

Article

An Infrastructure Cost and Benefits Evaluation Framework for Blockchain-Based Applications

Miguel Pincheira ^{1,*} , Elena Donini ² , Massimo Vecchio ¹  and Raffaele Giaffreda ¹¹ Digital Industry Center, Fondazione Bruno Kessler, 38123 Trento, Italy² Digital Society Center, Fondazione Bruno Kessler, 38123 Trento, Italy

* Correspondence: mpincheiracar@fbk.eu

Abstract: Blockchain is currently a core technology for developing new types of decentralized applications. With the unique properties of blockchain, unique challenges and characteristics are introduced to the system. Among these characteristics, the infrastructure costs and benefits of the system are critical to evaluate the feasibility of any system and have yet to be addressed in the current literature. This work presents a framework for evaluating blockchain applications' infrastructure costs and benefits. The framework includes a taxonomy to classify the related transactions, a model to evaluate the infrastructure costs and benefits in applications using public or private blockchains, and a methodology to guide the use of the model. The model is based on simple parameters that describe the systems, and the methodology helps to identify and estimate these parameters at any stage of the application life cycle. We quantitatively analyze three real use cases to demonstrate the framework's merit. The analyses highlight the model's accuracy by achieving the same results presented in the use cases. Furthermore, the use-case analyses emphasize the framework's potential to evaluate different scenarios across the entire life cycle of blockchain-based applications.

Keywords: blockchain; software; infrastructure; costs; benefits; evaluation



Citation: Pincheira, M.; Donini, E.; Vecchio, M.; Giaffreda, R. An Infrastructure Cost and Benefits Evaluation Framework for Blockchain-Based Applications. *Systems* **2023**, *11*, 184. <https://doi.org/10.3390/systems11040184>

Academic Editors: Philippe Mathieu, Juan M. Corchado, Alfonso González-Briones and Fernando De la Prieta Pintado

Received: 31 January 2023

Revised: 16 March 2023

Accepted: 31 March 2023

Published: 5 April 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Blockchain is recognized as one of the most important technology disruptions in the latest years [1]. Researchers and practitioners have shown increasing interest in leveraging blockchain technology's unique benefits and properties to empower new software systems [2]. Applications in several domains have adopted private [3] and public blockchain networks [4,5] to provide a software platform where interactions between actors can occur without intermediaries. Thus, from a software perspective, blockchain has been mainly characterized as an architecture component that provides immutable storage and computational capabilities in a decentralized way. The unique properties of this software component also raise new challenges across the systems development life cycle while introducing unexplored characteristics that need to be identified and evaluated [1,6]. Much of the current literature has focused on non-functional characteristics of blockchain-based systems, such as scalability, security, and performance [6,7].

However, blockchain is still in the early stages of adoption [8], and more is needed to know about other characteristics supporting the system development, deployment, and evaluation. Among these characteristics, the infrastructure to support the application is critical to evaluate the potential monetary value in terms of costs and benefits across the entire application life cycle. Costs and benefits can greatly increase the adoption of blockchain on software systems [1], as blockchain technology is a major target of investments for companies [7]. Although there has been some research on evaluating the infrastructure costs of blockchain applications [1,9], these findings are focused on particular use cases and are not extendable to other scenarios. Each application and use case may have unique

requirements and characteristics that must be considered to evaluate its blockchain infrastructure [1]. Therefore, further research is necessary to fully understand blockchain applications' potential costs and benefits across various use cases and contexts.

This work presents an infrastructure cost framework for blockchain-based software systems, extending a first version of the model presented in [10]. We propose a framework to evaluate application costs and benefits from the early stages of development to the exploitation phase, based on the blockchain software component of the system. Our goal is to provide a tool matching the existing literature on the cost and benefits of blockchain-based applications while also providing the tools to analyze more complex cost scenarios.

The framework comprises a transaction taxonomy, a cost and benefits model for public and private networks, and a methodology to apply the model. First, the taxonomy aims to classify and generalize typical transactions in the life cycle of blockchain applications. Then, the infrastructure cost and benefits model aims to create evaluation scenarios over the entire application's life cycle based on simple application parameters. The selection of these parameters aims to simplify the characterization of the system with minimal application domain knowledge. Finally, we propose a methodology for identifying and estimating the model parameters. To illustrate the proposed framework's usability, we quantitatively analyze the monetary costs and benefits of three blockchain-based applications from the current literature. These evaluations highlight the flexibility of the model to work for public and private blockchains, the simplicity of identifying the model parameters using the proposed methodology, and the benefits of the framework to analyze different scenarios for the entire application life cycle.

The contribution of this paper is three-fold. (i) We propose a transaction taxonomy for blockchain-based applications. (ii) We define an infrastructure cost model for private and public blockchains. (iii) We propose a methodology that identifies and estimates the model parameters to create and evaluated scenarios over the application's life cycle.

The rest of this paper is structured as follows: Section 2 describes similar works and highlights the gap in the literature; Section 3 formalizes the problem our framework addresses; Section 4 describes the proposed framework by detailing the transaction taxonomy, cost model, and methodology; Section 5 uses the methodology to apply the model and evaluate three real use cases. We finalize the paper with our conclusion and plans for future work.

2. Related Works

In recent years, there has been an increasing amount of literature on blockchain as a core component for developing new software systems. From a system and software-engineering perspective, blockchain applications present inherent challenges, such as scalability, security, and performance [6] given by a specific life cycle that introduces unique constraints and characteristics in the system [1]. Much of the current literature on blockchain-based systems pays particular attention to high-level characteristics such as the levels of permissions or types of actors to evaluate which software components can benefit from blockchain [11] or if other software components are sufficient [12]. However, monetary costs are still marginally addressed.

In this context, one of the first studies addressing the topic of costs for a blockchain-based system is presented in [13]. The authors aimed to quantify the current scalability limits of Bitcoin, and from that goal, they performed a small exploratory analysis of estimating monetary costs. However, since they focus on the scalability aspects of Bitcoin, they do not provide an approach compatible with a software-system perspective. Similarly, the authors of [14] address the cost of a blockchain-based system in the context of a blockchain-based digital payment. The authors rely on private Ethereum infrastructure and present a brief analysis of the costs of managing the private architecture. Their model focuses on the rewarding costs for miners and the costs of network resources, based on the same scalability metrics described in [13].

A different approach to cost is presented in [15] by describing an evaluation framework that identifies factors influencing a blockchain application from a financial perspective (i.e., cost savings and benefits). However, the proposal has a high-level approach that divides the framework into five focus areas: the purpose of the blockchain, the features of the blockchain, the cost reductions derived from using blockchain, environmental and motivational factors for using blockchain, and the actual implementation and operations costs. Therefore, infrastructure costs are linked to only one focus area and are addressed with a narrow perspective. Similarly, the authors of [16] present a cost comparison between the Ethereum blockchain and Amazon under the context of business process execution for supply chain applications. In this work, the authors focus on operation costs under different architecture choices and provide a good model for the Ethereum network. However, they focus on the comparison against Amazon Simple Workflow Service but need more generalization for other application scenarios or blockchain architectural choices.

Recently, authors of [17] presented an overview of blockchain-based applications with a focus on smart contracts in the Ethereum network. On the one hand, the work emphasized how Ethereum is becoming the preferred platform for developing blockchain-based applications, a fact also highlighted by surveys such as [18,19]. On the other hand, the study presented several metrics regarding the application, such as the level of open-source and the usage of patterns in smart contracts. Here, the authors focused only on analyzing existing applications, using gas usage as a metric to evaluate the costs of blockchain-based systems on public networks. Nonetheless, they do not provide a model or structure framework for this evaluation.

The importance of gas usage as a cost metric is also underlined by the authors of [20] in their study about developing cost-effective blockchain-powered applications. The authors emphasize that developers of these applications need to understand the gas of their smart contract through the entire application lifecycle (deployment and usage). Furthermore, the authors state that transactions with high gas usage will frequently have the same priority as transactions with low gas usage when using the same gas price, despite the difference in transaction fees. To leverage this topic, the authors propose a gas usage prediction models to help developers make informed decisions regarding gas prices. However, the authors do not address the cost of deploying or issuing the transaction in the context of a software system; they only focus on the gas price. The authors of [9] conducted preliminary work on infrastructure costs for blockchain-based systems based on gas usage. They provided a simple monetary cost model for the required infrastructure in a farm-to-fork case study but did not offer enough information to generalize to other architectures or case studies.

Summarizing, the infrastructure cost of blockchain-based applications has been marginally analyzed in current literature, which leans to focus on metrics such as scalability and performance or high-level concerns such as motivation or other environmental factors. Furthermore, only a few works provide frameworks or models for these cost analyses. There is a need for an approach that covers the entire lifecycle of blockchain-based applications or needs more details to provide enough generalization to evaluate different scenarios, particularly on public and private infrastructures.

3. Problem Definition

Our proposal aims to provide a model to allow actors and stakeholders to evaluate the economic feasibility of a blockchain-based application. The model seeks to estimate costs and monetary benefits during the entire life cycle of the application (i.e., from the early stages of development to a more advanced stage of exploitation). Furthermore, the model is general enough to evaluate a system using a private or a public blockchain network.

First, we consider a blockchain-based application as the software that supports the interactions of a group of actors identified only by their private/public keys. These actors have limited types of interactions to create and transfer information and value among them. We consider that the entire application logic is in the blockchain so that all information and interactions in the application are immutable, auditable, and accessible by anybody.

Further, we consider that the application has a two-phase life cycle: bootstrap and operation, similarly to the phases deployment and operation presented in [16]. It is important to notice that off-chain computations and side-chains are beyond the scope of this paper.

Using this definition of a blockchain-based application, we defined the cost of the application $C(m)$ as the infrastructure needed to process and store the interactions I between the actors A . These interactions generate a value unit K , which provides the monetary benefits $B(m)$ for a group of stakeholders S .

Therefore, we estimate $C(m)$ and $B(m)$ based on the characteristics of the application and the blockchain network that supports it, i.e., the nodes running the network. The number of nodes and configurations depends on the blockchains' technical implementation (e.g., consensus model). Furthermore, the configuration follows the application requirements (e.g., transaction life-span, latency, and throughput) and the existing trust between stakeholders [21]. However, the type of blockchain used for the application (i.e., public or private) makes a great difference in estimating the costs and benefits. On public blockchains, transactions that create new information (i.e., modify the state of the blockchain) have a monetary cost. Conversely, the transaction number does not directly affect the infrastructure cost in private blockchains. Therefore, we divided the model into two parts: the public blockchain cost model and the private blockchain cost model, described in the following section.

4. Proposed Cost and Benefit Model

In this section, we describe our proposed model to evaluate a blockchain-based application's infrastructure costs and benefits.

First, we define a transaction taxonomy since our model considers interactions among actors and stakeholders (i.e., transactions) as the functional units for an application's life-cycle assessment (LCA). The taxonomy is called *CRIV* and classifies the interactions between the actors in the framework of blockchain applications during all the development and exploitation phases.

Then, we define the costs and benefits model using the proposed taxonomy and a few simple system parameters. We model the costs of public and private blockchains separately, given that the transaction fees depend on the network type. Similarly, we model the monetary benefits for the stakeholders associated with a generic blockchain that can be public or private.

Finally, we proposed a methodology that guides the model's use by helping identify and estimate the model parameters.

Table 1 lists and describes the mathematical symbols used as the parameters for the model.

Table 1. Parameters for the proposed cost model.

Parameter	Description	Parameter	Description
A_0	Number of initial actors	S	Number of stakeholders
K	Value unit	L_k	K lifespan
$P_C(m)$	Cryptocurrency price	P_K	Value unit price
P_{node}	Node price	F_I	Interaction factor
μ	Time factor for transactions	F_g	Growth factor
O_C	Computational cost of T_C	F_V	Value factor
O_R	Computational cost of T_R	F_t	Trust factor
O_I	Computational cost of T_I	F_o	Operation factor
O_V	Computational cost of T_V	F_k	Benefit factor

4.1. Proposed Transaction Taxonomy for Blockchain Applications

As described in the previous section, the interactions among actors in a blockchain-based application are transactions. From the set \mathcal{I} of all possible transactions, we focus only on a subset $\mathcal{T} = \{T_i, i = [C, R, I, V]\} \subset \mathcal{I}$ of transactions that creates new information for the application and the actors. Considering that these transactions vary greatly from one application to another, we propose the *CRIV* taxonomy to easily identify core transactions and link them to the application's life cycle. *CRIV* categorizes the interactions into four types of transactions: creation T_C , registration T_R , interaction T_I , and value T_V . Figure 1 shows the four transactions in our taxonomy along with the general life cycle of the application. Each type of transaction is defined as follows:

- Creation transaction T_C deploys the application into the blockchain. T_C may include one or more transactions happening during the bootstrap phase.
- Registration transaction T_R is required at the first interaction of an actor with the system to make the actor part of the application.
- Interaction transaction T_I are the most common interactions between actors. They produce information to be stored in the blockchain, without including any value transfer.
- Value transaction T_V is the most important transaction as it includes the value transfer between unknown actors.

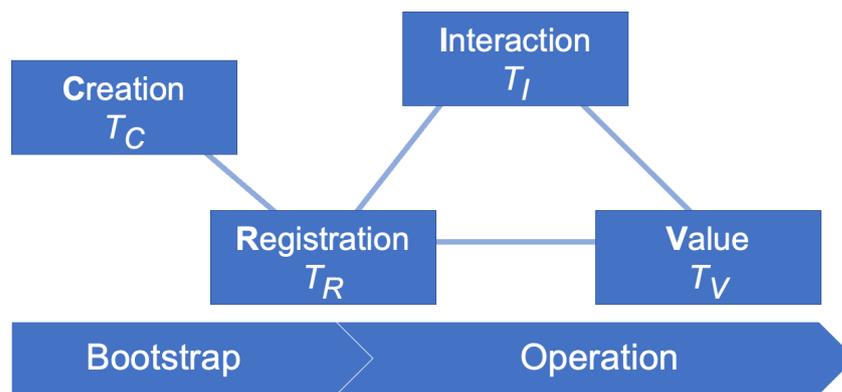


Figure 1. Life cycle of a blockchain-based application.

4.2. Proposed Public Blockchain Cost Model

Given the life cycle of an application shown in Figure 1, we divide the cost model into two components, i.e., the bootstrap C_B and operation C_O costs. C_B considers the transactions needed to deploy the application logic (T_C) and the transactions to register the initial actors (T_R). C_O considers the transactions for the registration of new actors (T_R), the interaction transactions between actors (T_I), and the transactions that transfer value (T_V). Here, considering the general life cycle of an application, C_B and C_O are evaluated in a given month m , defined as the minimal time window for assessing the systems. This window makes it easier to make comparisons with other types of monetary evaluations (e.g., budget planning). However, the cost model can easily adapt to shorter and longer windows. Hence, the initial month ($m = 0$) corresponds to the bootstrap phase, and any other month ($m > 0$) indicates the operation phase. We define the costs of a public blockchain infrastructure $C_{pub}(m)$ as:

$$C_{pub}(m) = \begin{cases} C_B(m) = CT_C(m) + CT_R(m), & \text{if } m = 0 \\ C_O(m) = CT_R(m) + CT_I(m) + CT_V(m) & m > 0 \end{cases} \quad (1)$$

where $CT_i(m)$ is the total monetary costs paid in a month m , for all the transactions of type i , where $i = \{C, R, I, V\}$. The monetary cost of a single transaction of type i links the computational cost O_i of the transaction with the price of the cryptocurrency $P_C(m)$ and a processing time factor μ . Finally, the total number of transactions of type i in a given month

m is given by $Q_i(m)$. The price of the cryptocurrency $P_C(m)$ is a function that the user must define by considering the high volatility of the cryptocurrency price and the scenario to evaluate. For instance, the user can use historic cryptocurrency prices to define $P_C(m)$ with a fixed value for any month (i.e., an average for all months). Similarly, the user can define $P_C(m)$ with a different monthly value (i.e., an average for each month). The factor μ is used to scale the price paid for each transaction (i.e., a transaction fee). Transactions with higher prices are more attractive for the node operators and typically are processed faster since the node that processes the transaction will receive the fee as a reward [20]. The total cost $CT_i(m)$ for each transaction i is given by:

$$CT_i(m) = O_i P_C(m) \mu Q_i(m), \quad \text{with } i \in \{C, R, I, V\} \quad (2)$$

In the operational phase, the total number of transactions of each type i in a given month m is directly related to the number of actors $A(m)$ in the system defined as:

$$A(m) = \begin{cases} A_0, & \text{if } m = 1 \\ A(m-1) F_g & m > 1 \end{cases} \quad (3)$$

where A_0 is the initial number of actors in the system and F_g is the actor growth factor that describes the growth of the system in terms of actors related to the time unit m such that:

$$F_g = (A(m)/A(m-1)) - 1 \quad (4)$$

For instance, a $F_g = 0.5$ means that if $A(m-1) = 100$, then at $A(m) = 150$. Finally, the number of each type of transaction $Q_i(m)$ in the operation phase ($m > 0$) with respect to the actor number is defined as:

$$Q_i(m) = \begin{cases} 0 & \text{for } i = T_C, m = 0 \\ A(m) - A(m-1) & \text{for } i = T_R, m \geq 0 \\ A(m) F_I & \text{for } i = T_I, m > 0 \\ A(m) F_V & \text{for } i = T_V, m > 0 \end{cases} \quad (5)$$

where $Q_C(m)$ is equal to 0 after the bootstrap phase ($m > 0$). $Q_R(m)$ is given by the number of new actors in that month. $Q_I(m)$ links the total number of actors in the system $A(m)$ using an interaction factor F_I . F_I relates to the expected number of interaction transactions T_I of each actor. For example, if each actor is expected to have at least two T_I in the time unit m (e.g., T_I per month), the factor is set to $F_I = 2$. Lastly, $Q_V(m)$ links the total number of actors with factor F_V , which represents the value transfer transactions T_V of each actor. For instance, when actors are expected to have at least one T_V every two months, the factor is set to $F_V = 0.5$. The user can estimate the values of F_I and F_V at an early stage of development. In a more advanced stage, the factors can be estimated from the current activity in the system.

4.3. Proposed Private Blockchain Cost Model

For applications based on a private blockchain, we define the infrastructure cost $C_{Pri}(m)$ in a given month m as divided into two components C_B and C_O , indicating the bootstrap and the operation phase, respectively. C_B is the initial investment to acquire N nodes (i.e., computers) with a price P_{node} to create the network infrastructure. C_O is the expense of running and operating the nodes. Similar to the traditional software systems, we estimate the operating costs C_O as a percentage of P_{node} using a scale factor F_o . F_o is estimated by considering the system characteristics in terms of the hardware and software required to run the nodes. The infrastructure cost $C_{Pri}(m)$ in a private blockchain is defined as:

$$C_{Pri}(m) = \begin{cases} C_B(m) = N P_{node} & \text{if } m = 0 \\ C_O(m) = N F_o P_{node} & m > 0, \end{cases} \quad (6)$$

In our model, the node number N is related to the number of stakeholders S by a trust factor F_t :

$$N = S (1 - F_t). \quad (7)$$

where F_t is the relation between the N and the total number of stakeholders. For instance, if 100 stakeholders agree that only 30 different nodes are required to support the infrastructure, there is a trust factor of 70%. Similarly, a 100% trust factor will translate into a centralized system. Here, for simplicity, one node represents one stakeholder.

4.4. Proposed Model for Monetary Benefits

For applications based on both public and private blockchains, the monetary benefits $B(m)$ for the stakeholders are derived from the value units K transacted in the application and are given by:

$$B(m) = F_k Q_V(m) P_K \quad (8)$$

where F_k is the benefit factor that indicates the expected value units for each value transfer transaction T_V , $Q_V(m)$ is the number of value transfer transactions in the month m , and P_K is the price (i.e., total monetary value) of the value unit K . P_K is the sum of all the monetary values assigned to each stakeholder S (i.e., the benefit for each stakeholder). In a public network, this value may also be linked to the price of the cryptocurrency, such as $P_K(m) = 0.4 P_C(m)$.

4.5. Proposed Methodology

From a software system perspective, a methodology is a procedure to help understand the steps needed to perform a task with such a system [22]. Here, we propose a methodology of five steps to guide the people behind the blockchain-based application (i.e., the user) to use our model to perform a monetary evaluation of the application. The methodology groups the model parameters into four categories, using the relations between the parameters. These four groups translate into four steps (S1–S4), providing a simplified incremental approach to identifying them. The last step of the methodology (S5) is the actual monetary evaluation of the application and includes a series of proposed analyses. The five steps of the methodology are:

- (S1) Define the blockchain setup: The first step aims at selecting the blockchain network on which the application will be implemented (e.g., Ethereum, Hyperledger). Assign the values for the network's node price P_{node} , the operation factor F_o , the cryptocurrency price $P_C(m)$, and the time factor μ .
- (S2) Identify the actors and stakeholders: Once the blockchain network is selected, the second step aims at identifying the actors A_0 (i.e., who uses the application), stakeholders S (i.e., who is interested in the application), the value unit K , and its lifespan L_k (i.e., what is transacted in the application and how long it lasts). Estimate how the number of actors will change F_g and how much trust is between the stakeholders F_t .
- (S3) Estimate the computational cost of interactions: The third step strives to apply the proposed CRIV taxonomy to identify the different types of transactions and estimate their computation costs O_c , O_r , O_i , and O_v . Estimate the interaction factor F_i and the value factor F_v .
- (S4) Identify the benefits: The fourth step aims to give a monetary value to the value unit P_k and estimate the benefits factor F_k . If these parameters can not be identified, the model can still be used for estimating the costs, but the benefits will not be available. At the end of this step, all the model parameters have been identified, even if some values could not be estimated.
- (S5) Evaluate scenarios: Once all parameters have been identified, the values assigned provide a scenario to evaluate the blockchain-based application's cost and benefits. Changing the model parameters value can provide different scenarios to compare (different

F_g , different $P_C(m)$. Some of the most common evaluations include: using Equation (1) to estimate the bootstrap and operation costs on a public network or using Equation (6) for a private network. Another example is using Equation (8) for estimating the benefits. Furthermore, equations and parameters can be combined to obtain other evaluations. For instance, dividing Equation (1) by the number of actors on a given month $C(m)/A(m)$ can provide an estimate of how much each actor will pay for the system operation. For each evaluation, varying the model parameters value can provide different scenarios to compare (different F_g , different $P_C(m)$). These are just a few examples of the model's usability.

5. Evaluation of the Proposed Model

To evaluate the correctness and goodness of our proposal, we evaluated a series of blockchain-based applications in the current literature [4,23–25]. We selected applications using Ethereum, as it is a reference implementation for smart contracts and can be used in both public and private scenarios [18]. However, our proposed model can be used with any other blockchain implementation.

We present three use cases: a water management system using a public network, a medical image system that can be used on public or private networks, and a manufacturing traceability application using a private network. For each use case, we show how to apply the proposed model following our methodology for defining the model parameters. Given that all applications run on the Ethereum network, the first step of the methodology is common for all use cases. Then, for each use case, we provide analyses highlighting our model's potential when evaluating the costs and benefits of blockchain-based applications.

5.1. (S1) Define the Blockchain Setup

We define Ethereum as the blockchain network for all use cases. The cryptocurrency price $P_C(m)$ is the price of Ethereum, using historical values that are available online (Etherscan¹). The computational cost O_i of the transactions is equal to the gas required for their execution, and μ is the gas price on Ethereum, expressed in gwei. For the cost of the node P_{node} , we consider the minimum hardware requirements for an Ethereum node². At the time of writing, this translates into a computer of USD 300. We consider the operation factor for the node as 40% of the cost of the node, this $F_o = 0.4$.

5.2. Use Case: Water Management System

The authors of [4] present an architecture for a blockchain-based IoT water management system. The authors implement a prototype using Ethereum as a public blockchain and constrained IoT devices as data sources. The authors evaluate focused on the IoT devices and provide implementation details of the smart contracts. Some parameters of our model are clearly expressed (i.e., A, S, K, L_k), while others require a brief analysis to be estimated (i.e., P_k). Thus, following our methodology, the steps are:

5.2.1. (S2) Identify Actors and Stakeholders

The group of actors A comprises farmers using IoT devices to measure water consumption (i.e., a valve). The stakeholders S are three organizations interested in encouraging water savings (i.e., an energy company, NGO, and certification authorities), thus $S = 3$. The value unit K is a cubic meter of the saved water, and the lifespan unit is a day, as water usage is reported daily. We estimated the initial number of actors is $A_0 = 100$ with a monthly growth of 5%, making $F_g = 0.5$, based on the information described in the paper and the references within it.

5.2.2. (S3) Estimate the Computation Cost of Interactions

The application is based on two types of smart contracts with four types of transactions. These transactions can be directly mapped to our taxonomy C_C, C_R, C_I , and C_V . The computation cost is calculated based on the gas usage reported for the transactions. The

farmers report their water consumption once a week, which translates into four $F_I = 4$ (four transactions T_I per month), and they receive their rewards once a month, thus, $F_V = 1$ (one transaction T_V per month).

5.2.3. (S4) Identify the benefits

Each actor saves on average 4 m^3 in 100 ha farm as described in similar studies [26]. Hence, the benefit factor is set to $F_k = 4$. The authors do not provide a monetary value for a m^3 of saved water (P_K), so it must be estimated based on the document and its references. We estimated that the energy company offers a discount of USD 0.2 for each saved m^3 . We considered NGO assigns to the savings a value equal to the cost of irrigation m^3 at USD 0.8 (according to [26]). Finally, we assume a “eco-friendly” label will translate into USD 10 additional monthly benefits. Thus, the total monetary value of K is USD 11. Table 2 summarizes the value of the parameters for the application, obtained following the proposed methodology, and that will be used for evaluating scenarios.

Table 2. Parameters for the cost model of the water-management system from [4].

Parameter	Value	Parameter	Value
A_0	100	S	3
K	Saved m^3 of water	L_k	1 day
$P_C(m)$	205	P_K	USD 11
P_{node}	USD 300	F_I	4
μ	2, 5, 10 gwei	F_g	0.05
O_C	3,343,572	F_V	1
O_R	143,947	F_t	0.01
O_I	26,821	F_o	0.4
O_V	156,580	F_k	1

5.2.4. (S5) Evaluate Scenarios

In their work, the authors present a brief evaluation of the transaction costs using three different gas prices (i.e., 2, 5, and 10 gwei) with a cryptocurrency price equal to USD 205 (based on the yearly average for 2019), as shown in Table 3. Our model uses Equation (2) and the proposed taxonomy to obtain these values.

Table 3. Transaction costs for the water management use case [4].

	Gas	2 Gwei	5 Gwei	10 Gwei
Valve Creation	143.947	USD 0.059	USD 0.148	USD 0.295
setValue()	26.821	USD 0.011	USD 0.027	USD 0.055
App Creation	3.343.572	USD 1.371	USD 3.427	USD 6.854
Reward	156.580	USD 0.064	USD 0.160	USD 0.321

Furthermore, our model can extend the author’s evaluation to different scenarios. Considering the monthly average price for 2019, we can use Equations (1), (6), and (8) to evaluate the costs and benefits of the application for a year. Figure 2 shows the benefits ($Benef$), the total monthly cost of a private network ($CPri$), and the total monthly for a public network using 2, 5, and 10 gwei ($CPub_2$, $CPub_5$, $CPub_{10}$, respectively), evaluated for the year 2019.

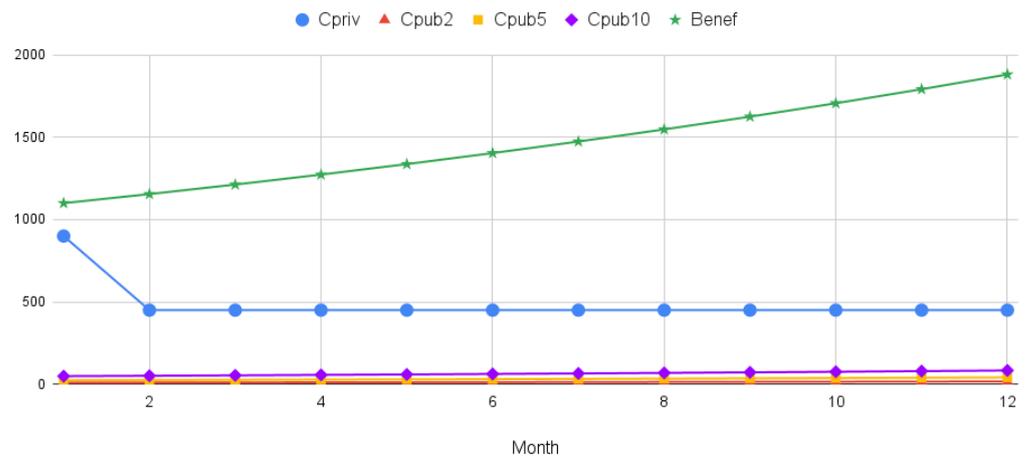


Figure 2. Monthly costs and benefits of the water management system.

With this use case, we highlight the correctness of our model to match the existing literature on cost. Furthermore, we highlight its potential for analyzing more complex cost scenarios with minimal additional information. For instance, those developing the blockchain-based application should easily identify the values we have estimated (i.e., K , P_k).

5.3. Use Case: Patient-Centric Image Management System

Jabarulla and Lee propose a blockchain-based patient-centric image management system [24]. They developed a proof-of-concept using the Ethereum blockchain and a distributed storage system. The authors validate their proposal with experiments and evaluate gas usage as a metric for executing functions. Furthermore, they assigned a monetary value to the gas to provide a price reference for the system.

5.3.1. (S2) Identify Actors and Stakeholders

The actors A are patients, doctors, and practitioners involved. The value unit K is a medical image with a lifespan L_K of 3 months. As presented in the paper, we define $A_0 = 4$ with a growth factor $F_g = 0.75$. However, more information is needed to identify the stakeholders S .

5.3.2. (S3) Estimate the Computation Cost of Interactions

Based on the source code provided by the authors, we can obtain the value for O_C . The authors provide the gas usage for the contract functions, and using our taxonomy, we can obtain the values for O_R and O_V . There needs to be more detail to estimate O_I based on the three functions described, so we average the values. Then, we define F_I and F_V as 1, meaning we consider sharing one image (F_I) and accessing one image (F_V).

5.3.3. (S4) Identify the Benefits

The authors state that an average transaction price of USD 0.11 is lower than existing solutions for managing patients' images. Therefore, we consider P_K as USD 0.11 and set F_K as 1. Table 4 summarizes the model's parameters.

Table 4. Parameters for the cost model of patient-centric image management system [24].

Parameter	Value	Parameter	Value
A_0	4	S	-
K	medical image	L_k	3 months
$P_C(m)$	USD 187	P_K	USD 0.11
P_{node}	USD 300	F_I	1
μ	2 gwei	F_g	0.75
O_C	1,611,435	F_V	1
O_R	67,397	F_t	-
O_I	113,510	F_o	0.4
O_V	170,412	F_k	1

5.3.4. (S5) Evaluate Scenarios

The authors use a cryptocurrency price of USD 187 and a gas price of 2 gwei to provide an average transaction price of USD 0.11. With our model, we can obtain the average transaction price by dividing the total costs Equation (2) by the number of actors Equation (3). This operation renders a value of USD 0.12, where the minimal difference is due to the estimation of the computation cost of interactions. Then, we extend the author's evaluation by analyzing the impact of adding more images per user (changing the parameter F_I) impact the costs. Figure 3 shows this cost with a baseline of USD 0.12.

**Figure 3.** Comparison of average transaction price using different values for F_V and F_I .

This evaluation highlights the correctness of our model to match existing approaches to evaluate costs. Furthermore, the model offers additional value by providing the tools to evaluate different scenarios even if a few parameters can not be estimated.

5.4. Use Case: Automotive Manufacturing Traceability

Kuhn et al. [23] propose a blockchain-based traceability architecture to process manufacturing data. The authors present an evaluation of gas used per transaction as a metric for scalability without a monetary evaluation. Compared with the other use cases, this application does not provide enough information to estimate several model parameters. However, the model can still be used as follows.

5.4.1. (S2) Identify Actors and Stakeholders

The value unit is manufactured (i.e., electrical contacts) with a lifespan of L_k of one day. The stakeholders S are the companies involved in the manufacturing process, each providing a node for the blockchain network with $N \in [10, 50]$. Since each stakeholder provides a node, the trust factor is F_t is 0. The actors A are the machines and devices in the

production process with $A \in [10, 100]$. Unfortunately, there is not enough information to estimate a growth factor F_g .

5.4.2. (S3) Estimate the Computation Cost of Interactions

All the interactions are managed through a single contract based on the ERC1155 token standard and deployed on a private Ethereum network. The paper provides results regarding gas usage for a single stakeholder, processing a batch of 3000 items, which means 3000 transactions. However, more details are needed for applying the taxonomy or estimating F_v and F_i .

5.4.3. (S4) Identify the Benefits

Based on the paper [23] and the references within, we can define P_K as USD 0.5 for each processed unit using a Benefit factor $F_k = 1$. Table 5 summarizes the value of the parameters for the use case.

Table 5. Parameters for the cost model of the architecture for automotive traceability [23].

Parameter	Value	Parameter	Value
A_0	50	S	25
K	electric contact	L_k	1 day
$P_C(m)$	-	P_K	USD 0.5
P_{node}	USD 300	F_I	-
μ	-	F_g	0
O_C	-	F_V	-
O_R	-	F_t	-
O_I	-	F_o	0.4
O_V	-	F_k	1

5.4.4. (S5) Evaluate scenarios

Although if this use case only provides enough information for estimating 6 of the 18 parameters, it can be used to calculate additional cost information. For instance, using Equation (6), the bootstrap cost is USD 7.500 for acquiring 25 nodes, and the operation cost is fixed at USD 3500. Then, making the monthly costs equal to the benefits described by Equation (8), the total number of transactions $Qv(m)$ should be 7000 to reach an equilibrium value.

This use case highlights the usability of the model, even when not all parameters can be defined or estimated. With only 6 of the 18 parameters, the model provided the tools to find a monetary balance point for a private Ethereum network.

5.5. Discussion

In the previous sections, we evaluated our proposed model and methodology with three use cases from the current literature. In the first use case, we demonstrated that our model could match the existing literature on cost and has the potential to analyze more complex cost scenarios. In the second use case, we further demonstrated the correctness of our model in matching existing cost evaluation approaches, even if a few parameters cannot be estimated. In the last use case, we highlighted the usability of our model with minimal available information. The results and rationale of using our framework in these use cases highlighted the usability of the selected parameters and the methodology to identify them. On the one hand, our selection of static parameters, even if it can be considered a limitation, strikes simplicity and effectiveness and proved useful, particularly with little application domain knowledge. On the other hand, the proposed methodology to guide the users was also demonstrated to be effective in maintaining a streamlined and straightforward approach while still being widely applicable. Finally, the quantitative information showcased by the example cost and benefits analysis performed on each use

case provides an empirical reference that can further enrich the growing field of studying blockchain-based systems.

6. Conclusions and Future Works

In this paper, we presented a framework for evaluating the costs and benefits of the blockchain-based system across its entire life cycle. The proposed framework includes a transaction taxonomy, a cost and benefit model, and a methodology to use the model. We used the methodology to apply the proposed model and quantitatively evaluate the cost and benefits of three use cases found in the current literature. The analyses highlight the model's accuracy and usability in evaluating different types of blockchain-based applications. In particular, the proposed methodology emphasizes the simplicity of identifying and estimating the model parameters. Once the parameters have been identified, the evaluation shows how to assess different scenarios by simply varying the values of some parameters. Furthermore, the diverse use cases provided different application details, showcasing the model's potential even when some parameters could not be identified or estimated. All these features make our proposed methodology a valuable tool for organizations that want to estimate the costs associated with implementing blockchain-based systems in various domains. By leveraging our model's usability and flexibility, they can make informed decisions about such systems' feasibility and expected returns on investment. Additionally, the empirical reference provided by our quantitative information can serve as a benchmark for future research in this field, enabling researchers to explore new areas of study more effectively. Overall, our model and methodology offer a powerful combination of simplicity, effectiveness, and versatility that can benefit industry practitioners and academic researchers.

Future works include assessing other use cases to improve the methodology and provide reference values for the model parameters. Similarly, studying which parameters can benefit dynamic values is an interesting research path. Finally, studying a possible hybrid model combining public and private blockchain networks could enhance our proposed framework.

Author Contributions: Conceptualization, M.P. and R.G.; Methodology, M.P. and E.D.; Validation, M.P. and E.D.; Formal analysis, M.P. and E.D.; Investigation, M.P. and E.D.; Data curation, M.P. and E.D.; Writing—original draft, M.P. and E.D.; Supervision, M.V. and R.G.; All authors have read and agreed to the published version of the manuscript.

Funding: This work was partly supported by the project "AI@TN" funded by the Autonomous Province of Trento.

Data Availability Statement: Data available on request due to restrictions. The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

Notes

¹ <https://etherscan.io/chart/etherprice>(accessed on 20th January 2023).

² <https://docs.ethhub.io/using-ethereum/ethereum-clients/geth/> (accessed on 20th January 2023).

References

1. Vacca, A.; Di Sorbo, A.; Visaggio, C.A.; Canfora, G. A systematic literature review of blockchain and smart contract development: Techniques, tools, and open challenges. *J. Syst. Softw.* **2021**, *174*, 110891. [[CrossRef](#)]
2. Fahmideh, M.; Grundy, J.; Ahmad, A.; Shen, J.; Yan, J.; Mougouei, D.; Wang, P.; Ghose, A.; Gunawardana, A.; Aickelin, U.; et al. Engineering Blockchain-Based Software Systems: Foundations, Survey, and Future Directions. *ACM Comput. Surv.* **2022**, *55*, 1–44. [[CrossRef](#)]
3. Mistry, I.; Tanwar, S.; Tyagi, S.; Kumar, N. Blockchain for 5G-enabled IoT for industrial automation: A systematic review, solutions, and challenges. *Mech. Syst. Signal Process.* **2020**, *135*, 106382. [[CrossRef](#)]
4. Pincheira, M.; Vecchio, M.; Giaffreda, R.; Kanhere, S.S. Cost-effective IoT devices as trustworthy data sources for a blockchain-based water management system in precision agriculture. *Comput. Electron. Agric.* **2021**, *180*, 105889. [[CrossRef](#)]

5. Pincheira, M.; Donini, E.; Giaffreda, R.; Vecchio, M. A Blockchain-Based Approach to Enable Remote Sensing Trusted Data. In Proceedings of the 2020 IEEE Latin American GRSS ISPRS Remote Sensing Conference (LAGIRS), Santiago, Chile, 22–26 March 2020; pp. 652–657. [\[CrossRef\]](#)
6. Destefanis, G.; Marchesi, M.; Ortu, M.; Tonelli, R.; Bracciali, A.; Hierons, R. Smart contracts vulnerabilities: A call for blockchain software engineering? In Proceedings of the 2018 International Workshop on Blockchain Oriented Software Engineering (IWBOSE), Campobasso, Italy, 20 March 2018; pp. 19–25. [\[CrossRef\]](#)
7. Xu, X.; Lu, Q.; Liu, Y.; Zhu, L.; Yao, H.; Vasilakos, A.V. Designing blockchain-based applications a case study for imported product traceability. *Future Gener. Comput. Syst.* **2019**, *92*, 399–406. [\[CrossRef\]](#)
8. Wöhrer, M.; Zdun, U. Architectural Design Decisions for Blockchain-Based Applications. In Proceedings of the The 3rd IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Sydney, Australia, 3–6 May 2021.
9. Pincheira, M.; Vecchio, M.; Giaffreda, R. Characterization and Costs of Integrating Blockchain and IoT for Agri-Food Traceability Systems. *Systems* **2022**, *10*, 57. [\[CrossRef\]](#)
10. Pincheira, M.; Donini, E.; Vecchio, M.; Giaffreda, R. Towards an Infrastructure Cost Model for Blockchain-Based Applications. In *Blockchain and Applications, Proceedings of the 4th International Congress, Lille, France, 11–13 July 2023*; Prieto, J., Benítez Martínez, F.L., Ferretti, S., Arroyo Guardado, D., Tomás Nevado-Batalla, P., Eds.; Springer International Publishing: Cham, Switzerland, 2023; pp. 345–355.
11. Wessling, F.; Ehmke, C.; Hesenius, M.; Gruhn, V. How much blockchain do you need? towards a concept for building hybrid dapp architectures. In Proceedings of the 2018 IEEE/ACM 1st International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB), Gothenburg, Sweden, 27 May 2018; IEEE: New York, NY, USA, 2018; pp. 44–47.
12. Wüst, K.; Gervais, A. Do you need a blockchain? In Proceedings of the 2018 Crypto Valley Conference on Blockchain Technology (CVCBT), Zug, Switzerland, 20–22 June 2018; IEEE: New York, NY, USA, 2018; pp. 45–54.
13. Croman, K.; Decker, C.; Eyal, I.; Gencer, A.E.; Juels, A.; Kosba, A.; Miller, A.; Saxena, P.; Shi, E.; Sirer, E.G.; et al. On Scaling Decentralized Blockchains. In *Financial Cryptography and Data Security*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 106–125.
14. Hu, Y.; Manzoor, A.; Ekparinya, P.; Liyanage, M.; Thilakarathna, K.; Jourjon, G.; Seneviratne, A. A Delay-Tolerant Payment Scheme Based on the Ethereum Blockchain. *IEEE Access* **2019**, *7*, 33159–33172. [\[CrossRef\]](#)
15. Demir, M.; Turetken, O.; Ferworn, A. A Financial Evaluation Framework for Blockchain Implementations. In Proceedings of the 2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, Canada, 17–19 October 2019; pp. 715–722. [\[CrossRef\]](#)
16. Rimba, P.; Tran, A.B.; Weber, L.; Staples, M.; Ponomarev, A.; Xu, X. Quantifying the Cost of Distrust: Comparing Blockchain and Cloud Services for Business Process Execution. *Inf. Syst. Front.* **2020**, *22*, 489–507. [\[CrossRef\]](#)
17. Wu, K.; Ma, Y.; Huang, G.; Liu, X. A first look at blockchain-based decentralized applications. *Softw. Pract. Exp.* **2021**, *51*, 2033–2050. [\[CrossRef\]](#)
18. Kondo, M.; Oliva, G.A.; Jiang, Z.M.; Hassan, A.E.; Mizuno, O. Code cloning in smart contracts: A case study on verified contracts from the Ethereum blockchain platform. *Empir. Softw. Eng.* **2020**, *25*, 4617–4675. [\[CrossRef\]](#)
19. Oliva, G.A.; Hassan, A.E.; Jiang, Z.M.J. An exploratory study of smart contracts in the Ethereum blockchain platform. *Empir. Softw. Eng.* **2020**, *25*, 1864–1904. [\[CrossRef\]](#)
20. Zarir, A.A.; Oliva, G.A.; Jiang, Z.M.; Hassan, A.E. Developing cost-effective blockchain-powered applications: A case study of the gas usage of smart contract transactions in the ethereum blockchain platform. *ACM Trans. Softw. Eng. Methodol. (TOSEM)* **2021**, *30*, 1–38. [\[CrossRef\]](#)
21. Schäffer, M.; Di Angelo, M.; Salzer, G. Performance and scalability of private Ethereum blockchains. In Proceedings of the International Conference on Business Process Management, Rome, Italy, 6–10 September 2019; Springer: Berlin, Germany, 2019; pp. 103–118.
22. Karat, J. Software Evaluation Methodologies. In *Handbook of Human-Computer Interaction*; Helander, M., Ed.; Elsevier: Amsterdam, The Netherlands, 1988; pp. 891–903. [\[CrossRef\]](#)
23. Kuhn, M.; Funk, F.; Franke, J. Blockchain architecture for automotive traceability. *Procedia Cirp* **2021**, *97*, 390–395. [\[CrossRef\]](#)
24. Jabarulla, M.Y.; Lee, H.N. Blockchain-based distributed patient-centric image management system. *Appl. Sci.* **2020**, *11*, 196. [\[CrossRef\]](#)
25. Kudva, S.; Norderhaug, R.; Badsha, S.; Sengupta, S.; Kayes, A. PEBERS: Practical Ethereum Blockchain based Efficient Ride Hailing Service. In Proceedings of the 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), Doha, Qatar, 2–5 February 2020; pp. 422–428. [\[CrossRef\]](#)
26. Giannakis, E.; Bruggeman, A.; Djuma, H.; Kozyra, J.; Hammer, J. Water pricing and irrigation across Europe: Opportunities and constraints for adopting irrigation scheduling decision support systems. *Water Supply* **2015**, *16*, 245–252. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.