Decentralized Autonomous Organizations en Blockchain: Análisis y Visualización
Decentralized Autonomous Organizations on Blockchain: Analysis and Visualization

Trabajo de Fin de Máster
Curso 2020–2021

Autor
Youssef El Faqir El Rhazoui

Director
Javier Arroyo Gallardo

Máster en Ingeniería Informática
Facultad de Informática
Universidad Complutense de Madrid

Decentralized Autonomous Organizations en Blockchain: Análisis y Visualización
Decentralized Autonomous Organizations on Blockchain: Analysis and Visualization

Trabajo de Fin de Máster en Ingeniería Informática
Departamento de Ingeniería del Software e Inteligencia Artificial

Autor
Youssef El Faqir El Rhazoui

Director
Javier Arroyo Gallardo

Convocatoria: *Febrero 2021*
Calificación: *8.5*

Máster en Ingeniería Informática
Facultad de Informática
Universidad Complutense de Madrid

5 de febrero de 2021

Resumen

Decentralized Autonomous Organizations en Blockchain

La tecnología blockchain se ha convertido en un paradigma para construir sistemas descentralizados, que se autoorganizan y no requieren una autoridad central para su gestión. Sin embargo, blockchain va más allá de las finanzas y se ha aplicado en muchos otros campos. Ese es el caso de la gobernanza descentralizada, dando lugar a la conocida como "Decentralized Autonomous Organization"(DAO). Las DAOs han surgido como una nueva forma de gobernanza colectiva y descentralizada, donde los usuarios pueden organizarse apoyándose en la tecnología blockchain. Además cada acción realizada en esta queda registrada en la blockchain e implica un coste asociado.

En este trabajo, presentamos el concepto de DAO, viendo para qué se utilizan y las principales plataformas disponibles para crearlas. Repasamos el estado del arte actual apoyándonos en los dashboards para DAOs como herramientas para mejorar el conocimiento sobre ellas. Sin embargo, no hay muchas de estas herramientas disponibles debido a la dificultad de acceder a datos de la blockchain. A su vez, proponemos nuestra propia herramienta de código libre, DAO-Analyzer, para esa tarea. Investigamos los entresijos de estas DAOs, haciendo un análisis comparativo de 48,886 usuarios agrupados en 1,935 DAOs de las principales plataformas DAO (Aragon, DAOhaus y DAOstack). Los resultados obtenidos muestran diferencias significativas en estas plataformas siendo Aragon la plataforma más grande de ellas y la que muestra un crecimiento sostenido. Sin embargo, la plataforma DAOstack muestra un claro estancamiento, sin apenas actividad.

En un segundo paso, estudiamos cómo algunos factores externos pueden influir en los comportamientos de una DAO. Para ello, nos enfocamos en el aumento de las tarifas de transacción (gas) desde mayo de 2020 en la red Ethereum. Buscando la influencia del gas en la actividad de las DAOs. Hemos analizado 15.977 transacciones de todos las DAOs de los principales ecosistemas y hemos usado herramientas econométricas con series temporales diarias del valor medio de la tarifa y las acciones DAO, con el objetivo de analizar la causalidad entre ambas. Los resultados muestran solo una tenue influencia menor del precio del la tarifa (gas) en la actividad de los usuarios de las DAOs. La insensibilidad de la actividad al precio de la tarifa es una anomalía en un mercado supuestamente autorregulado que debería solucionarse en el futuro.

Palabras clave

Blockchain, Decentralized Autonomous Organization, Aragon, DAOhaus, DAOstack, Gas, Dashboard, Causality

Abstract

Decentralized Autonomous Organizations on Blockchain: Analysis and Visualization

Blockchain technology has become a paradigm to build decentralized systems, which are self-organized and do not require a central authority for their management. However, blockchain span beyond Finance, and it was applied in many other fields. That is the case of decentralized governance, giving rise the well-know "Decentralized Autonomous Organization" (DAO). DAOs have emerged as a new form of collective and decentralized governance, where users may organize themselves but relying on blockchain technology, where every performed action is registered in it and implies an associated cost.

In this work, we introduce the concept of DAO, seeing what it is used for and the main platforms available to create a DAO. We review the current state of the art relying on dashboards for DAOs as tools to improve knowledge about them. However, not many of these tools are available due to the difficulty of accessing blockchain data, and the few that exist are proprietary. In turn, we propose our own open source tool, DAO-Analyzer, for this task. We investigated the ins and outs of these DAOs, doing a comparative analysis of 48,886 users grouped in 1,935 DAOs from the main DAO platforms (Aragon, DAOhaus and DAOstack). The results obtained show significant differences in these platforms, Aragon being the largest platform of them and the one showing sustained growth. However, the DAOstack platform shows a clear stagnation, with hardly any activity.

In a second step, we study how external factors can influence DAO's behaviours. For that, we focus on the transaction fees (gas) surge from May 2020 in the Ethereum network. In order to see if gas prices influence the DAO activity, we have analyzed 15,977 transactions from all the DAOs of the main ecosystems, and used econometric tools with daily time series of the average fee value and the DAO operations, in order to analyze the causality between them. Our results show just a minor influence of the fee (gas) price and the activity of DAO users. The insensitivity of the activity to the fee price is an anomaly in a supposedly self-regulated market that should be solved in the future.

Keywords

Blockchain, Decentralized Autonomous Organization, Aragon, DAOhaus, DAOstack, Gas, Dashboard, Causality

Contents

List of figures

List of tables

Chapter 1

Introduction

Since the surge of the cryptocurrency Bitcoin in 2008 (Nakamoto and Bitcoin, 2008), the technology that powers it, blockchain, has been growing in popularity and adoption. This technology enables new forms of online services which rely on a decentralized infrastructure. Some blockchain properties are: transparency of the operations performed; the immutability of the transactions; and resilience since the chain of blocks that contain the operation data is ensured through cryptographic means (Underwood, 2016; Filippi and Wright, 2018).

The blockchain sphere has evolved and beyond bitcoin and cryptocurrencies, the most widely used blockchain project is the Ethereum computing platform. Ethereum enables developers to write code in top of the general-purpose blockchain. This code or "smart contract" is used to build distributed apps, which are used in many fields (e.g., IoT, bank services, personal identification, etc.), without relying on a third-party to host (Filippi and Hassan, 2016).

These smart contracts are also being applied to governance with the advent of Decentralized Autonomous Organizations or simply DAOs. A DAO is a virtual entity with a certain number of members and non-central management, which have the right to collectively allocate, use, or spend the entity's resources. DAOs have brought a new way of organization, where people with common goals can join to achieve them under a virtual organization (e.g., companies, cooperatives or non-profit organizations). The entity's code is used to provide action autonomy to the DAO (e.g., employ people, make payments, buy/sell resources to the community) (Buterin et al., 2014).

Actions in a DAO are driven by the DAO community consensus, usually reached by a vote. Thus, voting conforms the main interaction between DAO members in the blockchain. But, they also use other non-blockchain tools (e.g., forums, chats) to interact among them.

The way to create a DAO has changed over time: firstly, writing the entire smart contract's code; But, now, many DAO platforms has arrived to ease this task, giving a DAO template which is configurable. Those are *Aragon* with 58,681 users grouped in 2,069 DAO communities, *DAOhaus* with 1,445 members from 225 DAOs, and finally, *DAOstack* which has 12,194 users from 59 DAO communities. Those numbers are currently growing.

Thus, we can observe other large commons-based peer production projects such as

Wikipedia or the Linux kernel (Benkler, 2006). Still, these communities face multiple challenges, such as the emergence of bureaucratization and elites (Shaw and Hill, 2014). Blockchain enthusiasts claim DAOs can reduce the cost of large-scale decentralized cooperation, incentivize cooperative behaviour, increase participation, and facilitate open democratic organizations (Rozas et al., 2018; ara). Hence, community aims, technologies developed to enable them, and systems for self-organization are attractive and active matters in the study of human-computer interaction (HCI). However, the DAO phenomenon is under-studied in the current literature due to several factors: its novelty and the lack of tools to get meaningful information about them.

In order to explore that, this master thesis aims to clear the way for those novelty blockchain communities, which have been growing in recent years and will keep growing following on. However, without a better knowledge about them, how the underlying technology affects them, and how is the software design they use, these DAO communities are doomed to disappear. So, to better understand them, we developed an interactive dashboard with those aims, and we go a step further exploring the causes which drive the DAO's activity.

This work is part of the project Chain Community (grant no.: RTI2018-096820-A-100) funded by the Spanish Ministry of Science and Innovation. In a first research (El Faqir et al., 2020), we described what is a DAO, the main DAO platforms, and we analyzed a specific DAO. In a second iteration (Faqir-Rhazoui et al., 2021b), we validated the voting system of one of the main DAO platforms. However, those works are not part of this work. For this thesis we cover two new kinds of research: first ties to how blockchain transaction fees affect DAO's activity (Faqir-Rhazoui et al., 2021a); The second is currently in-develop, and relies on a comparison between the main DAO platforms.

1.1. Objectives

The main aim of this master thesis is to define, characterize, and analyze the main factors which affect these blockchain communities. Firstly, we propose to analyze and compare the main DAO platforms, giving their main key features and differences. For the previous task, we have to investigate the current state of the art for tools which give meaningful data, information, and analytics about the DAOs. Finally, we also propose to investigate external factors which are susceptible to affect DAO behaviour. The intermediate objectives include:

- Compare the main DAO platforms in several dimensions like growth, activity or the resources they have. This should give us the differences and similitudes of how DAOs are managed in each platform, and how DAOs are influenced by the platform they use.
- Check the state of the art for tools which can help to the previous task. If there is not, then we will develop one which can give us the information needed.
- Analyze external factors which can drastically affect the DAO behaviour. Due to each DAO action performed is registered in the blockchain as a transaction. And, these transactions always have an associated monetary cost, which varies over time. We propose to analyze the causality of transaction costs and the DAO activity.

1.2. Work Plan

In order to achieve the above objectives we have organized ourselves in weekly sprints, iterating on the following list:

1. Weakly meeting to state the week goal from the backlog, and check the previous objective.
2. Analyze the goal requirements (e.g., required tools, APIs to fetch, etc.).
3. Design the problem solution, sometimes using UML diagrams, in other cases simply, writing a sketch down.
4. Software development or solution development.
5. Code testing if the solution was codified. In other case, we use pair reviewing to check the methodology and results.
6. If the solution was in-code, then deploy if it is necessary.
7. Iterate.

To accomplish this work methodology, we use several online tools. We group all the objectives and create a work flow in a Kanban board using Trello. For diagrams and documents, we use Google Drive and its tools (e.g., Docs, Diagrams, etc.). For coding control version, we use GitHub, and GitHub Actions to check the code integrity.

1.3. Memory organization

This work is structured as follows:

- **Chapter 1: Introduction**

  It introduces the master thesis matter, showing the motivation, the main work goals, and how was the work plan to accomplish them.

- **Chapter 2: An Introduction of Blockchain**

  In this chapter, we introduce how a blockchain is and works. Next, we dive in the transaction fees or gas. Finally, we explain what is a DAO, for what they are, and the available platforms to create a DAO.

- **Chapter 3: DAO-Analyzer: A tool for DAO's Metrics Visualization**

  The following chapter gives a brief view of the DAO tool's state of the art. Describes where to retrieve the DAO's data, and explains how is the DAO-Analyzer architecture.

- **Chapter 4: A Quantitative DAO Comparison**

  It uses the previous tool, DAO-Analyzer, to give a DAO comparison in dimensions like growth, activity or voting system.

- **Chapter 5: Effect of the Gas Price Surges on User Activity in the DAOs**

  This chapter analyzes external factors can alter the DAO activity. For that, we create a model to find a relationship between gas prices and the DAO activity.

- **Chapter 6: Conclusions and Future Work**

  This chapter concludes the work with some remarks, gives some indications for future work, and describes the work covered in this master thesis.

Chapter 2

An Introduction of Blockchain

2.1. Blockchain: A brief overview

Blockchain is a distributed ledger, which can be thought as a distributed append-only database with a synchronization mechanism. Like the Internet, the public blockchain has an open infrastructure, and it is not owned or controlled by one central authority. Generally, the ledger is copied in each of the network nodes, and thus can be viewed by all its users (Underwood, 2016; Zheng et al., 2018).

The ledger is a sequence of blocks (hence block-chain) that contains a set of transactions already performed.[1] Each block points to the previous block in the ledger, forming a chain. Figure 2.1 (which has CC BY license[2]) shows an example of this.

When a user wants to add a new transaction to the ledger, the transaction data is verified by the so-called miners. If there is consensus on the new block validity, it is added to the chain in a decentralized process (Filippi and Hassan, 2016; Zheng et al., 2018). This way to validate the transactions grants the blockchain immutability of its past records: nobody can delete and alter the data of the block placed (Hofmann et al., 2017).

The first implementation of the blockchain technology was *Bitcoin*, which is a "crypto-currency", i.e. decentralized digital currency validated through cryptography (Nakamoto and Bitcoin, 2008). After that, thousands of new cryptocurrencies have emerged with their own features (Hu et al., 2019).

A second wave of blockchain applications started with the advent of *Ethereum* in 2013 (Wood et al., 2014), which provides a distributed computing platform and a programming language, *Solidity* (Dannen, 2017). *Solidity* addressed several limitations of the *Bitcoin*'s scripting language, like the lack of Turing-completeness (Vujičić et al., 2018). This has enabled multiple types of decentralised applications (Dapps) and the so called "smart contracts",computational agreements between parties which may be self-executed and self-

[1] In cryptocurrencies, each block holds transactions, i.e. movements of cryptocurrency between accounts. In other more general applications such as Ethereum-based apps (and DAOs), blocks contain operations, akin to typical instructions in a computer program, that need to be executed.

[2] This figure was extracted from: `https://www.researchgate.net/figure/A-simplified-example-of-how-blocks-are-chained-to-form-a-blockchain-Notice-that-each_fig1_332215097`

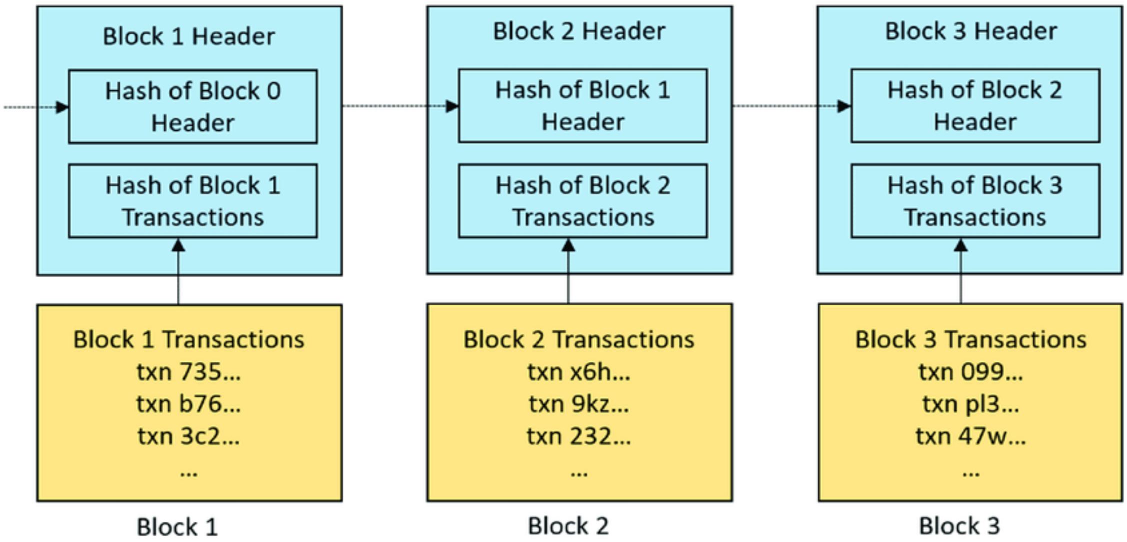

Figure 2.1: Simplified example of how blocks are chained in a blockchain.

enforced.

Those Dapps have been applied in wide-ranging fields (Casino et al., 2019; Metcalfe, 2020). These include financial applications such as general banking services (Cocco et al., 2017), cryptocurrency payment (Cawrey, 2014), or even Decentralized Finance (DeFi) applications, used to get crypto-savings, crypto-loans, or even trade with them (Liu and Szalachowski, 2020). Other fields, like IoT, are using blockchain as a common communication layer (Christidis and Devetsikiotis, 2016). The use of Dapps for community governance enables Decentralized Autonomous Organizations (DAOs), the aim of this.

2.2. Gas: The Ethereum transaction fee

Ethereum keeps a public, shared, and immutable ledger where all the transactions are registered by peers in the network. This fact grants that transactions are truthful. At first, the majority of those transactions were for the Ethereum cryptocurrency. However, nowadays, the Ethereum platform is broadly used to run smart contracts for more complex applications (e.g. DeFi, DAOs, etc.), which also use those transactions (e.g., to save their states in the blockchain). However, in Ethereum, every transaction must pay a fee, called Gas, (Buterin et al., 2014).

The idea behind the gas is to make the users pay for computational resources (e.g., CPU, energy) whenever a transaction has to be processed (Dannen, 2017). So, each time you want to run your program on the blockchain, you must pay a fee for it in order to compensate for the used resources that miners use to carry it out. However, the amount you must pay is not fixed, and it will be tied to the complexity of the code you want to run. For example, a bare transfer may use 21,000 gas, but a more complex transaction (e.g., decentralized finance apps) could increase its need to 1,000,000 gas (Kordyban, 2020b; ethereumprice, 2020).

Gas prices are denoted in GWei, where $1ETH = 1 * 10^9 GWei$. Thus, at the time we want to execute a simple transaction with a cost of 21,000 GWei. Also a market unit price of GWei is set to 50. So, we will need to pay $21,000*50 = 1,050,000GWei$ or $0.00105ETH$ for it. This means that just for a simple transaction we have to pay a fee of 1.15 USD.[3] So then, when a user wants to execute a transaction, he/she needs to take into account the amount of gas it demands and the price of gas at the execution time. The way to calculate the gas cost is analyzing the operations the transaction code uses (e.g., load, store, jump, etc.), where each of them spends an amount of gas (Wood et al., 2014).

On the other side, gas's unitary price (measured in Wei) is quite more complex to set, and it is usually estimated. That is due to the the transaction's demand in the Ethereum network. To better understand this, we have to introduce what is the "block gas limit." Each block is filled with a certain number of transactions, and in turn, this number is tied to the total number of gas which those transactions spend. So, each block has a limit of gas, which transactions can use, and this limit is set by the miners (Sousa et al., 2020). Increasing the block gas limit will also take more time to propagate the changes around the network. If those blocks take more time to be processed, it will also take more time to discover new blocks, decreasing the network's decentralization (0xNick, 2020).

As we previously said, there is no way to know how much gas fee you have to pay for a transaction. Users can set a gas limit to pay for their transactions. However, a low limit may lead to a long time to accomplish the transaction, or even worse, the transaction may not accomplish, and the user loses his/her money. On the contrary, a higher gas limit may derive from an overpriced fee. Due to that, there are many tools (e.g., ETH Gas Station,[4] or Etherchain[5]) to estimate the gas limit you need to set. Usually, those tools give you an estimated price depending on the speed and the cost you need. For example, the "safe low" transaction takes less than 30 minutes to be processed, and it is the cheapest one. The "fastest" transaction takes at most 30 seconds to be processed, but it is the most expensive one (Pierro and Rocha, 2019).

2.3. What is a Decentralized Autonomous Organization?

A formal definition of DAO states that "a DAO is an internet-native entity with no central management which is regulated by a set of automatically enforceable rules on a public blockchain, and whose goal is to take a life of its own and incentives people to achieve a shared common mission.".[6]

Commonly, a DAO is formed by a group of agents which are identified by a unique address. However, at the beginning, the way that these agents ingresses in the DAO, is by investing in the DAO to get an amount (depending on their investment) of the DAO participation (in the blockchain sphere named tokens), which is exactly as a corporation does. DAOs are considered autonomous because they work and are restricted by a set of rules, which are in simple terms, code that runs in the blockchain (e.g., hire people, buy/sell resources for/to the community) and gives the DAO with autonomy action. Therefore, this virtual entity is totally independent of its creators and cannot be influenced from the outside.

[3] The exchange was made on 5th January 2021, where $1ETH = 1097.69USD$

[4] `https://ethgasstation.info/`

[5] `https://www.etherchain.org/`

[6] Definition by Luis Cuende, co-founder of the Aragon DAO platform, in: `https://twitter.com/licuende/status/1263511552709267456?`

Regarding decentralization, it comes from the fact that all the decisions (and consequently the actions) a DAO takes, are by a consensus among the DAO members. This consensus is usually reached by a vote, where can participate all the agents who would like to participate.

Notice that those decisions do not just concern the allocation of the DAO resources (e.g., funding projects, payments) but also could change the DAO code. For example, even if a bug in the code of the smart contract came out, it would require a vote to change the code. Meanwhile, an attacker could still commit a robbery during the voting period (Martellucci, 2020).

The concept of DAO has been used with blockchain (Buterin et al., 2014; Buterin, 2014), and either its definition or its implementation is usually tied to this technology leading to some restrictions as follows.

The DAO activity[7] is recorded in the blockchain and, as a result, implies a cost. Validating and confirming transactions on the Ethereum blockchain requires a certain amount of work and computational resources, called *gas* (paid in cryptocurrency, see Section 2.2). This work is performed by blockchain miners in order to include transactions in a block. Gas ultimately translates into money and the amount of gas depends on the size and type of each transaction. As a result, it is expected that DAO activity is conditioned by this, since users are required to pay small amounts of cryptocurrency if they want their operation to be executed.

Bitcoin is considered one of the first implementations of the concept of DAO (Hsieh et al., 2018; Larimer, 2013). Although, there is a discussion if Bitcoin is actually a DAO or a DO (Decentralized Organization) (Buterin, 2014). We can consider that miners are the DAO agents, and the mining process is a consensus system, in which 51% of all miners have to be aligned to add a new block in the chain. However, you can also consider that Bitcoin is not "smart" enough because it is not an entity by itself and can not autonomously do things (except the mining protocol). These are who think that Bitcoin is DO-like.

One of the most Ethereum novel and promise project was *The DAO*, that was launched in April 2016 by a group of programmers in the Ethereum blockchain. It was the most successful investment crowdfunding at that time. *The DAO* was a sort of hedge fund, in which contributors could directly vote proposed projects. Investors would exchange Ether for tokens during an Initial Coin Offering (ICO). Then investors would vote new projects with their votes (weighted by tokens). In June 2016, due to an error in *The DAO* code, an attacker robbed a large part of its funds (Tikhomirov et al., 2018). However, due to *The DAO* representing the largest project in Ethereum at that time, the *Ethereum Foundation* decided to take action. After several days of discussions, the foundation finally decided to move forward with a hard fork,[8] and returned the stolen funds to *The DAO* investors. However, the concept of immutability of the ledger past records was damaged due to the fork (Mehar et al., 2019).

Despite that traumatic event, the endeavor of creating decentralized organizations to operate in the blockchain persists, but tempered with the knowledge that any DAO operates under the logic of smart contracts, and due to the nature of the code, it is subject of

[7]Those actions basically are transaction on the blockchain.

[8]A "hard fork" means the blockchain is copied in a new version with some difference (e.g. different rules or some blocks removed). This provokes a divergence in the two paths forward the blockchain can take. Typically, only one is considered the "valid" path; however, both can still be used.

security holes and risks.

Furthermore, creating a DAO from scratch requires highly specialized knowledge about blockchain programming. However, new ways to create a DAO have emerged, and there are platforms that offer DAO templates in order to create them, highly reducing the technical knowledge barrier.

2.4. Ethereum DAO platforms

Several platforms have emerged in recent times to facilitate the deployment of DAOs in the blockchain by significantly reducing the technological knowledge required. They provide DAO infrastructure *as a service*. These platforms allow users with scarce knowledge on how blockchain works to create a DAO using a template that typically can be customized. The main platforms are *Aragon*, *DAOstack*, *DAOhaus* and *Colony* (El Faqir et al., 2020).

However, the case of *Colony* will not be covered in this paper due to the lack of data availability (at the date of this work) and its novelty (released in February 2020); besides, the way that a *Colony* DAO works breaks with the "traditional" vote-driven DAO, where each action of the DAO must be voted. In *Colony*, they use the concept of work-driven, where works are published, and members accept them for a bounty (Mannan, 2018). So, comparing vote-driven and work-driven DAOs makes no sense due to the different nature of their conceptions.

2.4.1. Aragon

Aragon[9] is by far the largest DAO platform. *Aragon*'s approach is to extend the use of DAOs as a free and open-source technology to allow the creation and management of decentralised organizations (Aragon-Team, 2018). They are aimed to enable a wide-range organization forms, as companies, cooperatives, nonprofits, or open-source projects.[10]

Aragon provides a static template to make your own DAO, but it also allows you to create a customized one. Customization is enabled through "apps" (sets of smart contracts), which can be installed or removed from DAOs.[11] The purpose of apps widely varies: a *Finance* app to allocate the DAO's funds; a *Agent* app to interact with other Ethereum smart contracts; a *Token* app to manage the membership; or a *Vote* app used as a decision-making system (Valiente Blázquez et al., 2017; Wang et al., 2019). In addition, *Aragon* provides a *SDK*[12] to allow the community create and deploy their smart contracts, apps, and organizations templates (i.e. a set of predefined apps and a customized configuration for the template purpose).

In our case, we will focus on the *Vote* app, because voting is the main action in most DAOs. The default *Aragon Voting* app is the most used in *Aragon* DAOs, and its decision-making system work as follows. The app defines two conditions that any voting must fulfill to be approved.

[9] `https://aragon.org/`
[10] `https://help.aragon.org/article/4-about-aragon`
[11] Always by a vote.
[12] `https://hack.aragon.org/`

1. From all casted votes, the percentage of positive casts must be greater or equals than the *support required percentage*, that is a parameter.

2. The *minimum acceptance quorum* parameter states the minimum number of votes from all possible votes in the DAO.

The first one establishes the majority acceptance from the casted votes, while the second one establishes the minimum quorum. Both parameters can be changed by voting to adjust the majority required or the acceptance quorum.

There are other apps for voting. For example, there is an in-development app which implements the decision-making system of the *DAOstack* platform (Santander and One, 2019), or the *Dandelion voting* app which implements the decision-making system of *Moloch* (see section 2.4.3). The *Dot-Voting* app adds the possibility to vote with more than two answers instead of the typically binary answer(yes/no).

Even you can create a new organization template to change, not just the voting app, but the organization sort; for example the *Committee* template (Colombo, 2020a,b). This template was created to face the scalability problem (see section 2.4.2).[13] For that, they propose to break the DAO group into specialized sub-groups in charge of specific tasks, which can autonomously work without the interaction with the whole group.

But, one of the most ambitious decision-making systems is the *Conviction Voting* or *Social Sensor Fusion* (Zargham, 2018). In the CV, DAO members cast votes, with an amount of the DAO token they have, in a proposal. For example, member-A has fifty tokens, and she votes with twenty tokens in the proposal-A. So, now the member-A has just thirty tokens to vote in other proposals. Hence, if she wants to use more tokens than the remaining, she will have to remove tokens from proposal-A. That is the conviction. The second key of the CV is that proposals have to pass a dynamic threshold to be approved. This threshold is tied to the treasury funds. In other words, the more budget the proposal requests, the more conviction (tokens) the proposal has to gather. Notice that if the proposal-B is approved before the proposal-A, then the DAO funds decrease, and the proposal-A threshold will increase. On the other side, if the DAO funds increase, the threshold will decrease (Emmett, 2019; CVM, 2019). The *Conviction Voting* was tested with simulations,[14] and *Aragon* is testing it in a real environment (Association, 2020).

2.4.2. DAOstack

Unlike *Aragon*, the *DAOstack* platform[15] does not offer many customizations. *DAOstack* was designed to tackle the governance scalability problem, so to better understand the nature of this platform let's see what it is.

Matan Field, co-founder of DAOstack, states that the bigger a DAO is, the harder it is to manage it (Field, 2018a). Think in a DAO where all the decisions are taken by voting; in a DAO with few members, the number of decisions will be bound to the number of members, and the majority to approve those decisions will involve a sub-set of those few users. However, if we increase the number of those users, then the number of

[13] Scalability in terms of growing a DAO membership and its operations (i.e., votes, tasks, etc.)
[14] `https://github.com/1Hive/conviction-voting-cadcad`
[15] `https://alchemy.daostack.io/`

decisions and the number required users to approve decisions will also increase. That is important because it always requires enough mindful members to approve decisions. A naive solution to this could be reducing the required quorum (i.e., a relative majority), but it also introduces new flaws. For example, an attacker could spam lots of decisions in a small frame-time requesting the DAO funds, and it will be easier to get the funds using a lower quorum. Field states that increasing the DAO membership will reduce its resilience (Field, 2018a).

Therefore, to face this problem, *DAOstack* proposes its own decision-making system, the *Holographic Consensus* (Field, 2018b, 2019). The HC states three actors in the stage; the DAO members, the proposals, and the stakers. DAO members send and vote for proposals. Proposals are approved by absolute majority. And finally, the stakers are external actors[16] who stake their money trying to guess if a proposal will be approved. If it finally is, the staker will gain a bounty, if not they will lose the money staked. Besides, if a proposal is receives enough stakes, it will reduce its majority from absolute to relative.

With that mechanism, DAO members do not spend that much time deciding what proposals are really interesting for the community because, if staking works correctly, stakers actually filter the good proposals from the bad ones. To be rewarded, stakers need to be aligned with the DAO global opinion; otherwise, they will lose their investment.

Holographic Consensus was validated (Faqir-Rhazoui et al., 2021b), and the results show that usually, the larger a DAO is the better this mechanism work.

2.4.3. DAOhaus

DAOhaus[17] is the platform where to create and interact with the DAOs from the *Moloch* DAO framework.

Moloch DAOs implement a straightforward voting system, which is basically a non-quorum system, where always a relative majority is enough to approve a proposal. This way to proceed simplifies development and testing processes (Soleimani et al., 2019) of their voting system. A key aspect from these DAOs is the "rage quit", a mechanism to quit the DAO with your portion of the DAO resources if you do not agree with the result of a voting. After the voting outcome is achieved, there is a 'grace' period, when DAO members can commit rage quit if they are not agree with the outcome. Additionally, if there are more than ≈30% of rage quits, then the vote will be automatically rejected (Turley, 2020).

In addition, this voting system has two main attributes to consider: shares and tributes. Shares refer to an amount of resources that each DAO member has, independently of the cryptos the DAO has. And tributes refer to an amount of shares the proposal applicant pays to the DAO. Thus, in a proposal, the applicant can request shares and pay a tribute. Either shares or tribute defines the proposal sort, e.g. the same amount of shares and tribute defines a membership proposal, only share request defines a project proposal, and only tribute payment means a donation to the DAO (Duncan, 2019). If the proposal sent fails, then the applicant will get back his/her tribute offering.

However, *Moloch v2* has introduced a new variable in the equation, the sponsorship, which slightly changes the voting system. Now, when a proposal is sent, it requires the

[16] They can also be DAO members.
[17] `https://daohaus.club/`

sponsorship of any member of the DAO. This means that any DAO member makes a deposit confirming that it is trustful. All the proposals need to be sponsored before moving them to the regular queue, where the vote starts (Turley, 2020). When the vote ends, independently of the outcome, the sponsor will get a portion of his/her deposit back. This way to go avoids attackers to spam tons of proposals in order to exploit the non-quorum characteristic of *Moloch* voting system.

Nowadays, *DAOhaus* DAOs are split into two groups. Those created in the early stage of *DAOhaus*, v1 DAOs, and those created with new features of *Moloch*, v2 DAOs. Some changes introduced from v1 to v2 DAOs, include the ability to expel a DAO member from the community. It also includes some changes like the ability to send proposals by non-DAO members or the changes related to its voting system (Turley, 2020) already mentioned.

Chapter 3

DAO-Analyzer: A tool for DAO's Metrics Visualization

Online communities are studied and analyzed using visualization tools to show their features, model their state, or simply represent their main stats. In the wiki sphere, there are several tools to see at a glance how they are going. For example, *Wikimedia Statistics*,[1] shows the *Wikipedia* evolution among others (e.g., Wikimedia Foundation projects). Likewise, there are other tools like *WikiChron*,[2] which can give a more deep analysis on Mediawiki wikis (El Faqir et al., 2019).

However, when it comes to DAOs, there are not that many visualization tools. Especially due to the novelty of those blockchain communities. But, each DAO platform has their own dashboards, usually focused on the DAO user interaction (DAO managing), rather than in the study of DAOs.

Concerning *DAOstack*, they provide the *Alchemy*[3] dashboard which gives comprehensive information about DAOs (e.g., number of members, or proposals and their outcomes). In the case of *DAOhaus*, their tool[4] also shows how many communities this platform has, their members, or even the bank funds for each DAO. Regarding *Aragon*, the *Apiary*[5] tool offers an overview with some basic stats of the *Aragon* communities. However, all these tools provide so far little information that can be used to investigate DAO activity.

Besides our DAO-analyzer tool, *Scout*[6] is a proprietary tool which shows data about many Ethereum Dapps, among them *Aragon* DAOs. This tool gives a more global view of the ecosystem plotting data like the number of active DAOs, the number of DAOs created by template, or the amount of Ether they have.

More interestingly, *DeepDAO*[7] is another proprietary tool that aims to display analytics for financial data, members, proposals to all DAO platforms (including *Colony*). This

[1] `https://stats.wikimedia.org/#/all-projects`
[2] `http://wikichron.science/`
[3] `https://alchemy.daostack.io/`
[4] `https://daohaus.club/explore`
[5] `https://apiary.1hive.org/orgs`
[6] `https://scout.cool/aragon/mainnet`
[7] `http://deepdao.world`

tool was deployed at the same time of it provides a table with all the DAOs they have recorded, to directly compare them with some basic stats (number of users, or value of their crypto-assets). So far, you can also have a close look at each DAO, having a more detailed vision of their metrics.

Regarding our tool, *DAO-Analyzer*[8], its development started in February 2020 as part of the Chain Community project (RTI2018-096820-A-I00). This web-tool was designed to inspect the different DAO platforms, and the DAOs inside them. In the beginning, *DAO-Analyzer* was a proof of concept to show just *DAOstack*'s DAOs. Firstly, analyzing the main community in *DAOstack*, named *Genesis Alpha*, (El Faqir et al., 2020). And lately, we dove in the *DAOstack* voting system (*The Holographic Consensus*) showing how this voting system works as a predictor market and its utility for bigger DAOs (Faqir-Rhazoui et al., 2021b).

In a second iteration, due to the improvements of the data acquisition in the other DAO platforms (*DAOhaus*, *Aragon*), we could add DAOs from them.

In a third iteration, we also added DAOs from the *xDai* (covered in Chapter 4), which is a parallel network to *mainnet*,[9] and where many DAOs have moved due to the gas costs (the implications are analyzed in Chapter 5).

In a later iteration, we improved the application interface. Figure 3.1 shows how *DAO-Analyzer* looks like. Its look and feel was designed by Elena Martínez Vicente.[10] The *DAO-Analyzer* code is freely available.[11] with a copy-left license (GPLv3)

3.1. DAO-analyzer interface and use

The navigation through the tool is intended to be self-explanatory. Starting at the top left, you can select a DAO platforms from the available ones: *DAOstack*, *Aragon*, and *DAOhaus*. At the top right, you can select each DAO[12] from the platform selected.[13] Once you select the DAO name, the tool displays the plots with all the metrics available for the DAO platform. You also can pick the *"All DAOs"* selector, which computes the metrics for all the DAOs from the selected DAO platform. This is useful to see the numbers of the whole platform.

Regarding the metrics, the Table 3.1 summarizes the metrics calculated for each platform, and by their category. There are 45 metrics, where *DAOstack* has 18, *Aragon* has 12, and *DAOhaus* has 15.

In the case of *DAOstack*, their DAO users are so-called *Reputation holders*. Concerning the *Stakes* category, it is a special vote in this ecosystem (see Section 2.4.2). Stakes are also used to calculate the success rate metrics (e.g., *Total success rate of the stakes*, *Success rate of the stakes by type*), these show how good are the predictors (stakes) vs the real result (proposal outcome), this is better described in (Faqir-Rhazoui et al., 2021b).

[8] `http://dao-analyzer.science/`

[9] It is the central Ethereum blockchain's network, where all the transactions take place.

[10] Her LinkedIn: `https://www.linkedin.com/in/elenamartinezvicente/?originalSubdomain=es`

[11] `https://github.com/Grasia/dao-analyzer`

[12] DAOs from mainnet are marked with "(mainnet)". DAOs from xDai are marked with "(xDai)"

[13] For Aragon DAOs, there is not available their name. We show its hash address, which represents an Ethereum account.

Regarding *Aragon*, their DAO users are named *token holders*. In this case, there is non-available data about the user timestamp, that is why *Aragon* do not have as much metrics (in this category) as the other platforms.

Figure 3.1: DAO-Analyzer main look.

	DAOstack	**Aragon**	**DAOhaus**
DAO metrics	-Months with activity	-Months with activity	-Months with activity
DAO members	-New reputation holders -Reputation holders -Active rep. holders	-Active token holders	-New members -Members -Active members
Votes / casted votes	-Total votes option -Votes-for rate -Votes-against rate -Different voters -Rep. holders who vote -Votes per voter	-Casted votes by support -Casted votes-for rate -Casted votes-against rate -Active voters -Casted votes per voter	-Votes for and against -Votes-for rate -Votes-against rate -Active voters -Members who vote -Votes per voter
Stakes	-Total stakes -Different stakers		
Proposals / Votes	-New proposals -Closed proposal's majority outcome -Closed proposal's outcome -Approval proposal rate -Total success rate of the stakes -Success rate of the stakes by type	-New votes -Votes's outcome -Approval vote rate	-New proposals -Proposal outcome -Approval proposal rate -Proposals by type
Transactions		-New transactions	
Aragon apps		-Installed apps	
Rage quit			-Outgoing members

Table 3.1: DAO-Analyzer metrics by platform and metric category.

3.2. Data acquisition

As we previously said in Section 2.2, all the data in the blockchain is recorded as transactions. We can use tools like *Etherscan*[14] to fetch all the data we need. However, this way to proceed implies hard work to understand and translate transactions (as shown in Figure 3.2) into a more human-readable data.

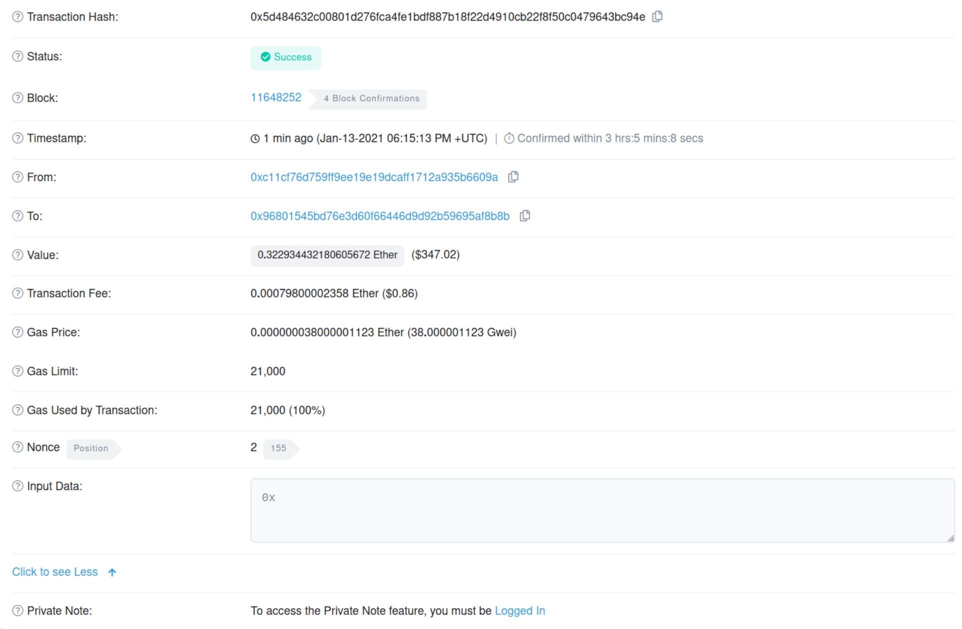

Figure 3.2: A random transaction from Etherscan.

Fortunately, DAO platforms are starting to provide friendlier ways to retrieve DAO data. More precisely, they are becoming to use *The Graph*[15] which is becoming in a standard for the Ethereum blockchain data acquisition. *The Graph* is a decentralized protocol for indexing and querying data from blockchains (not just for mainnet, but also for others like xDai). This protocol scans events in the blockchain to change the data stored and transformed in a different database.[16] Finally, *The Graph* provides a *GraphQL* API to query the data. Figure 3.3 shows how the data travel using this protocol to get it finally.

On the other side, a *GraphQL* API[17] defines a query language to fetch the data. It states different entities as schemes, e.g., User(name, age, email, etc.), for a unique endpoint. And it gives the ability to retrieve just the data fetched and not the whole endpoint data offering. This manner to get data reduces the times you need to call the API and gives you exactly the data you asking for.

Hence, because of the benefits of *The Graph*, we decide to use it in *DAO-Analyzer*. Besides, each platform (*DAOstack*,[18] *DAOhaus*,[19] *Aragon*[20]) provides its endpoint to fetch its data.

[14] `https://etherscan.io/`
[15] `https://thegraph.com/`
[16] Those databases usually use IPFS as a file system to keep the data decentralized.
[17] `https://graphql.org/`
[18] `https://thegraph.com/explorer/subgraph/daostack/master`
[19] `https://thegraph.com/explorer/subgraph/odyssy-automaton/daohaus`
[20] Aragon exposes various endpoints related to each app.

At the beginning, we implemented *DAO-Analyzer* to fetch the data in real-time. However, due to the increase on the number of analytics to be retrieved, the web-tool loading times also increased. So, we finally decided to create a local data warehouse and update it once per week.

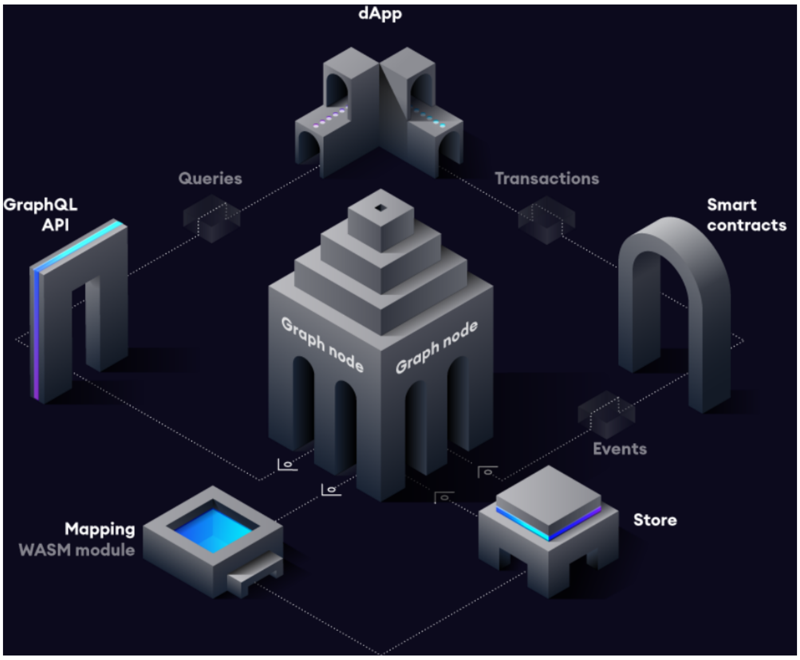

Figure 3.3: The Graph protocol architecture, extracted from their webpage `https://graphql.org/`.

3.3. Client-Server architecture

The web-tool has been developed using *Python* and the *Dash*[21] framework. *Dash* is built in top of *Flask*, and uses *React.js* and *Plotly.js* to power the client-side. This means that you have to program the whole code in Python (either server- or client-side) and *Dash* is in charge of translate the client-side code.

It works as follows, at first you define the *HTML* web structure using either a ".html" file or via Python using objects that the framework provides. Next, you state that this *HTML* code is the web page calling a *Dash* function (which in turn is a *Flask* function wrap) that sends it as a plain text to the client-side. If you want to add events in the client-side, you will need to define them with the framework functions, and finally, it will translate for you those events into *React.js* code. This code will fire an asynchronous event on the server-side, it will be processed, and the server will return a response that should change the layout in the client-side. One of the main advantages of *Dash* is that it already has defined many components (e.g., plots, HTML buttons, HTML selectors, etc.) which wraps the layout, and the events, although, you can also define yours if you need extra features. Figure 3.4 summarizes how the web-app works.

Besides that, we will dive into the data acquisition, introduced in the previous section.

[21] http://dash.plotly.com/introduction

Before to up and running the server needs the data warehouse prepared to get the application data. So we need to launch before a data process to fetch the data from all the GraphQL APIs for each DAO platform in a JSON format. This data is processed and transformed in a *CSV* format, in order to later be used as a dataframe with the library *Pandas*. The data warehouse keeps the historic DAO's data and follows a schema on read, meaning that it is prepared to easily add new data, although, change existing data would imply more computational effort.

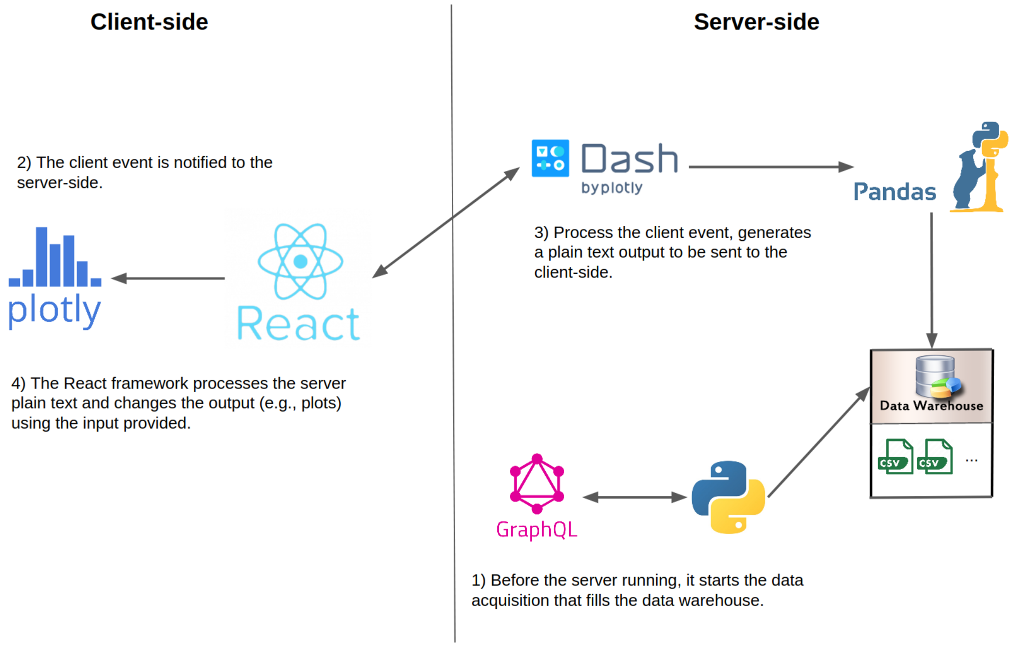

Figure 3.4: DAO-Analyzer Client-server architecture and technologies.

3.4. The in-application architecture and design patterns

DAO-Analyzer defines an application per platform. However, most of the code is commonly used for all the applications, and the main code differences boil down to the metrics. In this section, we are going to show the *DAOhaus* class diagram, but the other diagrams are also available in the repository.[22]

The application is split into three layers; presentation, business, and data. Where the presentation layer is in charge of keeping the layout and notify the client events to the business layer, it follows a *MVC* pattern. However, in this case, controller classes just notify events using interface functions of the object they are in charge of (e.g., each plot has its own controller).

The business layer is in charge of mediating between the presentation and data layers. It defines what metrics are available, how the metrics are presented (e.g., plot sort) and asks for the data. This layer gets transfer classes from the data layer and transforms them into dictionary data to send them to the presentation layer.

Finally, the data layer which basically is Data Access Objects,[23] transforms the ware-

[22] `https://github.com/Grasia/dao-analyzer/blob/master/src/apps/aragon/class_diagram.png`
[23] In this section, Data Access Objects are called daos, and Decentralized Autonomous Organizations are

house data into transfer classes (e.g., metrics, organizations). So, the main application calculations are done in this layer and are tied to the transformation of the raw warehouse data into the metrics we already presented in the Table 3.1.

The full diagram we are going to show is available at the *DAO-Analyzer* repository.[24] In the Figure 3.5, we can see the *"DAOhausService"* class used as main class of the business layer. It configures the layout and binds it to the controller class ("DashboardView" and "DashboardController"). Besides that, the *"DashboardView"* class has a set of classes which also are bonded to their controllers (*"ChartController"* class). The *"ChartPaneLayout"* class in turn wraps the *"Figure"*(graph), which has different types by inheritance (e.g., MultipleBarFigure, BarFigure, DoubleScatterFigure). Either *"ChartPaneLayout"* or *"Figure"* are configured using the decorator pattern (see also Figure 3.6).

Figure 3.7 shows that *"OrganizationListDao"* uses the *"CacheRequester"* class to get the raw data from the warehouse. This class transforms the data and returns the *"OrganizationList"* transfer class. In a similar way that *"MetricDao"* does. Although, this class is created using the factory pattern (*"MetricDaoFactory"*). *"MetricDao"* also uses the strategy pattern (see Figure 3.8) to process the raw data into the transfer classes (e.g., *"StackedSeries"*, *"NStackedSeries"*). In the end, those metric transfer classes can not be used directly to fill the *"Figure"* classes with data, for that we use the adapter pattern (e.g., *"ProposalOutcome"*, *"BasicAdapter"*, etc.), which takes *"StackedSeries"* and *"NStackedSeries"* classes and transforms them into a dictionary data that the *"Figure"* class can undestand.

3.5. Testing

During the development, we created a set of unit tests[25] to cover the application hot spots, that are those concerning the metrics calculation (e.g., metric strategy classes). For that, we have created a specific mock-up to deal with UNIX dates (*"UnixDateBuilder"*), which implements the builder pattern. Another mock-up is related to the class that directly access the raw data, *"ApiRequesterMock"*. Those mocks are used for dependency injection providing premade data (testing corner cases) to the classes to be tested.

In order to improve the testing coverage, we have also included some *property-based tests* or random tests. This sort of test defines properties that the function's output must satisfy (e.g., length of an array, positive values, etc.), and runs the function with random inputs (Claessen and Hughes, 2011). In our case we have used the Python implementation, *Hypothesis*.[26]

named as organizations.

[24] `https://github.com/Grasia/dao-analyzer/blob/master/src/apps/daohaus/class_diagram.png`

[25] `https://github.com/Grasia/dao-analyzer/blob/master/test`

[26] `https://hypothesis.readthedocs.io/en/latest/`

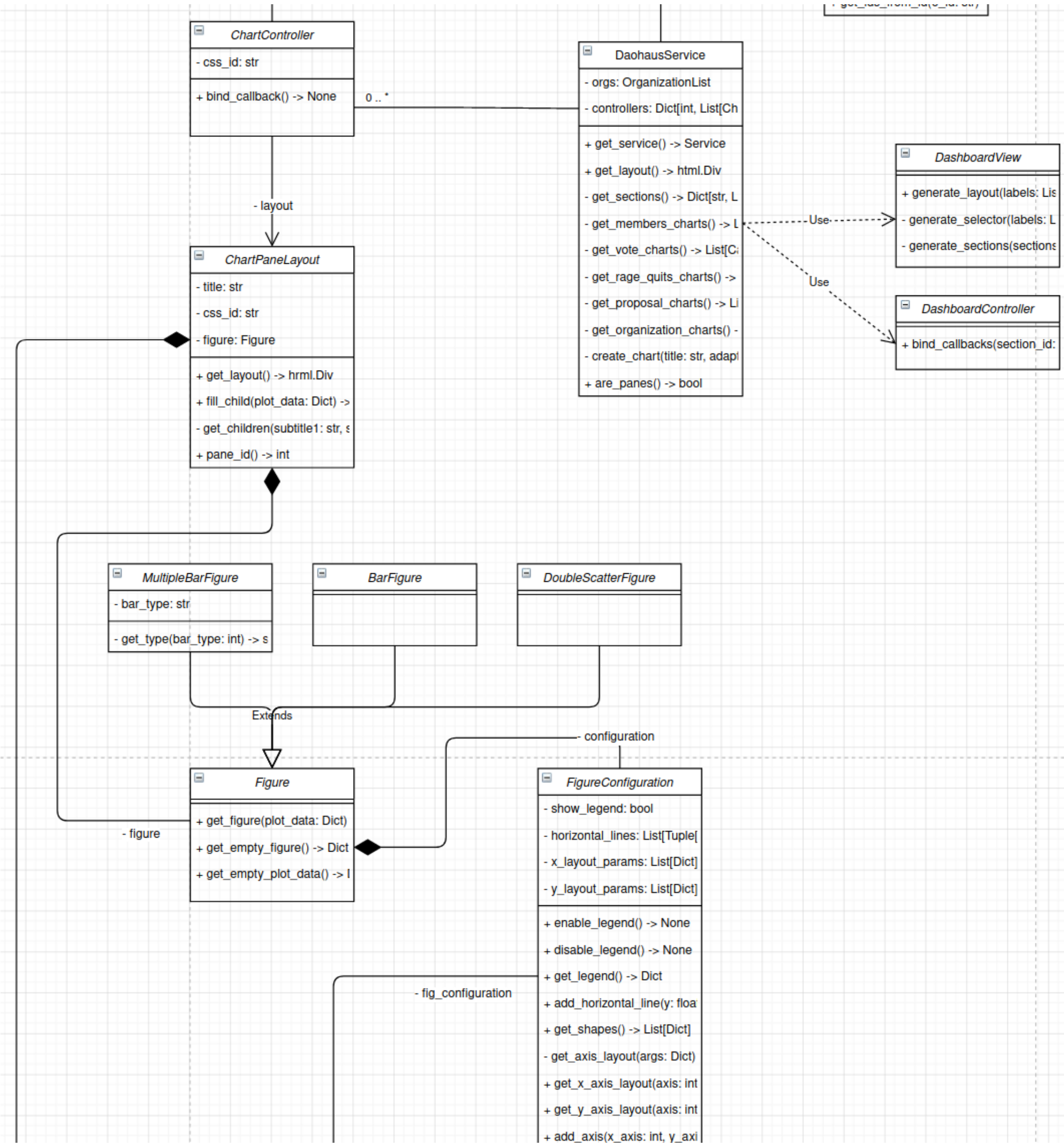

Figure 3.5: DAOhaus app: presentation and business layers.

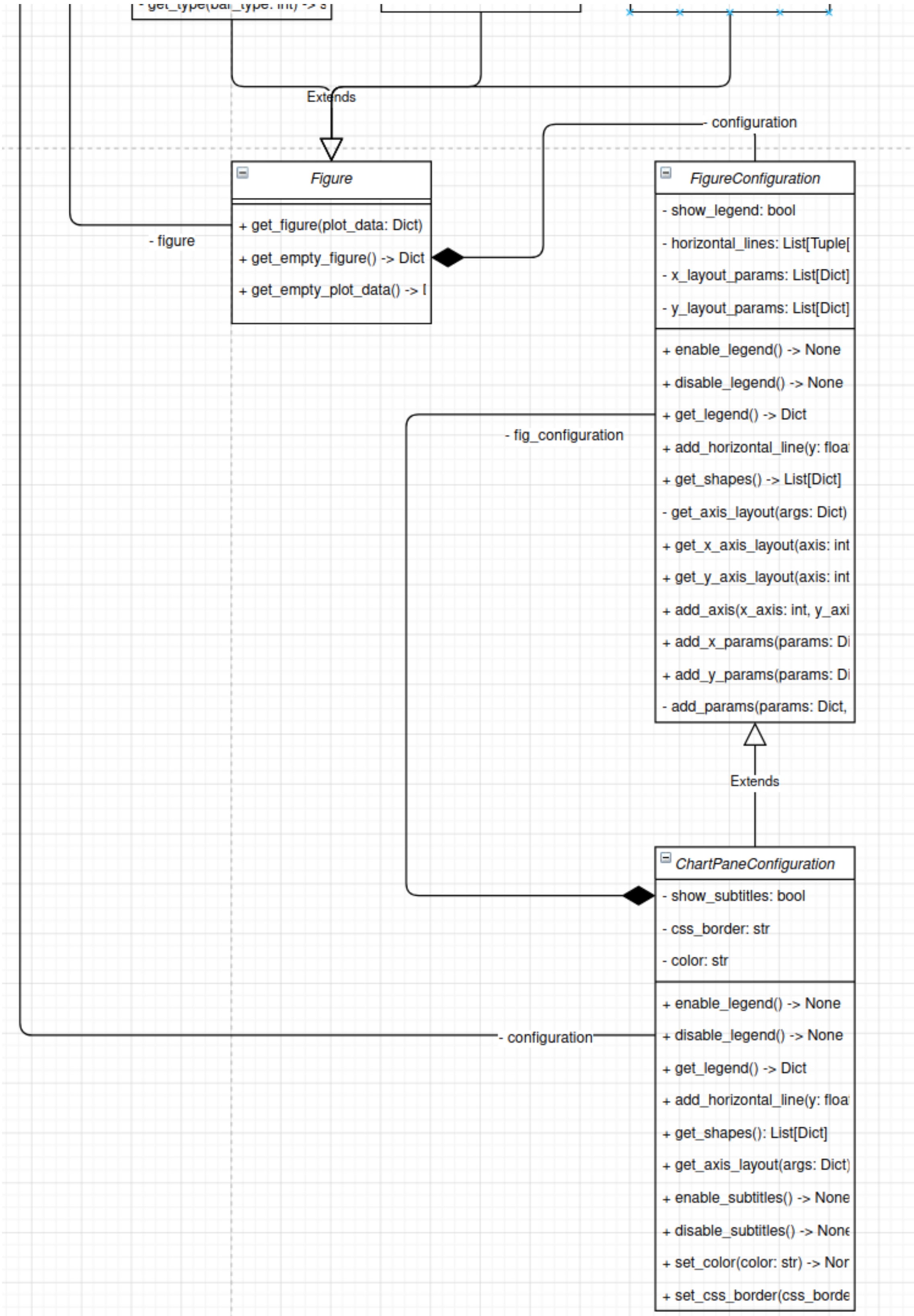

Figure 3.6: DAOhaus app: chart classes.

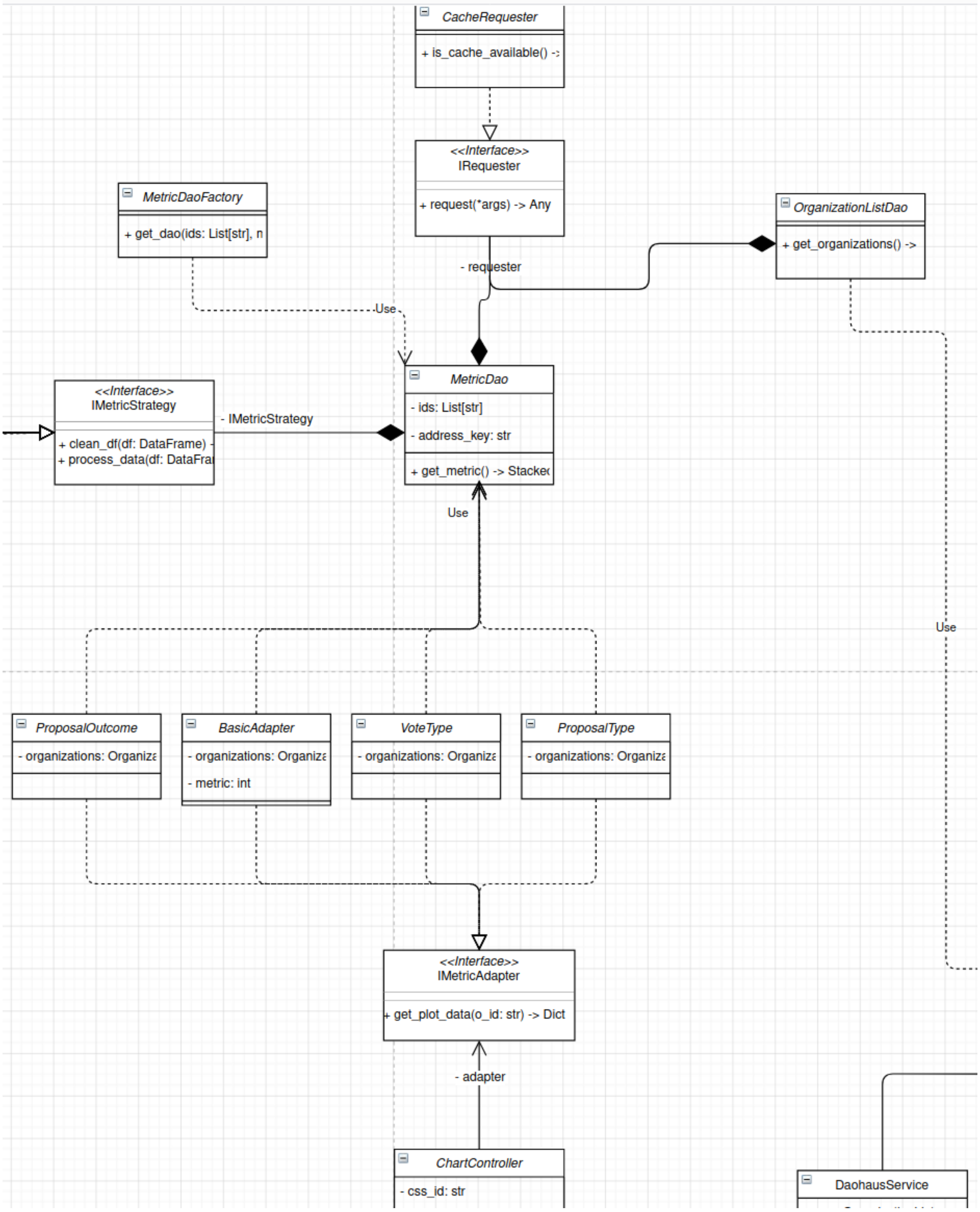

Figure 3.7: DAOhaus app: data access objects, and adapter classes.

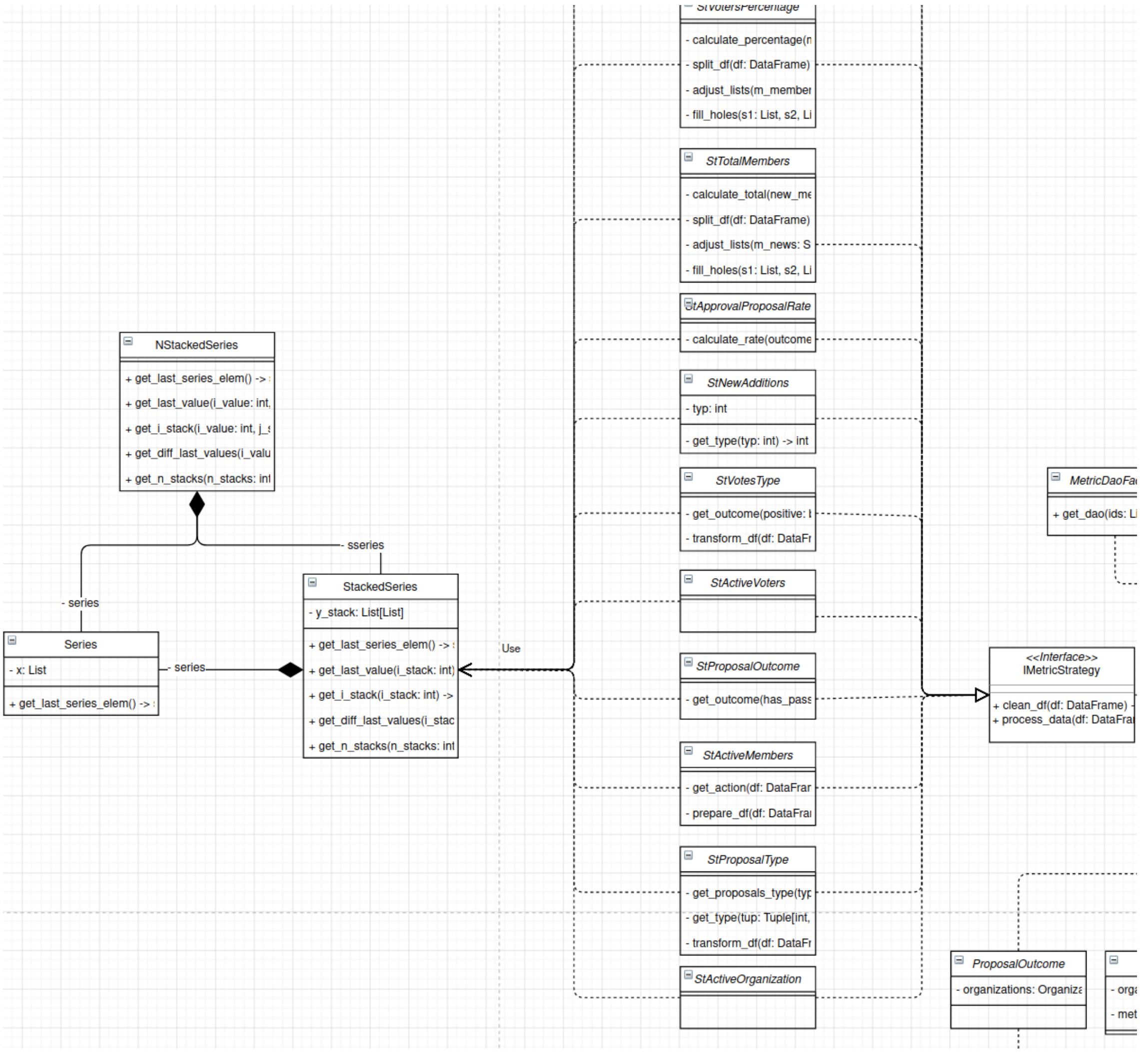

Figure 3.8: DAOhaus app: strategy metric classes and series transfer classes.

Chapter 4

A Quantitative DAO Comparison

The three DAO platforms introduced in Section 2.4, i.e., *Aragon*, *DAOstack*, and *DAOhaus*, are going to be compared using *DAO-Analyzer*. However, we also will use *DeepDAO* for the comparison. All the data used to this comparison covers until November 30th.

The comparison will tackle four topics; growth, activity, voting system, and the cryptocurrencies DAOs own. In our comparison, we will also consider a new phenomenon, the *xDai* network. As already explained the use of the *Ethereum mainnet* implies the payment of a fee (e.g., gas cost), and this fee is tied to the network's use. Nowadays, the *Ethereum mainnet* is overused, so the fee you need to pay for any transaction (e.g., vote, create a DAO, etc.) is quite high. Even though *Ethereum 2.0* (currently in-develop) will drastically reduce these prohibited fees, other alternatives have emerged while it arrives. That is the case of the *xDai* network, a now available solution to the gas cost problem. *xDai* uses a bridge with *mainnet* to bring tokens or to give them back.

Table 4.1[1] shows the costs to create a DAO, and to vote, in the DAOhaus ecosystem. They are much more expensive in *mainnet* than *xDai*.

	Summon	Vote	Speed	Security
mainnet	$80	$5	5 tps	Hyper secure
xDai	$0.01	$0.001	70 tps	Secure

Table 4.1: Network comparison

Table 4.2 shows the number of DAOs, users, and proposals per platform and by network (mainnet and xDAI). *Aragon* is by far the biggest platform: its *xDai* DAOs represents 15.71% (from all *Aragon* DAOs), and its *xDai* users are 30.1%. *Aragon* started using *xDai* from July 2020. Because of that, there is not still available data about its voting app.

In the case of *DAOhaus*, it is the second one in terms of number of DAOs. The *DAOhaus* *xDai* DAOs are 24.89% from its total, the *xDai* users are 18.34%, and the proposals represent 24.93% of all proposals. *xDai* is used in *DAOhaus* since July 2020.

Finally, *DAOstack* is the second one in term of users. *xDai* DAOs are 62.71% from its total, *xDai* users are 45.51%, and the number of proposals are 19.16% from the total.

[1] The table was took from `https://daohaus.club/help#xDAI`

DAOstack has started using *xDai* since February 2020, so it is widely used.

Due to the importance of the *xDai* use, we will take it into account for the ecosystems comparison.

	Aragon		DAOhaus		DAOstack	
	mainnet	xDai	mainnet	xDai	mainnet	xDai
DAOs	1,744	325	169	56	22	37
Users	41,021	17,660	1,180	265	6,645	5,549
Proposals	10,246	-	1,668	554	1,954	463

Table 4.2: Comparison of the three ecosystems in terms of number of DAOs, users and proposals

4.1. Growth

For the growth comparison, we have chosen two metrics; the number of DAOs and the number of users. However, the DAO's timestamp creation is not currently available for *DAOstack*'s DAOs. Similarly, user's timestamp creation is not available for *Aragon*'s DAOs.

Figure 4.1 shows that *Aragon* has kept a stable growth and also boosted by *xDai*'s new DAOs. On the other side, *DAOhaus* has been slowly growing, even with the addition of *xDai*. The new DAOs created in *xDai* can be brand new DAOs or 'migration' of *mainnet* DAOs; however, the DAO migration implies the creation of a new DAO with a different id and account. So there is no way to automatically know when a DAO is migrated, or it is a new DAO.

Figure 4.2 involves the number of user of *DAOhaus* and *DAOstack*. In the case of *DAOstack*, there are two peaks in the series. The first peak was in June 2019 (5397 new users), and it concerns the early months of *mainnet* deploy. The *DAOstack* user number has remained stable till February 2020, the second peak (2822 new users). This peak matches with the launch of *xDai* network (the gap between the dark and light green), but since then, the number of users keeps without significant variations. By contrast, the users in *DAOhaus* have been continuously increased without relevant peaks. However, as we previously explained in Section 2.4.3, *DAOhaus* users can do 'rage quit,' so they can quit from the DAO if they want to. Because of this phenomenon, the number of users could be higher if 'rage quit' didn't exist.

4.2. Activity

We will consider active (either DAO or user) if it has made an action at least once in a given month as El Faqir et al. (2020) state. The actions depend on the platform. For *DAOstack* we will consider as actions; create a proposal, vote in a proposal, and stake in a proposal. In the case of *DAOhaus*, we took; create a proposal, vote in a proposal, and the rage quit, which is the final action performed by a user. Finally, due to the customization of an *Aragon* DAO (by installing apps), it is quite difficult to state their main actions because all apps are not installed in all the DAOs, and some apps have not available data

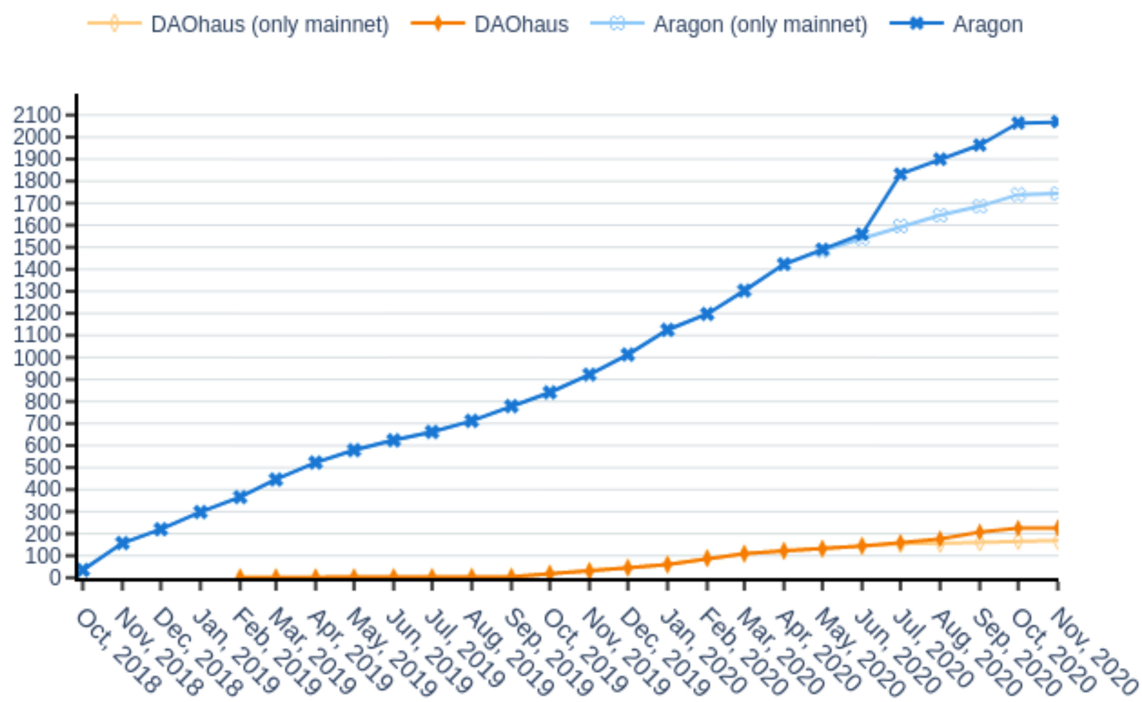

Figure 4.1: Number of DAOs of *Aragon*, and *DAOhaus*. Lighter colors just refer mainnet DAOs. Darker colors take into account DAOs from mainnet and xDai.

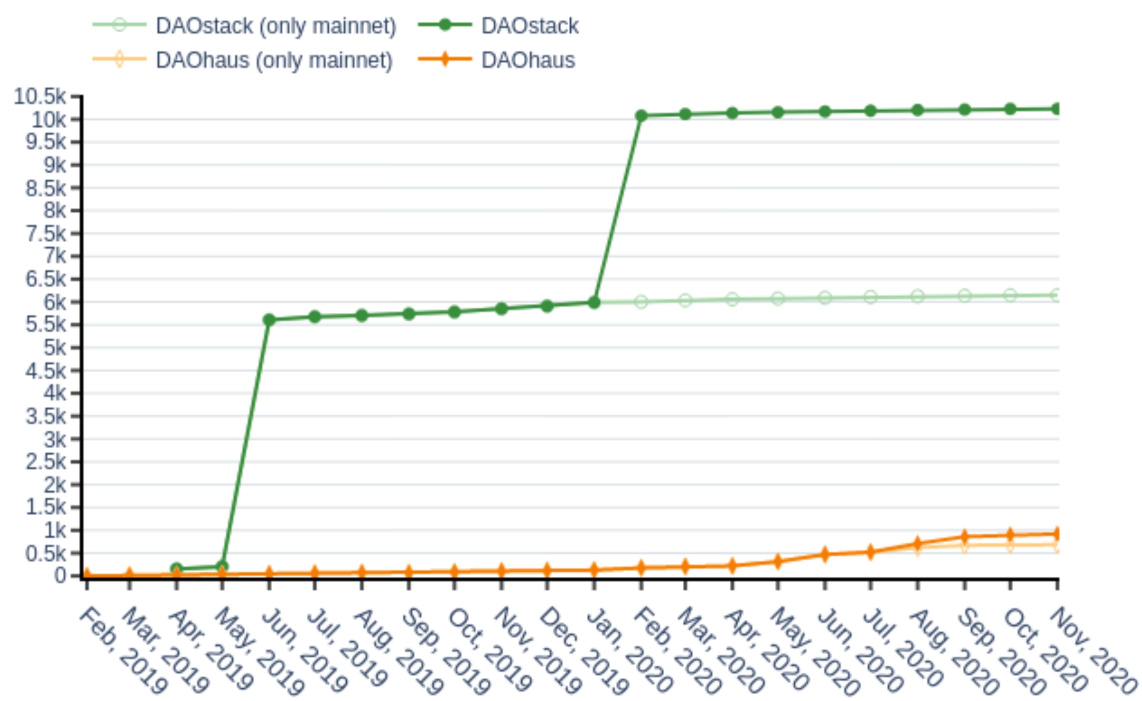

Figure 4.2: Number of users of *Aragon*, and *DAOhaus*. Lighter colors just refer mainnet users. Darker colors take into account users from both networks.

to fetch. So, in *Aragon* we will just consider data from the basic voting app (create a vote, and cast a vote), and from the transaction app (used for donations or payments) is used the transaction as an action. Although for *Aragon xDai* actions, we can just consider transactions because there is not still available data for the *xDai* voting app.

Figure 4.3 shows that active DAOs in *Aragon* have been increased by the *xDai* boost (the difference between the light and dark blue lines), even taking just transactions, so the real number of active DAOs could be much higher. Surprisingly, *DAOstack* has been reduced this number even with *xDai*, to values below to ten DAOs actives per month. *DAOhaus* has benefited from *xDai* network, greatly increasing its active DAOs to forty per month.

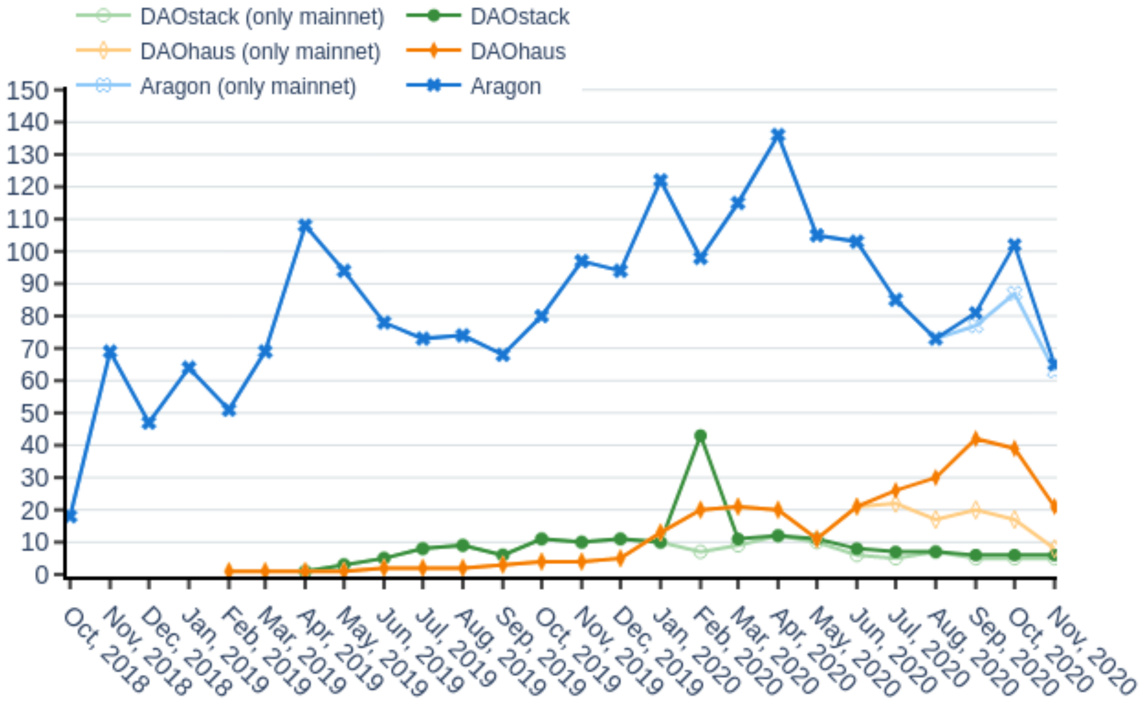

Figure 4.3: Number of active DAOs of the three platforms. Lighter colors just refer mainnet DAOs. Darker colors take into account DAOs from mainnet and xDai.

Figure 4.4 presents activity in terms of users. Neither *DAOstack* nor *DAOhaus* have benefited from *xDai*. However, while *DAOhaus* has been continuously increased active users along the time series, *DAOstack* has been continuously decreased its number of active users. On the other side, before *Aragon* introduced *xDai* it had a stable number of active users (around 350). But since then, its active users have greatly increased, even just counting transactions and not the other apps.

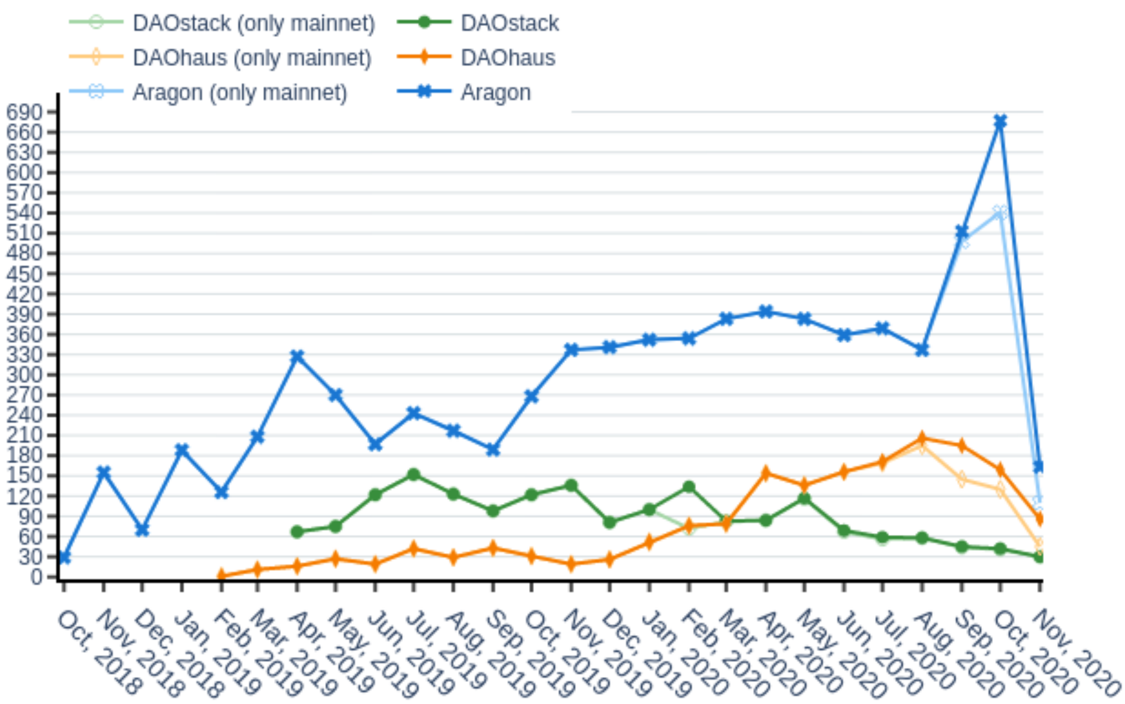

Figure 4.4: Number of active users of the three platforms. Lighter colors just refer mainnet users. Darker colors take into account users from mainnet and xDai.

4.3. Voting system

All the platforms are vote-driven, and each of them have their own voting system, as we covered in Section 2.4. In the case of *Aragon* DAOs, they can have many voting systems; however, for the sake of simplicity, we will take just data from the basic voting app.

In order to compare these decision-making systems, we have chosen four metrics; The percentage of proposals that are approved, this will show how the voting system could bias the result of a vote. The percentage of casted votes-for will help us to understand whether the proposals failed because of the system or the vote outcome. The number of casted votes per voter, a value close to one will mean that votes are equally distributed, but increasing it will mean that votes are not normally distributed. Finally, the percentage of users who vote can help us to see the engagement of the community with their project (or DAO). Additionally, we will also consider the *mainnet*, *xDai* network, and the sum of them. In the table 4.2 you can see the number of users, and proposals by platform and network.

Starting with *DAOstack*, Table 4.3 shows that its votes are biased due to the small number of users who have voted at least once (4.5% or 549 users), and also due to the 4.6 votes per voter. However, those voters usually vote positively 86%, but the percentage of approved proposals are 74%, this could result from the *DAOstack* voting system, which needs an absolute majority by default, or enough staking in a proposal to achieve a relative majority. So, usually, non-boosted proposals are rejected. The implications to have a parallel voting system (the boosting process) are covered in (Faqir-Rhazoui et al., 2021b).

Nevertheless, notice that even with *xDai*, *DAOstack* has reduced its activity (see Section 4.2), only the 2.1% of *xDai* users have ever voted. So the gas cost does not seem the reason for the low activity.

Moving to *DAOhaus*, we can see that it has the most engagement community, where 38.37% of users have voted at least once. Interestingly, that number is higher in *mainnet* than in *xDai*, maybe because of the novelty of the network in this platform (deployed in July 2020). Regarding the casted votes, these are 91% positives, and the percentage of approved proposals is 92%, which is also the highest one. These numbers can be explained because of the *DAOhaus* voting system, which just counts 'yes' votes versus 'no' votes meaning that there are non-quorum. Additionally, this system needs the sponsorship,[2] which is an amount of tokens staked by a DAO member. Due to that, users usually send proposals which they think could be passed. Notice that we are not stating that a voting system is better than the others, we are just comparing the influence of those decision-making systems on the results.

In the case of *Aragon*, there are not still available data for the *xDai* voting system, so we can just analyze the *mainnet* data. In *Aragon*, the number of users who have ever voted is quite low 6.18%, but it is bigger than the *DAOstack*, taking into account that *Aragon* has 3.3 times more users (41,021 vs 17,660) in *mainnet* than *DAOstack*. The percentage of positive votes is the highest (94%), and the approved proposals are 81%. The difference between positive votes and approved proposals is related to the variability of the quorums (minimum vote quorum[3] and minimum positive votes quorum[4]) which each DAO has. However, that percentage is higher than the *DAOstack*'s and lower than the *DAOhaus*'s. Maybe due to the quorums, which is more restrictive in *DAOstack*, and non-restrictive in *DAOhaus*.

	Approved props.	**Votes-for**	**Votes/voter**	**Users who vote**
DAOstack	74%	86%	4.6	4.5%
DAOstack (mainnet)	74%	86%	4.64	6.3%
DAOstack (xDai)	76%	95%	3.64	2.1%
DAOhaus	92%	91%	4.26	38.37%
DAOhaus (mainnet)	93%	90%	3.96	39.5%
DAOhaus (xDai)	87%	98%	7.28	24.32%
Aragon	-	-	-	-
Aragon (mainnet)	81%	94%	4.08	6.18%
Aragon (xDai)	-	-	-	-

Table 4.3: Voting comparison by platform and network

[2]It is just for v2 *DAOhaus* DAOs.
[3]The minimum DAO members who have to participate.
[4]From the members who have participated, the minimum positive votes required to approve it.

4.4. Cryptocurrencies

DAO funds are an important topic to cover, because those DAOs are usually used to employ people, to fund proposals that can benefit the community (i.e. physical meetings, hackathons), or even invest to make a profit for the DAO. We have selected the top ten cryptocurrencies and DAOs to exemplify those ideas. We have used *DeepDAO* for this task; however, it does not offer all the DAOs from the three ecosystems we cover, but it has the main DAOs of each platform.

Table 4.4 shows the top ten cryptocurrencies by the number of DAOs which have them. Many of those cryptos are stablecoins like DAI, SAI, USDC, or USDT, this means they were designed to keep a stable value during volatile market periods and reduce transaction fees. Additionally, those cryptos also have differences between them. The fiat-collateralized type (e.g. USDC, USDT) is the most common stablecoin, and they usually rely on centralized institutions. In the case of crypto-collateralized (e.g. DAI, SAI) coins do not depend on traditional finance infrastructure and use crypto assets as collateral. For example, DAI and SAI cryptos are created by *MakerDAO*, a non-platform DAO. To get an amount of those cryptos you exchanges ETH to get them (sta, 2019).

It is worth to mention that ETH is one of the most used cryptos, despite its market volatility. However, not all DAOs can use ETH as an asset. That is the case of *DAOhaus*'s DAOs, which can not use non-ERC20 cryptocurrencies. Due to that, there are solutions like WETH, that wraps ETH in an ERC20 smart contract. There are other cryptos like ANT or GEN, which are specific tokens for *Aragon*, and *DAOstack* platforms. In the case of GEN, it is used for the proposal boosting process. And the ANT token is used to govern the *Aragon Network* DAO. Besides that, each DAO can have its own crypto, for example, *PieDAO* has DOUGH, a coin with 44,291,262 USD of market capitalization, but it is just owned by *PieDAO*.

Token name	Token symbol	DAOs	USD value DAOs have
Dai stablecoin	DAI	51	6,229,754$
Ethereum	ETH	50	14,714,446$
Sai stablecoin v1.0	SAI	21	15,013$
USD Coin	USDC	20	5,878,148$
Wrapped Ether	WETH	18	9,303,476$
Aragon	ANT	15	12,824,896$
Panvala pan	PAN	11	20,552$
DAOstack	GEN	9	37,553$
Tether USD	USDT	8	1,158,129$
Balancer	BAL	6	331,744$

Table 4.4: Top 10 currencies by number of DAOs which use them.

Table 4.5 has the top ten DAOs with most crypto resources in USD. Most of those DAOs are from *Aragon*, they also have a small number of members compared to the resources they manage. For example, *mStable* is a DAO which provides autonomous and non-custodial stablecoin infrastructure, where to exchange stablecoins without additional fees.[5] *PieDAO* is focused on bringing market accessibility and economic empowerment,

[5] `https://docs.mstable.org/`

it allocates DAO resources by governance, in order to gain profit for its investments. [6] *dxDAO* is a *DAOstack* DAO, which takes its revenues from its DeFi services or products they have and/or develop[7]. *MetaCartel Ventures* (a *DAOhaus* DAO) is a for-profit DAO created for the purposes of making investments into early-stage Decentralized Applications (DApps).[8]

DAO name	DAO platform	Resources in USD	Members
PieDAO	Aragon	73,829,906$	2,881
mStable	Aragon	38,263,266$	8
dxDAO	DAOstack	17,581,208$	444
Airalab	Aragon	13,263,696$	11
Aragon Trust	Aragon	7,015,477$	5
Aragon Network Budget	Aragon	5,903,309$	3
MetaCartel Ventures	DAOhaus	5,619,718$	99
Aavegotchi	Aragon	5,059,662$	3
API3 DAOv1	Aragon	2,991,833$	30
Aragon Network	Aragon	2,932,121$	5

Table 4.5: Top 10 DAOs by a total of cryptocurrencies in USD.

4.5. Discussion

We have compared the three main DAO ecosystems using four dimensions: growth, activity, their voting system, and cryptocurrencies they have. *Aragon* has shown good numbers showing continuous growth, in terms of users and DAO deploy in it, this effect was also slightly boosted due to the *xDai* adoption in the last months. Also, eight of the top ten richest DAOs are from *Aragon*, which means that this platform has been established as the main option to deploy a DAO, due to its capabilities.

Regarding *DAOhaus*, we also appreciate a growing effect, although, this effect is quite low than the *Aragon* case. In terms of activity, the data of this platform shows that its activity was strongly boosted due to *xDai* network. The *DAOhaus* voting system shows that this is the most engagement ecosystem where 38% of users have voted at least once. Also, these results show that most of proposals (93%) are accepted, which is the highest acceptance rate.

However, the *DAOstack* case is special. Their result shows that it is stagnated. The number of users has registered two peaks: first in its early months, and the second peak ties to the adoption of *xDai*. But, this adoption, which is supposed to reduce the transaction fees and increase the activity, does not show any effect.

Surprisingly, in all the cases, we did not find that the use of *xDai* network has a strong effect on the platform's activity, as we previously expected. So, a limitation of this work ties to know if there is a consensus in the DAO sphere to move to the *xDai* network, or there are other networks also used to reduce the gas costs. Hence, this could alter the adoption results of non-mainnet networks. However, with the results we get the high gas

[6] `https://docs.piedao.org/`
[7] `https://dxdao.eth.link/#/faq`
[8] `https://metacartel.xyz/about`

cost does not seem to affect the DAO activity directly, because of the adoption of *xDai*. But, the gas implications will be covered in the next chapter.

Chapter 5

Effect of the Gas Price Surges on User Activity in the DAOs

In this chapter, we study the 2020 surge of transaction fee price in the Ethereum network and analyze how that affected user activities. In the previous Chapter 4, we have superficially seen how is the *xDai* network adoption, which is directly related to gas cost. Although, we did not find evidence that proves the lowest the gas is, the higher the activity is.

We still consider DAOs a good case of study due to their nature, where people need to organize themselves using votes, which implies the network use for non-profit actions a priori. Thus, we hypothesize that increases of the gas price would impact negatively the number of operations performed by users. Particularly, we want to ascertain whether such increases affect to governance operations in DAOs, not including other Ethereum operations. For that, we analyzed 15,977 transactions from 48,886 users grouped in 1,935 DAO communities, using a Vector AutoRegression (VAR) model with daily time series of the average gas value (in mainnet) and the DAO operations.

5.1. Related work

The issues we want to tackle have been under-studied in the current literature. However, there are some researches which can help to better understand the gas fluctuations.

Sovbetov (2018) has investigated the factors that influence several cryptocurrencies price, like Bitcoin or Ethereum. And he found that those cryptocurrencies are much more volatile in short- and long-run than other markets (e.g., *SP500*). Möser and Böhme (2015) study the Bitcoin transaction fees until December 2014. They want to see how those fees change over time. However, they found that the volume of transaction fees is driven by social conventions formed by actors in the sphere (e.g., miners) rather than the market protocol. Easley et al. (2019) build a model to explain the factors leading the Bitcoin transactions fees, concluding that the transaction's waiting time is a significant factor in the transactions fees. de Azevedo Sousa et al. (2020) evaluates 7.2 million of Ethereum's transactions to investigate if there is a relationship between the fees and the pending time. They found that despite the common belief, the transaction pending time is not directly related on the value of the fee offered. Pierro and Rocha (2019) has found a Granger causality between the total amount of miners in the Ethereum network and the final gas price.

5.2. Research Question

The transaction fees have been greatly increased recently, from 12GWei in January 2020 to 100GWei in September 2020. Figure 5.1 shows the average gas price (Wei) along 2019 (blue) compared to 2020 (red) until December. We can appreciate that from May 2020 the average transaction fee price has risen to values that are too prohibitive to many users. This is caused by the network congestion, due to the increment of decentralized finance (DeFi) applications[1] in March 2020 (Kordyban, 2020b; 0xNick, 2020). Those DeFi apps requires more complex transactions, which means more gas and fewer transactions per block due to the gas limit restriction. Likewise, Ethereum is the main network for those applications (Liu and Szalachowski, 2020; Kordyban, 2020a). So, all in all, DeFi products increase gas cost. Hence, the use of the Ethereum network for other applications (e.g., DAOs) is more difficult due to the fees.

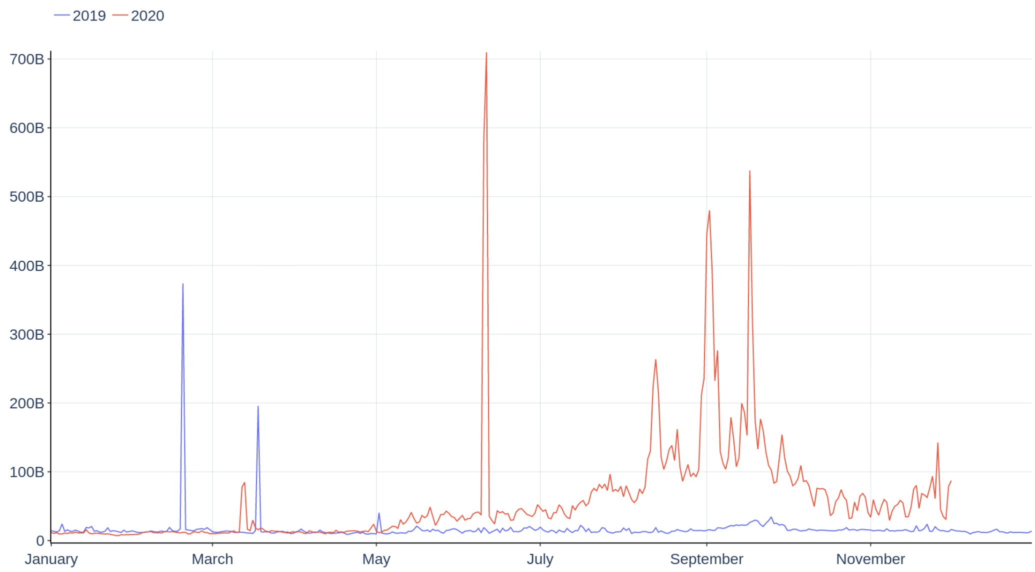

Figure 5.1: Average gas price in Wei along 2019 (in blue) and 2020 (in red) until November.

Because of the fee's cost, many solutions have appeared until the launch of Ethereum 2.0 (Cortes-Goicoechea et al., 2020). For example, there are ERC20[2] contracts that tokenize gas, storing gas when it is cheap and deploying it when the gas is expensive (e.g., GST[3]) (Nadler, 2020). Other solution ties to parallel networks (e.g., xDai) to Ethereum mainnet where the gas cost is quite low, and other Dapps are migrating to them, that is the case of DAOs in order to minimize the fees to participate in their communities. However, there are also DAOs which use DeFi products to gain profit for their community (e.g., MakerDAO,[4]

[1] DeFi is used to get profit from decentralized finance products (e.g., loans, financial services, cryptocurrencies exchanges, etc.).

[2] ERC20 is a Token Standard for tokens within Smart Contracts.

[3] https://gastoken.io/

[4] https://makerdao.com/en/

PieDAO[5]). So, how those prohibitive fee prices can affect transactions of other Dapps? Especially, does the gas cost directly related to the DAO's activity?

5.3. Data and Methods

5.3.1. Data description

Our analysis is divided into two sub-analysis. The first one considers the activity (actions) as we previously defined. The second analysis considers the daily ratio of activity per registered user, that is, the quotient between the number of actions performed in a DAO ecosystem during a given day and the number of members registered in the DAO ecosystem until that day. The reason to use this ratio is later discussed (Section 5.5). Besides, we could not consider *Aragon*, because so far the API does not provide the user registration date needed to compute the ratio. In both cases, we used data from May 2020 to December 2020. We used May 2020 to start the time series because it is when the gas price significantly surges, which could affect the activity in DAOs, according to our research hypothesis. Table 5.1 summarizes analysis we implement and coverage stats they have.

	Platforms	**Period**	**Stats**
Activity analysis	- Aragon - DAOhaus - DAOstack	- May 2020 - December 2020	- Trans.: 15,977 - Users: 48,886 - DAOs: 1,935
Ratio analysis	- DAOhaus - DAOstack	- May 2020 - December 2020	- Trans.: 5,580 - Users: 7,825 - DAOs: 191

Table 5.1: Analysis comparison summary.

To get the DAO's data we have used *DAO-Analyzer*'s data warehouse. We retrieved the daily average gas prices from Etherscan.[6] Since the gas price time series had trend and volatility, we transformed it to logarithmic returns, as it is usually done with price time series in Finance. Thus, we dealt with the logarithmic returns time series that is easier for model fitting and also has a clear economic interpretation. The daily log-returns of the gas price, i.e., $ret\_gas_t = ln(gas\_price_t/gas\_price_{t-1})$, incorporate information on the growth and decrease movements of prices that is roughly similar to percentage increases and decreases.

5.3.2. Methods

We analyzed the possible dependence between gas price and DAO activity using Econometric's methods. The original time series have weekly seasonality, which is the presence of regular weekly variations. The seasonal component can bias the results, so we performed a seasonal decomposition of the time series and removed the seasonal component from the original series. We considered several models for the seasonal decomposition (e.g.,

[5]`https://docs.piedao.org/`

[6]`https://etherscan.io/chart/gasprice`

Seasonal-Trend decomposition procedure), though we finally applied an additive model because of its effectiveness.

Before the model estimation, we need to ensure the time series's stationarity, which is required by the model. This means that the mean, variance, and the autocorrelation structure of the series do not change over time. We use two unit-root tests: Augmented Dickey-Fuller (ADF) and a non-parametric alternative, Phillips-Perron (PP) test, their critical values are derived from (MacKinnon, 1994, 2010).

For the dependence analysis, we used the Vector AutoRegression (VAR) models, introduced by Sims in macroeconomics (Sims, 1980). In simple terms, in a VAR model of order p, each of the p variables is modeled as a linear combination of past values of itself and the past values of the other variables considered. Our VAR(p) is:

$$y_t = v + A_1 y_{t-1} + ... + A_p y_{t-p} + u_t, \tag{5.1}$$

where the vector of variables is $y_t = (ret\_gas_t, activity_t)'$, the $(2x2)$ matrices of parameters are $\nu, A_1, ..., A_p$, and the error process $u_t = (u_{1t}, u_{2t})'$ is 2-dimensional white noise, with covariance matrix $E(u_t u_t') = \sum_u$, that is $u_t \sim (0, \sum_u)$.

VAR models represent the dependence patterns between the set of variables considered and offer the possibility of evaluating the statistical "causality" existing between them. For that purpose, we used these three econometric tools:

- Orthogonalized and accumulated Impulse Response (IR) functions
- The decomposition of the variance of the forecast error
- Granger's causality test under the fitted model structure

The IR's plots represent the dynamics generated in the system variables. This IR system will generate a random shock on the gas price, and those fluctuations should affect the activity, in the case of causality between both variables. We will quantify these shocks' effects, their sign, and evaluate whether they are transient or permanent.

The decomposition of the variance determines how the forecast error in each variable can be attributed to its innovations or innovations of other variables. It will help us to assess the sensitivity of the activity rate to price changes.

Finally, the statistical evaluation known as Granger-causality (Granger, 1969), investigates the existence and direction of the relationship between the variables studied. In this case, we evaluated whether the activity is Granger-caused by gas price movements, meaning a *predictive* causality.

We estimated a VAR(p) for each DAO ecosystem, choosing p by optimizing the Akaike information criterion (AIC). We evaluated the model by checking that the residues behave as white noise. For the latter, we used the autocorrelation function graphs and the Portmanteu test for checking the whiteness of the residuals for up to 20 lags.

For further details, the methodology is explained in (Lütkepohl, 2005). For the statistical analysis we used the *statsmodels* package in Python (Seabold and Perktold, 2010). The analysis data and the code of the statistical analysis are publicly available under *MIT* license.[7]

[7] GitHub repository: `https://github.com/Grasia/gas-dao-activity`

5.4. Activity results

As we previously stated, the first step to build the model is to assess the series stationarity. Table 5.2 shows the results we get from the ADF and PP tests. These tests state that either the gas price or the activity's series are stationary at all significance levels. As a result, the stationarity condition of the series is fulfilled, and we are able to estimate the VAR model.

Variables	ADF Test	PP Test
gas price	-11.759***	-22.866***
DAOstack activity	-4.624***	-9.282***
DAOhaus activity	-4.666***	-12.485***
Aragon activity	-3.766***	-11.050***

*** sig at 1%, **sig at 5%, *sig at 10% level

Table 5.2: Stationarity test results by DAO ecosystem.

Table 5.3 shows the VAR results given. First, we built a VAR(2) model for the three platforms. The Portmanteau test shows that we can reject the hypothesis of white noise, this means that the models account for all the lineal information in the data. For *DAOhaus* and *Aragon* at all significance levels, in the case of *DAOstack* at 5% of the significance level. Regarding the Granger causality, there is statistical evidence that *DAOhaus* is influenced by the gas price fluctuations (significant at 5%). Surprisingly, that is not the case of the other platforms (*DAOstack*, *Aragon*).

Statistical model information	DAOstack	DAOhaus	Aragon
VAR(p)	2	2	2
Portmanteau test (20 lags)	90.800*	79.830	63.990
Granger causality test	1.831	3.047**	0.122

*** sig at 1%, **sig at 5%, *sig at 10% level

Table 5.3: Model results by DAO ecosystem.

Moving to the decomposition of the variance of the forecast error. Table 5.4 presents it for the activity rate short term ($h = 1, 5$), and medium and long term forecasts ($h = 10, 20$). Those results reveal that most of the changes produced in the activity are due to their own change. For example, in the case of *DAOstack* in a short term run, 99.3% ($h = 1$) of the shocks produced are by its own, and just 0.7% are produced by a gas surge. Therefore, the activity is small affected due to the gas price changes.

Now we will focus on the accumulated IR plots. Figures 5.2, 5.3, and 5.4 show how they look like. In the left figure side, we plotted the accumulated IR where the gas is affected by itself. In the right figure plot, we have the gas as impulse and the activity as a response. Also, there are two dotted lines above and below the IR line (blue), these are the confidence intervals set at 95%. In other words, we are finding that both dotted lines move up the zero line, or down to it. If that occurs, then it will mean that at 95% of confidence level, we can assure that a gas shock influence the activity.

All the ecosystem IR plots (left-side) have a positive gas shock of 0.35, which means that the gas increases 35% in the daily price. On the other hand, the activity plots (right-side) show an alike structure between them. In relation with *DAOstack* (Figure 5.2), we can see

	DAOstack		DAOhaus		Aragon	
Forecast horizon h	gas p.	activ.	gas p.	activ.	gas p.	activ.
1	0.007	0.993	0.012	0.988	0.009	0.990
5	0.032	0.968	0.024	0.976	0.013	0.987
10	0.034	0.966	0.024	0.976	0.013	0.987
15	0.034	0.966	0.024	0.976	0.013	0.987
20	0.034	0.966	0.024	0.976	0.013	0.987

Table 5.4: Forecast error variance decomposition for the activity rate by DAO ecosystem. Proportions of forecast error variance h periods ahead, accounted for by innovations in gas returns and activity rate.

that it is negatively affected by the gas shock, which is statically significant (95%) in the second day and the effect remains until the day twelve. Despite the significance level, we can see the weakness of the effect in the *DAOstack* activity.

Concerning *DAOhaus* (Figure 5.3) IR, there is non-significant effect in its activity due to a gas shock (dotted bands). Contrary to the other ecosystems, this platform positively response at a gas shock at the beginning, but it goes to negative values from the second day and so on. This effect seems counter-intuitive, it makes sense if we consider that some users may hasten to perform their actions because they expect the gas price to keep increasing in the short term. Naturally, these results are non-statically significant, and the explanation of the gas effect in *DAOhaus* is just a hypothesis.

The *Aragon* case is also similar to the other ecosystems. We can appreciate a negative effect due to a price shock. However, there is not a significant effect which confirms the gas effect.

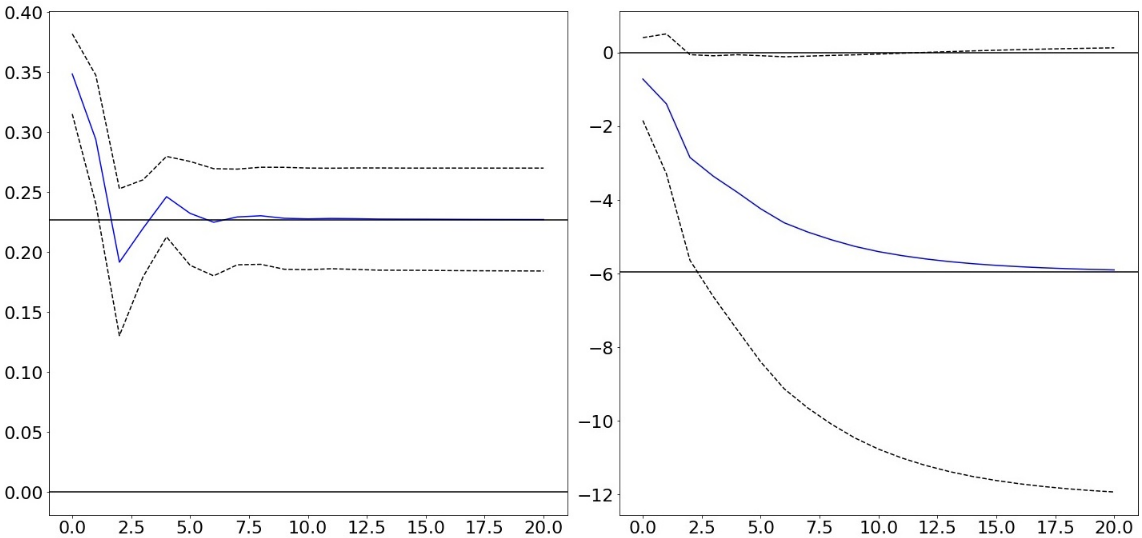

Figure 5.2: DAOstack accumulated impulse-response functions (Left: gas → gas, Right: gas → activity)

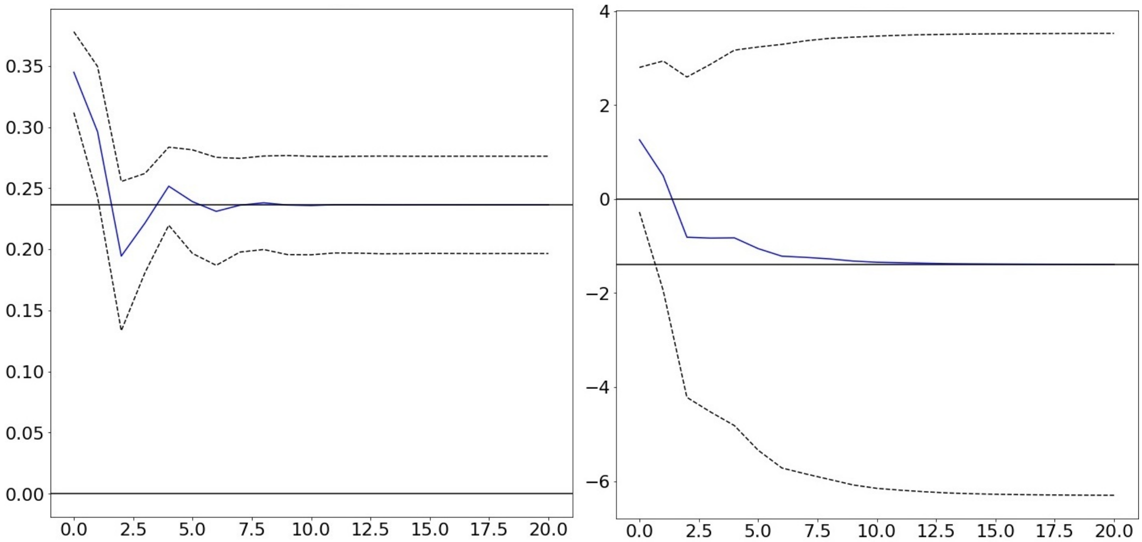

Figure 5.3: DAOhaus accumulated impulse-response functions (Left: gas → gas, Right: gas → activity)

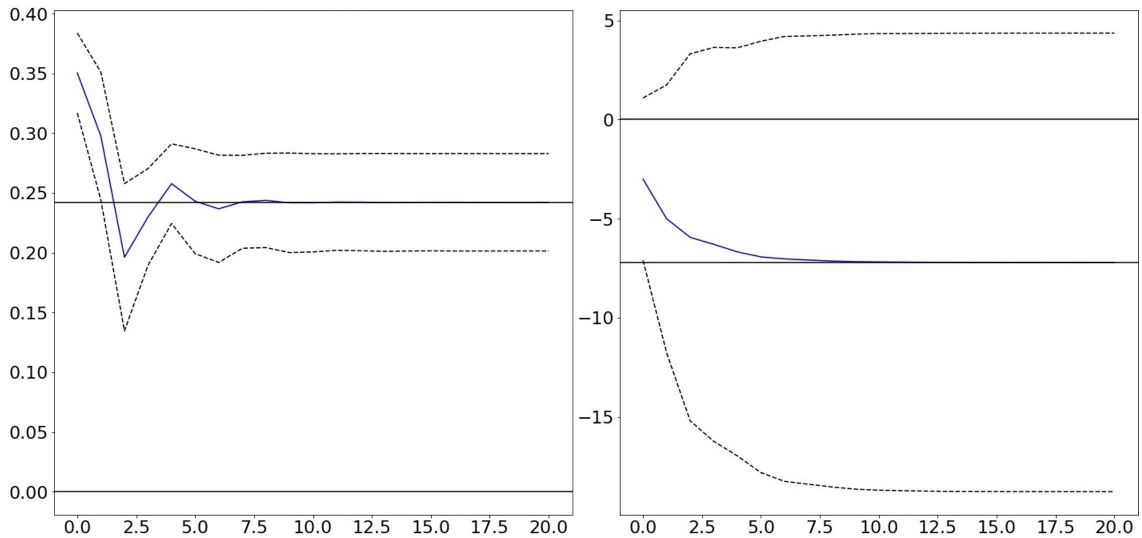

Figure 5.4: Aragon accumulated impulse-response functions (Left: gas → gas, Right: gas → activity)

5.5. First results discussion

On the contrary, we thought, these results suggest a weak causality due to the gas price surges. In particular, *DAOhaus* has shown a Granger causality drove by gas prices (significant at 5%), but no significant effect was seen in the IR plot. On the contrary, *DAostack* does not show a Granger causality, but an IR effect with a 95% of significance and a negative sign. However, this effect was minimal. On the other hand, the case of *Aragon* is special, due to its dimension we expected to find a more clear relation with the gas turbulences. But it did not show any effect caused by gas prices, neither Granger causality nor IR plot.

However, we want to go a step further, analyzing the activity but removing the user number, which could affect the previous analysis. The number of DAO members increased significantly in one of the ecosystems (*DAOhaus*) considered during the period studied. The lack of causality could be produced by incrementing DAO members, who produce more activity in the platform, reducing the gas cost effect. For that, we propose the activity user ratio or the number of actions per user in the whole platform. As we previously said in Section 5.3, we have to exclude *Aragon* for the next analysis due to the lack of a user creation timestamp in the data.

5.6. User-activity rate results

As the first step to build the model, we present stationarity test results in the Table 5.5. In both cases, the activity rate time series do not have a unit root. In the case of *DAOstack* at a significance level of 1%, and *DAOhaus* at a significance level of 5%. The gas price series is still the same series as we discussed in 5.4, so it is also stationary.

Variables	ADF Test	PP Test
gas price	-11.759***	-22.866***
DAOstack activity rate	-4.577***	-9.225***
DAOhaus activity rate	-2.899**	-12.031***

*** sig at 1%, **sig at 5%, *sig at 10% level

Table 5.5: Stationarity test results by DAO ecosystem.

Once we build the VAR model, it gives us the results in the Table 5.6. We built a VAR(2) for *DAOstack*, and a VAR(3) model for *DAOhaus*. In both cases, we test the model's white noise, and we can assess that it is at 5% of significance level in both models. Regarding the Granger causality derived from the model. It states that in the case of *DAOhaus* there is a weak causality (at 10% of significance level), meaning that *DAOhaus*'s activity is influenced by the gas price.

Table 5.7 shows the decomposition of the variance of the forecast error for the activity rate short term ($h = 1, 5$), and medium and long term forecasts ($h = 10, 20$). The results show quite similar values for both ecosystems, where the variance of the forecast error in the activity rate is mainly due to own shocks in at least 96.5%, and maximum in 3.5% to shocks on gas price returns. In other words, the changes associated with the activity rate forecast of the VAR model are mainly from its own shocks in both DAO ecosystems. Meaning that the gas price movements little influence the activity in those platforms.

Statistical model information	DAOstack	DAOhaus
VAR(p)	2	3
Portmanteau test (20 lags)	90.630*	81.460
Granger causality test	1.694	3.385*

*** sig at 1%, **sig at 5%, *sig at 10% level

Table 5.6: Model results by DAO ecosystem.

	DAOstack		DAOhaus	
Forecast horizon h	gas price	activity rate	gas price	activity rate
1	0.008	0.992	0.016	0.984
5	0.033	0.967	0.034	0.966
10	0.035	0.965	0.034	0.966
15	0.035	0.965	0.034	0.966
20	0.035	0.965	0.034	0.966

Table 5.7: Forecast error variance decomposition for the activity rate by DAO ecosystem. Proportions of forecast error variance h periods ahead, accounted for by innovations in gas returns and activity rate.

Let us focusing now in the impulse response plots in the Figures 5.5 and 5.6. In both cases, the plot shapes are much alike, also compared with the activity IR plots in the previous results. In *DAOstack* activity plot there is a significance effect after two days of the gas price shock and it even remains two weeks after. However, the effect is small but statistically significant (the dashed lines are the confidence intervals). On the other side, *DAOhaus* does not show clear evidences of the price shock, according to the confidence bounds (95%).

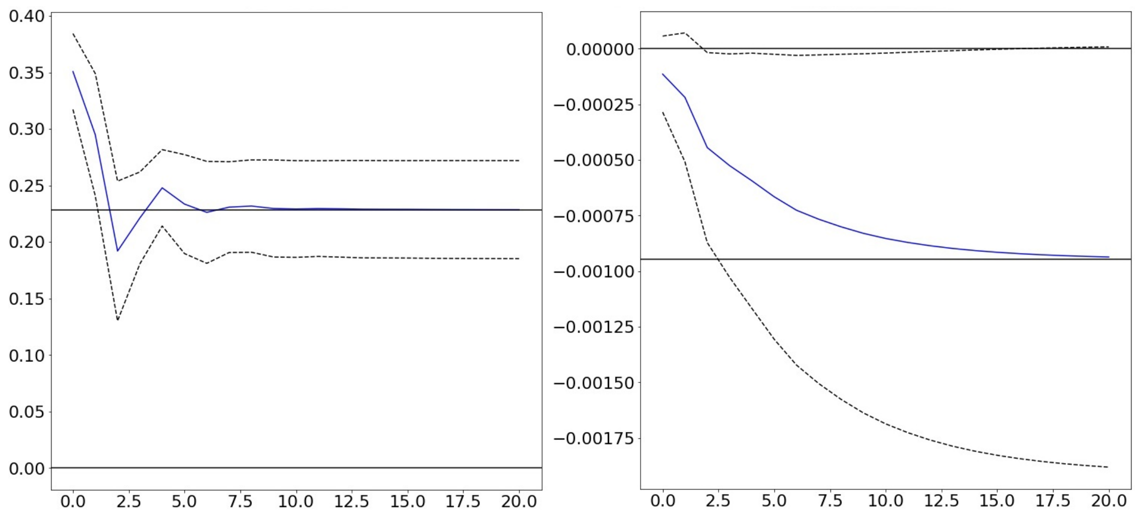

Figure 5.5: DAOstack accumulated impulse-response functions (Left: gas → gas, Right: gas → user-activity rate)

In this case, we can also conclude that a gas increment weakly causes a decrease in the activity rate, although, it is not statistically significant. Those values chime with the previous found in Section 5.4. So we can conclude that there is a negative sign in the activity due to a surge gas price, but not significant.

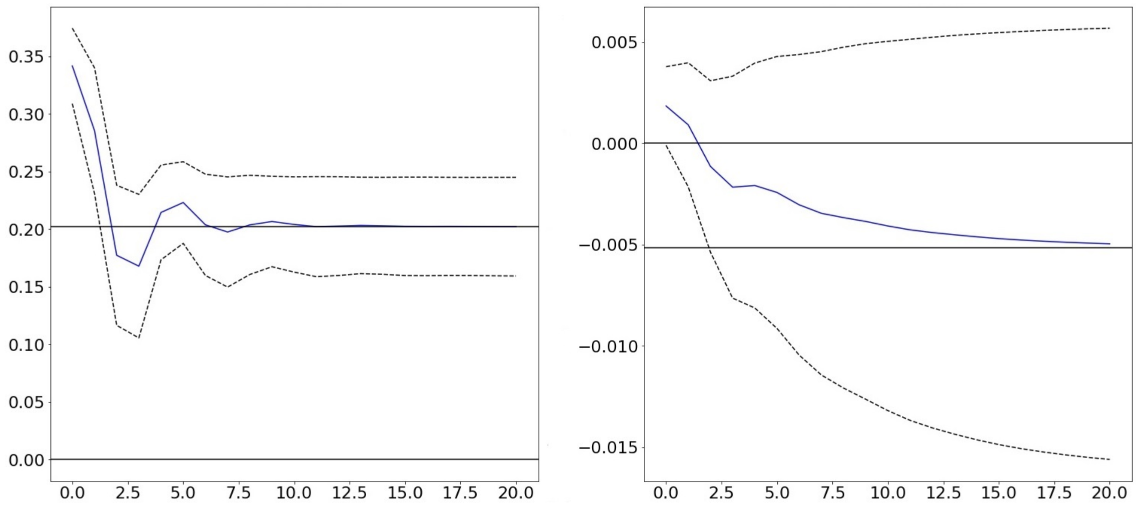

Figure 5.6: DAOhaus accumulated impulse-response functions (Left: gas → gas, Right: gas → user-activity rate)

5.7. Final discussion and concluding remarks

By comparison, both analyses have retrieved quite a similar result. Hence, on the contrary, we previously thought, the user growth in the platforms does not condition the result we got in Section 5.4.

In the DAO sphere, there is a common belief that gas surges strongly decrease the DAO activity. However, we found these effects are weak, and in most cases insignificant (e.g., *Aragon*). We have also checked the pre-May 2020 time period, which we did not include in this work. But similarly, we also did not found strong evidence of the alleged gas causality. The interested reader could find these results in the previously mentioned repository.[8]

A limitation of this study is that it did not account other phenomena related to decreasing gas surges, like the *xDai* network, we previously mentioned, or even gas tokenizers (e.g., GST) we mentioned in Section 5.2. As well as, the fact that many DAOs have an interest to use mainnet to get profit for their communities, for example using DeFi products (see the cryptocurrency comparison in Chapter 4).

Nevertheless, the fact of this small effect shows that most DAO users need to cope with the surge of fees and absorb the cost individually. This is counter-intuitive in theoretical self-regulated markets, in which raises in the cost of a product (in this case, transactions) typically would reduce its demand until it drops again. The behavior studied is more similar to medicine markets such, where price is inelastic since most consumers will buy the medicine regardless of the price. This is a market signal often understood as the convenience to intervene or regulate the studied market, in order to benefit consumers Sassi et al. (2018); Brekke et al. (2007). The mentioned impact is relevant in the development of these platforms particularly in their early years, since it may disincentive DAO use or even trigger abandonment.

[8] `https://github.com/Grasia/gas-dao-activity`

Chapter 6

Conclusions and Future Work

6.1. Conclusions

The main goals of this Master thesis were mainly achieved. In doing so, we have contributed to generate understanding about the blockchain communities. For that, we have built and deploy a free and open-source tool, *DAO-Analyzer*, which is able to show several metrics of all the DAOs of the main DAO platforms; *Aragon*, *DAOhaus*, *DAOstack*.

We also have used our web-tool to compare the current state of the three main DAO platforms, revealing their differences in terms of growth, activity, voting system, and crypto-assets adoption. The results show that all the platforms differ, even in the adoption of the xDai network, which was conceived to overcome the problems motivated by the surge of transactions fees in the mainnet Ethereum network.

Going further, we decided to understand how the transaction fees could affect the activity in a DAO. We used econometric techniques to tackle the matter. Building a model which infers the relations between the gas cost and the activity time series. The effect we found suggests that the gas fee cost barely causes the decrease of the DAO's activity in two of the three DAO ecosystems. Those results show that the main cause of the DAO activity is related to their own internal factors.

All in all, this Master thesis has contributed adding open-source tools to the academic- and DAO sphere-community. Proving its utility in three academic papers (El Faqir et al., 2020; Faqir-Rhazoui et al., 2021b). Also, the gas fee results clear the way to investigate as the main cause of the DAO activity, the internal human factors that occur in these communities.

6.2. Future Work

There some features that can be added for *DAO-Analyzer*. But also, many kinds of research can be done to study the DAO sphere and its behaviours.

- DAO-Analyzer: The *Colony* DAO platform is currently developing its API[1] for The Graph protocol. Once it is deployed, it would be straightforward to add it to DAO-Analyzer.
- DAO-Analyzer: There are many other Aragon applications that can be analyzed with the web-tool. For example, the govern app[2] or the dot-voting app.[3]
- Furthermore, a comparison of the cost and time complexity of the smart contracts used in the different DAO ecosystems would also be interesting. Specially, given the problems with Ethereum transaction fees. For that, it would be possible to use the tools that provide information of the smart contract's time complexity (Albert et al., 2020, 2018; Chen et al., 2017).
- It would also be interesting to investigate how the different DAO voting systems are used by the communities. For example, whether there are coalitions and oppositions, whether the voting is under or over represents the most wealthy members, etc. For that purpose, it would be suitable to use network analysis, similarly as we did in a previous work for communication and collaboration wiki communities (El Faqir et al., 2019).
- Another topic that could deserve research attention is the study of the money flow (incomes and expenditures, or distribution among members) in a DAO and compare it with traditional organizations.

6.3. Contribution

This master thesis covers the app's development of *Aragon* and *DAOhaus* in *DAO-Analyzer*, but not the *DAOstack* development. However, the following metrics were added to the *DAOstack* app, as a part of this thesis:

- Months with activity
- Reputation Holders
- Reputation holders who vote
- Votes per voter
- Approval proposal rate

The *DAO-Analyzer* data architecture (datawarehouse) was also added for this work.

The work of the chapter "A quantitative DAO comparison" was part of a journal article written together with Javier Arroyo and Samer Hassan. As of the date of this thesis writing, the article is under development.

The work of the chapter "Effect of the Gas Price Surges on User Activity in the DAO" was also used in the article (Faqir-Rhazoui et al., 2021a), which is under review for the Conference on Human Factors in Computing Systems 2021.[4]

[1] `https://github.com/JoinColony/subgraph`
[2] `https://thegraph.com/explorer/subgraph/aragon/aragon-govern-mainnet`
[3] `https://thegraph.com/explorer/subgraph/aragon/aragon-dvoting-mainnet`
[4] `https://chi2021.acm.org/`

Bibliography

Decentralized autonomous organizations. the next frontier of human coordination. https://aragon.org/dao, ????

Mathematically Formalizing the Conviction Voting Algorithm. https://nbviewer.jupyter.org/github/BlockScience/Aragon_Conviction_Voting/blob/master/algorithm_overview.ipynb, 2019.

What Is DAI? https://cryptonews.com/coins/dai/, 2019.

0xNick. We Need To Talk About The Block Gas Limit. https://ethereumprice.org/newsletter/we-need-to-talk-about-the-block-gas-limit/, 2020.

Albert, E., Correas, J., Gordillo, P., Román-Díez, G. and Rubio, A. Gasol: Gas analysis and optimization for ethereum smart contracts. In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, 118–125. Springer, 2020.

Albert, E., Gordillo, P., Rubio, A. and Sergey, I. Gastap: A gas analyzer for smart contracts. *CoRR, vol. abs/1811.10403*, 2018.

Aragon-Team. The Aragon Manifesto. 2018.

Association, A. Introducing the Conviction Funding community grants pilot. https://aragon.org/blog/introducing-the-conviction-funding-pilot, 2020.

de Azevedo Sousa, J. E., Oliveira, V., Valadares, J., Dias Gonçalves, G., Moraes Villela, S., Soares Bernardino, H. and Borges Vieira, A. An analysis of the fees and pending time correlation in ethereum. *International Journal of Network Management*, e2113, 2020.

Benkler, Y. *The Wealth of Networks: How social production transforms markets and freedom*. Yale University Press, 2006.

Brekke, K. R., Königbauer, I. and Straume, O. R. Reference pricing of pharmaceuticals. *Journal of Health Economics*, Vol. 26(3), 613 – 642, 2007. ISSN 0167-6296.

Buterin, V. DAOs, DACs, DAs and More: An Incomplete Terminology Guide. 2014.

Buterin, V. et al. A next-generation smart contract and decentralized application platform. *white paper*, Vol. 3(37), 2014.

Casino, F., Dasaklis, T. K. and Patsakis, C. A systematic literature review of blockchain-based applications: current status, classification and open issues. *Telematics and Informatics*, Vol. 36, 55–81, 2019.

Cawrey, D. 37Coins Plans Worldwide Bitcoin Access with SMS-Based Wallet. 2014.

Chen, T., Li, X., Luo, X. and Zhang, X. Under-optimized smart contracts devour your money. In *2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, 442–446. IEEE, 2017.

Christidis, K. and Devetsikiotis, M. Blockchains and Smart Contracts for the Internet of Things. *IEEE Access*, Vol. 4, 2292–2303, 2016.

Claessen, K. and Hughes, J. Quickcheck: a lightweight tool for random testing of haskell programs. *Acm sigplan notices*, Vol. 46(4), 53–64, 2011.

Cocco, L., Pinna, A. and Marchesi, M. Banking on Blockchain: Costs Savings Thanks to the Blockchain Technology. *Future Internet*, Vol. 9, 25, 2017.

Colombo, P. Committees: an application to delegate certain operations of DAOs. 2020a.

Colombo, P. How does Committees work? 2020b.

Cortes-Goicoechea, M., Franceschini, L. and Bautista-Gomez, L. Resource analysis of ethereum 2.0 clients. *arXiv preprint arXiv:2012.14718*, 2020.

Dannen, C. *Introducing Ethereum and solidity*, Vol. 1. Springer, 2017.

Duncan, J. MolochDAO: a primitive solution. `https://medium.com/metacartel/molochdao-a-primitive-solution-d11cc522b18e`, 2019.

Easley, D., O'Hara, M. and Basu, S. From mining to markets: The evolution of bitcoin transaction fees. *Journal of Financial Economics*, Vol. 134(1), 91–109, 2019.

El Faqir, Y., Arroyo, J. and Hassan, S. An overview of decentralized autonomous organizations on the blockchain. In *Proceedings of the 16th International Symposium on Open Collaboration*. Association for Computing Machinery, New York, NY, USA, 2020. ISBN 9781450387798.

El Faqir, Y., Arroyo, J. and Serrano, A. Visualization of the evolution of collaboration and communication networks in wikis. In *Proceedings of the 15th International Symposium on Open Collaboration*, 1–10. 2019.

Emmett, J. Conviction Voting: A Novel Continuous Decision Making Alternative to Governance. `https://medium.com/giveth/conviction-voting-a-novel-continuous-decision-making-alternative-to-governance-aa746cfb9`, 2019.

ethereumprice. Ethereum Gas Charts. `https://ethereumprice.org/gas/`, 2020.

Faqir-Rhazoui, Y., Ariza-Garzón, M. J., Arroyo, J. and Hassan, S. Effect of the Gas Price Surges on User Activity in the DAOs of the Ethereum Blockchain. 2021a.

Faqir-Rhazoui, Y., Arroyo, J. and Hassan, S. A scalable voting system: Validation of holographic consensus in daostack. In *Proceedings of the 54th Hawaii International Conference on System Sciences*, 5557. 2021b.

FIELD, M. Decentralized Governance Matters. 2018a.

FIELD, M. Holographic consensus—part 1. 2018b.

FIELD, M. Holographic Consensus — Part 2. `https://medium.com/daostack/holographic-consensus-part-2-4fd461e8dcde`, 2019.

FILIPPI, P. D. and HASSAN, S. Blockchain technology as a regulatory technology: From code is law to law is code. *First Monday*, Vol. 21(12), 2016.

FILIPPI, P. D. and WRIGHT, A. *Blockchain and the Law: The Rule of Code*. Harvard University Press, 2018.

GRANGER, C. W. J. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, Vol. 37(3), 424–438, 1969. ISSN 00129682, 14680262.

HOFMANN, F., WURSTER, S., RON, E. and BÖHMECKE-SCHWAFERT, M. The immutability concept of blockchains and benefits of early standardization. In *2017 ITU Kaleidoscope: Challenges for a Data-Driven Society (ITU K)*, 1–8. IEEE, 2017.

HSIEH, Y.-Y., VERGNE, J.-P., ANDERSON, P., LAKHANI, K. and REITZIG, M. Bitcoin and the rise of decentralized autonomous organizations. *Journal of Organization Design*, Vol. 7(1), 1–16, 2018.

HU, Y., VALERA, H. G. A. and OXLEY, L. Market efficiency of the top market-cap cryptocurrencies: Further evidence from a panel framework. *Finance Research Letters*, Vol. 31, 138–145, 2019.

KORDYBAN, S. What is DeFi Yield Farming? `https://hedgetrade.com/defi-yield-farming-defined/`, 2020a.

KORDYBAN, S. Why Ethereum's Gas Prices have Skyrocketed. `https://hedgetrade.com/ethereums-gas-prices-skyrocket/`, 2020b.

LARIMER, D. The Hidden Costs of Bitcoin. 2013.

LIU, B. and SZALACHOWSKI, P. A first look into defi oracles. *arXiv preprint arXiv:2005.04377*, 2020.

LÜTKEPOHL, H. *New introduction to multiple time series analysis*. Springer Science & Business Media, 2005.

MACKINNON, J. G. Approximate asymptotic distribution functions for unit-root and cointegration tests. *Journal of Business & Economic Statistics*, Vol. 12(2), 167–176, 1994. ISSN 07350015.

MACKINNON, J. G. Critical Values For Cointegration Tests. Working Paper 1227, Economics Department, Queen's University, 2010.

MANNAN, M. Fostering worker cooperatives with blockchain technology: Lessons from the Colony project. *Erasmus Law Review*, Vol. 11, 190, 2018.

MARTELLUCCI, E. What are DAOs? 2020.

Mehar, M. I., Shier, C. L., Giambattista, A., Gong, E., Fletcher, G., Sanayhie, R., Kim, H. M. and Laskowski, M. Understanding a revolutionary and flawed grand experiment in blockchain: the dao attack. *Journal of Cases on Information Technology*, Vol. 21(1), 19–32, 2019.

Metcalfe, W. Ethereum, Smart Contracts, DApps. In *Blockchain and Crypt Currency*, 77–93. Springer, Singapore, 2020.

Möser, M. and Böhme, R. Trends, tips, tolls: A longitudinal study of bitcoin transaction fees. In *International Conference on Financial Cryptography and Data Security*, 19–33. Springer, 2015.

Nadler, M. A quantitative analysis of the ethereum fee market: How storing gas can result in more predictable prices. 2020.

Nakamoto, S. and Bitcoin, A. A peer-to-peer electronic cash system. *Bitcoin.–URL: https://bitcoin. org/bitcoin. pdf*, 2008.

Pierro, G. A. and Rocha, H. The influence factors on ethereum transaction fees. In *2019 IEEE/ACM 2nd International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB)*, 24–31. IEEE, 2019.

Rozas, D., Tenorio-Fornés, A., Díaz-Molina, S. and Hassan, S. When Ostrom meets blockchain: Exploring the potentials of blockchain for commons governance. *Available at SSRN 3272329*, 2018.

Santander, A. and One, A. My First Aragon App: Voting supercharged with DAOstack's Holographic Consensus (Part 4). 2019.

Sassi, F., Belloni, A., Mirelman, A. J., Suhrcke, M., Thomas, A., Salti, N., Vellakkal, S., Visaruthvong, C., Popkin, B. M. and Nugent, R. Equity impacts of price policies to promote healthy behaviours. *The Lancet*, Vol. 391(10134), 2059 – 2070, 2018. ISSN 0140-6736.

Seabold, S. and Perktold, J. statsmodels: Econometric and statistical modeling with Python. In *9th Python in Science Conference*. 2010.

Shaw, A. and Hill, B. M. Laboratories of oligarchy? how the iron law extends to peer production. *Journal of Communication*, Vol. 64(2), 215–238, 2014.

Sims, C. A. Macroeconomics and reality. *Econometrica*, Vol. 48(1), 1–48, 1980. ISSN 00129682, 14680262.

Soleimani, A., Bhuptani, A., Young, J., Haber, L. and Sethuram, R. The Moloch DAO. 2019.

Sousa, J. E. d. A., Oliveira, V., Valadares, J., Vieira, A. B., Villela, S., Bernardino, H. and Dias, G. An analysis of the fees and pending time correlation in ethereum. *Int. J. Netw. Manag.*, 2020.

Sovbetov, Y. Factors influencing cryptocurrency prices: Evidence from bitcoin, ethereum, dash, litcoin, and monero. *Journal of Economics and Financial Analysis*, Vol. 2(2), 1–27, 2018.

TIKHOMIROV, S., VOSKRESENSKAYA, E., IVANITSKIY, I., TAKHAVIEV, R., MARCHENKO, E. and ALEXANDROV, Y. Smartcheck: Static analysis of ethereum smart contracts. In *Proceedings of the 1st International Workshop on Emerging Trends in Software Engineering for Blockchain*, 9–16. 2018.

TURLEY, C. Moloch Evolved: V2 Primer. `https://medium.com/raid-guild/moloch-evolved-v2-primer-25c9cdeab455`, 2020.

UNDERWOOD, S. Blockchain beyond bitcoin. 2016.

VALIENTE BLÁZQUEZ, M.-C., HASSAN, S. and PAVÓN MESTRAS, J. Results and experiences from developing daos with aragon: A case study. 2017.

VUJIČIĆ, D., JAGODIĆ, D. and RANDIĆ, S. Blockchain technology, bitcoin, and Ethereum: A brief overview. In *2018 17th international symposium INFOTEH-JAHORINA*, 1–6. IEEE, 2018.

WANG, S., DING, W., LI, J., YUAN, Y., OUYANG, L. and WANG, F.-Y. Decentralized autonomous organizations: concept, model, and applications. *IEEE Transactions on Computational Social Systems*, Vol. 6(5), 870–878, 2019.

WOOD, G. ET AL. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper*, Vol. 151(2014), 20, 2014.

ZARGHAM, M. Sensor Fusion and Social Choice. `https://github.com/BlockScience/conviction/blob/master/social-sensorfusion.pdf`, 2018.

ZHENG, Z., XIE, S., DAI, H.-N., CHEN, X. and WANG, H. Blockchain challenges and opportunities: A survey. *International Journal of Web and Grid Services*, Vol. 14(4), 352–375, 2018.