Generic DAO primitives for Full Academic Decentralization and Scalability

Brian Planje
b.o.s.planje@student.tudelft.nl
Delft University of Technology
Delft, The Netherlands

***Abstract*—This thesis describes a new architecture for a completely decentralized and scalable decentralized autonomous organization based on multi-signature and thresh-hold signature schemes. To demonstrate the feasibility, we design, implement, evaluate, and deploy a DAO centered around music where artists can share their music in a decentralised manner and listeners can invest in artists using the DAO.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

Decentralized autonomous organizations (DAOs) are a mechanism for economic activity by an unbounded group of people within an adversarial environment. Many of such organizations have already been deployed successfully. For instance, Uniswap, a decentralized exchange, reached transaction volumes to up to $85.5 billion in November 2021 [uniswap˙volume]. The token associated with the DAO can be utilized for the collective management and modification of the exchange's protocols. Prior to this, (partly) decentralized protocols and platforms such as BitTorrent and Wikipedia have enabled millions of individuals to collaborate in file sharing and information accumulation.

Decentralized autonomous organizations (DAOs) have been widely deployed, yet many of them exhibit forms of centralization in their governance structure and infrastructure. This centralization is reflected in the lack of true managerial decentralization in many DAOs. For example, the second-largest DAO by market capitalization, APE DAO, is characterized by an initial token distribution in which 38% of tokens were distributed to various founders, who now hold a disproportionate amount of voting power. Additionally, proposals in APE DAO are vetted by a centralized moderation team, and all execution of proposals is carried out off-chain by the foundation members of the DAO. Another example is Solend, one of the largest decentralized lending systems. In 2022, it was plagued by an incident of concern. After a DAO vote, the development team took control of and liquidated the account of a whale with approximately $170 million worth of cryptocurrency, as it allegedly posed a systemic risk to the ecosystem at the time. This highlights that the ownership of 1% of the tokens is able to take control of take control of 80% of the protocol's overall liquidity.

The root cause of the failure of contemporary DAOs to decentralise lies in the underlying blockchain. Proof-of-work and proof-of-stake have failed to scale, despite a full decade of attempts to boost transaction rates, without the loss of decentralisation. Attempts to circumvent this by working with fewer miners which process more transactions, bring us back to square one to VISA-like central systems. Centralization might even be inevitable, with Cong et al. showing that in the long run, due to centralized mining pools, Bitcoin will have a centralized market structure [cong2021decentralized]. Proof-of-stake distributed ledgers run the risk of reinstating a centralized elite. To validate the network, a substantial amount of capital must be placed at risk. This set of validators can then be subjected to regulatory pressure or collide with one another to alter transaction validation rules at the infrastructure layer. They run the risk of moving to a new centrality with a new elite, who can afford to buy enough tokens to put up to stake to validate the network.

In this paper, we propose a new architecture for decentralized autonomous organizations (DAOs) that is completely decentralized and scalable. To demonstrate the feasibility of this architecture, we design, implement, and evaluate a prototype for a DAO centered around music, referred to as the Music DAO. This implementation solely utilizes smartphones and is currently live. We conduct a real-world test with users and analyze the performance of our voting mechanism. The results show that our proposed architecture is a viable and sustainable solution. We argue that pure academic decentralisation within a viable and sustainable DAO represents a key milestone in the evolution of Web3. We believe an as-simple-as-possible DAO with basic governance, membership voting, and treasury management is a key step forward in achieving this goal.

1) **A Simple DAO Architecture** We design and justify an infrastructure for DAOs which is completely decentralized and scalable. To achieve this, we propose a set of technologies and principles that must be followed. In particular, we separate the settlement mechanism and validation of rules using multi-signature and thresh-hold signature schemes.
2) **Music DAO: a true decentralised DAO** We design and implement a real-world DAO that revolves around the music industry using the proposed infrastructure. We use a combination of networks, including the TU Delft created IPv8, to create a music platform where artists can share music and receive funds from a flexible DAO crowdfund structure. This DAO runs on smartphones

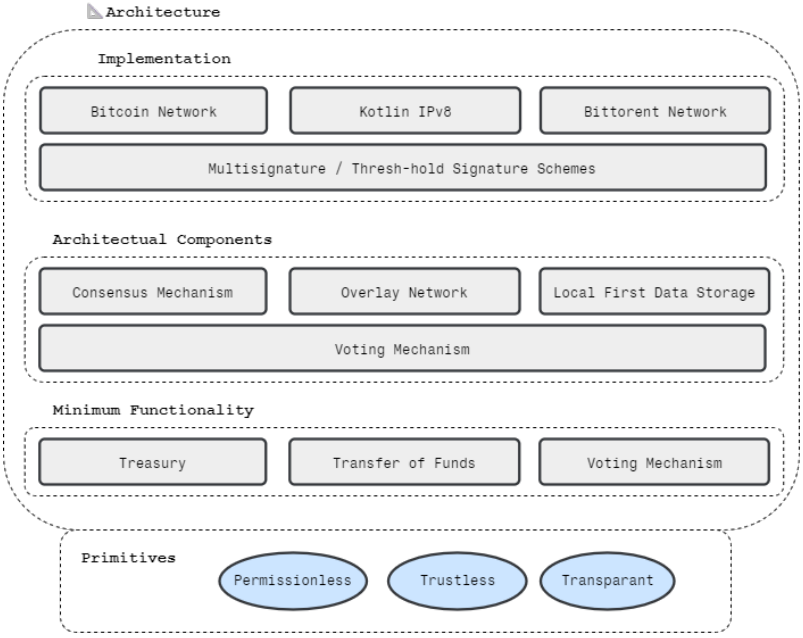

Fig. 1. The complete architecture of minimal DAO

only, has no central components and is deployed on the Android Play store.

3) **Evaluation** To evaluate the proposed infrastructure and implementation, we perform a real-life deployment test amongst a set of participants who work closely with DAOs. In addition, we perform a set of performance tests on our voting and joining mechanism to see assess the performance in a real-world deployment. The results of these tests provide insights into the feasibility and effectiveness of our proposed architecture and implementation.

II. Problem Description

The goal of this study is to develop and deploy an academically pure decentralised DAO. We define a DAO as *a mechanism for economic activity by an unbounded group of people in a competitive environment devoid of infrastructure, leadership, and legal centralized authority.* An organisation which relies on no central intermediary nor central authority and one which is truly unstoppable.

In DAOs the rules are transparent and enforced by an underlying decentralized protocol, such as a public blockchain. The rules of such organizations can be changed collectively by its members through the voting in a governance protocol. While such organizations are autonomous to an extent, they will still rely on human individuals to perform certain tasks. A alternative recent definition proposed by Vitalik, one of the founders of Ethereum, for DAOs is ït is an entity that lives on the internet and exists autonomously, but also heavily relies on hiring individuals to perform certain tasks that the automaton itself cannot do [dao˙blog˙foundation].

The need for pure academic decentralisation arises from the fact that any deviation from this leads to the mechanism inheriting the problems associated with centralized traditional organizations. In traditional organizations, individuals work towards a common objective, but the rules are enforced by a central authority. Third-parties such as institutions, large technology companies, governments, and legal systems ensure that individuals can trust one another and cooperate, providing efficiency gains through their top-down control. However, their interests may not align with the interest of the participants. They may alter the rules in alignment with their own interest or not follow them at all. Even if participants have some influence on this process, it often is outdated and slow (democracy) or relegated to a select wealthy group (share-holders). For example, commercial companies, such as big-tech companies, are ultimately primarily interested in maximizing their own profits. They often use increase user retention rate, at the expense of social and economic problems,. This problem is exacerbated when power becomes concentrated more among a small group of people.

In the field of decentralized autonomous organizations (DAOs), developing a mechanism that simultaneously achieves trust, pure academic decentralization, and scalability is a major challenge. Real DAOs only exist in theory. Every technology claiming to be a DAO has central points of control and critically relies on central servers. Bitcoin and Bittorrent are the only examples of technology stacks which are not reliant on central infrastructure. Numerous startups claim to offer a DAO with decentralisation. To date, all DAOs are still centralised to some extend. The problem is to actually engineer what has been dubbed the future of the firm. The challenge is to incrementally realise a new organisational method to coordinate socio-economic activities. In theory a true DAO will be more efficient than a traditional company, replace

middleman with code, and scale beyond any work-from-home company operating on informal email exchanges. In principle, a DAO should be able to replace current Big Tech companies. This requires scalability beyond 1 billion contributing users. Irrefutable proof that a decentralised DAO is possible is the first near-term problem.

III. Related Work

The concept of DAOs in academia is relatively new, it has mostly been developed by open source developers in the blockchain sphere. One of the first deployed and successfully used DAOs was created in 2016 by Christoph Jentzsch and was called T̈he DAO.̈ The goal of the project was to create a new business model for non-profit enterprises. With an internal capital of 150 million
USD from 11.000 investors at its peak, it was extremely large for its time. It however suffered from an exploit in the smart contract [dao˙memorial], after which the Ethereum blockchain was forked to return the money to investors.

There has been considerable effort invested in observing and researching the phenomenon of deployed DAOs. Shuai et al. have developed a comprehensive framework for DAOs that identifies their characteristics, problems, implementations, and upcoming trends [8836488]. In addition, they suggest a five-layer architecture for DAOs. They do not, however, give a concrete implementation of such a DAO utilizing the design.

Hassan et al. conducted a similar study with the objective of identifying the largest unresolved issues in DAO research [hassan2021decentralized]. They pose the questions of which DAO layers should be decentralized, to what extent a DAO should be autonomous, and whether a DAO should be considered a legal entity. The identification of these obstacles eases the entry of new researchers into the field.

IV. A Simple DAO Architecture

We propose a generic and simple as possible architecture for DAOs. We deliberately remove all unnecessary features and complexity in order to provide a flexible and strong building block. Our building block represents a milestone within the evolution of actual DAO realisations: it is the first to achieve hyper decentralisation. Our minimal function decomposition leads to the following three arcitectual principles, the minimal functionality a DAO handling activity should have and the accompanying components which should be implemented.

A. Architectual Principles

All accompanying components should adhere to these architectural principles in order to satisfy the definition of a decentralized autonomous organization.

1) Trustless: Any decision made in the organization should not depend on any third-party or intermediary. The trust that the decisions are created in a fair manner according to a set of voting rules and the execution of the decisions should be established through cryptographic, verifiable means.

2) Permisionless: Any person should have the opportunity available to participate or access in the organization, without needing any approval of intermediaries. They should not be discriminated based on factors which are not relevant for the workings of the DAO. This does however mean that members in the organization can still collectively decide to block or not allow a person in the organization.

3) Transparent: All information regarding the organization, its decision making process and decisions made should be available to access for everyone, inside and outside the organization. Transparancy is important to instill confidence that the other principles are adhered to, since they can be verified.

B. Architectual Minimum Functionality

The DAO must have a minimum set of functions which provide the ability for participants to coordinate economic activity among each other.

1) Treasury: There must be some internal capital by which activities can be funded with. There must be a way for people to join the treasury.

2) Transfer of Funds: There must be a way that the participants can spend from the treasury

3) Voting Mechanism:

C. Architectural Components

1) Consensus Mechanism: A secure and decentralized blockchain is essential to enable participants who do not trust each other to coordinate econimic activity. The blockchain acts as a foundation of trust upon which participants rely to enforce the existing rules of the DAO and possibly also provide a mechanism to change the rules according to a set of meta-rulesm, i.e. a vote to change the rules. It is important that such a blockchain must have the capabilities for validating transactions using at-least multi-signature and thresh-hold signature schemes in order to facilitate off-chain transaction settlements.

A blockchain network is a network wherein participants come to consensus on a set of transactions. The network ensures the 1) validity and 2) ordering of the transactions. Transactions are grouped in blocks, which contain a set of transactions and the hash of the previous block. This makes it hard for the chain to be tampered with. In order to agree on the same chain (ordering of transactions), consensus mechanisms are used. These are a collection of rules and in combination with financial incentives to determine which chain is favored and thus which ordering is used. In the case of Bitcoinm Proof-of-Work is used, where the chain with the most work is preferred over the others.

2) Local First Data Storage: A decentralized data storage solution is required for the effective storage of digital assets which are located in the DAO. These assets may include are media files or other documents. Due to their large size and storage requirements, it is not feasible to replicate these assets entirely on every node in a blockchain network. The validation of these assets may not necessarily require complex rules, such as those used for validating normal transactions. In this

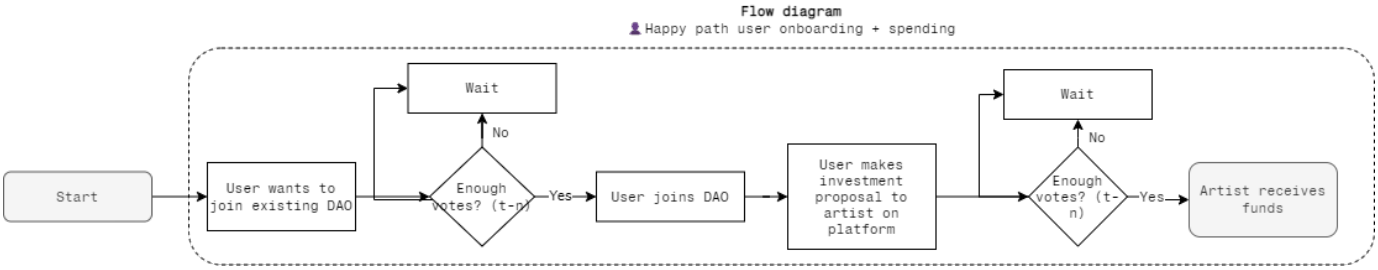

Fig. 2. Spending process

approach, nodes on the network run by participants themselves will host and store the assets.

3) Overlay Network: A peer-to-peer communication solution is necessary for enabling individuals to effectively communicate with each other and coordinate activities without intermediaries. This includes both protocol-level communication, as well as communication related to the organization's internal operations. The creation and dissemination of proposals for instance must be communicated among all members. This information however does not necceserily need to be stored in an immutable blockchain, since there is no relevant double-spending attack possible. Instead, a peer-to-peer communication solution would be sufficient for transmitting information that does not need to be permanently stored.

4) Voting Mechanism: A voting mechanism is necessary in order to facilitate decision-making within in a DAO and allowing participants to come to reach on consensus on decisions that require a vote. This includes decisions on modification of existing rules, and decisions regarding current rules, such as the election of new members. The mechanism should be transparant and accesible to all members. The design of meta-rules should also be fair, however the definition of fairness is subjective and varies depending on the context and organization. This is still an unsolved problem and subject to ongoing research.

We propose a voting mechanism based on thresh-hold signature schemes. Thresh-hold signatures are a signature scheme where a minimum amount of partial signatures are combined in order to create a valid signature for a public key over a message. Each member possesses a shared public key. A secure Distributed Key Generation (DKG) protocol generates this key collectively using a predetermined threshold value. Members hold their respective portions of the corresponding private key. To sign a message, members of a t-n must participate in a thresh-hold signature signing protocol. A collective decision is simply the signing of an arbitrary message, since implicitly t-n members are required to sign a message that indicates t members have agreed on a proposal for a decision.

The implicit governance structure exhibited here is founded on the ownership of private key shares. A one-token-one-vote [weyl2022decentralized] model can be implemented using sybil-resistance mechanisms. In the absence of this restriction, a single user can create sybils to acquire additional shares based on the required criteria for membership. This can be desirable if, for instance, the members of the DAO wish to incentivize greater participation in the DAO (financial or otherwise), which can be rewarded with additional private key shares.

V. A Scalable Voting Mechanism

In order to make decisions among a large number of participants possible, it is essential that there is some mechanism in place which off-loads the work from the blockchain. A typical blockchain which is actually decentralized and secure currently still has a small throughput. A trivial solution would be casting every vote in a proposal as a transaction on the blockchain. This would quickly become infeasible if the number of participants increase,.

Our proposed scaling solution aims to address the issue of scalability by avoiding the need for transitioning between complex smart contract states on a blockchain with global consensus for making decisions. Instead, we leverage the use of threshold signature schemes among the DAO participants to achieve consensus on what state changes and decisions should be made. The key idea is only the relevant participants should validate whether the state transition rules have been followed, by participating in the group signature scheme for a particular proposed transaction. This approach reduces the complexity and computational requirements while still ensuring that decisions are made in a decentralized and trustless manner. By reducing the reliance on global consensus, we can improve the scalability and efficiency of the platform.

A. Blockchain Model

We make assumptions about how our blockchain works and provide some formal specification based on Al-Bassam's work [al2019lazyledger]. We assume a blockchain model consisting of blocks $b_0, b_1 ..., b_n$. Every block contains a header h_i and a set of transactions $T_i = \{t_0 ... t_n\}$. This header contains a merkle root m_i of the set of transactions T_i.

B. Voting Mechanism OLD

A DAO DAO_i consists of a ordered set of signed transactions $T_{ivalidated}$ which must be published on a blockchain. Let $state(T_{ivalidated})$ be the state of the DAO at some point in time. This function determines both the valid set of participants currently in the DAO $P_i = p_0 ... p_n$ and the current value of the treasury. The DAO also consists of a set of not yet signed

transactions $T_{inot\_validated}$. The current treasury content of a DAO is equal to the unspent UTXO of the set of validated ordered signed transactions $T_{ivalidated}$, akin to a normal user wallet.

In the most minimal form, a DAO consists of a set of signed transactions which are published

the most minimal form, we define a DAO as a set of signed transactions which are published in an ordered manner on a secure blockchain. A transaction consists of a message, a public key, and a valid signature created with the corresponding private key. We define the last transaction in this set as t_n.

$$DAO_i \ = \ \{ \ t_1, \ ..., \ t_n \}$$
$$t_i \ = \ \{ \ m_i, \ pk_i, \ s_i \ }$$

In order to transition from one state to another, a new signed transaction must be published to the blockchain according to the rules specified in $verify$, which takes in the current DAO state and the new transaction.

$$verify(DAO_i, \ t_{n+1}) \ = \ true$$
$$DAO_{i+1} \ = \ \{ \ t_1, \ ..., \ t_n, \ t_{n+1} \ }$$

We define the $verify$ rule according to the two base cases of capital management in the DAO based on the UTXO model. In the UTXO model, transactions have inputs and outputs. In order to spend an input, a valid signature must be created over the a message which spends the input to a new address.

All of the transactions in the DAO are signed by group signatures. This signature is created by the members collectively, using a n-k thresh-hold signature scheme. In order to create a valid signature, additional information thus is required.

$threshhold(s_1, \ ..., \ s_n, \ pk_i, \ n, \ k) \ =$ true iff n valid signatures are given

$$members \ = \ \{ \ p_1, \ ..., \ p_n \ }$$
$$p_i \ = \ \{ \ pk_i, \ sk_i \ }$$

The on-chain state of the DAO can be derived from the set of signed transactions. This state consists of the currently used public key pk_i and the current total capital c_i.

$$state(DAO_i) \ = \ \{ \ daokey_i, \ c_i \ }$$

The currently used public key is equal to the public key used to sign the latest transaction.

$$daokey_i = \ pk_n \ \in \ t_n$$

C. Voting Mechanism New

DAO State

In the most minimal form, we define a DAO as a set of signed transactions $\{ \ t_1, \ ..., \ t_n\}$ which are published in an ordered manner on a secure blockchain. A transaction consists of a message m_i, a public key pk_i, and a valid signature sk_i created with the corresponding private key. We define the last transaction in this set as t_n. The messages needs to be a valid transaction for the blockchain that is used.

$$DAO_i \ = \ \{ \ t_1, \ ..., \ t_n \}$$
$$t_i \ = \ \{ \ m_i, \ pk_i, \ sk_i \ }$$

All of the transactions are signed with a shared public key created by a thresh-hold signature scheme, of which the share keys are shared among the participants. The parameters of this signature scheme can be changed, a higher thresh-hold will require more participants to participate which increases the effort needed to commit fraud.

In this set of transactions, multiple types of information pertaining to the DAO can be stored. Most importantly, in a UTXO based blockchain, the transactions can lock up some financial value: the DAO treasury. The total locked up value c_i is equal to the treasury amount.

$$state(DAO_i) \ = \ \{ \ daokey_i, \ c_i \ }$$

Each participant in the DAO corresponds to a particular private key share.

DAO Transitions

In order to make a decision in the DAO, its state needs to transition from one state to another. A new group signed transaction must be published to the blockchain. Anyone can propose to sign a new transaction. This transaction must follow 2 rules:

$$DAO_i \ \rightarrow \ transition(DAO_i) \ \rightarrow \ DAO_{i+1}$$

1) It must be a valid transaction on the blockchain.
2) It must follow a set of client-side rules if specified, which all participants must verify. We define these rules as a function: $verify(DAO_i, \ t_x) \ =$ TRUE OR FALSE, which depends on the current DAO state and the new transaction.

Every transaction in the DAO state logically already follows (1), if it is published on a blockchain it is verified by its consensus mechanism. In addition to this, to follow (2), we rely on the other participants. If a thresh-hold number of participants commit fraud and sign while $verify(DAO_i, \ t_x) \ = \ FALSE$, the transaction will still be executed on the blockchain.

The current group signature key of the DAO key is defined as the key the last transaction is signed with, as that transaction is guaranteed to include any newly joined member.

$$daokey_i = \ pk_n \ \in \ t_n$$

Based on these rules, there are two main ways for the state to transition:

1) Treasury re-allocation: this transaction transfer funds from the DAO treasury to an abritrary address, to fund some type of economic activity.

a) (1) a valid transaction from old outputs to a target input, with the rest of the funds sent to the DAO treasury
b) (2) verify is empty

2) Thresh-hold signature inclusion: this transaction adds a new members to the DAO, by moving all the treasury funds from old locked up outputs, to a single new locked up output which is signed with a group signature where the new members is included. The new member should send sufficient coins as a entrance fee to the DAO treasury in the transaction.
   a) (1) a valid transaction where all funds are sent to the DAO treasury using the new key
   b) (2) verify should check whether the new members sent sufficient coins to the DAO treasury, before signing

D. Security Model

In this proposed architecture, the security model differs significantly from that of a smart contract platform run on a blockchain with global consensus. In a traditional blockchain, transactions are validated according to a set of rules that are determined by a group of miners. If 51% of all miners agree to, for example, commit fraud, it is possible for them to do so. In other words, the validator set consists of all the miner nodes in the network and the accompanying hash-rate.

In contrast, our security model rests on the number of particiapnts in the DAO that are part of the group signature group. If 51% of the people (or any other percentage, depending on the n-k threshold) want to commit fraud, it is possible for them to do so. The main advantage of this model is that the complexity of the client-side rules can be arbitrarily complex and is essentially free to compute, since we only need to verify the transaction on the client side. The other nodes in the network, which do not have anything to do with the DAO, do not have to validate the client-side rules. 51% of the DAO members can run the client-side rules, verify their correctness, and if they are valid, participate in the threshold signature scheme. If they do not verify, they can simply not participate, after which no signature will be created.

In this design, we do not rely on advanced turing-complete smart contract capabilities. Instead, we use a blockchain of choice, namely Bitcoin, which is simple and secure, and does not require advanced smart contract capabilities. In this way, we can achieve a high level of security and scalability, while keeping the complexity of the system at a minimum.

VI. Music DAO: a truly decentralised DAO

We have created an implementation of a DAO centered around music using our proposed architecture. This implementation uses all the specified architectual components and adheres to the architectural principles that we have laid out, through which we achieve full academic decentralization. We will describe the functionality of the DAO and which technologies we have used in what way to realise this.

In contrast to many other works in this field, our proposed architecture and implementation of the Music DAO is actually deployed. However, it is important to note that due to the limited time frame in which it was deployed, it may still contain some bugs and not perform optimally. Nevertheless, this live deployment provides valuable insights which can inform future research and development in this area.

1) Overlay Network: kotlin-ipv8
2) Blockchain: Bitcoin
3) Local First Data Storage: BitTorrent and DHT
4) Voting Mechanism: thresh-hold signatures
5) Application Layer: Kotlin, JVM and Jetpack Compose

The implementation is created using Kotlin and Android on the JVM platform. This allows for deployment on the Play Store and accessibility for hundreds of users. Cross-platform mobile application is outside the scope of our use case, due to many of our libraries not being available, such as our chosen overlay network IPv8. Android additionally provides extensive service APIs that allow services to continously run in the background, allowing for the upkeep of the network.

We chose to limit our implementation to smartphones only for several reasons, all of which align with our principle of creating a permissionless system. Additionally, smartphones have a lower barrier to entry, as almost everyone has a phone, especially in developing countries, and not everyone has a PC. The zero-architecture server stack also supports the idea that smartphones are the superior device for maintaining and using P2P networks.

The DAO consists of two main components: the music platform, and the crowdfund platform. The music platform enables the dissemination and availability of music and it's meta-data. The crowdfund platform enables the collective management of funds by listeners to support musicians.

The requirements for the music platform component of our implementation are as follows:

1) Music Publishing: Artists can publish music to the platform. Published music is shared on the IPv8 peer-to-peer overlay network. The music is first encoded to the correct format and an accompanying torrent file/torrent meta-data is created for the formatted data. This meta-data is then published on the personal trustchain of the user and gossiped around to other users. At the same time, the torrent file is published on the BitTorrent DHT network and is available to seed from the phone. Additional meta-data such as album art cover is also included in the published music and is displayed in the GUI.
2) Music Listening: Different users on the network can receive the signed trustchain blocks and add them to their local storage of published music. They use the meta-data in the block to query the DHT network and download peer information to download the torrent from seeders. After the music has been downloaded, everything is verified, and the listener can listen to the music with the accompanying data.

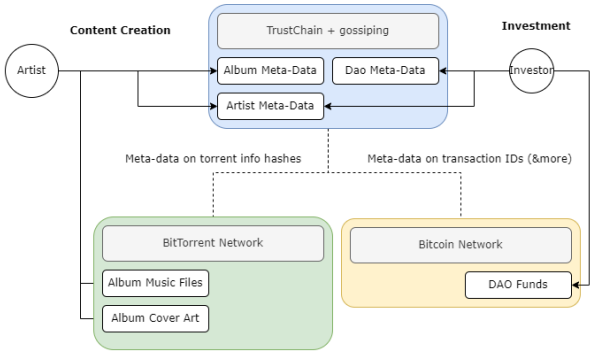

Fig. 3. Architectural components of the Music DAO

Listeners keep seeding a part of their music according to some type of a set of rules, for instance based on popularity. The optimization of this process is out of scope for this work. For this implementation, the most popular music and a selection of the less popular music (tail-end) is randomly selected and seeded.

3) Reputation: Through the use of linked trustchain blocks, there is a notion of reputation. A person with a public key can demonstrate that they have consecutively published music on the network, indicating that they are an active musician.
4) Content Discovery
5) Content Search

The use of BitTorrent in our implementation is due to its reliability and decentralization. BitTorrent has a proven track record of stability and security, with 19 years of incremental improvements to the protocol. While other technologies such as IPFS offer similar functionality, BitTorrent is more widely adopted and has a larger user base.0 By extracting torrent info hashes from the platform, we can facilitate mass seeding of the network, or allow users to download content using popular torrent clients without the need for our application. The use of the accompanying Distributed Hash Table (DHT) network in our implementation is to remove the need for tracker servers, which are centralized and may be taken down by law enforcement agencies. DHT networks are much harder to take down and only require a simple bootstrap node, which can be any node with sufficient knowledge, after which you can get almost any swarm info about a info-hash in the network.

VII. Performance Evaluation

In the previous sections, we have discussed the infrastructure of our DAO and the design and implementation of the Music DAO. In this section, we will perform both a qualitative and quantitative evaluation of our DAO in terms of usability and performance. We deploy our DAO on the Android play store and do a real life usability test amongst a set of participants who work closely with DAOs. Then we do an experimental analysis on the performance of the multi-signature voting scheme.

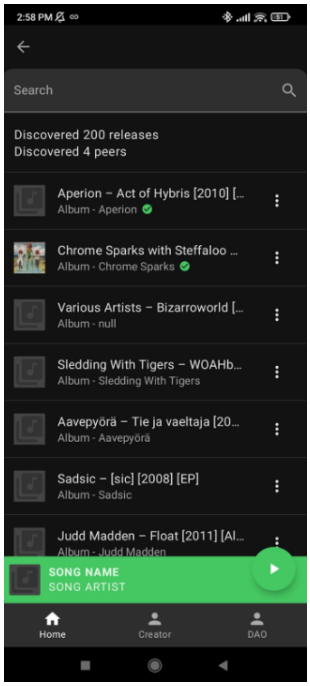

Fig. 4. Homepage of the Music DAO

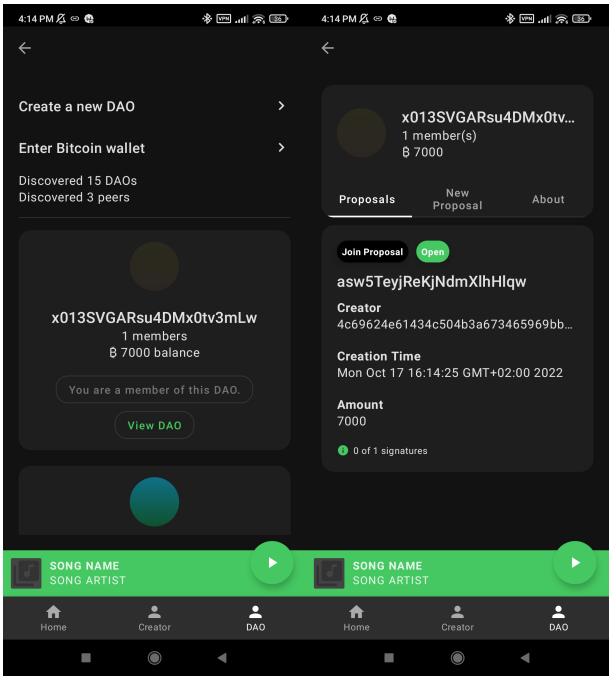

Fig. 5. All available DAOs in the Music DAO

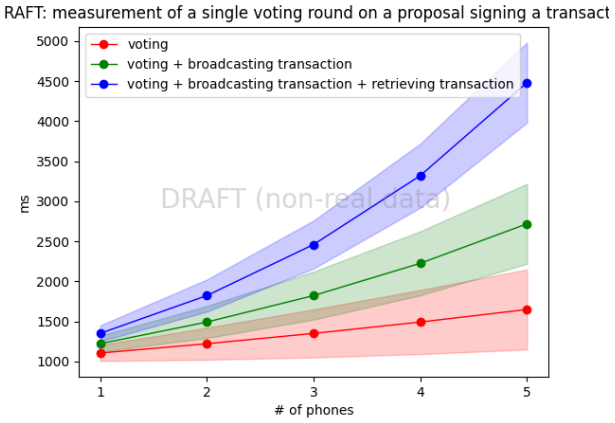

Fig. 6. Performance of our voting mechanism

A. Real-life deployment test

In order to evaluate the usability of our tests, we have additionally do a real-life deployment test. Participants are given a presentation on DAOs and were subsequently provided access to the application, which is deployed on the Google Play Store. This allows us to gather valuable insight on the usability and user experience of our solution in a real-world setting.

B. Performance Experiment

For the performance experiment, we wish to determine whether the DAO can scale in a deployed, real-world environment. Specifically, we wish to examine how the voting mechanism scales with the number of voters. In a deployed environment, many factors are at play, including phone performance, network type and connectivity, and implementation of the various technology layers. With these experiments, the interaction between the IPv8 overlay network, the multi-signature scheme, and the Bitcoin network will be evaluated.

The initial experiment will utilize actual phones. To measure the time between the creation of a DAO and the addition of a new member, a benchmark script is developed. All existing DAO members will be required to sign the new members into the DAO.

The second experiment will be done locally using a set of local IPv8 nodes running on a computer.

VIII. Conclusion

IX. Acknowledgment

References

[1] [uniswap˙volume] "Uniswap combined metrics." [Online]. Available: https://dune.com/danrobinson/uniswap-combined-metrics

[2] [cong2021decentralized] L. W. Cong, Z. He, and J. Li, "Decentralized mining in centralized pools," *The Review of Financial Studies*, vol. 34, no. 3, pp. 1191–1235, 2021.

[3] [dao˙blog˙foundation] E. Foundation, "Daos, dacs, das and more: An incomplete terminology guide." [Online]. Available: https://blog.ethereum.org/2014/05/06/daos-dacs-das-and-more-an-incomplete-terminology-guide/

[4] [dao˙memorial] "A call for a temporary moratorium on the dao." [Online]. Available: https://hackingdistributed.com/2016/05/27/dao-call-for-moratorium/

[5] [8836488] S. Wang, W. Ding, J. Li, Y. Yuan, L. Ouyang, and F.-Y. Wang, "Decentralized autonomous organizations: Concept, model, and applications," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 5, pp. 870–878, 2019.

[6] [hassan2021decentralized] S. Hassan and P. De Filippi, "Decentralized autonomous organization," *Internet Policy Review*, vol. 10, no. 2, pp. 1–10, 2021.

[8] [al2019lazyledger] M. Al-Bassam, "Lazyledger: A distributed data availability ledger with client-side smart contracts," *arXiv preprint arXiv:1905.09274*, 2019.

[7] [weyl2022decentralized] E. G. Weyl, P. Ohlhaver, and V. Buterin, "Decentralized society: Finding web3's soul," *Available at SSRN 4105763*, 2022.

[pouwelse˙towards˙2020] J. Pouwelse, "Towards the Science of Essential Decentralised Infrastructures," in *Proceedings of the 1st International Workshop on Distributed Infrastructure for Common Good*. Delft Netherlands: ACM, Dec. 2020, pp. 1–6. [Online]. Available: https://dl.acm.org/doi/10.1145/3428662.3429744

[buterin2014next] V. Buterin *et al.*, "A next-generation smart contract and decentralized application platform," *white paper*, vol. 3, no. 37, pp. 2–1, 2014.

[8962150] Q. Zhou, H. Huang, Z. Zheng, and J. Bian, "Solutions to scalability of blockchain: A survey," *IEEE Access*, vol. 8, pp. 16 440–16 455, 2020.

[noauthor˙state˙nodate] "The state and future of Decentralized Autonomous Organizations (DAOs) including 6 leading examples - Ross Dawson." [Online]. Available: https://rossdawson.com/futurist/companies-creating-future/top-decentralized-autonomous-organizations-dao/

[wang˙decentralized˙2019] S. Wang, W. Ding, J. Li, Y. Yuan, L. Ouyang, and F.-Y. Wang, "Decentralized Autonomous Organizations: Concept, Model, and Applications," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 5, pp. 870–878, Oct. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8836488/

[zhou˙solutions˙2020] Q. Zhou, H. Huang, Z. Zheng, and J. Bian, "Solutions to Scalability of Blockchain: A Survey," *IEEE Access*, vol. 8, pp. 16 440–16 455, 2020. [Online]. Available: https://ieeexplore.ieee.org/document/8962150/

[xu˙taxonomy˙2017] X. Xu, I. Weber, M. Staples, L. Zhu, J. Bosch, L. Bass, C. Pautasso, and P. Rimba, "A Taxonomy of Blockchain-Based Systems for Architecture Design," in *2017 IEEE International Conference on Software Architecture (ICSA)*. Gothenburg, Sweden: IEEE, Apr. 2017, pp. 243–252. [Online]. Available: http://ieeexplore.ieee.org/document/7930224/

[komlo2020frost] C. Komlo and I. Goldberg, "Frost: flexible round-optimized schnorr threshold signatures," in *International Conference on Selected Areas in Cryptography*. Springer, 2020, pp. 34–65.

[githubBipsbip0340mediawikiMaster] "bips/bip-0340.mediawiki at master · bitcoin/bips — github.com," https://github.com/bitcoin/bips/blob/master/bip-0340.mediawiki, [Accessed 30-Jun-2022].

[stallings1987handbook] W. Stallings, *Handbook of computer-communications standards; Vol. 1: the open systems interconnection (OSI) model and OSI-related standards*. Macmillan Publishing Co., Inc., 1987.

[coindeskSwedenDiscussed] J. Schickler, "Sweden, EU Discussed Bitcoin Proof-of-Work Ban: Report — coindesk.com," https://www.coindesk.com/policy/2022/04/21/sweden-eu-discussed-bitcoin-proof-of-work-ban-report/, [Accessed 30-Jun-2022].

[heliumHeliumx2013] "Helium x2013; Introducing The Peopleapos;s Network — helium.com," https://www.helium.com/, [Accessed 30-Jun-2022].

[nakamoto2008bitcoin] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Dec 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[otte2020trustchain] P. Otte, M. de Vos, and J. Pouwelse, "Trustchain: A sybil-resistant scalable blockchain," *Future Generation Computer Systems*, vol. 107, pp. 770–780, 2020.

[torbensen2019tuning] A. Torbensen and R. Ciriello, "Tuning into blockchain: Challenges and opportunities of blockchain-based music platforms," in *Twenty-Seventh European Conference on Information Systems (ECIS2019), Stockholm-Uppsala, Sweden*, 2019.

[gervais2016security] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the security and performance of proof of work blockchains," in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 2016, pp. 3–16.

[scharnowski2021bitcoin] S. Scharnowski and Y. Shi, "Bitcoin blackout: Proof-of-work and the centralization of mining," *Available at SSRN 3936787*, 2021.

[beikverdi2015trend] A. Beikverdi and J. Song, "Trend of centralization in bitcoin's distributed network," in *2015 IEEE/ACIS 16th international conference on software engineering, artificial intelligence, networking and parallel/distributed computing (SNPD)*. IEEE, 2015, pp. 1–6.

[jentzsch2016decentralized] C. Jentzsch, "Decentralized autonomous organization to automate governance," *White paper, November*, 2016.

[chohan2017decentralized] U. W. Chohan, "The decentralized autonomous organization and governance issues," *Available at SSRN 3082055*, 2017.