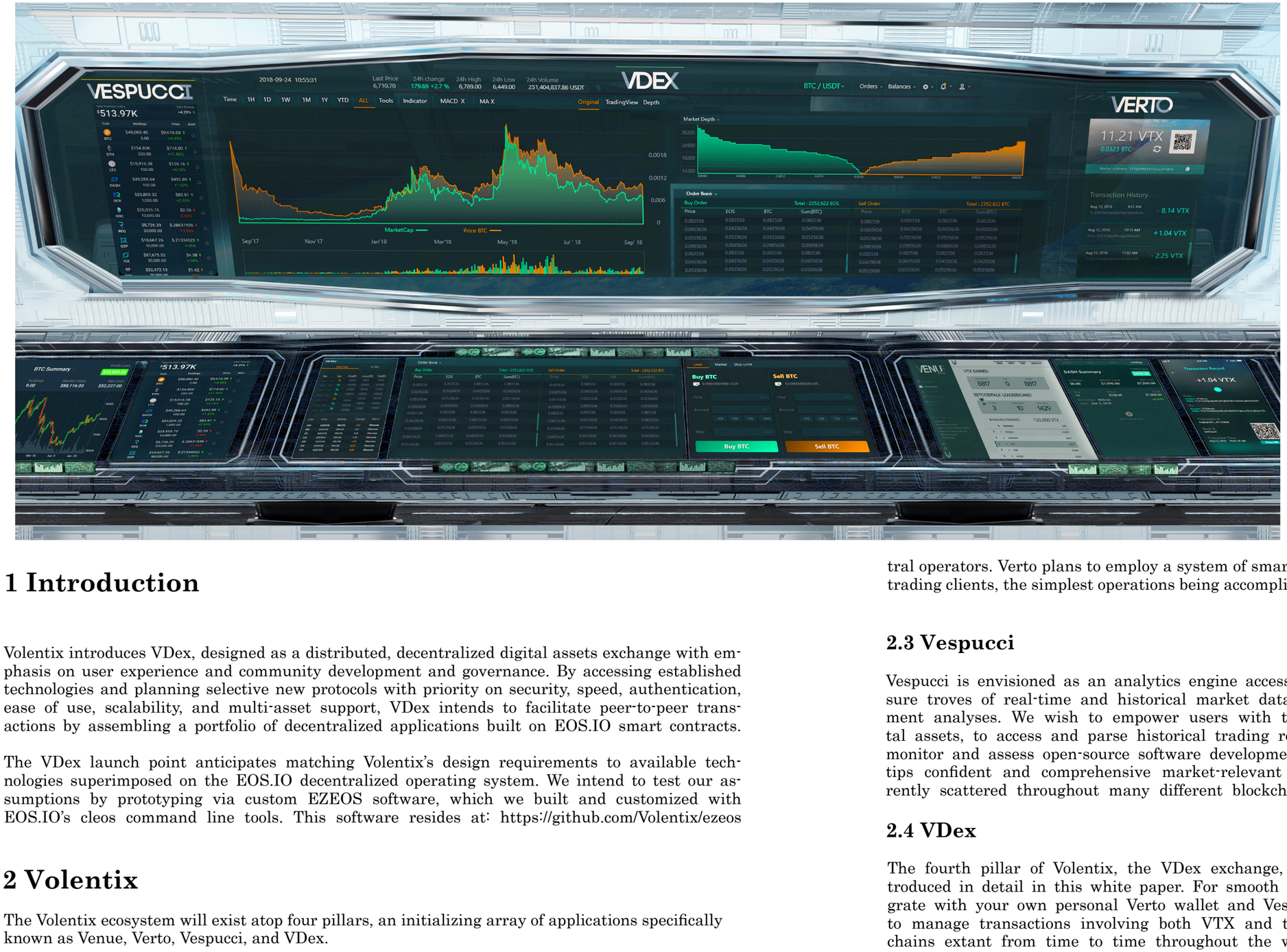
Volentix is envisioned to be a community-powered decentralized autonomous organization (DAO).

THE VOLENTIX VDEX WHITE PAPER V0.1.4

10242018

TABLE OF CONTENT



1 Introduction

Volentix introduces VDex, designed as a distributed, decentralized digital assets exchange with emphasis on user experience and community development and governance. By accessing established technologies and planning selective new protocols with priority on security, speed, authentication, ease of use, scalability, and multi-asset support, VDex intends to facilitate peer-to-peer transactions by assembling a portfolio of decentralized applications built on EOS.IO smart contracts.

The VDex launch point anticipates matching Volentix's design requirements to available technologies superimposed on the EOS.IO decentralized operating system. We intend to test our assumptions by prototyping via custom EZEOS software, which we built and customized with EOS.IO's cleos command line tools. This software resides at: https://github.com/Volentix/ezeos

2 Volentix

The Volentix ecosystem will exist atop four pillars, an initializing array of applications specifically known as Venue, Verto, Vespucci, and VDex.

2.1 Venue

Venue is planned as a dynamic community platform that recruits and aligns members of the Volentix community to facilitate distribution of VTX, the native digital asset of the Volentix ecosystem, and to promote awareness of Volentix initiatives.

Recently launched in beta testing, Venue enables users to receive VTX in exchange, for example, for participating in developing dedicated communities, submitting bug fixes, and claiming bounties. Leaderboards and live metrics reflect user participation. The first signature campaign was launched on the https://bitcointalk.org/ forum on July 13, 2018. Please visit https://venue.volentix.io for more information.

2.2 Verto

Verto is being built as a multi-currency wallet for use with the VDex decentralized exchange, and intends to facilitate personal custody and local management of private and public keys in peer-to-peer transactions, with the goal of eliminating the risks of devastating losses of stake associated with traumatic failures of central operators. Verto plans to employ a system of smart contracts to maintain the state between two trading clients, the simplest operations being accomplished with atomic swaps.[1]

2.3 Vespucci

Vespucci is envisioned as an analytics engine accessible via a user-friendly interface with treasure troves of real-time and historical market data, such as digital assets ratings and sentiment analyses. We wish to empower users with tools to graph and compare tradeable digital assets, to access and parse historical trading records, to plot trends and patterns, and to monitor and assess open-source software developments. Vespucci seeks to bring to your fingertips confident and comprehensive market-relevant data by aggregating the information currently scattered throughout many different blockchains, websites, chat rooms, and exchanges.

2.4 VDex

The fourth pillar of Volentix, the VDex exchange, is the tradable digital assets platform introduced in detail in this white paper. For smooth and secure usability, we plan VDex to integrate with your own personal Verto wallet and Vespucci interface. We expect VDex to be able to manage transactions involving both VTX and the vast array of digital assets and blockchains extant from time to time throughout the world. We are developing Venue as a complementary adjunct primarily in order to incentivize and drive native VTX-based initiatives.

3 Architecture

3.0.1 Operating system

We have evaluated various operating systems as candidates for the substructure of our VDex exchange. Though we honor the work done by a number of the established leaders in digital assets and blockchain technology, among those trailblazers the work of EOS.IO as an operating system-like framework upon which decentralized applications can be built stands out, in our opinion, as exemplary. The software provides accounts, authentication, databases, asynchronous communication, and scheduling across clusters. Components and protocols are already built into the platform, and a subset can be used to satisfy our VDex requirements. VDex will initially benefit from the standard features offered by EOS.IO such as account and wallet creation and the recovery of stolen keys, but we plan subsequently to implement protocols for creation of a decentralized exchange through EOS contracts and other tools.[2] Here is a summary of encouraging methodologies:

Context Free Actions

Most of the scalability techniques proposed by Ethereum (Sharding, Raiden, Plasma, State Channels) become more efficient, parallelizable, and practical while also ensuring speedy inter-blockchain communication and unimpaired scalability. A Context Free Action involves computations that depend only on transaction data, and not upon the blockchain state.

Binary/JSON conversion

EOS contracts combine the human readability of JSON with the efficiency of binary.

Parallelization and optimization

Separating authentication from application allows faster transaction times and increases bandwidth. EOS.IO blocks are reportedly produced every 500ms.

Web Assembly(WASM)

Web Assembly enables high-performance Web applications and also secures each application in its own sandbox, through which functionalities VDex can obtain network access, filesystem namespace restrictions, and enforced rule-based execution.

Rust/C++ contracts

The well-known and popular programming language C++ appears highly suitable for WASM. C++ has highly mature debugging support and libraries. The EOS codebase uses templates liberally, and C++ allows the use of templates and operator overloading to define a runtime cost-free validation of units. The program reinitializes to clean state at the start of every message, a distinct advantage that streamlines formulation of smart contracts. The WebAssembly framework automatically rejects any transaction addressing memory inaccurately. In case dynamic memory allocation is necessary, users can depart to smart pointers because EOS.IO contracts use C++14. It is noteworthy that the

first implementation of PARSEC Directed Acyclic Graph (DAG) technology is expected to be in Rust.

3.0.2 Schema defined messages and database

Service contracts are standardized to provide a baseline measure of interoperability between and among disparate systems by harmonization of data models. Indeed, the Standardized Service Contract design principle advocates that service contracts be based on standardized data models. Analysis is done on the service inventory blueprint to find out the commonly occurring business documents that are exchanged between services. These business documents are then modeled in a standardized manner. The Canonical Schema pattern reduces the need for application of the data model transformation design pattern. [3]

3.0.3 Inter-Contract Communication

Data is shared between contracts via an oracle, which, “in the context of blockchains and smart contracts, is an agent that finds and verifies real-world occurrences and submits this information to a blockchain to be used by smart contracts.” [4] Every node will have an identical copy of these data, for use in smart contract computation. Rather than the smart contract functioning to pull the data, instead the oracle pushes the data onto the blockchain. In the instance of a blockchain, most reading of the data is done via polling “models” in order to monitor blockchain state and to perform certain responsive actions.

3.0.4 Sidechains

In EOS.IO, issuance of a digital asset creates a sidechain, which is an emerging mechanism permitting digital assets from one blockchain to be securely used in a separate blockchain and then moved back to the original blockchain. Efficiency of processing is promoted by creating multiple sidechains. A TCP-like communication channel between different blockchains evaluates proofs. For each shard (a unit of parallelizable execution in a cycle), a balanced merkle tree is constructed of these action commitments to generate a temporary shared merkle root; this is done for speed of parallel computation. The block header contains the root of a balanced merkle tree the leaves of which are the roots of these individual shard merkle trees. [2]

3.0.5 Liquidity

A digital asset is liquid if it is easily sold or purchased in ordinary trading volumes without a significant short-term impact on its prevailing market price. In order to achieve such a status, traditionally any tradable asset must surmount a trading volume threshold sufficient to support stability. Specifically, we anticipate adopting the following methodologies:

Loopring protocol with the use of EOS.IO contracts acting as nodes.[5]

Bancor algorithm used to bring stability to the digital asset.[6]

Toggles between these protocols and HTLC (atomic swaps) according to Vespucci analyses on the VDex network.

3.0.6 Hashed Timelock Contracts (Atomic Swaps)

A Hashed Timelock Contract (HTLC)[1] is a smart contract enabling the implementation of time-bound transactions. Users will be offered a variable lock-in period for their transactions, with a discount on transaction fees in exchange for choosing a longer lock-in period.

3.1 Network Topology

3.1.1 Nodes

Nodes are the endpoints of the VDex exchange. Their functions are:

1. Act as portals to VDex through the Verto wallet.

2. Merge order book information.

3. Settle order book.

4. Manage order cancellation.

5. Assign timeouts for the RAFT protocol.

6. Initiate contracts for orders that have been filled.

Nodes earn a portion of the fee for each transaction. If a user has sufficient funds and possesses a good track record, his or her Verto wallet can act as a node.

3.1.2 Aggregators

The VDex aggregators are dedicated Volentix servers for simulator and security purposes. One of their functions is to pull logs and order book data from nodes into sparse distributed representations for hierarchical temporal memory as intrusion [7] analysis for detecting anomalies in the system. The aggregators will also be host to other components such as metachain ledgers and blockchain scrapers.

3.1.3 Latency

EOS.IO has low latency block confirmation (0.5 seconds).[5] This degree of latency can be maintained in transactions with other blockchains if those chains admit of similar latency. But fundamentally the transaction is only as rapid as the lesser-rapid chain in the equation. It is well known, for example, that a Bitcoin block requires approximately ten minutes for processing. Receiving a transaction hash does not mean the transaction is confirmed; it means only that a node accepted the transaction without error, although there is generally a high probability other block producers will accept it.

3.2 Order Book

The order book is the list of buy-and-sell orders VDex records from interested users. A matching engine uses an order book to determine which orders can be fulfilled. The Loopring protocol allows for customizing

the order book data structure.[5] Containers provided by EOS.IO can be used for optimal performance.[8]

3.2.1 Data structures

Using the Loopring Protocol FIFO (first-in first-out) circular buffer, nodes can design their order books to display and match a user's order. This method follows an OTC model, where limit orders are positioned based on price only.[5]

Referencing the EOS.IO persistence API, the order book is able to take advantage of the powerful multi-index container shared among nodes through the same EOS.IO account.

3.2.2 On-Chain order book

An on-chain order book is a record of offers residing on the wallet (node) chosen to settle the order book. It resides in a persistent database on each node subscribing to the same account as all the other nodes.

3.2.3 Off-Chain order book

Residing on the aggregator, offline order books serve for simulator and security purposes.

3.2.4 Decentralization process of order book settlement

For decentralization purposes, nodes will take turns to settle the order book. The settling node must be designated by the protocol and all order book entries from all nodes must be available to the settling nodes. We believe the RAFT[9] and PARSEC[10] consensus mechanisms offer effective solutions. RAFT is a well-established algorithm and is easy to implement.[7] PARSEC is more recent and more efficient, using Directed Acyclic Graph (DAG) technology and eliminating the need for copying logs.

3.3 Order settlement

Order settlement contains familiar elements of conventional financial market transactions. Utilizing FIFO technology to design the order book, VDex intends to check order, inventory, and fill rate, as well as limit orders and cancellations.

3.4 VTX

3.4.1 VTX Issuance and Use

VTX is the native digital asset to be issued and used on the VDex decentralized exchange. We currently plan to use an eosio.token contract from the EOS.IO framework to issue 2.1 billion EOS.IO-compliant VTX tokens with a supply of 1.3 billion. VTX will have a diverse array of uses, for example:

To reward participants in the consensus process and in Venue campaigns.

To pay and redistribute transaction fees on the VDex exchange.

To submit and vote on proposals to the Volentix ecosystem, using the voting rights allocated to VTX holders.

To stake support for reviewing proposals and implementing projects.

To incentivize users to participate in order book settlement by becoming nodes via their Verto wallets.

To incentivize users to lock funds in for >24 hours by HTLC time-bound transactions.

3.4.2 VTX Allocation

A digital assets ecosystem requires an array of certain fundamental human constituents who shepherd the project forward.[11] It is essential to compensate those individuals for their participation. Subject to adjustment, Volentix currently anticipates the following allocations:

1. Contributors. 12%. An array of individuals, akin to founders, who contribute insights, time and talent, though often work without early compensation.

2. Supporters.

Phase 1. 5%. Early passive seed funders.

Phase 2. 28%. Funders via qualified private pre-sales and possible public sale.

3. Facilitators. (Advisors, Developers, Promoters, Custodians). Note that requirements for assistance from the sub-categories in this category may differ significantly before and after the project receives substantial funding support, but certain individuals may serve during both phases.

Phase 1. 10%.

Phase 2. 10%.

4. Decentralized treasury. 35%. Community members incentivized and rewarded for participation in progressive development of a decentralized autonomous organization (DAO). A decentralized treasury is anticipated to be administered by smart contracts and community consensus.

3.4.3 VTX Distribution

In light of market conditions at the time of this writing, Volentix is considering timing, means, and terms and conditions of VTX distribution as a function of private pre-sales and possible public sale. Please monitor our website for updates.

3.5 EOS.IO platform deployment

The following considerations are relevant to our deploying the VDex exchange on the EOS.IO platform:

Deploying a contract has a cost but is free to use.

Developers stake EOS.IO-compliant tokens to deploy a smart contract. After the contract is deployed, the locked tokens are returned.

Decentralized applications allocate memory, CPU, bandwidth, and other resources to their contracts.

Multiple messages and multiple accounts can be assigned to the same thread.

3.6 Blockchain Interaction

3.6.1 Inter-Blockchain Communication

EOS.IO is designed to make Inter-Blockchain Communication (IBC) proofs lightweight. For chains with insufficient capacity for processing the IBC proofs and establishing validity, there is an option to default to trusted oracles/escrows. With an EOS.IO-based smart contract, a trusted multi-signature wallet holding the asset in escrow can be used to persuade the signing/publishing of the transaction based on IBC proofs from the originating chain.

3.6.2 Multi-Blockchain Information

Comprehensible multi-blockchain information can be obtained by aggregating blockchain timelines in parallel order (with variance in the frequency of change of state). This system can trigger multi-chain load balancers, transfer states, draw data outputs from smart contracts, and foreign blockchain transaction execution. Relative block distance, relative global state, and timestamped events are recorded on a global ledger to optimize and confirm transactions before they actually happen on the native chain. This approach could also be used to determine block production coincidence between chains to access greater liquidity.[12]

3.7 Security concerns

To shake out certain assumptions, we intend to commence security testing following the prototyping phase. Security concerns are of paramount importance to users and must be addressed. Threats include, for example, an attacker executing malicious code within a transaction or manipulating the order of transactions or the timestamps of blocks. In the following sections, we address certain security measures and specific security threats and remedies.

3.8 Security measures

3.8.1 Contract security

Retain vast majority of funds in a time-delayed, multi-signature-controlled account.

Use multi-signatures on a hot wallet with several independent processes/servers double-checking all withdrawals, with the concomitant benefit of creating a trusted list of accounts.

Deploy a custom contract that allows withdrawals only to accounts verified by KYC/AML.

Deploy a custom contract that accepts only deposits of known assets from accounts verified by KYC/AML.

Deploy a custom contract that requires a mandatory 24-hour waiting period for all withdrawals.

Utilize contracts with hardware wallets for all signing, including for automated withdrawals.

Upgrade broken contracts.

Include ability to pause the functionality of a contract.

Include ability to delay an action of a contract.

3.8.2 Auditing rogue processes

Artificial intelligence analysis of rogue processes will accumulate and reside on the aggregators.

3.8.3 Randomization

The RAFT protocol promotes randomization by utilizing varying lengths of timeouts.

3.8.4 Log inspection

Log inspection is facilitated by the RAFT protocol, Numenta anomaly detection, and script investigations.

3.8.5 Transaction as Proof of Stake (TaPoS)

Prevents a replay of a transaction on forks that do not include the referenced block.

Signals the network that a particular user and stake are on a specific fork.

3.9 Security threats and remedies

3.9.1 Double spend

A double spend is an attack in which a particular cryptocurrency stake is spent in more than one transaction.

A race attack occurs when two conflicting transactions are sent in rapid succession into the network.

A Finney attack pre-mines one transaction into a block and spends the same tokens before releasing the block to invalidate that transaction.

A 51% attack can be mounted by anyone owning >50% of the total computing power of a network. A majority ownership position permits reversal of any transaction and allows total control of selection of transactions appearing in blocks. EOS.IO, Loopring, and RAFT appear to prevent this problem. If a block producer takes an unreasonable amount of runtime or is not sufficiently profitable, then the process is blacklisted.[5]

3.9.2 Front running

A front runner steals one or more orders from a pending order book settlement transaction. Both EOS.IO and Loopring offer remedies in which keys are protected because they are not part of the on-chain transaction, and therefore remain unknown to parties other than the owner. Only the order book settling node is possessed of the sensitive information, and each node uses a different solution for resolving the order books, introducing yet another level of complexity to promote security.

3.9.3 Forged identities

Malicious users create forged identities to send a large number of small orders to attack Loopring nodes. However, most of these orders will be rejected for not yielding satisfying profit when matched.

3.9.4 Insufficient Balance

Malicious users sign and spread orders the value of which is non-zero but the address of which has a zero balance. Nodes monitor actual balances, update these order states accordingly, and then discard them.

3.9.5 Timing attack

Timing attacks are a class of cryptographic attacks through which a third-party observer can deduce the content of encrypted data by recording and analyzing the time taken to execute cryptographic algorithms. The RAFT algorithm prevents timing attacks by using randomness of timeouts.

3.10 User experience

Our focus on user experience is primary. We wish to make VTX and the four pillars of Volentix -- Venue, Verto, Vespucci, and VDex -- easily accessible to and useable by all those who wish to join our community. We expect the experience continually to be educational as well, with templates and simulators to support a superior UX/UI relationship.

3.11 True decentralization

EOS.IO is an open-source, scalable infrastructure for decentralized applications. Its goal is a fair and transparent block producer (BP) election process utilizing a democratic delegated proof of stake (DPoS) consensus. Particularly as such a system just begins to proliferate, there will be glitches. Therefore, some degree of retained centralization is inevitable and necessary. But our guiding philosophy is one of decentralization, and our ongoing efforts are targeted to promoting a reduction in dependence on central authority.

For example, initially we plan to erect a system for electing nodes (when solving order books) that will not use a shared central clock or DPoS but instead will be based either on random timeouts for the determination of leaders in an election (RAFT) or on Directed Acyclic Graph (DAG) in the PARSEC protocol.

3.12 System recovery

The RAFT and PARSEC protocols provide a robust system for recovery in the case of node failure. Security measures are also provided for trading between and among native blockchains. If a chain defies identification, the system defaults to the next block or a short time lock.

3.13 Evolving architecture

Daily announcements of fresh code developments impacting on use of digital assets reveal the tremendous benefit of the open-source code philosophy. We at Volentix recognize we are the beneficiaries of the enormous financial resources dedicated by many early movers to developing digital assets applications over the past decade. We now have an opportunity to take the next step by creating VDex, a decentralized exchange for the next generation of digital assets transactions.

4 Concluding Thoughts

All of us at Volentix are dedicating our work and insights to developing a program premised on empowerment and independence. If you are of a mind to join us, in whatever capacity, then please do so and please become educated on the topics contained in this white paper and additional Volentix publications as we share them with our community.

5 Timeline

Please monitor our website and social media for updates and other important announcements.
Thank you very much for your attention and interest.

DISCLAIMER

This white paper was prepared, and is presented, for information purposes only. The information presented does not purport to be comprehensive. The information is subject to change in whole or in part at any time without notice. Volentix Labs reserves the right to amend, replace, remove, or delete any and all information at the sole and exclusive discretion of Volentix. Volentix Labs makes no representation or warranty, expressed or implied, concerning the accuracy or completeness of the information and expressly disclaims any and all liability of any and all kinds whatsoever for the information contained or not contained. Volentix Labs requests each and every reader to read the information fully and carefully, and to undertake independent investigation and analysis of the information, and to seek and obtain professional advice for purposes of evaluating the information. To the knowledge of Volentix Labs, no regulatory agency, government, or other third-party enforcement entity has reviewed, evaluated, or approved any part or all of the information. This information is not an offer or solicitation of any kind whatsoever and does not form the basis for any contract or commitment of any kind whatsoever. Any statement considered to be forward-looking is purely a matter of opinion, and no viewer should rely on any such statement or on any part or all of the information in any way whatsoever.

Footnotes

1. K. Kurokawa, Atomic cross chain transfer, an overview, (2015).

2. EOS.IO, Eos.io technical white paper v2, (2018).

3.. T. Earl, Soa principles of service design, (2016).

4. blockchainhub.net, blockchain-oracles, (2017).

5. F. Zhou, Wang, Loopring: A decentralized token exchange protocol, (2018).

6. G. B. Eyal Hertzog, Guy Benartzi, Bancor protocol: Continuous liquidity for cryptographic tokens through their smart contracts, (2018).

7. L. Lamport, The part time parliament, (1998).

8. D. Larimer, eosio.boot telegram chat, (2018).

9. J. O. Diego Ongaro, In search of an understandable consensus algorithm, (2018).

10. F. H. Q. M. S. S. Pierre Chevalier, Bart lomiej KamiÂtnski, Protocol for asynchronous, reliable, secure and efficient consensus (parsec), (2018).

11. Dane Keller Rutledge, Fundamental Human Constituents of a Digital Assets Ecosystem (DAE). (2018).

12. BlockColliderTeam, Block collider white paper, (2018).

Supplemental References

Aelf, A multi-chain parallel computing blockchain framework, (2018).

ARK, A platform for consumer adoption, (2018).

V. Buterin, Ethereum: a next generation smart contract and decentralized application platform, (2013).

S. Cormier, A machine based societal model for curbing citizen cynicism, (2017).

M. Duncan, Quale, Halo platform, (2018).

S. D. K. M. T. S. H. Garcia-Molina, The eigentrust algorithm for reputation management in p2p networks, (2018).

M. R. Garrick Hileman, Global cryptocurrency benchmarking study, (2017).

Komodo, An advanced blockchain technology, focused on freedom, (2018).

Q. Liquid, Providing liquidity to the non-liquid crypto economy, (2018).

S. R. M.P.M-S, Aniket Kate Matteo Maffei, Concurrency and privacy with payment-channel networks, (2017).

SingularityNET, A decentralized, open market and inter-network for ais, (2018).

M. M. Timo Hanke and D. Williams, Dfinity technology overview series consensus system, (2018).

A. B. Will Warren, 0x: An open protocol for decentralized exchange on the ethereum blockchain, (2017).

G. Wood, Ethereum: A secure decentralised generalised transaction ledger.ethereum project yellow paper, (2014).

Dane Keller Rutledge, Creating a Comprehensive Digital Assets Ecosystem (DAE), (2018).

END OF PAPER