Decision Framework for Improved Distributed Ledger Technology Utilization

Tomas Bueno Momcilovic
fortiss GmbH, momcilovic@fortiss.org

Matthias Buchinger
fortiss, mbuchinger@outlook.de

Vincent Delitz
Technical University of Munich, vincent.delitz@tum.de

Dian Balta
fortiss, balta@fortiss.org

Decision Framework for Improved Distributed Ledger Technology Utilization

Emergent Research

Abstract

Distributed ledger technology (DLT) has been salient in research and practice for over a decade, with substantial investments in numerous areas. Still, the absence of a rapid, industry-wide success fuels skepticism and numerous decision frameworks emerged focusing on how to scaffold DLT utilization. However, a consideration of needs, added value, and integrative design of DLT-based systems remains overlooked. By analyzing existing frameworks and DLT Proof-of-Concepts, we provide a research-in-progress decision framework for making evidence-based decisions on whether to use DLT and how to design a technology bundle for specific cases. Our main contribution centers on the focus on rapid collaborative prototyping. For applicability and validation, we implement the framework in an online questionnaire-like tool that generates a detailed report as a basis for an informed decision. While beneficial for academia and practice, our framework draws clear directions for future research on complementary tools, enhanced recommendations and the design of feasible DLT solutions for real world challenges.

Keywords (Required)

DLT, blockchain, decision framework, software architecture

Introduction

Since the publication of the Bitcoin white paper (Nakamoto, 2008), distributed ledger technology (DLT) has become a topic of much interest and contention. Beyond cryptocurrencies, DLT implementations have generated successes in industries such as supply chains (e.g., Jensen et al., 2019), but also disappointment in terms of wider sustained suitability (Sodhi et al., 2022) and usefulness (Ammous, 2016) that would justify large investments. The contradictions can be summarized as the challenge of the law of the instrument: a potential bias towards searching for need and value of a tool, instead of questioning the tool itself.

As a countermeasure to both hype and disillusionment, a common approach has been to enable managers and engineers to evaluate whether the DLT would generate value and/or fit their technological capabilities. Tools such as decision trees or questionnaires have been put forward in academia and practice (cf. e.g., Koens & Poll, 2018), but they seem not to fulfill all existing needs. A predominant focus has been put on gatekeeping (should you use DLT?) or high-level suggestions (should you use private, hybrid, or public DLT?). This scattered and limited insight overlooks how architectural components fit with the business case, which can increase sunk costs (Bogers & Horst, 2014). In such context, rapid, collaborative and iterative prototyping of DLT systems based on integrated design is potentially highly beneficial.

We aim at addressing this gap by combining both managerial and engineering perspectives into a decision framework. Where the systematic review of literature provides the basis for determining added value, our contribution centers on providing a complete canvas with DLT and non-DLT components, according to needs and technological capabilities of the user organization in question. Our framework thus suggests research-driven design decisions, for managers and engineers to develop existing or future applications collaboratively, which is the key to minimize investment risks (Bogers and Horst, 2014), or alternatively, systems engineers with process ownership. The audience of framework are thus users who are not necessarily proficient in DLT prototyping, yet intend to implement changes in their business process based on evidence from research and practice, and a tangible prototype.

Background and Related Work

DLT, including blockchain (Burkhardt et al., 2018), is a combination of computer science, cryptography, and economic concepts: linked lists, distributed networking, hashing, digital signatures, asymmetric encryption, ledgers, and incentive mechanisms for coordination of participants towards building consensus (Wattenhofer, 2017). It allows "secure processing of transactions between untrustworthy parties in a decentralized system" (Kannengießer et al. 2019), while maintaining a single point of truth (Wattenhofer, 2017). Thereby, DLT is especially relevant for interorganizational cases, where many parties interact (Zheng et al. 2018). While the number of DLT systems with diverse architectures grows (Ballandies et al. 2018), a core difference is the support for smart contracts (Clack et al. 2017): computer programs inside DLT networks that automatically execute transactions if pre-specified contractual terms are fulfilled.

With respect to extant research on DLT utilization frameworks, our theoretical contribution lies in three areas. First, earlier frameworks filter questions based on experience or experts' validation (cf. e.g. Betzwieser et al., 2019) or build on informal sources (e.g. Twitter) which add little substance. We rely on the formal review process for quality but include all questions in common groups. This maximizing approach means that rarely discussed but important issues such as version compatibility, legality and lifecycle maintenance are covered. Second, to our knowledge, existing frameworks do not provide clear architectural component recommendations, nor bind them to the determined business or societal need; our approach tackles that by introducing close-ended component recommendations which are traceable to the user's domain. Third, only three articles address some issues related to architecture design: Betzwieser et al. (2019), Gourisetti et al. (2019), and Abdo & Zeadally (2020). We build on their insights and complement their work with best practices from technical papers and case studies to create a novel comprehensive approach to design, which users can embed into their preferred workflows.

Research Approach

We conduct an argumentative-deductive analysis (Wilde & Hess, 2007), which comprised theoretically-founded concept and prototype development. We conducted a hermeneutic review (Boell et al., 2014) of the accessible body of knowledge of 289 articles, retrieved from journals, books, conference or white papers using keyword search and the snowball method for mining citations (Wohlin, 2014). As a starting point for analysis, we combined the systematic review method (Shepperd, 2013) with the method by Koens & Poll (2018), which breaks down DLT frameworks into questions and outcomes and groups them according to similarity, and expanded the groups using the work of Colomo-Palacios et al. (2020). The survey resulted in 36 decision frameworks of more than 400 questions for evaluating DLT use cases, which we grouped according to similarity and reformulated into core underlying questions. Our systematic review of non-framework literature resulted in 32 technical papers for architecture design recommendations based on performance measures, industrial case studies or best practices in DLT-oriented software development.

To expand our analysis beyond secondary data, we sampled primary data from 19 case studies on DLT prototypes for real-life challenges, which were elaborated by graduate students (teams sized 2-6 members) together with a partner from practice in a 4-month course in the period 2019-21 at the Technical University of Munich (TUM). All cases were conducted using the Balta et al. (2019) three-step method of need analysis, feasibility study, and architecture design. Reports and prototype repositories (see: https://git.fortiss.org/Blockchain) from the case studies were used to identify best practices. We relied on the iterative approach to qualitative research (Srivastava and Hopwood, 2009) to code (i) the background of the challenge-providing organization, their stakeholders, and the use case for elicitation purposes; and (ii) design decisions and steps which students took to build a prototype for architectural recommendations.

Framework for DLT Utilization

Our framework (cf. Figure 1) centers on three consecutive industry-agnostic stages, formed as a three-step integrated and iterative process with clear inputs and outputs. The entire process comprises 86 self-contained questions and close-ended outcomes, based on evidence from literature and case studies. It is available as an online tool at http://dlt.fortiss-demo.org/, to be more easily usable in practice.

Inputs and outputs are document-based artifacts. The **feasibility** input is an umbrella category of written knowledge which is collected by managers using existing tools to map the business landscape and technical constraints to the yes/no questions in the first stage. For example, a Stakeholder Dependency Matrix is a tool for diagramming the roles and interdependencies of stakeholders (Balta et al., 2015) which helps both determine feasibility of a DLT-oriented project and crystallize some shared needs and points of contention. The result is a documented **value-based decision** that DLT would in fact bring value through efficiency, security, financial gains and/or societal benefit, based on the fulfilled constraints of both the use case and on-chain technical capabilities. Building on the answers from the previous stage, the particular **needs** are written specifications that are further elaborated in deeper analysis and collaboration with all relevant stakeholders to answer yes/no questions. The resulting list of technical requirements is an output of **business-technology alignment** on the public-hybrid-private DLT path (i.e. high-level architecture), feeding into the final stage. With additional multiple-choice questions that require the technical expertise of engineers, the third stage translates the high-level **architecture** and corresponding requirements into an output canvas on a component level, forming a **blueprint** for the complete DLT application. The final omitted implementation stage is no different from traditional software engineering.

Should I use DLT in my use case in the first place? In the first stage "**EVALUATION**", users evaluate the suitability of their use case with the goal to satisfy all necessary preconditions and understand potential risks of DLT. **Business process** questions arise from the DLT properties that would unavoidably affect organizational processes and stakeholders, including explicit measures that fix misaligned interests and trustless relationships at the expense of third parties (cf. e.g. Koens & Poll, 2018). Constraints imposed by stakeholders may create obstacles for DLT use, but also a potential to prove benefits of the change provided the constraints are relaxed. **Existing IT** questions arise from the technical constraints concerning hardware, throughput and network scalability (cf. e.g. Hribernik et al., 2020), as well as the immutability of data entered into the ledger (cf. e.g. Lo et al., 2017). Finally, **subjective fitness** includes aspects for which DLT is more appropriate than the alternatives. Users would need to decide whether certain benefits (e.g. avoiding censorship; cf. e.g. Lapointe & Fishbane, 2019) are worth the costs. The output, a value-based decision, means that users have demonstrated - or disproved - the value of DLT for case under study. This stage corresponds to the grouped evaluation-centered questions from extant frameworks.

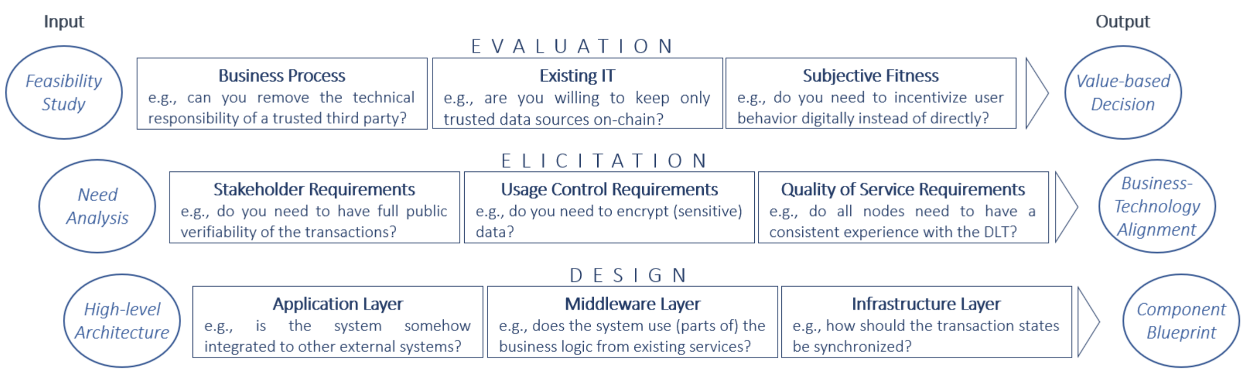

Figure 1. Towards evidence-based decisions for DLT utilization based on our framework

Which DLT type is the best fit? The second stage, "**ELICITATION**" helps users to elicit detailed requirements that can be translated to traceable architectural design decisions. Managers and engineers preferably elicit these together by specifying the most important questions represented by an appropriate approach - public, hybrid or private DLT. **Stakeholder requirements** center on the participants in the process, their interactions, and their incentives (cf. e.g. Gourisetti, Mylrea & Patangia, 2019). Analysis is conducted on who determines consensus, writes and audits data, among other aspects. **Usage control requirements** address privacy or security reasons for compartmentalizing the flow of data and managing permissions (cf. e.g. Belotti et al., 2019), where e.g. private DLT offer increased control at the expense of public verifiability of transactions. **Quality of service requirements** provide initial points for determining scalability and performance needs (cf. e.g. Betzwieser et al., 2019), to be later specified as on-chain and off-chain design components. Once the stage is completed and decisions are traced back to the needs, the output is a confirmation of business-technology alignment.

How could the software architecture for my DLT-based use case look like? The third stage "**DESIGN**" centers on three layers of a DLT-based software architecture. The application layer deals with data representation, the extension layer with data dissemination, and the protocol layer with data verification (Luo & Yan, 2021, Zhao et al., 2021). The **application layer** covers typical software application topics, such as the interaction with the system and users. It also includes classical system administration, such as monitoring or logging. Offering DLT-specific functionality for users is the main task of the **middleware layer**, where programmable and automatable business logic is codified in smart contracts. The backbone of the DLT-based system architecture is described in the **infrastructure layer** through three sub-layers: network, processing, and storage. Details about the deployment of components, consensus, scalability and security are covered in the network layer, linking to the answers of previous stages. The storage layer deals with data management, specifically the types of on-chain and off-chain storage and the links between them. The output is a suggested software architecture on component-level.

The framework allows for users to switch between the stages (depicted in Figure 1 as small arrows) and thus validate the results in previous steps. Main contributions center on 1) having a holistic and comparatively rapid approach for getting to a prototype from an unelaborated need; and 2) being able to iterate continuously, so as to incorporate changing context and environment over time, or help managers secure stakeholder buy-in for the changes whose value and technical details were specified in the first iteration.

Conclusion und Future Work

We present a preliminary version of a decision framework whose comprehensiveness refers to maximally representative evaluation of DLT value from extant literature, and novel component recommendations for the suitable software architecture that matches the requirement of a use case. This contribution with a special focus on architecture design represents a beginning in the further investigation of the design of DLT-based software systems with a clear focus on adding value based on technology. Finally, its implicit role is to enable user organizations to motivate change using a tangible output, either to test and implement it, to proving value of DLT to stakeholders in multiple iterations.

Still, we present a work-in-progress with corresponding limitations, mainly that the results are not fully validated through research and practices yet. Moreover, the framework guides users through important questions and derives conclusions out of the answers, but does not provide assistance for answering them. Given the number of possible tools for each topic, e.g. mapping stakeholder incentives, our contribution remains general without limits to potential methods (e.g. requirements engineering) or technology components (e.g. Hyperledger Fabric). Finally, strategies to replace trusted third parties are not explicitly tackled, but we urge users to delineate technical responsibilities which can be automated and substituted (e.g. executing transactions), from social or legal responsibilities (e.g. auditing) which cannot.

To address some of the limitations, we developed an online tool[1] with all 86 questions from the framework. With the tool, users can get a rapid and realistic answer to the same overarching questions in each stage. The tool allows users to share and save answers, enabling collaboration between managers with necessary domain knowledge for the first and second stages, and engineers who can translate the business needs to technical specifications in the third stage.

We believe that our framework and tool serves as a good starting point and “one-stop-shop” for developing suitable DLT-based architectures from combined evidence of scattered literature and multiple prototypes. This can help to speed up the development process and increases the likelihood of success of a DLT project in the future. Moreover, our preliminary results can steer a vivid discussion in academia towards a more evidence-based utilization of DLT cross-industries and beyond hype cycles.

REFERENCES

Abdo, J. B. & Zeadally, S. (2020). Neural network-based blockchain decision scheme. Information Security Journal: A Global Perspective, 30(3), 173 - 187.

Ammous, S. (2016). Blockchain Technology: What is it Good for? SSRN. Retrieved from https://ssrn.com/abstract=2832751

Ballandies, M.C., Dapp, M.M., Pournaras, E.: Decrypting Distributed Ledger Design - Taxonomy, Classification and Blockchain Community Evaluation. Computers and Society (2018)

[1] cf. http://dlt.fortiss-demo.org/

Balta, D., Kalogeropoulos, A., Kuhn, P. & Krcmar, H. (2019). In Search For Consensus. In: Shefali V. (ed), Proceedings of the International Conference EGOV-CeDEM-ePart 2019.

Balta, D., Greger, V., Wolf, P., & Krcmar, H. 2015. E-government stakeholder analysis and management based on stakeholder interactions and resource dependencies. 2015 48th Hawaii International Conference on System Sciences, pp. 2456–2465. IEEE.

Belotti, M., Božić, N., Pujolle, G. & Secci, S. 2019. A Vademecum on Blockchain Technologies: When, Which, and How. IEEE Communications Surveys & Tutorials, 21(4), 3796 - 3838.

Betzwieser, B., Franzbonenkamp, S. et al. (2019). A decision model for the implementation of blockchain solutions. 2019 Americas Conference on Information Systems.

Boell, S. K., and Cecez-Kecmanovic. D. (2014). A hermeneutic approach for conducting literature reviews and literature searches. Communications of the Association for Information Systems, 34 (1), 1 - 12.

Bogers, M. & Horst, W. (2014). Collaborative prototyping: Cross-fertilization of knowledge in prototype-driven problem solving. Journal of Product Innovation Management, 31 (4), 744-764.

Burkhardt, D., Werling, M., Lasi, H.: Distributed Ledger. Definition & Demarcation. In: IEEE (ed.) Conference Proceedings ICE/IEEE ITMC 2018. IEEE (2018)

Clack, C.D., Bakshi, V.A. and Braine, L.: Smart Contract Templates: foundations, design landscape and research directions, https://arxiv.org/abs/1608.00771 (2017)

Colomo-Palacios, R., Sanchez-Gordon, M. & Arias-Aranda, D. (2020). A critical review on blockchain assessment initiatives: A technology evolution viewpoint. Software: Evolution and Process, 32(11), 1 - 11.

Gourisetti, S. N. G., Mylrea, M. & Patangia, H. (2019). Evaluation and Demonstration of Blockchain Applicability Framework. IEEE Transactions on Engineering Management, 67(4), 1-15.

Hribernik, M., Zero, K., Kummer, S. & Herold, D. M. (2020). City logistics: Towards a blockchain decision framework for collaborative parcel deliveries in micro-hubs. Transportation Research Interdisciplinary Perspectives, 8, 1 - 8.

Jensen, T., Hedman, J. & Henningsson, S. (2019). How TradeLens delivers business value with blockchain technology. MIS Quarterly Executive, 18 (4), 221-243.

Kannengießer, N., Lins, S., Dehling, T., Sunyaev, A.: What Does Not Fit Can be Made to Fit! Trade-Offs in Distributed Ledger Technology Designs. In: Bui, T. (ed.) Proceedings of the 52nd Hawaii International Conference on System Sciences, pp. 7069–7078 (2019)

Koens, T. & Poll, E. (2018). What Blockchain Alternative Do You Need?. In J. Garcia-Alfaro et al. (eds.), Data Privacy Management, Cryptocurriencies and Blockchain Technology, 113 - 129.

Lapointe, C. & Fishbane, L. (2019). The Blockchain Ethical Design Framework. Innovations Technology Governance Globalization, 12(3-4), 50 - 71.

Lo, S. K., Xu, X., Chiam, Y. K. & Lu, Q. (2017). Evaluating Suitability of Applying Blockchain. 2017 22nd International Conference on Engineering of Complex Computer Systems (ICECCS).

Luo, H. & Yan, D. (2021): Blockchain architecture and its applications in a bank risk mitigation framework. In: Economic Research-Ekonomska Istraživanja, S. 1–19.

Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. Retrieved from https://bitcoin.org/bitcoin.pdf

Shepperd, M. (2013). Combining Evidence and Meta-analysis in Software Engineering. In De Lucia A., Ferrucci F. (eds), Software Engineering, ISSSE 2009/2010/2011. Lecture Notes in Computer Science, vol 7171. Berlin: Springer.

Sodhi, M. S., Seyedghorban, Z., Tahernejad, H. & Samson, D. (2022). Why emerging supply chain technologies initially disappoint: Blockchain, IoT, and AI. Productions and Operations Management, published online. DOI: 10.1111/poms.13694

Srivastava, P. & Hopwood, N. (2009). A Practical Iterative Framework for Qualitative Data Analysis. International Journal of Qualitative Methods, 8 (1), 76 – 84.

Wattenhofer, R. (2017). Distributed Ledger Technology: The Science of the Blockchain. Scotts Valley, CA: CreateSpace.

Wohlin, C. (2014). Guidelines for snowballing in systematic literature studies and a replication in software engineering. 18th International Conference on Evaluation and Assessment in Software Engineering.

Zhao, H., Zhang, M., Wang, S., Li, E., Guo, Z. & Sun, D. (2021): Security risk and response analysis of typical application architecture of information and communication blockchain. Neural Computing and Applications, 33 (13), 7661–7671.

Zheng, Z., Xie, S., Dai, H.-N., Chen, X., Wang, H. (2018). Blockchain challenges and opportunities: a survey. Int. J. Web and Grid Service, 14, 352–375