A Domain-specific e^{3}value Extension for Analyzing Blockchain-based Value Networks*

Simon Curty[1][0000−0002−2868−9001]✉ and Hans-Georg Fill[1][0000−0001−5076−5341]

University of Fribourg, Digitalization and Information Systems Group,
Bd de Pérolles 90, 1700 Fribourg, Switzerland
{simon.curty,hans-georg.fill}@unifr.ch
https://www.unifr.ch/inf/digits/en/

Abstract. Adopting blockchain technologies in organizations has multiple implications for business models. To make adoption successful, both the business as well as the technical perspectives must be carefully aligned. However, understanding the impact of the technological changes on business models is a challenge due to the technological complexity, the lack of knowledge in the organization, and regulatory requirements. Further, domain-specific modeling methods that inherently deal with blockchain concepts in business models are currently missing. To address this gap, we present an extension of the *e^{3}value* modeling method to depict blockchain-specific aspects in value networks, including the automatic inference of transparency based on blockchain usage and configuration. The extended modeling method was implemented on the ADOxx metamodeling platform and applied to three exemplary use cases for a first evaluation.

Keywords: Blockchain · Business model · Enterprise modeling · e^{3}value · Value network

1 Introduction

In recent years, major advances have been made in distributed ledger technologies (DLT), commonly known as blockchains. Improvements, e.g., in transaction volumes and energy efficiency give rise to a potential wider adoption [8,30]. The intrinsic qualities of this novel family of technologies, such as decentralized and tamper-proof storage [11], promise opportunities for the digital transformation of businesses and enable new business cases. However, the successful adoption and integration of blockchain technology in an enterprise in real-world scenarios remains challenging. This is on one hand due to: a. the complexity and the comparatively low maturity of the technological ecosystem [13], as is evident by the few standards that have been established so far, and b. by raising concerns, e.g., regarding interoperability with existing IT systems or cross-blockchain operations. On the other hand, organizational barriers such as the involvement of

* This research was partially funded by the Swiss National Science Foundation [196889]

regulatory requirements, the availability of financial and human resources, and insufficient knowledge about the technology in the organization requiring the development of new skills and competencies may prevent adoption in practice [2]. A major challenge in designing and realizing a business model that reverts to some blockchain-based system is to deal with the complexity in aligning institutional, market, and technology factors [24]. This includes for example leveraging the unique properties in the business model, the positioning on the market, and the implementation and engineering challenges specific to this technology [26].

Techniques and methods incorporating concepts dedicated to blockchains help to address these business challenges. However, such approaches have not been explored extensively. Modeling support for the design and analysis of blockchain business models is comparatively sparse [6]. For this reason, we propose a domain-specific extension of the e^3value method for supporting the design of blockchain-based business models. Thereby, an emphasis lies on the increased transparency of blockchain-based applications and the analysis of its propagation through value networks. We consider the transparency of records as one of the most desirable properties of blockchains from a business perspective. On a more general level, transparency is an economic measure to alleviate the information asymmetry between parties, ensuring that no side may have a potentially unfair advantage due to the availability of information [22]. For example, transparency of information may be offered as value proposition to customers and partners (e.g., [3]) or to facilitate trust between parties enabling collaborative efforts (e.g., [21]). For reducing undesirable effects of asymmetric information, blockchains have been proposed previously, e.g., in the context of bank credits [36], and commodity markets [28].

The remainder of this paper is structured as follows. In Section 2 we will introduce foundations on blockchains and blockchain-based systems, and related work regarding business models and the design of blockchain-based business models in particular. In Section 3 we will present a domain-specific extension of the e^3value modeling method to support the design of blockchain-based business models, which will be applied to exemplary use cases in Section 4. The paper will conclude with a discussion of the approach in Section 5 and an outlook to further research in Section 6.

2 Foundations and Related Work

In the following we present brief foundations on blockchain technologies that are necessary for our approach, the representation of business models, and outline prior work on domain-specific languages for blockchain-based business models.

2.1 Blockchain

Blockchain-based applications rely on distributed ledger technologies (DLT), which store transactions between authorized parties in a decentralized, distributed, immutable and trustful way [12]. This is achieved through so-called consensus algorithms that guarantee the validity of transactions. Further, so-called *smart contracts* may be added to transactions in some blockchains for the decentralized execution of algorithms [1]. The access to the ledger may either be restricted to certain parties (*permissioned blockchain*) or it may be openly accessible (*public blockchain*).

According to the recent ISO standard 23635 on Blockchain Governance, blockchain-based applications should follow several principles for ensuring the effective, efficient, and acceptable use of DLT systems [23]. These include for example the support of *openness and transparency* so that stakeholders can observe and audit the dynamics of the system, the *alignment of incentive mechanisms* with the used consensus algorithms and the application's objectives, the provision of *security mechanisms* and the consideration of *privacy impacts and compliance obligations*, or requirements regarding the *interoperability* with other DLT or non-DLT systems.

2.2 Representation of Business Models

The concept of a business model is today commonly regarded as an integrated view on the organization of an enterprise for contributing to the successful management in the decision-making process [38]. It includes information about the interplay of an organization's strategy, resources, customers, market offerings, and revenues, as well as underlying processes and services. For dealing with the complexity of these aspects, various approaches have been proposed. These include formal and semi-formal representations as found in enterprise modeling methods such as 4EM [34], MEMO [15], ArchiMate [20], or SOM [9] as well as specialized approaches such as e^{3}value [18] or the business model canvas (BMC) [31], which is a popular but only graphical way of representing and analyzing these aspects that can however be transitioned to a semi-formal representation as well [37]. Whereas enterprise modeling approaches take a holistic perspective on business and IT aspects, e^{3}value or the BMC focus on the exchange of value between actors. Thereby, e^{3}value is a language and set of techniques for representing and analyzing value networks, i.e., who exchanges what kind of value with whom and what expenses and revenues are created for each actor [18].

2.3 Languages for Blockchain-based Business Models

Although domain-specific languages and extensions of existing languages for the modeling of blockchain-based business models is a sparsely researched topic [6], several business ontologies and modeling methods have been proposed. In the following we present a selection of these works focusing on organizational and enterprise modeling, and the representation of business models. For a comprehensive overview of modeling methods in the context of DLT, we refer readers to a recent literature survey [6].

An approach to overcome the challenges of integration and adoption of DLT on the organizational level is to revert to enterprise and business ontologies. The Resource-Event-Agent model [29] describes a general business ontology about the relationships of economic concepts. In combination with DEMO, a methodology for enterprise modeling, de Kruijff et al. described a domain ontology for blockchains as a common terminology for business and technical actors alike [7]. Another ontological approach has been presented by Kim et al., where the Toronto Virtual Enterprise Ontology was extended with concepts for provenance tracking in supply chains [27]. Such ontologies may then for example be used as foundation for the design of smart contracts.

Further approaches can be found in the field of Enterprise Architecture for depicting the integration of DLT into the IT architecture or for representing views on the organization in relation to DLT concepts. Jiang and Ræder used a combination of ArchiMate strategy and motivation models for modeling value chains built on blockchain technology [25]. Another approach based on ArchiMate includes a holistic top-down methodology for the design of blockchain-based applications that reverts to ArchiMate's core layers (*business, application, technology*) [3]. Thereby, business models are first explored by drafting a business model canvas [31] and process models. The business model is then reflected on the business layer of the integrated ArchiMate model created subsequently. The alignment of a blockchain business model and the underlying software and IT infrastructure was discussed in [5], where an NFT use case was modeled with ArchiMate.

In the context of blockchain, e^3value concepts have been mapped to Solidity code constructs with the aim to generate blueprints for services [16]. In contrast, Poels et al. [33] apply e^3value to analyze the viability of DLT business cases. They propose a model pattern to identify business cases where the implementation of DLT could be beneficial. An extension of e^3value for blockchain business models was described by [32]. In particular, this work introduces decentralized autonomous organizations as model element and various DLT-related attributes to existing elements. Interactive analysis and the visualization of blockchain value networks regarding transparency and privacy are not considered. Our work is based on the implementation of this extension. However, the design goals and conceptualizations differ substantially.

In summary, the representation of blockchain-based business models is not a well-explored topic. Graphical modeling languages and tools for modeling support are in need as to facilitate the understanding of DLT on organizations. In particular, methods for interactive analysis of transparency, data privacy issues, and system interactions in blockchain-based business models are amiss. Rather than creating a standalone domain-specific language to address these issues, we propose a domain-specific extension to the well-known e^3value method.

3 Extending e^3value for Blockchain-based Applications

E^3value is a modeling methodology for representing and analyzing electronic business models based on an ontology for supporting the development of e-commerce systems [17]. A business model is seen as a set of actors exchanging value to increase their economic utility, thereby forming a so-called *value net-*

work. In the following, the core concepts of e^{3}value are briefly summarized [18]. Figure 1 shows the basic graphical notation of the modeling elements.

Actors are independent entities that hold the responsibility to ensure their own survival, well-being, and success—the definition of which varies for individual actors.

Value objects are things of value, that is, something valuable to at least one actor. Such an object can be of any nature, e.g., a physical product, a service, currencies, an experience, etc.

Market segments present groups of actors who individually have the same notion on the value of a value object.

Partnerships (or composite actors) are collaborations in which actors co-operate with the goal to offer some value to other actors or market segments.

Value activities are performed by actors to increase utility, generate profit, or advance their mission.

Value ports represent the intent to offer or accept a value object. Value ports are either incoming or outgoing.

Value interfaces group together ports to define atomic value exchanges. Actors, market segments, and value activities may have value interfaces, indicating willingness to exchange the specified value objects. A value interface has at least one outgoing and one incoming port.

Value transfers (or value exchanges) connect an outgoing value port to an incoming one, representing transfers of value objects between the two ports.

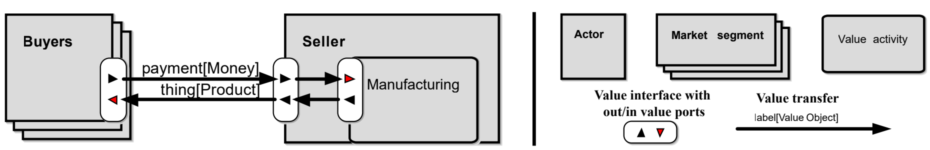


Fig. 1: Standard graphical notation of e^{3}value showing a sample model on the left and the graphical notation on the right.

For the development of a domain-specific extension of e^{3}value to support the design of blockchain-based applications we revert to the *macro process* by Frank [14]. This process consists of seven cyclic phases (*micro processes*), providing guidelines for designing a domain-specific modeling language as summarized in the following:

1. *Clarification of scope and purpose*: extension of the e^{3}value method to support the design and analysis of value networks involving blockchain-based software components.
2. *Analysis of generic requirements*, i.e., requirements that apply to every DSML. We consider the generic requirements of the catalog outlined by Frank [14]. Since our contribution is an extension of an existing modeling language, and for the sake of brevity we omit a discussion on the generic requirements.

3. *Analysis of specific requirements*, i.e., requirements that apply to the artifact in particular. We will present these in Section 3.1.
4. *Language specification*, in particular specification of the metamodel and constraints: The metamodel and related inference mechanisms will be presented in Section 3.2.
5. *Design and documentation of graphical notation*: The graphical notation of the extensions will be shown in Section 3.3.
6. *Development of modeling tool*: The implementation is discussed in Section 3.4.
7. *Evaluation and refinement*: The extensions were evaluated and continuously refined throughout the development against the requirements. In Section 4 we show the feasibility of the extension for modeling blockchain value networks by means of three exemplary use cases.

3.1 Requirements

At first, we derived seven specific requirements ($\mathbf{SR}_{1-7}$) for the extension on the basis of use scenarios. This includes tasks for which the method should be applicable, as well as descriptions of use cases that the models should capture. The fundamental use case is based on the intended application of e^{3}value, namely the modeling of value networks as part of a business model. We extend this with analysis tasks regarding blockchain specific properties and their impact on the value network. More specifically, we were interested in depicting the *diffusion of transparency of transactions* in a blockchain network. To refine and analyze collected requirements, mock diagrams were created. This allows for example to clarify visualization and usability concerns. The specific requirements were:

- $\mathbf{SR}_1$: The core semantics of e^{3}value elements should not be altered fundamentally. Users familiar with e^{3}value should find the extension to be straightforward in its use of core elements.
- $\mathbf{SR}_2$: A central benefit of e^{3}value is the comparatively low number of elements. Thus, only a minimal set of elements and attributes should be added.
- $\mathbf{SR}_3$: The modeling concepts should be technology-agnostic, i.e., the domain-specific extensions should be applicable to various blockchain technologies.
- $\mathbf{SR}_4$: Concepts should be provided for specific distributed ledger constructs such as smart contracts, oracles, and decentralized autonomous organizations (DAO) as most DLT use cases involve some of these constructs.
- $\mathbf{SR}_5$: Concepts for blockchain networks, sub-networks, and access control should be provided as access restrictions on the network level determine who can inspect or modify the ledger (public/permissioned ledgers). Networks may be segmented to allow for more fine-grained access control, e.g., to form complex consortia in permissioned blockchains.
- $\mathbf{SR}_6$: The user should be provided with mechanisms to analyze what value transfers are transparent to which parties through an automatic inference of transparency across the value network – see the aforementioned ISO standard [23]. That is, a value transfer is transparent to a party (actor, market segment, partnership) when information on the occurrence of the transfer is visible to a party upon inspection.

- **SR**$_7$: Transparent transfers on a blockchain involving some sensitive information render a business model non-viable without the necessary precautions as also mentioned in ISO 23635 [23]. Thus, modeling support should be provided for detecting the exchange of sensitive information in a blockchain-based business model.

3.2 Metamodel

The collected requirements formed the basis for the specification of the language metamodel. For this, we reverted to the original e^3value metamodel as published in the user guide [18] and extended it with DLT-specific concepts as follows (see Figure 2):

- **MM**$_1$: The concepts *smart contract*, *oracle* and *DAO* are represented as types of the *Actor* class (**SR**$_4$). This design choice prevented the change of the e^3value actor concept and rather extended it to preserve its original semantics (**SR**$_1$). Further, this allows to form partnerships of DLT actors, and subsequently capture the structure of DAOs in detail.
- **MM**$_2$: A class dedicated to representing the concept of a blockchain network is introduced. Further, we defined two access modes, *public*, for networks without any access restrictions, and *permissioned*, for networks where such restrictions are in place in some form (**SR**$_5$).
- **MM**$_3$: Two super-classes have been added to allow for elements to be aware of partaking networks. Notably, aggregations are introduced for grouping elements, e.g., a network groups actors.
- **MM**$_4$: The relation *part of* specifies the participation of an element in some network. Similarly, the *carrier* relation denotes that a value transfer is carried out over the related network (**SR**$_6$).
- **MM**$_5$: Value transfers have been extended with three additional attributes: The attribute *Sensitive* denotes that the transfer involves information that should not be disclosed to some or all parties (**SR**$_7$). *Off-chain* explicitly denotes that the transfer is not carried out over a network, whether or not the source and target would suggest otherwise. Finally, *Override Network* allows to manually specify the carrier network. This is of use in complex modeling scenarios, or when the automatic selection of a carrier network is ambiguous due to modeling restrictions.

Deciding on the carrier network is the main concern for analyzing the transparency of value transfers (**SR**$_6$), as this property directly depends on the access mode of the network. Actors with access to a network can inspect the ledger and all recorded blockchain transactions. The ledger of a public network is accessible to everyone, whether or not the actor directly participates in the network, e.g., by operating a node. Participation in a network is expressed with the *part of* relation. Consuming a business service hosted by a public network does not require participation. Consequently, value transfers carried out by a public network are transparent, but this property does not propagate beyond the first value interface of an actor or market segment outside of a network.

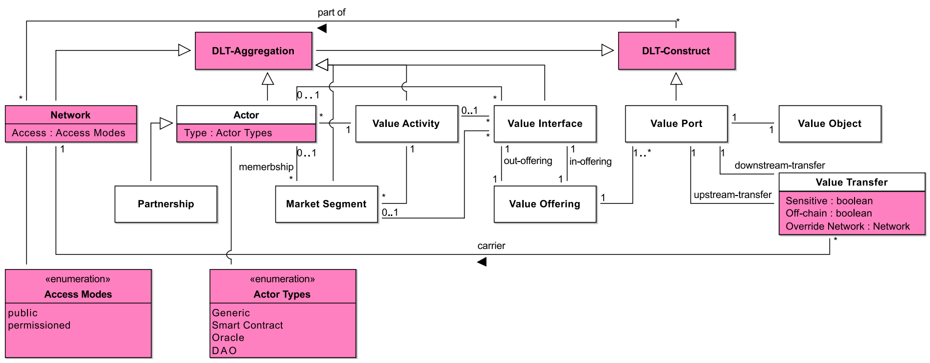
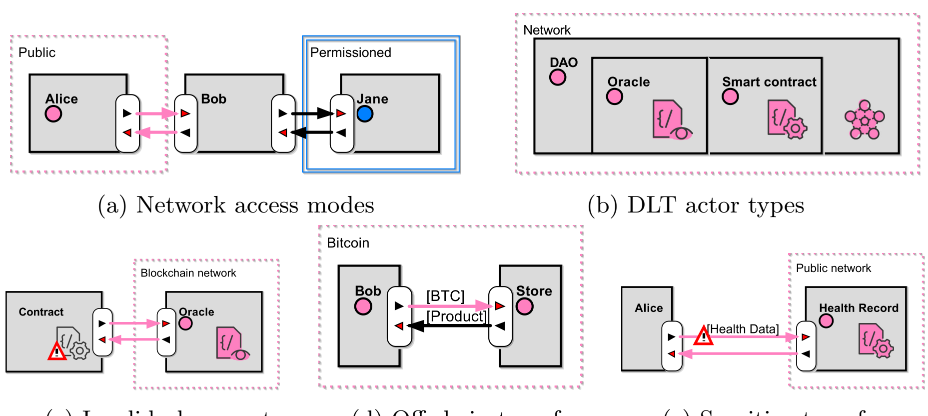


Fig. 2: Extensions of the simplified metamodel of e^{3}value. The colored classes and attributes have been added. Attributes of e^{3}value are omitted.

In a permissioned network, the ledger is accessible only to authorized parties. As such, a transparent value transfer requires the outgoing and receiving value interfaces to be in the same (permissioned) network. Propagating the transparency property beyond the network boundary is impossible in this case. In case multiple networks qualify as carrier, the network lowest in the hierarchy is selected, e.g., a sub-network.

3.3 Graphical Extensions and Modeling Patterns

Of all concepts introduced by the proposed extension, only the *Network* class requires a new dedicated modeling element ($\mathbf{SR}_{2,5}$). A network is thereby simply represented as a rectangle, aggregating contained elements. A network's access



(a) Network access modes (b) DLT actor types

(c) Invalid placement (d) Off-chain transfer (e) Sensitive transfer

Fig. 3: Graphical patterns of the extensions of e^{3}value to support the modeling and analysis of blockchain-based value networks.

mode is signified by a variation of the rectangle's border style. Figure 3a illustrates the basic transfer patterns between actors ($\mathbf{MM}_2$). The notation of existing elements corresponds to the examples shown in the e^3value user guide [18]. Each participation of an actor in some (sub-)network is expressed by a circular badge in the same color as the related network ($\mathbf{MM}_4$). An icon is displayed that corresponds to the actor's type (see 3b). The individual actor types are depicted in Figure 3b, where a DAO is a partnership of a smart contract and an oracle ($\mathbf{MM}_1$). The color of the icon relates to the main network, e.g., the network in which a smart contract is deployed. It is impossible for DLT entities to exist outside of a network. To make modelers aware of such an invalid placement, a warning symbol is displayed (see Figure 3c). Value transfers have the same color as the carrier network if one exists (see Figures 3a, 3d). That is, *carrier* relations are represented by matching coloring of transfers and their carrier network ($\mathbf{MM}_4$). Value transfers that are *off-chain* ignore this coloring rule (see Figure 3d). If a value transfer is *sensitive* and transparent to all ($\mathbf{SR}_7$), i.e., carried out over a public network, the transfer is decorated with a warning symbol (see Figure 3e). Thereby, the transfer patterns shown in Figures 3d and 3e correspond to the metamodel extension ($\mathbf{MM}_5$). Combining these features results in the five basic graphical patterns of the transfer properties and actor types, shown in Figure 3. These patterns commonly occur in blockchain-based business models. That is, models are a combination of these basic patterns.

3.4 Implementation

The e^3value language with the extension has been prototypically implemented using the ADOxx metamodeling platform [10]. ADOxx was chosen for its maturity, acceptance in academia and industry, and suitability for prototyping modeling methods. Further, the implementation is based on an existing e^3value library for modeling blockchain-inspired businesses [32]. Elements containing others are realized as ADOxx aggregations, notably actors, networks and market segments. Thereby, a binary *is inside* relation is automatically derived for visually contained elements. As such, the *part of* relation (see Section 3.2) needs no special implementation. The *carrier* relation between a value transfer and a network is established by an ADOxx expression attribute on the value transfer relation, which computes at run-time the information required for the display of the graphical patterns. Actor types, access modes and some visual parameters, e.g., the color of networks, are implemented as user modifiable attributes on their respective elements. The modeling library for ADOxx is openly available [4].

4 Exemplary Use Cases

In accordance with the *macro process*, the extension was continuously evaluated. Among other measures, the continuous analysis of use cases contributed to refining the method. By means of three fictitious use cases that are inspired by previous, informal discussions with industry experts, we present the application of the extension for designing and analyzing blockchain-based business models.

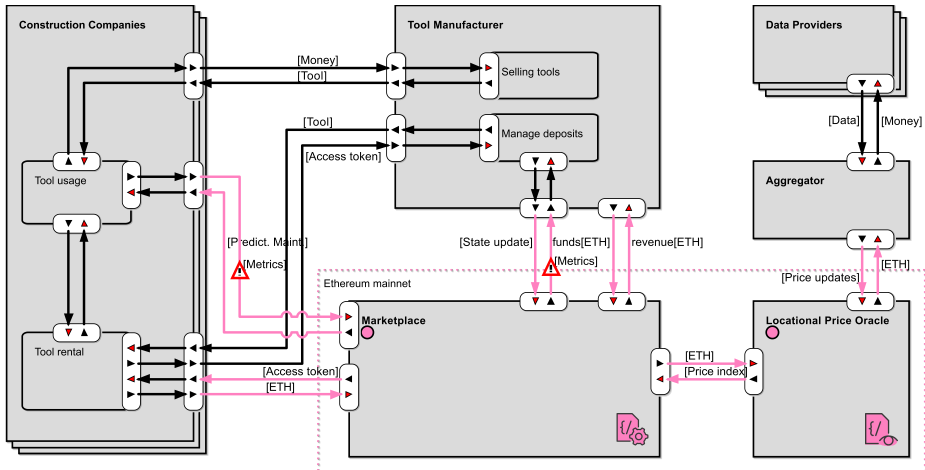


Fig. 4: Value network of a tool rental service realized as decentralized marketplace. The tool supplier operates the marketplace as a smart contract on the public blockchain of the Ethereum mainnet. The pricing model is supported by an oracle service that provides up-to-date regional pricing data.

4.1 Decentralized marketplace for tool rental

A supplier of construction tools explores options for a tool rental marketplace for professionals. This use case is inspired by existing tools-on-demand programs[1]. In the scenario shown in Figure 4, the marketplace is realized by a smart contract on the public Ethereum mainnet[2] blockchain, thereby leveraging an existing, highly available and durable infrastructure. The marketplace contract contains a registry of all tools for rental. Construction companies may acquire a personal token for retrieving a tool by paying some amount of Ether (ETH), which is the cryptocurrency of Ethereum. The contract supports regionally adjusted pricing. This is achieved by fetching a regional price index for this product category through an oracle. The tool may then be fetched with the access token from a deposit operated in the region of the manufacturer. Further, the tools function as smart devices, i.e., collecting data and metrics. These are used for analysis purposes, e.g., for development or predictive maintenance. In this scenario, the metrics are collected through the smart contract so that no central server must be operated. Apart from technical issues, this is problematic as the metrics contain personal information of the customer. As such, the related value transfers have been marked as sensitive and a warning sign is shown.

In this modeling scenario, the extension offers insights that would not be apparent otherwise: First, it is immediately clear which actors are blockchain-

[1] See for example the one by Hilti: https://www.hilti.com/content/hilti/W1/US/en/business/business/equipment/fleet/tools-on-demand.html

[2] https://ethereum.org

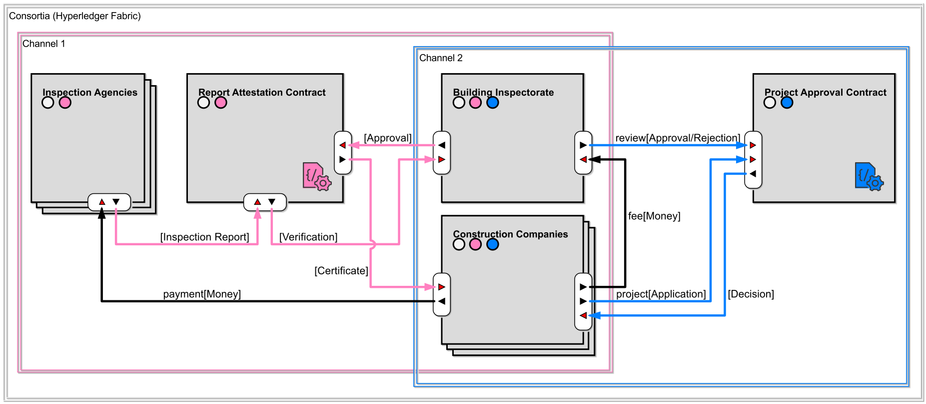


Fig. 5: Value network of a consortium for the construction project proposals and management of inspection reports using two channels.

based. The nature of an actor is expressed by its actor type, indicating its role in the value network. Transparency of value transfers is automatically visualized at design time. It is thus clear, for whom transfers are transparent and how this property propagates. For example, everyone can inspect, what regional price index was provided by the oracle. However, the data set used by the aggregator to calculate the index remains unknown. Equally, the exposure of sensitive data is evaluated at design time. This offers immediate feedback on the viability of the business model: transmission of sensitive information over a public network could violate regulations, cause privacy issues, or result in economic risks.

4.2 Building inspectorate consortium

The building inspectorate, a government agency, wishes to modernize their IT systems for approving construction project proposals and submitting building inspection reports. This use case is inspired by a real-world application of blockchain in construction planning [35]. Regulations demand that these systems be kept separate. Construction companies, external inspection agencies and the building inspectorate form a consortium connected through a permissioned blockchain using the Hyperledger Fabric platform as shown in Figure 5. The two services are operated as smart contracts in dedicated sub-networks, i.e., so-called channels in Hyperledger, which only authorized consortium members can access. As such, information in one sub-network is not exposed to another one. This scenario does not require any kind of cryptocurrency. Instead, payments are made traditionally, off-chain.

Here, the concept of *networks and sub-networks* as additional modeling element allows to separate parts of the value network. Hierarchies of (sub-)networks enable the modeling of complex value networks where actors may participate in several networks, visually indicated by colored badges. This, together with the

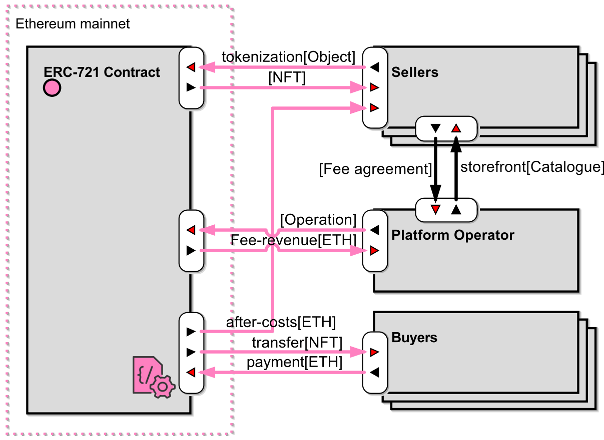


Fig. 6: Value network of an NFT marketplace enabled by an ERC-721 smart contract on the Ethereum mainnet.

automatic inference of the carrier network for each value transfer, clarifies value network partitions and actor interactions at first glance. The coloring of value transfers eases the distinction between on-chain and off-chain value transfers.

4.3 Non-fungible token marketplace

So called non-fungible tokens (NFT) are unique cryptographically verifiable representations of an underlying asset. The asset can be physical, e.g., a property, or digital, e.g., an image. The NFT then encapsulates some type of rights transfer of the associated asset, e.g., for digital art and photography [5]. Here, we regard a derivation of the business model of existing marketplaces[3].

Oftentimes, NFTs may be traded on decentralized marketplaces, whereby sales and transfers of funds and tokens are handled by smart contracts. Such a scenario is shown in Figure 6. The marketplace contract is based on ERC721[4], a token standard for the implementation of NFTs for Ethereum. It handles the creation, the so-called *minting*, and transfer of tokens. Additionally, it serves as payment channel. The platform operator offers an off-chain web-based storefront for advertising tokens available for sale. Usage of the marketplace and storefront requires payment of a fee. When a sale closes, the token, and thereby the ownership of the associated digital art, is transferred by the contract to the buyer upon receipt of sufficient payment in cryptocurrency. After deducting the platform fees, funds are transferred to the seller. In summary, the business model of the platform operator involves operation of the marketplace as a service.

Most value transfers in this scenario are carried out over the blockchain. The e^{3}value extension makes this immediately apparent by automatically coloring the

[3] https://opensea.io/
[4] https://eips.ethereum.org/EIPS/eip-721

value transfers. Thus, one can derive implications for the viability of the business model: blockchain transactions in public networks generally incur costs in form of a transaction fee, required to pay for infrastructure operation. This could discourage both sellers and buyers from using such a marketplace. Furthermore, a traditional storefront, e.g., a website, is still required as user-friendly interface for the customers.

5 Discussion

Value networks are well-suited for representing blockchain-based business models. There is conceptual overlap of what is represented in a value network and how a blockchain network operates. However, considering blockchain-specific concepts is required for facilitating a comprehensive reasoning on these business models. The proposed extension aims to fill this gap. By reverting to e^3value, a known method for representing electronic business models, we leverage its qualities, e.g., the manageable number of modeling elements. As such, the method is suitable for users already familiar with e^3value. A main challenge in dealing with complexities of blockchain-based business models is to convey the necessary domain knowledge so it can be communicated among non-experts. Our method supports this by inferring and visualizing the propagation of trust—one of the main benefits organizations hope to reap by adopting DLT [2]. The modeling process can be further simplified by reverting to basic modeling patterns, combining them as needed to represent the business model. Through exploring various modeling scenarios, we have found that these patterns are general and applicable in many business cases.

The presented approach originates from the idea of increasing the transparency of information through blockchain networks. The motivation behind relies on the theory of asymmetric information [22] and the problem of incomplete contracts [19]. On the other hand, information transparency is a value proposition in a business model, given its benefit for customers or partners. While transparency of information via public records is one of the major properties of blockchain technology, only vague claims can be made regarding resulting benefits for a particular business model without considering the network boundaries of blockchains. Thereby, a network boundary is defined by the hierarchical network structure and access policies. One could argue that the former is a matter of architecture and the latter is a configuration issue, and therefore has no bearing on the business perspective. However, designing viable blockchain-based business models presents a significant challenge due to the interdependent effects of business and apparently exclusive technical decisions, such as network access policies. This further complicates aligning business and technical factors [24]. We argue that the presented solution considers technical concepts to a degree that results in an effectual trade-off that significantly enriches the expressiveness of blockchain-based value networks.

This work is not without limitations: dependency paths of e^3value are currently not supported in the prototype, as these are not essential for analyzing transparency of value transfers. However, for simulating blockchain value networks, these would be required to relate value interfaces and coordinate their firing. Further, value transfers do not translate into blockchain transactions. This would be an oversimplification, as not every interaction, resulting in some value with a blockchain incurs a transaction. That is, an *occurrence* of a value transfer may represent an instance of a blockchain transaction in the network. A risk lies in users potentially assuming blockchain transactions and value transfers to be the same. We consider this however a training issue that we plan to investigate in user workshops for further evaluating the method.

6 Conclusion and Outlook

In this paper we proposed a domain-specific extension of the e^3value method for supporting the design and analysis of blockchain value networks. Thereby, an emphasis is placed on transparency as property of blockchain networks and how this can be conceptually represented in value networks. For this, an inference mechanism visualizes the transparency of value transfers based on the carrier network and network access configuration. Based on the extended graphical notation, we drafted five basic modeling patterns for blockchain-based business models.

In future work, we plan to further evaluate the method together with domain experts in a user study and add extensions for simulating the value transfers in blockchain value networks as well as for supporting e^3value's dependency paths.

References

1. Antonopoulos, A.M., Wood, G.: Mastering ethereum: building smart contracts and dapps. O'reilly Media (2018)
2. Clohessy, T., Acton, T., Rogers, N.: Blockchain Adoption: Technological, Organisational and Environmental Considerations. In: Business Transformation through Blockchain, pp. 47–76. Springer (2019)
3. Curty, S., Fill, H.G.: Exploring the Systematic Design of Blockchain-based Applications Using Integrated Modeling Standards. In: Proceedings of the PoEM 2022 Workshops. CEUR, vol. 3298. CEUR-WS.org (2022)
4. Curty, S., Fill, H.G.: BLEVEX - Blockchain e3value extension - ADOxx library (Sep 2023). https://doi.org/10.5281/zenodo.8380715
5. Curty, S., Härer, F., Fill, H.G.: Towards the comparison of blockchain-based applications using enterprise modeling. In: Proceedings of the ER Demos and Posters 2021. CEUR, vol. 2958, pp. 31–36. CEUR-WS.org (2021)
6. Curty, S., Härer, F., Fill, H.G.: Design of blockchain-based applications using model-driven engineering and low-code/no-code platforms: A structured literature review. Software and Systems Modeling (Jun 2023). https://doi.org/10.1007/s10270-023-01109-1
7. de Kruijff, J.T., Weigand, H.: Understanding the blockchain using enterprise ontology. In: CAiSE Conference. Lecture Notes in Computer Science, vol. 10253, pp. 29–43. Springer (2017). https://doi.org/10.1007/978-3-319-59536-8_3

8. Fernando, Y., Saravannan, R.: Blockchain Technology: Energy Efficiency and Ethical Compliance. Journal of Governance and Integrity **4**(2), 88–95 (Mar 2021). https://doi.org/10.15282/jgi.4.2.2021.5872
9. Ferstl, O.K., Sinz, E.J.: Modeling of business systems using som. In: Handbook on architectures of information systems, pp. 347–367. Springer (2006)
10. Fill, H.G., Karagiannis, D.: On the Conceptualisation of Modelling Methods Using the ADOxx Meta Modelling Platform. Enterprise Modelling and Information Systems Architectures (EMISAJ) **8**(1), 4–25 (2013). https://doi.org/10.18417/emisa.8.1.1
11. Fill, H.G., Meier, A. (eds.): Blockchain: Grundlagen, Anwendungsszenarien und Nutzungspotenziale. Edition HMD, Springer Fachmedien, Wiesbaden (2020). https://doi.org/10.1007/978-3-658-28006-2
12. Fill, H., Meier, A.: Blockchain Kompakt. Springer (2020), https://link.springer.com/book/10.1007/978-3-658-27461-0
13. Flovik, S., Moudnib, R.A., Vassilakopoulou, P.: Determinants of blockchain technology introduction in organizations: an empirical study among experienced practitioners. Procedia Computer Science **181**, 664–670 (2021)
14. Frank, U.: Domain-Specific Modeling Languages: Requirements Analysis and Design Guidelines. In: Domain Engineering: Product Lines, Conceptual Models, and Languages, pp. 133–157. Springer (May 2013). https://doi.org/10.1007/978-3-642-36654-3_6
15. Frank, U.: Multi-perspective enterprise modeling: foundational concepts, prospects and future research challenges. Software & Systems Modeling **13**, 941–962 (2014)
16. Gómez, C., Pérez Blanco, F.J., Vara, J.M., De Castro, V., Marcos, E.: Design and development of Smart Contracts for E-government through Value and Business Process Modeling. In: HICSS (2021). https://doi.org/10.24251/HICSS.2021.254
17. Gordijn, J., Akkermans, H., Van Vliet, H.: What's in an Electronic Business Model? In: Knowledge Engineering and Knowledge Management Methods, Models, and Tools, vol. 1937, pp. 257–273. Springer (2000). https://doi.org/10.1007/3-540-39967-4_19
18. Gordijn, J., Wieringa, R.: E3value User Guide. The Value Engineers B.V, Soest, NL, first edn. (Feb 2021)
19. Grossman, S.J., Hart, O.D.: The Costs and Benefits of Ownership: A Theory of Vertical and Lateral Integration. Journal of Political Economy **94**(4), 691–719 (Aug 1986). https://doi.org/10.1086/261404
20. Group, T.O.: ArchiMate® 3.1 Specification (2019), https://pubs.opengroup.org/architecture/archimate3-doc/, last accessed: 21.06.2023
21. Härer, F.: Process modeling in decentralized organizations utilizing blockchain consensus. Enterprise Modelling and Information Systems Architectures (EMISAJ) International Journal of Conceptual Modeling **15**, 13:1–13:17 (2020). https://doi.org/10.18417/emisa.15.13
22. Hillier, B.: The Economics of Asymmetric Information. Macmillan Education UK, London (1997). https://doi.org/10.1007/978-1-349-25485-9
23. ISO/TC 307 Technical Committee: ISO/TS 23635:2022: Blockchain and distributed ledger technologies – Guidelines for governance. Standard, International Organization for Standardization, Geneva, CH (2022)
24. Janssen, M., Weerakkody, V., Ismagilova, E., Sivarajah, U., Irani, Z.: A framework for analysing blockchain technology adoption: Integrating institutional, market and technical factors. International Journal of Information Management **50**, 302–309 (2020). https://doi.org/10.1016/j.ijinfomgt.2019.08.012

25. Jiang, S., Ræder, T.B.: Experience on using ArchiMate models for modelling blockchain-enhanced value chains. In: EASE 2022 Conference. pp. 375–382. ACM (2022). https://doi.org/10.1145/3530019.3531346
26. Kannengiesser, N., Lins, S., Sander, C., Winter, K., Frey, H., Sunyaev, A.: Challenges and common solutions in smart contract development. IEEE Transactions on Software Engineering (2021). https://doi.org/10.1109/TSE.2021.3116808
27. Kim, H.M., Laskowski, M.: Toward an ontology-driven blockchain design for supply-chain provenance. Intell. Syst. Account. Finance Manag. **25**(1), 18–27 (2018). https://doi.org/10.1002/isaf.1424
28. Lakkakula, P., Bullock, D.W., Wilson, W.W.: Asymmetric information and blockchains in soybean commodity markets. Applied Economic Perspectives and Policy **44**(1), 273–298 (2022). https://doi.org/10.1002/aepp.13159
29. McCarthy, W.E.: The REA Accounting Model: A Generalized Framework for Accounting Systems in a Shared Data Environment. The Accounting Review **57**(3), 554–578 (1982)
30. Nguyen, C.T., Hoang, D.T., Nguyen, D.N., Niyato, D., Nguyen, H.T., Dutkiewicz, E.: Proof-of-Stake Consensus Mechanisms for Future Blockchain Networks: Fundamentals, Applications and Opportunities. IEEE Access **7**, 85727–85745 (2019). https://doi.org/10.1109/ACCESS.2019.2925010
31. Osterwalder, A., Pigneur, Y.: Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers. John Wiley and Sons (2010)
32. Perrelet, S., Fill, H.G., Dibbern, J.: A Modeling Approach for Blockchain-inspired Business Models: An Extension of the E3 -Value Method. In: HICSS (2022). https://doi.org/10.24251/HICSS.2022.558
33. Poels, G., Kaya, F., Verdonck, M., Gordijn, J.: Early identification of potential distributed ledger technology business cases using e3value models. In: ER 2019 Workshops. vol. 11787, pp. 70–80. Springer (2019). https://doi.org/10.1007/978-3-030-34146-6_7
34. Sandkuhl, K., Stirna, J., Persson, A., Wißotzki, M.: Enterprise Modeling - Tackling Business Challenges with the 4EM Method. The Enterprise Engineering Series, Springer (2014). https://doi.org/10.1007/978-3-662-43725-4
35. Sigrist, N.: Innovation in the construction planning phase – how to prevent building models from being subsequently modified. https://www.swisscom.ch/en/business/blockchain/news/innovation-in-der-bauplanung.html (2023)
36. Wang, R., Lin, Z., Luo, H.: Blockchain, bank credit and SME financing. Quality & Quantity **53**(3), 1127–1140 (May 2019). https://doi.org/10.1007/s11135-018-0806-6
37. Wieland, M., Fill, H.: A domain-specific modeling method for supporting the generation of business plans. In: Modellierung 2020, 19.-21. Februar 2020, Wien, Österreich. LNI, vol. P-302, pp. 45–60. Gesellschaft für Informatik e.V. (2020), https://dl.gi.de/20.500.12116/31846
38. Wirtz, B.W., Pistoia, A., Ullrich, S., Göttel, V.: Business models: Origin, development and future research perspectives. Long Range Planning **49**(1), 36–54 (2016). https://doi.org/10.1016/j.lrp.2015.04.001