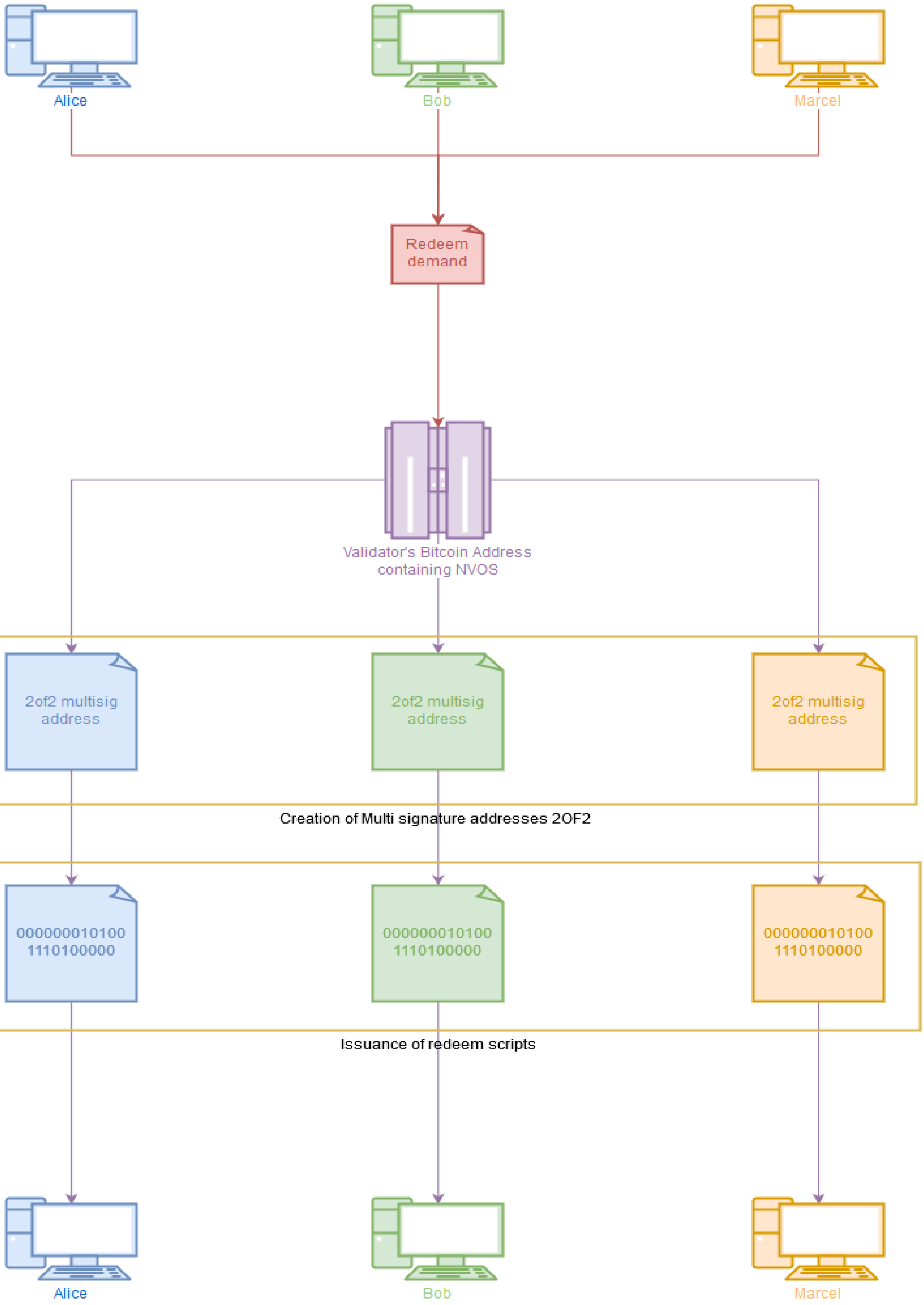
tx size =~ nr of inputs*147 + nr of outputs*34 + 10 + number of inputs

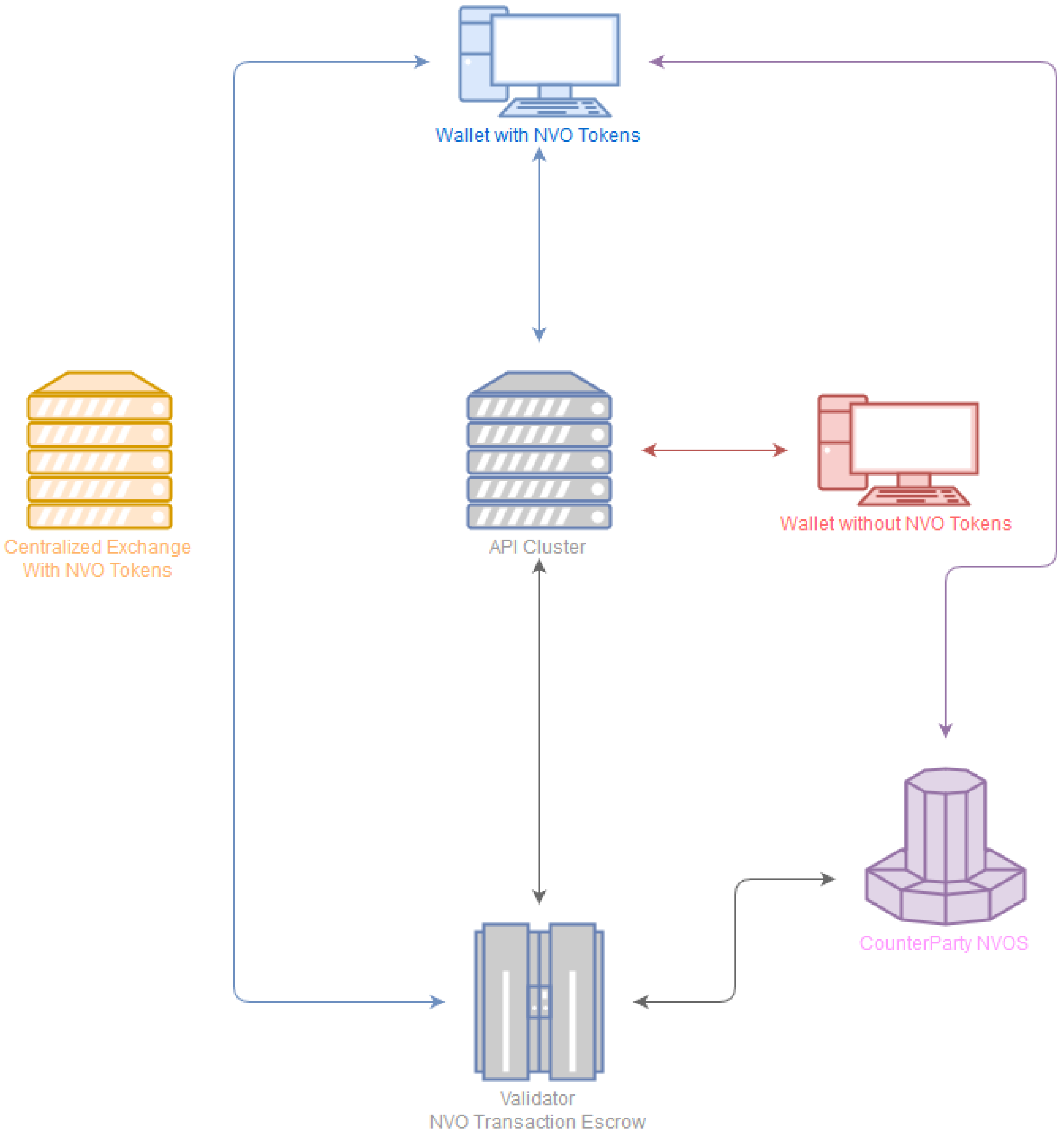
translated => tx size =~ (1 x 147) + (100 x 34) + 10 + 1 = 3558 bits

Considering the average confirmation time, the paid fees should be around 200 sat/bits in order to be confirmed in the next 3 hours, which means the fees will be =~ 0.00711600 BTC.

This calculation is based on the bitcoin blockchain, it may be more or less as counterparty isn't taken in count.

4.2 Overview of distribution process





4.3 Counterparty

Safenetwork will be used for the exchange while counterparty will be used for token creation and fees distribution because of the exciting features they are working on. Soon NVO will be able to use smart contracts on the bitcoin blockchain using serpent or solidity. These smart contracts will burn XCP instead of gas, thus it will be an implementation of smart contracts into the bitcoin blockchain using Counterparty.

Smart contracts are not the only unique great feature. Counterparty will also enable the usage of payment channels, Lightning which will let NVO add another layer of security to the exchange, and enable a proper response to the double spending issue.

NVO will take advantage of these features to automate and secure the NVO Exchange and provide an enhanced experience to end users.